Timely and Non-Disruptive Response of Emergency Vehicles: A Real-Time Approach

ABSTRACT

Facilitating timely traversal of emergency response vehicles (ERVs), especially through an urban road network remains a challenging task. ERV preemption is a technique commonly used to facilitate prioritized ERV traversal. Unplanned deployment of ERV preemption can induce network-wide traffic delays. This paper presents an ERV preemption strategy (considering ERV urgency levels) that proactively manages the traffic before, during and after ERV traversal by leveraging connected infrastructure to guarantee timely ERV travel while minimizing traffic delays. We also provide worst-case wait time bounds for non-emergency vehicles during ERV preemption. Our proposed approach does not introduce any additional delay in ERV traversal over existing approaches while also showing up to 43% reduction in average non-emergency traffic delays.

1 INTRODUCTION

With 70% of the world's population expected to live in major urban cities by 2050 [3], traffic congestion continues to be one of the most pressing issues of urbanization. Most urban cities across the globe have observed a 46–71% increase in traffic congestion [31]. Limited space for urban expansion, especially for existing roadways creates a bottleneck in meeting increasing traffic demands, due to which efficient traffic management has become of prime importance.

Congested roadways not only lead to increased commute times, fuel consumption and pollution, but also hamper timely deployment of emergency response systems [2]. With over 240 million 911 calls being made every year just in the US alone [1], emergency response vehicles (ERVs) often fail to meet their target response time due to congested roads and thereby affecting the hospitalization and mortality rates [14]. Congestion further lead to *queue spillbacks* and collisions disrupting the entire road network. Queue spillbacks occur when there is a queue downstream of an intersection that disrupts the discharge of vehicles even when the light is green, thereby propagating the congestion and causing a gridlock [23].

As a common practice, the non-emergency vehicle (non-ERV) drivers are required to pull over to the edge of the roadway, when an ERV is approaching, to allow its safe traversal [29] through congested roads. However, in dense urban areas, the edges of the roadways are often occupied by moving traffic or parked vehicles, leading to confusion among the drivers when the ERV approaches. In such situations, it is not only infeasible to clear the traffic for the ERV but potentially dangerous, causing the non-ERVs and ERVs to collide. The chances of an ERV getting into a crash are even higher when it enters an intersection block with cross-traffic movements [12]. Such situations make it difficult for the ERVs to quickly and safely traverse through the road network and meet their response times. Collisions further exacerbate the network-wide traffic flow, making it difficult to restore smooth operations. Thus, the traffic lights at an intersection must detect and facilitate safe ERV traversal, known as *ERV preemption*, while reducing the traffic delays for non-ERV movements caused by the preemption.

In most existing ERV preemption techniques, the intersections independently control the traffic lights upon detecting an approaching ERV [34]. However, due to its reliance on the range of detection and local decisions, failure to detect an ERV especially in heavy traffic, hampers the ERV's travel. The network-wide ERV preemption approaches [4] switch multiple traffic lights to green as soon as an ERV is expected to arrive while turning the lights for cross-traffic movements to red. While such techniques facilitate ERV traversal, they fail to minimize its impact on non-ERV traffic leading to propagated spillbacks and gridlocks. Spillbacks can also worsen the traversal time of other imminent ERVs [5].

Existing work either provide ERV preemption mechanisms without reducing its impact on the non-ERVs [20] or optimize traffic flows through a network without facilitating ERVs [22]. However, by leveraging the connected infrastructure, ERV arrival information can be acquired well in time to not only facilitate smoother and safer ERV traversal but also reduce the queues in the cross-traffic movements such that the delays after ERV preemption are minimized. By translating a typical traffic network with both, non-emergency and emergency vehicles into a real-time task scheduling problem, we present an end-to-end ERV preemption technique that enables safe and quick ERV traversal. Using a deadline-driven approach, we also **optimally** control the network-wide traffic *before*, *during* and *after* the ERV traversal to minimize queues and reduce traffic delays. We highlight the following contributions of this paper:

- We provide an ERV preemption strategy that provides a safe and timely ERV passage through a road network of multiple intersections while optimally managing non-emergency traffic.
- By leveraging real-time ERV arrival and traffic information, our approach optimally manages the network-wide traffic before the ERV arrives and after the ERV safely crosses the intersection, such that guaranteed timely response of the ERV can be accompanied with reduced queues and spillbacks.
- By using the triage scale [8] (Table 1), we propose a priority-based deadline-driven ERV preemption mechanism.
- We provide worst-case performance analysis for the non-ERV in presence of an ERV to enhance the predictability of emergency dispatch and traffic management systems.
- We analyze the adaptability of our approach using large-scale simulations and hardware-in-loop (HIL) testbed with robots that mimic human driving vehicles through urban environment.

2 RELATED WORK

The majority of the incidents involving ERVs occur due to collisions within the intersection [5]. Traffic lights in the US are therefore equipped with detectors, such as the 3M Opticom™ [24] using

Conference'17, July 2017, Washington, DC, USA
2022. ACM ISBN 978-x-xxxx-xxxx-x/YY/MM…$15.00
https://doi.org/10.1145/nnnnnnn.nnnnnnn

infrared and GPS communication, or IoT-based approaches with multiple sensors [17]. Such approaches are not only vulnerable to interference, but also lead to longer queues at the intersections downstream in urban areas due to the lack of coordination [18].

ERV traversal through a network of intersections is enabled by green wave coordination [15] where consecutive traffic lights along the ERV's path are turned green after a constant offset-based time delay or a fuzzy decision-making process [4]. However, such heuristic mechanisms, lead to prolonged red lights for the conflicting flows causing long wait-times and possible spillbacks [16]. We discuss in our evaluation (Section 6) that our approach improves the ERV travel time as compared to the existing work, while also reducing the network-wide non-ERV traffic delays.

ERV preemption with connected and autonomous vehicles (CAVs) present in the traffic is studied in [13, 33] that utilize vehicle-to-everything (V2X) and cloud connectivity to gather real-time traffic information and control each vehicle for ERV preemption. The presence of 100% CAVs on road however, is not expected until 2050 [9]. While we leverage basic connectivity between intersections within our road network to disseminate ERV information, our approach is applicable for conventional, mixed and autonomous traffic.

Conversely, the existing traffic approaches that optimize traffic for a road network do not consider ERV preemption [19]. The existing work [21, 22] proposed a task model to represent traffic through an intersection or a network of intersections. Specifically, in [22], a recovery mechanism to counter unexpected traffic disruptions using optimal strategies was provided. However, facilitating ERV movements through the road network was not addressed. In this paper, we leverage the real-time task model for a road network to provide timeliness and safety guarantees for ERV traversal while ensuring predictable traffic flow through the network.

The available ERV preemption mechanisms are heuristic and fail to minimize the ERV's impact on network-wide traffic and vice-versa. The existing traffic control strategies minimize travel time through the road network but they do not allow ERV preemption. In our work, we propose an end-to-end solution that utilizes real-time traffic and ERV arrival information to a) ensure timely ERV traversal, b) proactively organize traffic before ERV arrival to reduce traffic delays and c) clear congestion after ERV traversal.

3 SYSTEM MODEL

A real-time task model to represent traffic flows through a road network was proposed in [22]. In this paper we extend the task model to accommodate for prioritized ERV traversals.

3.1 Road Network and Traffic Lights

For our model, we use a Manhattan grid-like $m \times n$ road network formed by m and n arterials travelling along the east-west and north-south direction, respectively. Therefore, there are mn intersections in our road network. Such a network can either be standalone or a part of a larger urban area. An intersection is formed by four arterials, each traveling in a different direction, where the traffic flow within each arterial is controlled by a traffic light. Each arterial passes through multiple intersections across the network forming *links* that connect two consecutive intersections. For example, in a 3×3 network, there are nine intersections formed by three arterials traveling along the north-south and east-west directions each (Figure 1). We use the following notations to identify the links, arterials, and intersections within the network [22]:

- The direction of travel is denoted by D, such that, $D \in \{N, E, W, S\}$. Here, N, E, W and S stand for north, east, west and south, respectively. Additionally, conflicting directions are denoted by D and D' such that if $D = \{S, N\},\ D' = \{E, W\}$ and vice-versa.
- We use a dummy operator α such that $\alpha = ij,\ i \in [1, m],\ j \in [1, n]$, where m and n are the arterials for east-west and north-south traffic respectively.
- $A^D_{\alpha_r}$ denotes the r^{th} arterial for traffic in the direction D where,

$$\alpha_r = \begin{cases} ir, \text{where } i \in [1, m], 1 \leq r \leq n, \text{ if } D = \{S, N\} \\ ri, \text{where } 1 \leq r \leq m, i \in [1, n], \text{ if } D = \{E, W\}. \end{cases}$$

- Links that connect consecutive intersections collectively form an arterial. Therefore, an arterial $A^D_{\alpha_r}$ can be represented as a set of links $\{L_i^{D*}\}, \forall i \in [1, \lambda]$ where, $\lambda = m$, if $D = \{S, N\}$ and $\lambda = n$, if $D = \{S, N\}$. Here, $*$ denotes one or more lanes within each link and is dropped when the context is clear.
- $I_{rr'}$ is the intersection formed by arterials $A^D_{\alpha_r}$ and $A^{D'}_{\alpha'_r}$.

