Real-time Traffic Congestion Management and Deadlock Avoidance for Vehicular Ad Hoc Networks

Syed Rafiul Hussain, Ala Odeh, Amrut Shivakumar, Shalini Chauhan, and Khaled Harfoush
Department of Computer Science
North Carolina State University
Email: {shussai5, aaodeh, ashivak, schauha, kaharfou}@ncsu.edu

***Abstract*—Traffic congestion is common in heavily populated cities. In this paper, we introduce a novel protocol to guide vehicles to their destinations while managing congestion and avoiding deadlock situations in urban city grids. The protocol relies on vehicle-to-infrastructure (V2I) communication, infrastructure-to-infrastructure (I2I) communication as well as GPS information to disseminate congestion information to city grid intersections. Our proposal pro-actively aims at avoiding congestion scenarios and reacts to arising congestion in case it happens due to unpredictable events such as collisions. Simulation results reveal that vehicles enjoy reduced