In this $m \times n$ network, each traffic flow through an arterial (and hence links) is a part of a traffic flow pair such that the flows within each pair are *non-conflicting* to each other. Non-conflicting flows can cross the intersection at once without disrupting or hampering each other, i.e., the vehicles traveling through arterials (and the corresponding links) A^E_α and A^W_α, or A^N_α and A^S_α can utilize the intersection I_α simultaneously. Traffic lights at each intersection within the network allow only the non-conflicting flows to enter and access the intersection at once, as per the *phase sequences*.

Similar to [21, 22], our model considers that the traffic flow within the $m \times n$ network is managed by a *traffic manager* (TM), denoted by $TM_{m,n}$, that aggregates real-time traffic information within the network from traffic sensors, forecast data and/or from neighboring TMs. The TMs then control the traffic lights as per some traffic control technique. We assume that the relevant data is made available to the TM through minimal connectivity between the intersections. Explicitly defining how such data is acquired is out of scope of this work. The connected infrastructure is also used to acquire ERV information which will be discussed in detail (Section 5.2). Additionally, the size of the $m \times n$ network is determined as per the feasibility of reliable connectivity among the traffic infrastructure, communication latency overhead and available computational resources to perform calculations in real-time.

3.2 Non-Emergency Traffic Flow

The traffic lights at each intersection within the network change between green-yellow-red states as per the assigned timings. Each traffic light changes its state once in a cyclic pattern based on the phase sequence and *cycle time (T_c)*. All intersections within the purview of a TM have a fixed T_c value to synchronize the traffic patterns and is selected as per the network capacity [28]. An incoming link L^D_α is characterized by the tuple $\{a^D_{\alpha,k}, q^D_{\alpha,k}, z^D_\alpha\}$, where $a^D_{\alpha,k}$ is the incoming non-emergency vehicle flow rate, $q^D_{\alpha,k}$ is the number of vehicles queued in the link during the k^{th} traffic cycle,

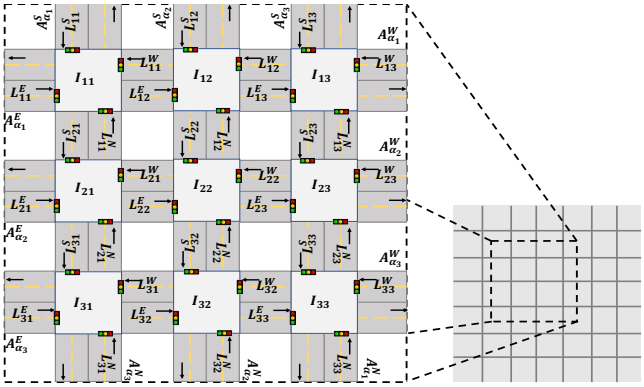

Figure 1: A typical 3×3 road network [22]

and z_α^D is the *link capacity*, i.e., the maximum number of vehicles that can enter a link before spillback occurs, and the remaining notations are as explained earlier. As per [21], the link capacity can be estimated as $z_\alpha^D = \frac{l_\alpha^D}{v_l + d_{safe}}$, where l_α^D is the length of the link L_α^D and v_l and d_{safe} denote the average vehicle length and safe distance required between any two consecutive vehicles, respectively.

The incoming non-emergency vehicle flow rate, $a_{\alpha,k}^D(t)$ is a time-varying quantity, however, in a given traffic cycle, under dynamic traffic conditions, the worst-case traffic flow can be bounded based on traffic information from the intersections. We thus refer to $a_{\alpha,k}^D(t)$ as $a_{\alpha,k}^D$ for the rest of the paper to determine non-ERV flow rates. When the traffic lights change and vehicles are allowed to cross the intersection, the discharge pattern is represented by the *saturation headway model* [27] and is given by,

$$T_k = h \cdot n_{\alpha,k}^D + t_l. \tag{1}$$

Here, T_k denotes the time required to discharge $n_{\alpha,k}^D$ vehicles from link L_α^D during the k^{th} traffic cycle when the saturation headway $h = 2$ s and the lost time $t_l = 4$ s are considered, respectively [11].

3.3 Emergency Response Vehicles

The emergency vehicles are often guided by a back-end dispatch system with various routing mechanisms to calculate the optimum path from its origin to the emergency location [25]. By leveraging V2X connectivity between the ERV (or the back-end system) and the connected infrastructure, the intersections within our $m \times n$ road network that will be affected by the ERV traversal can be identified in advance. The ERVs' arrival information can then be disseminated to the TM to allow safe and timely traversal of the ERV. In our model, an incoming ERV is denoted by v_e and is represented by the tuple, $\{l^e, s^e, \pi^e\}$. Here, l^e, s_e, π^e denote the current location coordinates, the desired speed and the priority level of an ERV. We schedule the traffic such that the ERV maintains its fixed desired speed. The arrival times and the deadlines are assigned to the ERV based on the priority level defined by the triage scale [8]. Preempting multiple emergency vehicles at once is left for future work.

Setting priorities as per the triage levels. Triage scale [8] provides critical ERV information that is already being utilized in emergency and disaster management to prioritize and establish time frames for emergencies. Since the ERV response time is directly influenced by the traffic conditions, it is crucial to utilize this data in traffic planning as well [6]. In our approach, we relate each urgency level to a priority value and a delay tolerance(Table 1). The lower the priority value, the smaller the delay tolerance for the ERV to cross our $m \times n$ network. When $\pi^e = 1$, the ERV must travel as quickly as possible (zero delay tolerance) and any additional delay may lead to missed target response time. However, for lower priorities (larger π^e value), we can slightly delay the ERV travel through our network (corresponding δ value) and still meet the ERV deadlines while allowing the conflicting flows to cross the intersections.

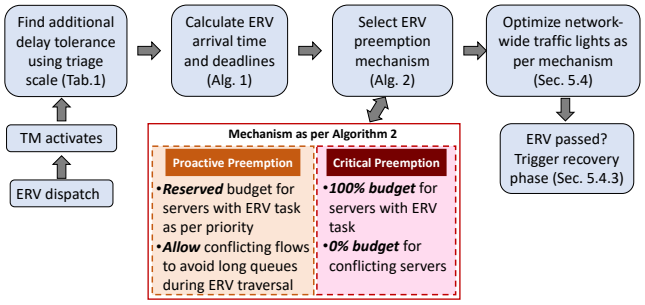

Figure 2: TM task flow and preemption mechanisms

4 SYSTEM OVERVIEW

As discussed in Section 2, the existing preemption strategies deploy a green wave to facilitate ERV traversal. However, its greedy mechanism leads to spillbacks in the road network. With the real-time task scheduling approach presented in this paper, by providing the timeliness guarantees of a real-time approach, we need not necessarily enable the green wave as soon as an ERV is detected to facilitate its traversal. By leveraging connectivity among the traffic infrastructure and real-time traffic information, we can *proactively preempt* the ERV traversal such that the ERV deadlines are met, while providing some green time to the conflicting flows using an optimal strategy to minimize the queues caused by ERV preemption.

As mentioned in Section 3, the traffic manager (TM) is responsible to acquire real-time traffic and ERV information to control the traffic lights within the entire network. When an ERV dispatch information arrives, the traffic manager (TM) obtains the priority and the delay tolerance for the ERV as per the triage scale (Table 1). The TM then calculates the arrival time and the deadline for the ERV task(Algorithm 1). Depending on the arrival time of the ERV and the traffic conditions within the network, our proposed approach selects a suitable preemption mechanism (Figure 2).

Proactive preemption provides timeliness guarantees for the ERV preemption while also optimally managing the traffic through the network *before* and *during* the ERV traversal. By providing additional green times to the conflicting traffic flows, this approach not only guarantees ERVs response times as per their priority, but also reduces traffic delays across the network.

Critical preemption is triggered when the traffic flow in the network is heavy and the ERV deadline does not allow for a *proactive preemption*. Our critical preemption approach enables green wave through the arterial that the ERV traverses, while optimally adjusting the traffic flow for the rest of the network to accommodate for the green wave to alleviate its impacts on traffic delays. Depend-

Table 1: Relating triage scale with traffic delay tolerance

Urgency Status	Assessment Time	Priority (π_e)	Traffic Delay Tolerance (s)	Example (s)
Resuscitation	0 minutes	1	δ_1	0
Emergent	15 minutes	2	δ_2	45
Urgent	30 minutes	3	δ_3	90
Less Urgent	60 minutes	4	δ_4	135
Nonurgent	120 minutes	5	δ_5	180

ing on the chosen preemption mechanism for the intersections impacted by the ERV traversal (Algorithm 2), an *optimal* network-wide strategy is formulated to avoid spillbacks and maximize traffic flow (Section 5). The preemption is active until the ERV safely passes through the road network and its deadlines are met. We then activate our *recovery phase* (Section 5.4), where we modify the per-arterial recovery approach presented in [22] to stabilize traffic across the network where we provide additional green times to the links prone to spillbacks. Once the traffic operations are back to normal and there are no ERV traversal requests, the TM task follows a default traffic strategy, such as [22]. As per results (Section 6), our approach ensures that the ERV travel time is not hampered while reducing the traffic delays in the network.

5 REAL-TIME SCHEDULING OF EMERGENCY VEHICLES THROUGH A ROAD NETWORK

We model an $m \times n$ road network (Figure 1) controlled by a traffic manager, $TM_{m,n}$, that schedules non-emergency traffic and emergency vehicles through this network, as a set of real-time tasks. We first describe the real-time task model in [22] for traffic flow control through an $m \times n$ network, but lacks the preemption mechanisms and modelling of emergency vehicles.

5.1 Review of Real-Time Task Model for a Network of Intersections

As shown in Figure 3, the non-emergency vehicles are represented as aperiodic tasks (τ_i) that have unknown arrival times (r_i) and known execution times (c_i) as per the traffic flow rates and the saturation headway model to travel though the intersections, respectively. If the traffic lights are red, the vehicles form a queue until the lights turn green. This resembles aperiodic tasks (vehicles) joining an aperiodic task queue (links) waiting to be executed on the shared resource (intersections). Since the traffic lights at each intersection modulate the vehicle flow, they are equivalent to sporadic servers (S_S) serving each task queue as per the assigned budget (B_S) and its inter-arrival time (T_S) [7]. The budget and the arrival time for each sporadic server is decided by the traffic manager task (TM) at the start of each traffic cycle. For example, in an $m \times n$ traffic network, with each intersection having four incoming flows from each direction $D \in \{N, E, W, S\}$, there are mn shared resources, $4mn$ aperiodic task queues and therefore, sporadic servers managed by the $TM_{m,n}$ task. The budgets at each intersection are assigned by the $TM_{m,n}$ task such that no two conflicting flows get to enter into the intersections at once and thereby ensuring safety. The servers arrive as per their arrival times are executed on the shared resource until the assigned budget is exhausted.

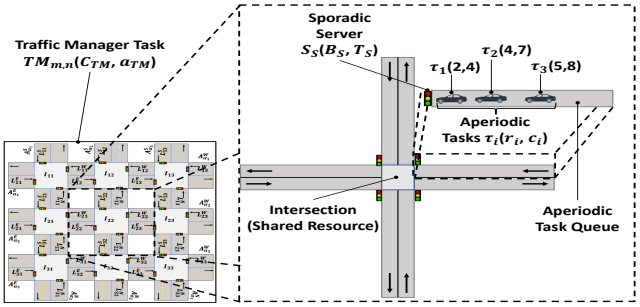

Figure 3: Real-time task model for an intersection in a road network [22]

5.2 Emergency Vehicles as Real-Time Tasks

The following information is required by the TM in our approach. (i) The route that the ERV will access within our $m \times n$ network, and (ii) the information represented by the tuple $\{l^e, s^e, \pi^e\}$ that contains the location coordinates, the desired speed, and the priority level of the ERV, respectively. Explicitly defining how such information is made available to the TM is out of scope of this work, however, we rely on existing communication between connected infrastructure, connected ERVs and/or the back-end dispatch systems that plan for the ERV's route and response times. Since the arrival of the ERV is at random, it resembles an *aperiodic* task.

From the route information, the TM determines the location coordinates of the intersections (I^e) and links (L^e) that the ERV will traverse. For simplicity of representation, we assume that the ERV traverses only in the east-west or the north-south direction through our road network, similar to the non-emergency traffic. However, our approach is applicable to the general case. For example, consider an ERV v^e, entering a 3×3 network (Figure 1) and traveling through the arterial $A^D_{\alpha_2}$. The intersections and links accessed by the ERV are $I^e = \{I_{21}, I_{22}, I_{33}\}$ and $L^e = \{L^D_1, L^D_2, L^D_3\}$, respectively. Thus the TM task must assign the budget to the servers that serve the links L^D_1, L^D_2, and L^D_3 to facilitate preemption.

The ERV task τ^e is represented by the tuple $\{a^e, d^e\}$ where a^e and d^e denote the arrival time and the deadline of ERV task. The arrival time indicates the time at which the ERV is expected to enter our road network, and the deadline indicates the time before which the ERV must exit our network to meet its priority-based response time. The values of a^e and d^e are calculated by the TM task using Algorithm 1. Since the TM task is responsible to schedule non-ERV traffic while also facilitate ERV traversals arriving at random, it resembles an *aperiodic* task with unknown arrival time and fixed execution times to perform the calculations to assign the server budgets (Figure 3).

The ERV task's deadline is calculated as per the delay tolerance for each urgency level (Algorithm 1). If the ERV has the highest priority, it has no delay tolerance ($\delta_1 = 0$, as per Table 1). Therefore, the deadline for a highest priority ERV task is equal to the travel time of the ERV from its current location to the end of the network at its desired speed (Line 6). The TM task schedules the non-ERV traffic and the ERV through the network such that the ERV meets the deadline since its failure leads to increased response times to attend life-threatening emergencies.

Algorithm 1: Calculating the ERV task parameters: arrival time and deadline using Manhattan distance [30]

Input: L_1^D, L_λ^D ▷ coordinates to the entry and exit points of the preemption arterial
l^e, s^e, π^e ▷ ERV parameters
Output: a^e, d^e ▷ arrival time and deadline for τ^e

Function GetERVTaskParams($L_1^D, L_\lambda^D, l^e, s^e$):
  $dist_1$ = ManhattanDistance(L_1^D, l^e) ▷ distance to L_1^D
  $dist_\lambda$ = ManhattanDistance(L_λ^D, l^e) ▷ distance to L_λ^D
  δ^e =GetDelay(π^e) ▷ From Table 1
  $a^e = \frac{dist_1}{s^e}$ ▷ arrival time as per desired speed
  $d^e = \frac{dist_\lambda}{s^e} + \delta^e$ ▷ deadline as per triage scale
  return a^e, d^e
End Function

To capture an ERV traversing through the arterial $A_{\alpha_r}^D$ in a real-time model, each *instance* or a job J^e of an ERV task τ^e consists of λ sub-jobs $j_i^e, \forall i \in [1, \lambda]$ where $\lambda = n$ if $D \in \{E, W\}$ and $\lambda = m$ otherwise. Here, the execution of the i^{th} sub-job j_i^e represents traversal of ERV in the link L_{ir}^D until it crosses the intersection I_{ir}. Each sub-job j_i^e is represented by the tuple, $\{a_i^e, C_i^e, d_i^e\}$ denoting the arrival time, the execution time and the deadline of the i^{th} sub-job, respectively. As shown in Figure 4, the arrival time of the sub-job indicates approaching ERV, the execution time indicates the time the ERV takes to traverse the corresponding section of the route, and the deadline indicates the time before which the sub-job must complete execution to avoid missed ERV response times.

Property 1. *An ERV task τ^e and its instance J^e consisting of λ sub-jobs $j_1^e, j_2^e, \ldots, j_\lambda^e$, has the following properties: (i) The arrival time a_1^e of the sub-job j_1^e is equal to the arrival time a^e of the task τ^e. (ii) The deadline d_λ^e of the sub-job j_λ^e is equal to the deadline d^e of the task τ^e. And finally, (iii) the arrival time a_i^e of the sub-job j_i^e is equal to the completion time of the previous sub-job $j_{i-1}^e, i = 2, \ldots, \lambda$.*

Property 1 ensures that the entire trajectory of the ERV is accounted for by our real-time model. (i) and (ii) ensure that the network-level arrival times and the deadlines of the ERV are translated to is sub-jobs. If the first sub-job j_1^e arrives at any time after the task arrives, it leads to the ERV making abrupt stop and go movements which could lead to unsafe driving conditions. Alternately, if the last sub-job j_λ^e has a deadline earlier than the task itself, the ERV will have to travel faster than its desired speed to meet its deadline leading to safety violations. Similarly, (iii) ensures the continuity of the ERV's trajectory. The i^{th} sub-job must arrive as soon as the i-1^{th} (previous) sub-job completes execution (Figure 4) to maintain the continuity in ERV traversal.

Assigning local deadlines to the sub-jobs: From Figure 4, the ERV task τ^e will meet its deadline only if each of its sub-job meet their respective deadlines. Once the TM receives the ERV information, we assign the deadlines to each sub-job to maintain the overall deadline for the ERV task as per Lemma 1.

Lemma 1. *Consider an ERV task τ^e with the arrival time and deadline of t^e and d^e respectively, representing ERV traversal through an $m \times n$ road network across the arterial $A_{\alpha_r}^D$, with λ sub-jobs j_i^e $i = 1, \ldots, \lambda$ corresponding to each segment consisting of the link L_i^D and intersection I_{ir}. If the traffic flow rate from each direction D at*

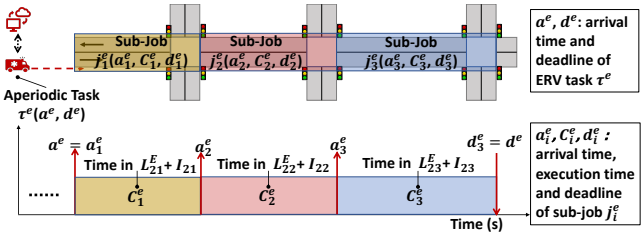

Figure 4: Modelling an ERV as an aperiodic real-time task and its sub-jobs

the intersection I_{ir} is given by a_{ir}^D, then the local deadline d_i^e for the i^{th} sub-job is given by,

$$d_i^e \propto \sum_{D \in \{N,E,W,S\}} a_i^D, \tag{2}$$

$$\text{such that, } \sum_{i=1}^{\lambda} d_i^e = d^e, \tag{3}$$

where, $\lambda = m$ if $D \in \{S, N\}$ and n otherwise.

Proof: The higher the traffic flow in a link, the longer is will be the execution time of the corresponding sub-job (Figure 4). To ensure that there is fair distribution of time to execute the sub-jobs, the local deadlines must be proportional to the traffic flows within the intersection (and hence the link). As per Equation 2, a higher deadline will be assigned to the sub-job if the net traffic flow rate through an intersection is higher. Similarly, since the arrival times of the sub-jobs are co-dependent as shown in the Property 1, missing the deadline for one sub-job causes a cascading effect and end-to-end deadline misses. If the summation of all the local deadlines is less than the total end-to-end deadline, the ERV may traverse quickly through the road network, but the conflicting flows suffer from resource starvation. Alternately, if the summation of the local deadlines is greater than the total end-to-end deadline it would clearly lead to missed end-to-end deadlines due to the sub-job precedence established in Property 1. Therefore, the deadline distribution in Equation 2 must be constrained by Equation 3.

As shown in Figure 5, an ERV with a lower priority has a higher delay tolerance, hence its deadline is later than that of the ERV with the highest priority. Upon scheduling, while the ERV with the highest priority must be executed as soon as it arrives, a lower priority ERV task execution can be delayed to schedule the conflicting flows and still meet the deadline.

With this task model for the ERV and the non-ERV traffic, we now discuss the scheduling strategy to provide timely traversal of the ERV through the road network.

5.3 Non-emergency Traffic Flow Scheduling

When there is no ERV dispatch information, the TM deploys a default traffic management scheme for optimal traffic flow. For our work, we use the network-wide traffic control strategy provided in [22] as a default scheme. However, any schedule-based traffic strategy maybe deployed at the TM. We will now briefly summarize the budget allocation strategy provided in [22] and then provide improvements required to accommodate the ERVs.

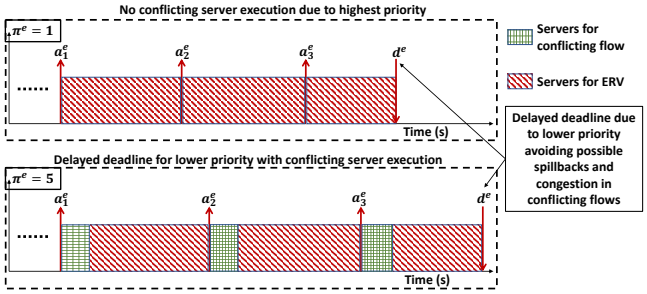

Figure 5: An example server execution for ERV preemption with varying priorities

In [22], the TM gathers the real-time traffic information from each intersection within the $m \times n$ network and allocates some budget and the arrival times for the sporadic servers in the network such that no two conflicting flows enter the intersections at the same time. For a non-emergency traffic flow only, there is no priority involved and therefore all sporadic servers arrive at the same time, at the start of each traffic cycle and are executed on the shared resource in a round-robin fashion. As mentioned, when there is no ERV arrival, for our approach, the TM controls the non-emergency traffic using the strategy defined in [22] which ensures that all traffic lights within the $m \times n$ network are *optimally* managed such that spillbacks do not occur at any place within the network, when possible. To do so, the budget allocation across all the servers in the network for each traffic cycle is formulated as an optimization problem with the following objective and constraints:

$$\text{maximize} \sum_{\substack{i=1\ldots m,\\ j=1\ldots n,\\ D\in\{SN,EW\}}} U_\alpha^D \tag{4}$$

$$\text{subject to } U_{min_\alpha}^D \le U_\alpha^D \le U_{max_\alpha}^D, \tag{5}$$

$$U_{\alpha-1}^D - U_\alpha^D \le \frac{z_\alpha^D h}{T_c}, \tag{6}$$

$$U_\alpha^D + U_\alpha^{D'} \le 1, \tag{7}$$

$$\text{where, } U_{min_\alpha}^D = \frac{h \cdot (a_k \cdot T_c + q_k - z) + t_l}{T_c}, \tag{8}$$

$$U_{max_\alpha}^D = \frac{h \cdot (a_k \cdot T_c + q_k) + t_l}{T_c}. \tag{9}$$

Here, $U_{min_\alpha}^D$ and $U_{max_\alpha}^D$ correspond to the minimum and the maximum utilization demands for the server at the intersections I_α, respectively. The minimum utilization demand is the budget required to avoid spillbacks within any given link, while the maximum demand represents the budget to dispatch all vehicles and clear a link. Similarly, U_α^D corresponds to the assigned budget for the server at intersection I_α. Since two non-conflicting flows can access the intersection at once, $U_\alpha^D = U_\alpha^S = U_\alpha^N$, when $D = SN$ and $U_\alpha^D = U_\alpha^E = U_\alpha^W$, otherwise. Since the budget allocation translates to the green time for the traffic lights, maximizing the total budget allocation across the network using Equation 4 implies increased traffic flow and thereby reduce wait times. Similarly, the constraint in Equation 5 enforces that resource starvation does not occur by assigning at least the minimum utilization requirement ($U_{min_\alpha}^D$) for each server, while avoiding over-utilization by keeping the assignment up to the maximum requirement ($U_{max_\alpha}^D$). The constraint in Equation 6 ensures that the budget allocation between consecutive intersections is such that no spillbacks occur in the system, when possible, thereby avoiding any additional delays and network-wide gridlocks caused by traffic congestion. Finally, the constraint in Equation 7 maintains the schedulability and avoid resource overloading. That is the budgets assigned to the servers at any given intersection do not exceed 100%.

Algorithm 2: Preemption mechanism based on ERV deadline and traffic flow

Input: t_{start} ▷ ERV dispatch notification received
t^e, d^e, π^e ▷ ERV arrival time, deadline and priority
$\{a_i^D, q_i^D, z_i^D\}$ ▷ traffic of links in preemption arterial
Output: preemption ▷ selected mechanism

```
Function SelectPreemption(t_start, t^e, d^e, a_i^D, q_i^D):
    t_nw = d^e − t_start
    n_arr = a_1^D t_nw                ▷ vehicles expected to arrive within t_nw
    q_sum = Σ_{i=1}^{λ} q_i^D         ▷ total vehicles in queue at t_start
    n_nw = n_arr + q_sum              ▷ total vehicles expected within t_nw
    t_clear = n_total h + t_l         ▷ time to clear n_total vehicles (Eq. 1)
    if t^e − t_start > t_clear then
        return proactive preemption
    else
        return critical preemption
End Function
```

5.4 Enabling ERV Preemption: Scheduling the ERV tasks and its Sub-Jobs

As mentioned in Section 4, our goal is to ensure that the ERV meets its required response time as per the urgency level of the situation while leveraging the possible delay tolerance of the ERVs to schedule non-emergency conflicting traffic flows as well(Figure 5). This avoids formation of longer queues and spillbacks due to the prolonged red lights for the conflicting traffic flows, once the ERV enters the network while adhering to the ERV timeliness. To facilitate ERV preemption through the traffic network, we propose and ERV-aware traffic control strategy that, when there is no ERV expected to enter in the network, the default optimal traffic control strategy as in [22] is implemented. However, an ERV preemption request may arrive at any time and hence, we deploy a polling mechanism that constantly checks for ERV dispatch notifications via the connected infrastructure such as the back-end dispatch system, RSUs or an equipped ERV itself. Once a preemption request is received, as shown in Figure 2, the TM task is activated to enable an ERV-aware traffic optimization approach which makes budget allocation and scheduling decisions by selecting one of the two preemption mechanisms (Algorithm 2), i.e., proactive preemption or critical preemption. As discussed, selection of the preemption mechanism depends on the traffic flow as well as the ERV deadlines.

Upon activation, at time t_{start}, the TM calculates the ERV task's arrival time (t^e) and the deadline (d^e) using Algorithm 1. Let us consider, that the ERV requests preemption through the arterial $A_{\alpha_r}^D$ to access our network, which comprises of links L_i^D with the parameters $\{a_i^D, q_i^D, z_i^D\}$ and intersections $I_{ir}, i = 1, \ldots, \lambda$. Based on the traffic flow rate (a_i^D) and existing queues (q_i^D) within each

Algorithm 3: Queues for ERV compensation phase

Input: t_{start} ▷ ERV dispatch notification received
$\{t_i^e\}, \{d_i^e\}$ ▷ arrival time and deadline of all sub-jobs
$\{a_i^D, q_i^D, z_i^D\}$ ▷ traffic of links in preemption arterial
T_c ▷ traffic cycle time
Output: $\{q_i^e\}$ ▷ set of queues for each link

1 **Function** GetQueues($t_{start}, \{t_i^e\}, \{d_i^e\}, a_i^D, q_i^D, z_i^D, T_c$):
2 $t_r = (d_1^e - t_{start})\%T_c$
3 $n_c^e = \left\lfloor \frac{d_1^e - t_{start}}{T_c} \right\rfloor$
4 **if** $t_r < t_l$ **then**
5 $t_r += T_c, n_c^e -= 1$
6 ▷ t_l : human reaction lost time
7 $n_r = \frac{t_r - t_l}{h}$
8 $n_{r_{in}} = a_1^D t_r$
9 $q_1^e \leftarrow \min(n_r - n_{r_{in}}, z_1^D)$
10 **for** $i = 2 \ldots \lambda$ **do**
11 $c_{i-1}^e = d_{i-1}^e - t_{i-1}^e$
12 $q_i^e \leftarrow \min(\frac{c_{i-1}^e - t_l}{h}, z_i^D)$
13 **return** $\{q_i^e\}$
14 **End Function**

link of the arterial $A_{\alpha_r}^D$, at the time of receiving the ERV information (t_{start}), the TM decides which mechanism, i.e., proactive or critical preemption to trigger. As shown in Algorithm 2, the TM calculates the duration (t_{nw}) between t_{start} and the ERV deadline, d^e (Line 2). It then estimates the total number of vehicles, n_{nw}, and the corresponding time, t_{clear} (as per Equation 1) to dispatch n_{nw} vehicles and clear the arterial for the ERV traversal (Line 5 and 6). Finally, if the time taken to clear the network is not less than the arrival time of the ERV task, we do not have any additional budget to proactively control the conflicting flows, and critical preemption must be triggered immediately (Line 7) to meet the deadlines. Else, the proactive preemption can be safely triggered (Line 9) to alleviate resource starvation for the conflicting links, while ensuring that the ERV deadlines are met. As per the selected mechanism, the TM task now allocates the budget as well as the arrival times for the corresponding servers that serve the links and intersections through which the ERV must pass while also optimizing the traffic flow through the entire network.

By leveraging the connected infrastructure, we expect that the TM will be notified about an incoming ERV, multiple traffic cycles before it arrives in the traffic network, i.e, $t^e - t_{start} > t_{clear}$ to maximize the benefits of the proposed proactive compensation to traffic congestion during and after the ERV traversal. However, in case $t^e - t_{start} \leq t_{clear}$, there is not enough bandwidth to proactively distribute budget for the conflicting flows and critical preemption is activated. However, by switching the traffic lights as per the arrival time and the deadline of the ERV task and its sub-jobs, and optimally controlling traffic for the entire network instead of a greedy green wave, we ensure that the ERV traversal is not hampered while reducing overall traffic delays.

5.4.1 Scheduling with critical preemption. Critical preemption indicates that the servers must dedicate their entire resources towards facilitating the ERV traversal through the network (Figure 2). Let the arterial through which the ERV requests for traversal be denoted by $A_{\alpha_e}^D$ and the links, intersections and the corresponding servers be denoted by L_i^e, I_i^e and $S_i^e, i = 1, \ldots, \lambda$ respectively. The servers that are serving the conflicting flows to this arterial are hence denoted by $S_i^{e'}$. Once critical preemption is triggered, the TM employs an ERV-aware optimization approach to calculate the budget allocation for the servers within the $m \times n$ network, which is formulated as follows:

$$\text{maximize} \sum_{\substack{i=1\ldots m,\\ j=1\ldots n,\\ ij \notin \alpha_e,\\ D \in \{SN, EW\}}} U_\alpha^D \tag{10}$$

$$\text{subject to } U_{\alpha_e}^D = 1, U_{\alpha_e}^{D'} = 0 \tag{11}$$

$$U_{min_\alpha}^D \leq U_\alpha^D \leq U_{max_\alpha}^D, \alpha \neq \alpha_e \tag{12}$$

$$U_{\alpha-1}^D - U_\alpha^D \leq \frac{z_\alpha^D h}{T_c}, \alpha \neq \alpha_e \tag{13}$$

Equation (7).

If $D = SN, D' = EW$, and if $D' = SN, D = EW$. Since the main goal of ERV preemption is not maximizing traffic flow but to facilitate ERV traversal first, through the intersections the ERV passes through, the objective function (Equation 10) only maximizes the traffic flow through the network for the servers that are not serving the ERV task ($ij \notin \alpha_e$). Since the arterial $A_{\alpha_e}^D$ is impacted by ERV preemption, all servers along this arterial are directly assigned 100% of the budget to facilitate rapid traversal of the ERV (Equation 11). Further, the TM does not assign any budget to the servers for the conflicting traffic flows to ensure the safety of the ERV. Constraints given by the Equations 12 and 13 ensure that the rest of the network stays live and the flow is still maximized while avoiding under-/over-utilization of the resources and spillbacks. This ensures that the effect of the ERV preemption remains localized and does not spread through the network as much as possible. Finally, the schedulability constraint (Equation 7) that ensures safety of the traffic is enforced for all servers within the network.

5.4.2 Scheduling in proactive preemption. Proactive preemption is activated, as per Algorithm 2 when the ERV does not necessarily require immediate preemption and the TM utilizes this time to provide additional budget to the flows conflicting the ERV movement. The budget in this mechanism are assigned, such that the vehicle queues in the ERV's arterial and its links are maintained using Algorithm 3. The vehicle queues are chosen based on the arrival time, deadline and the delay tolerance of the ERV task. The arrival time and the budget for the sporadic servers are set such that when the servers are executed, the vehicle queues are dispatched just in time to enable smooth and safe ERV traversal and thereby meeting the local sub-job deadlines (hence the ERV deadline). Upon selecting the proactive preemption, we calculate the number of traffic cycles, n_c^e of T_c time in the time span of receiving the ERV information and the deadline, d_1^e of the first sub-job j_1^e (Line 3). Then, we find the remaining time to j_1^e's deadline (Line 2). If the remaining time (t_r) is less than the start-up lost time (t_l) which represents the reaction time to the green light for the first vehicle in the queue, then t_r is not enough to dispatch even one vehicle, and hence, a cycle time (T_c) worth of time is added to t_r while subtracting one cycle from n_c^e (Line 5). This operation ensures that there is enough time for human driven vehicles to react to green lights and incoming ERVs without

making hasty traffic decision and thereby reinforcing safety. We then find the number of vehicles (n_r) that we can dispatch within t_r as well as the number of vehicles ($n_{r_{in}}$) that enter the arterial $A^D_{\alpha_e}$ at the worst-case (Line 6 and 7). This indicates that a queue of $q^e_1 = n_{r_{in}} - n_r$ in the first link can exist and still meet the first sub-job's deadline if we start ERV preemption at time n^e_c. The queue is also bounded by the link capacity to avoid spillbacks (Line 8).

Enabling green-wave coordination. : Since missing one of the sub-job's deadlines for the ERV task leads to a cascading effect for all sub-jobs (Property 1), when the sub-job j^e_i is being executed, the next link (corresponding to sub-job j^e_{i+1}) must be ready by dispatching all vehicles within, before j^e_i reaches its deadline and finishes execution. Since completion time for a sub-job j^e_i is equal to the arrival time for the next sub-job j^e_{i+1}, the queues in each link must be maintained (not exceeding the link capacity) such that the links can be cleared (all vehicles dispatched) by the time the sub-job for the previous link completes execution (Line 10 and 11). This ensures that by the time the previous sub-job reaches its deadline, the next sub-job is ready for execution.

Budget assignment for proactive preemption. Consider that the ERV preemption and traversal is requested in the arterial $A^D_{\alpha_e}$ and using Algorithm 3 we can determine the queues we need to maintain such that all sub-jobs of the ERV task meet their deadline, the ERV task hence meets its end-to-end deadline and green-wave coordination is also enabled to ensure minimum stop-and-go traffic for higher priority ERVs as well as the non-ERVs. Thus, the utilization bounds for the server S^e_i corresponding to the link L^e_i within the arterial $A^D_{\alpha_e}$ with queue estimated as q^e_i can be formulated as,

$$U^e_{min} = \frac{h \cdot q^e_i + t_l}{T_c}. \tag{14}$$

The ERV-aware traffic control with proactive preemption is formulated as follows.

$$\text{maximize(Equation 4)}$$

$$\text{subject to } U^D_{\alpha_e} \geq U^D_{min_{\alpha_e}} \tag{15}$$

$$U^{D'}_{\alpha_e} \leq U^{D'}_{max_{\alpha_e}} \tag{16}$$

$$\text{Equations (7), (12), (13).}$$

Unlike critical preemption, in proactive preemption constraints in Equations 15 and 16 enable dedicating some budget to the conflicting flows as long as the minimum utilization demands of the servers serving the ERV are satisfied, which ensures that the ERV deadlines are met for each sub-job. Constraints in Equation 12 and 13 ensure that the under-/over-utilization of resources for servers serving the links not affected by the ERV is avoided and the schedulability of the system (Equation 7) is maintained. Thus, by assigning sub-job specific deadlines for the ERV task based on priority and delay tolerance given by the triage scale, proactive preemption provides a more *planned* approach towards the ERV traversal by avoiding long queues and spillbacks before and during the ERV traversal. The preemption mechanisms (critical and proactive) stay in effect until the deadline of the corresponding sub-job has passed, i.e., until d^e_i.

In all, the critical preemption as well as the proactive preemption provide strategies to enable ERV traversal such that their deadlines are met and also control network-wide traffic to reduce queues and traffic delays. Since ERV preemption still remains a priority in both the preemption mechanisms discussed, there may be scenarios where long queues and potential spillbacks are unavoidable. Specifically, we show in Section 6, that when there are heavy traffic flows in the network, enabling ERV traversal may lead to potential spillbacks. For such cases, we extend the per-arterial *recovery approach* presented in [22] to stabilize the entire network.

Table 2: Flow types simulated as per the traffic flow rates and its description as per the FHWA [10]

Flow Type	Flow rate per lane (Net flow in 3 × 3 network)	Description
Low	1–4 veh/min (540–2100 veh/hr)	Network running under capacity with reduced travel delays
Medium	4–6 veh/min (2100–3200 veh/hr)	Network nearing capacity with longer queues
Heavy	up to 8 veh/min (up to 4300 veh/hr)	Unstable traffic flow with long wait times

5.4.3 Scheduling in recovery phase. A recovery approach addressing a single arterial experiencing spillbacks was presented in [22]. Due to the lack of ERV preemption in their approach, their recovery strategy fails to address cases when multiple arterials within the network have long queues and potential spillbacks. Since multiple conflicting flows experience prolonged red lights due to the preemption, especially when critical preemption is deployed, the recovery approach must be extended for multiple links and arterials. We therefore formulate a network-wide recovery approach as follows.

$$\text{maximize(Equation 4)}$$

$$\text{subject to } U^D_{\alpha_e} = U^D_{min_{\alpha_e}} \tag{17}$$

$$U^{D'}_{\alpha_e} = 1 - U^D_{min_{\alpha_e}} \tag{18}$$

$$\text{Equations (7), (12), (13).}$$

In the recovery phase, we assign minimum budget requirement to the servers that recently serviced the ERV (Equation 17) while maximizing traffic flow for the conflicting servers (Equation 18). Along with the constraint in Equation 13, the traffic through all arterials conflicting the ERV flow will be maximized (Equation 4) since assigned budget at U^D_α depends on $U^D_{\alpha-1}$. The recovery phase is replaced with the default traffic strategy after one traffic cycle.

5.5 Worst-Case Performance Analysis for ERV-Aware Traffic Control

We now provide a worst-case analysis to predict the impact of the ERV traversal on the non-ERV flow which can help the traffic routing systems to predict existing delays in the network. Worst-case traffic delay accounts for the total time that a vehicle has to stop while traversing through the network. Theorem 1 [22], presents a worst-case traffic delay analysis without ERV traversal.

Theorem 1 ([22]). *Assuming that the traffic flow rate for vehicles entering the arterial $A^D_{\alpha_r}$, traveling through all links L_i is a_i, the cumulative wait time $W^D_{\alpha_r,p}$ for a vehicle at the p^{th} position in the queue upon entering the arterial in the k^{th} cycle is bounded by*

$$W^D_{\alpha_r,p} \in \left[0, \sum_{k=1}^{\left\lfloor \frac{p}{\overline{n}_{out_{1_k}}} \right\rfloor} \left(\lambda T_c - \lambda U_{min_{1,k}} T_c + \sum_{i=2}^{\lambda} (\lambda - i + 1) z_i h \right)\right] \tag{19}$$

Table 3: ERV travel times in a 3×3 road network

Preemption Strategy	ERV Travel Times (s)		
	Low	Medium	Heavy
No preemption	78	89	100
Localized	63	71	96
Green wave-based	60	60	60
Proactive preemption	60	60	60

where $\lambda = n$, if $D = \{S, N\}$, and $\lambda = m$, otherwise. $n_{out_{1_k}}$ is the minimum number of vehicles dispatched from link L_1 of arterial $A^D_{\alpha_r}$ in the k^{th} cycle, $U_{min_{1,k}}$ denotes the minimum utilization demand for link L_1 in the arterial, in the k^{th} cycle, and all other variables are as defined previously.

We now present the cumulative worst-case non-ERV traffic delay bounds caused by ERV traversal in Theorem 2.

Theorem 2. *Assuming that an ERV traverses through the arterial $A^D_{\alpha_e}$ which conflicts the traffic flow within the arterial $A^D_{\alpha_{e'}}$ consisting of links L_i. Then the cumulative wait time $W^{e'}_{\alpha_{e'},p}$ for a vehicle at the p^{th} position in the queue upon entering the arterial in the k^{th} cycle is bounded by*

$$W^D_{\alpha_{e'},p} \in \left[0, W^D_{\lambda_i} + W^D_{\lambda_e} + W^D_{preempt} + W^D_{\lambda}\right] \tag{20}$$

where $\lambda = n$, if $D = \{S, N\}$, and $\lambda = m$, otherwise, and,

$$W^D_{\lambda_i} = \sum_{k=1}^{\left\lfloor \frac{p}{n_{out_{1_k}}} \right\rfloor} \left(\lambda_i T_c - \lambda_i U_{min_{1,k}} T_c + \sum_{i=2}^{\lambda_i} (\lambda_i - i + 1) z_i h \right),$$

$$W^D_{\lambda_e} = \sum_{k=1}^{\left\lfloor \frac{p}{n_{out_{1_k}}} \right\rfloor} \left(d^e_{e'-1} - t^e_{e'-1} - t_l \right),$$

$$W^D_{preempt} = d^e_{e'} - t^e_{e'} \text{ and}$$

$$W^D_{\lambda} = \sum_{k=1}^{\left\lfloor \frac{p}{n_{out_{1_k}}} \right\rfloor} \left((\lambda - e) T_c - (\lambda - e) U_{min_{1,k}} T_c + \sum_{i=e}^{\lambda} (\lambda - i + 1) z_i h \right).$$

Proof:

The arterials $A^D_{\alpha_e}$ and $A^D_{\alpha_{e'}}$ will meet at the intersection $I_{ee'}$. Consider that the ERV preemption is requested at $t_{dispatch}$, when the p^{th} vehicle has reached the intersection $I_{\lambda_i e'}$. Until the vehicle reaches the intersection $I_{\lambda_i e'}$, its cumulative worst-case delay is given by Theorem 1 [22] as,

$$W^D_{\lambda_i} = \sum_{k=1}^{\left\lfloor \frac{p}{n_{out_{1_k}}} \right\rfloor} \left(\lambda_i T_c - \lambda_i U_{min_{1,k}} T_c + \sum_{i=2}^{\lambda_i} (\lambda_i - i + 1) z_i h \right), \tag{21}$$

where, $\lambda_i \in [1, e]$ and other notations are as before.

Let the local deadline set by the TM task for the ERV to reach the intersection $I_{ee'}$ be given as, $t^e_{e'}$. The vehicles traveling across the arterial $A^D_{\alpha_{e'}}$ can be described by two situations:

Case 1: The non-ERVs in $A^D_{\alpha_{e'}}$ successfully cross the intersection $I_{ee'}$ before $t^e_{e'}$ (through proactive preemption). The budget, $U^e_{e'}$ assigned to the server serving the link $L_{e'}$ in the arterial $A^D_{\alpha_e}$ is given by the Equation 14. Therefore, the budget assigned to the server serving the link L_e in the arterial $A^D_{\alpha_{e'}}$ is given as $U^{e'}_e = 1 - U^e_{e'}$.

Table 4: Effect of priority-based deadlines on overall traffic flow. c^e_i and d^e_i are the completion times and the deadlines for the sub-job j^e_i of the ERV task arriving at 1100 s.

Priority (π^e)	c^e_1 (s)	d^e_1 (s)	c^e (s)	d^e_2 (s)	c^e_3 (s)	$d^e_3 = d^e$ (s)	Average Traffic Delay (s)
1	1255	1255	1285	1285	1315	1315	39.97
2	1255	1274	1288	1296	1320	1330	36.75
3	1275	1285	1315	1315	1345	1345	35.45
4	1282	282	1315	1324	1350	1360	31.48
5	1307	1320	1345	1352	1383	1385	30.72

Therefore, the worst-case delay for the vehicle at the p^{th} position is calculated from the Algorithm 3 and the Equation 14 as,

$$W^D_{\lambda_e} = \sum_{k=1}^{\left\lfloor \frac{p}{n_{out_{1_k}}} \right\rfloor} \left(U^e_{e'} T_c \right) \Rightarrow W^D_{\lambda_e} = \sum_{k=1}^{\left\lfloor \frac{p}{n_{out_{1_k}}} \right\rfloor} \left(d^e_{e'-1} - t^e_{e'-1} - t_l \right) \tag{22}$$

After crossing the intersection $I_{ee'}$, the vehicle will not be affected by the ERV and the wait-time is given as (Theorem 1),

$$W^D_{\lambda} = \sum_{k=1}^{\left\lfloor \frac{p}{n_{out_{1_k}}} \right\rfloor} \left((\lambda - e) T_c - (\lambda - e) U_{min_{1,k}} T_c + \sum_{i=e}^{\lambda} (\lambda - i + 1) z_i h \right), \tag{23}$$

The worst-case delay under proactive preemption is given by,

$$W^D_{\alpha_{e'},p} = W^D_{\lambda_i} + W^D_{\lambda_e} + W^D_{\lambda}.$$

The best-case occurs when the vehicles do not experience any delays with green lights at each intersection before the ERV arrives. The cumulative worst-case can thus be bounded by,

$$W^D_{\alpha_{e'},p} \in [0, W^D_{\lambda_i} + W^D_{\lambda_e} + W^D_{\lambda}]. \tag{24}$$

Case 2: The non-ERVs in $A^D_{\alpha_{e'}}$ enter intersection $I_{ee'}$ at $t^e_{e'}$ (critical preemption). The maximum delay occurs if the p^{th} vehicle is unable to cross the intersection $I_{ee'}$ before the ERV arrives at $t^e_{e'}$. To prioritize the ERV, the conflicting flows receive no budget until $d^e_{e'}$, i.e., the deadline for the sub-job that corresponds to the ERV's traversal through the intersection $I_{ee'}$ with the wait-time given by,

$$W^D_{preempt} = d^e_{e'} - t^e_{e'}, \tag{25}$$

which adds to the wait-time determined in case 1. The cumulative worst-case delay for this case is thus bounded by,

$$W^D_{\alpha_{e'},p} \in [0, W^D_{\lambda_i} + W^D_{\lambda_e} + W^D_{preempt} + W^D_{\lambda}]. \tag{26}$$

6 SIMULATIONS AND HARDWARE VALIDATION

We now present the large-scale traffic simulations and hardware-in-loop (HIL) validation of our proposed approach.

6.1 Simulation Setup

Using a tick-based traffic simulator developed in Python 3, we simulate 1) vehicle flows varying from 60–600 veh/hr resembling the critical volume-to-capacity ratios for signalized intersections [10] as shown in Table 2, 2) desired 3×3 road network architecture (Figure 1), and 3) different traffic algorithms, for one hour. An ERV is introduced at around 1000 s. The length of each link and ERV's desired speed are set to 400 m and 45 mph respectively.

ERV travel times with different approaches: Table 3 shows that our approach provides 15–37% faster ERV travel in medium and

Table 5: Network-wide traffic delays under different ERV preemption approaches *(shaded cells indicate spillbacks)*

ERV Arrival Information (s)	Traffic Flow	Proposed Approach						Green Wave-based		
		Proactive Preemption + Recovery			Proactive Preemption Only					
		Delay in Conflicting Flows (s)		Average Network-wide Delays (s)	Delay in Conflicting Flows (s)		Average Network-wide Delays (s)	Delay in Conflicting Flows (s)		Average Network-wide Delays (s)
		Average	Worst-Case		Average	Worst-Case		Average	Worst-Case	
60	Low	68.65	77.12	37.56	70.54	75.53	38.23	89.06	90.12	39.56
	Medium	71.23	81.12	40.12	76.34	82.12	42.12	104.21	119.31	42.42
	Heavy	88.64	92.12	46.42	93.23	103.12	47.21	100.21	112.21	49.21
120	Low	48.12	56.23	39.48	49.12	57.64	39.76	110.42	111.66	45.46
	Medium	58.56	69.24	38.86	71.77	87.63	40.93	171.36	186.66	55.42
	Heavy	69.65	89.66	39.24	74.86	90.21	49.98	134.21	135.43	53.67
180	Low	47.83	56.11	35.85	59.06	68.92	38.42	93.23	103.10	44.88
	Medium	42.07	53.12	35.29	54.48	65.61	38.26	107.85	113.44	44.64
	Heavy	52.16	64.76	34.59	61.41	67.30	38.22	167.83	169.62	61.24

heavy traffic, when compared to the localized preemption [4]. Our approach also shows 40% faster ERV travel time when compared to traffic control without ERV preemption [22]. Henceforth, since localized preemption causes ERV travel delays, we only compare our approach with the greedy green wave preemption [34] that always provides best-case ERV travel times. *We never see any difference in ERV travel times with our approach and green wave approach.*

Effect of preemption and recovery on overall traffic delays (Table 5): When the ERV arrival information is received at 60, 120 and 180 s before its arrival, the proactive preemption with recovery shows 17.9%, 33.79% and 61.82% reduction in worst-case traffic delays in conflicting flows, respectively, when compared to the green wave approach. Therefore, the earlier the ERV information is received, the better *proactive* decisions our approach makes. Further, having a recovery phase after preemption improves the performance by up to 21.48%, highlighting the need for an end-to-end preemption. For all scenarios shown in Table 5, the ERV requests highest priority and traverses the network without any deadline misses. Our approach switches to *critical preemption* when the ERV information is received < 60 s of its arrival, and we still see 8–26% improvement in traffic delays while avoiding spillbacks when compared to the green wave-based approach.

Triage levels and its impact on network-wide traffic performance: Table 4 shows the completion time and the deadline for the sub-jobs of the ERV task with the delay tolerance shown in Table 1 for each priority level. Using priority information in traffic planning and preemption could help achieve up to 23% improvement in network-wide traffic delays between the highest and the lowest priority ERV with 100% deadlines met for all priority levels.

6.2 Hardware Validation

We also implement our proposed approach on a hardware-in-loop (HIL) testbed consisting of 30 small robots each representing a human-driven vehicle (Figure 7). Our robots, affixed with infrared (IR) markers, are tracked using the Optitrack motion capture system. The position data is streamed to a Robot Operating System-based (ROS) [26] framework that implements the traffic manager (TM) to control the traffic light timings and vehicle flow. Two intersections of the 3×3 network are physically mapped in the testbed, while the rest of the network is simulated in software, due to limited physical space. We use the intelligent driver model (IDM) [32] to mimic human driving in urban areas. The desired velocities are communicated to the robots using Zigbee communication. One of the vehicles in the network is labelled as an ERV which sends its information to the TM on the ROS framework. The road architecture and vehicle dynamics are scaled to the size of the robot. Interestingly, while our simulations show no difference in ERV traversal times between our approach and the green wave approach, our HIL experiment (Figure 6), that considers non-ideal human reactions, shows our approach outperforms green wave preemption by ensuring non-stop ERV traversal at the desired speed of 0.3 m/s.

7 CONCLUSION AND FUTURE WORK

In this paper, we represented ERV preemption and traffic flow through a road network as a real-time scheduling problem. Using a deadline-driven approach, our proposed preemption strategy provided guaranteed timely ERV response while also minimizing traffic delays. Our approach shows up to 43% reduction in traffic delays without adding any delay to the ERV travel, by proactively controlling the traffic before, during and after the ERV traversal, through the entire network. By leveraging the real-time properties of our model, we provided worst-case wait time bounds for non-emergency vehicles during ERV preemption which could further help in planning predictable routes for the ERV dispatch systems. Hardware-in-loop (HIL) experiments showed that with our proposed approach, the ERV travels through the network without deviating from its desired speed, thus adapting to urban driving environment. Managing multiple ERV traversals through the road network is left for future work.

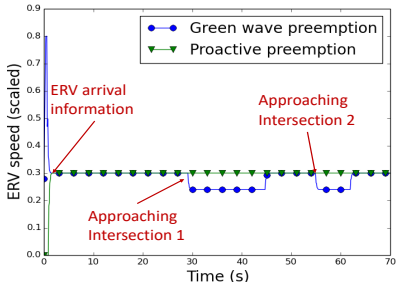

Figure 6: ERV speeds with various preemption approaches

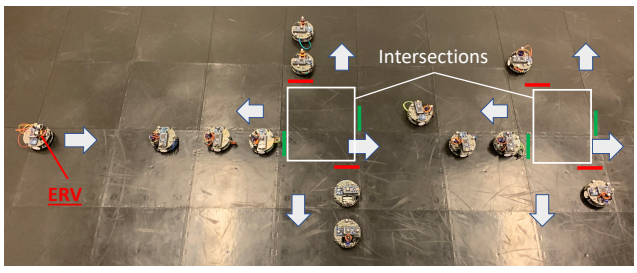

Figure 7: HIL framework with two intersections and an ERV

REFERENCES

[1] National Emergency Number Association. 2020. *9-1-1 Statistics*. Retrieved November 27, 2020 from https://www.nena.org/page/911Statistics

[2] Daniel Brent and Louis-Philippe Beland. 2020. Traffic congestion, transportation policies, and the performance of first responders. *Journal of Environmental Economics and Management* (2020).

[3] Wolf-Rüdiger Bretzke. 2013. Global urbanization: a major challenge for logistics. *Logistics Research* (2013).

[4] Miaomiao Cao, Qiqi Shuai, and Victor OK Li. 2019. Emergency Vehicle-Centered Traffic Signal Control in Intelligent Transportation Systems. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*. IEEE, 4525–4531.

[5] National Safety Council. 2018. *Emergency Vehicles Crash Statistics*. Retrieved November 27, 2020 from https://injuryfacts.nsc.org/motor-vehicle/road-users/emergency-vehicles/

[6] Jackson D Déziel. 2019. Ambulance transport to the emergency department: A patient-selected signal of acuity and its effect on resource provision. *The American journal of emergency medicine* 37, 6 (2019), 1096–1100.

[7] Dario Faggioli, Marko Bertogna, and Fabio Checconi. 2010. Sporadic server revisited. In *Proceedings of the 2010 ACM Symposium on Applied Computing*. ACM, 340–345.

[8] Nasim Farrohknia, Maaret Castrén, Anna Ehrenberg, Lars Lind, Sven Oredsson, Håkan Jonsson, Kjell Asplund, and Katarina E Göransson. 2011. Emergency department triage scales and their components: a systematic review of the scientific evidence. *Scandinavian journal of trauma, resuscitation and emergency medicine* 19, 1 (2011), 42.

[9] Yiheng Feng, K Larry Head, Shayan Khoshmagham, and Mehdi Zamanipour. 2015. A real-time adaptive signal control in a connected vehicle environment. *Transportation Research Part C: Emerging Technologies* 55 (2015), 460–473.

[10] FHWA-USDoT. 2019. Signalized intersections: Informational guide. https://www.fhwa.dot.gov/publications/research/safety/04091/07.cfm

[11] HCM. 2010. Highway capacity manual, 2010. *Transportation Research Board, National Research Council, Washington, DC* (2010).

[12] Hongwei Hsiao, Joonho Chang, and Peter Simeonov. 2018. Preventing emergency vehicle crashes: status and challenges of human factors issues. *Human factors* 60, 7 (2018), 1048–1072.

[13] Subash Humagain and Roopak Sinha. 2020. Dynamic Prioritization of Emergency Vehicles For Self-Organizing Traffic using VTL+ EV. In *IECON 2020 The 46th Annual Conference of the IEEE Industrial Electronics Society*. IEEE, 789–794.

[14] Anupam B Jena, N Clay Mann, Leia N Wedlund, and Andrew Olenski. 2017. Delays in emergency care and mortality during major US marathons. *New England Journal of Medicine* (2017).

[15] Wenwen Kang, Gang Xiong, Yisheng Lv, Xisong Dong, Fenghua Zhu, and Qingjie Kong. 2014. Traffic signal coordination for emergency vehicles. In *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*. IEEE, 157–161.

[16] Borna Kapusta, Mladen MiletiC, Edouard Ivanjko, and Miroslav Vujić. 2017. Preemptive traffic light control based on vehicle tracking and queue lengths. In *2017 International Symposium ELMAR*. IEEE, 11–16.

[17] Yeong-Lin Lai, Yung-Hua Chou, and Li-Chih Chang. 2018. An intelligent IoT emergency vehicle warning system using RFID and Wi-Fi technologies for emergency medical services. *Technology and health care* 26, 1 (2018), 43–55.

[18] M. Masoud and S. Belkasim. 2018. WSN-EVP: A Novel Special Purpose Protocol for Emergency Vehicle Preemption Systems. *IEEE Transactions on Vehicular Technology* 67, 4 (2018), 3695–3700. https://doi.org/10.1109/TVT.2017.2784568

[19] Zhaolong Ning, Jun Huang, and Xiaojie Wang. 2019. Vehicular fog computing: Enabling real-time traffic management for smart cities. *IEEE Wireless Communications* 26, 1 (2019), 87–93.

[20] Hamed Noori, Liping Fu, and Sajad Shiravi. 2016. A connected vehicle based traffic signal control strategy for emergency vehicle preemption. In *Transportation Research Board 95th Annual Meeting*.

[21] P. Oza and T. Chantem. 2019. A Real-Time Server Based Approach for Safe and Timely Intersection Crossings. In *2019 IEEE 25th International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA)*. 1–12. https://doi.org/10.1109/RTCSA.2019.8864584

[22] P. Oza, T. Chantem, and P. Murray-Tuite. 2020. A Coordinated Spillback-Aware Traffic Optimization and Recovery at Multiple Intersections. In *2020 IEEE 26th International Conference on Embedded and Real-Time Computing Systems and Applications (RTCSA)*. 1–10. https://doi.org/10.1109/RTCSA50079.2020.9203582

[23] Prahlad D Pant, Yizong Cheng, Arudi Rajagopal, Nagaraju Kashayi, et al. 2005. Field testing and implementation of dilemma zone protection and signal coordination at closely-spaced high-speed intersections. *Rep. No. FHWA/OH-2005/006, Ohio Dept. of Transportation, Columbus, OH* (2005).

[24] V. Paruchuri. 2017. Adaptive Preemption of Traffic for Emergency Vehicles. In *2017 UKSim-AMSS 19th International Conference on Computer Modelling Simulation (UKSim)*. 45–49. https://doi.org/10.1109/UKSim.2017.34

[25] Jin Qin, Yong Ye, Bi-rong Cheng, Xiaobo Zhao, and Linling Ni. 2017. The emergency vehicle routing problem with uncertain demand under sustainability environments. *Sustainability* 9, 2 (2017), 288.

[26] Morgan Quigley et al. 2009. ROS: an open-source robot operating system. In *ICRA workshop on open source software*. Kobe, Japan.

[27] Chang Qiao Shao and Xiao Ming Liu. 2012. Estimation of saturation flow rates at signalized intersections. *Discrete Dynamics in Nature and Society* (2012).

[28] Aleksandar Stevanovic, Cameron Kergaye, and Peter T Martin. 2009. Scoot and scats: A closer look into their operations. In *88th Annual Meeting of the Transportation Research Board. Washington DC*.

[29] Virginia Legislative Information System. 2020. *Code of Virginia: Emergency Vehicles*. Retrieved November 27, 2020 from https://law.lis.virginia.gov/vacode/title46.2/chapter8/section46.2-829/

[30] Fred Szabo. 2015. *The linear algebra survival guide: illustrated with Mathematica*. Academic Press.

[31] TomTom. 2019. TomTom Traffic Index. https://www.tomtom.com/en_gb/traffic-index/

[32] Martin Treiber, Ansgar Hennecke, and Dirk Helbing. 2000. Congested traffic states in empirical observations and microscopic simulations. *Physical review E* 62, 2 (2000), 1805.

[33] Jiaming Wu, Balázs Kulcsár, Soyoung Ahn, and Xiaobo Qu. 2020. Emergency vehicle lane pre-clearing: From microscopic cooperation to routing decision making. *Transportation Research Part B: Methodological* 141 (2020), 223–239.

[34] Jiao Yao, Kaimin Zhang, Yuanyuan Yang, and Jin Wang. 2018. Emergency vehicle route oriented signal coordinated control model with two-level programming. *Soft Computing* 22, 13 (2018), 4283–4294.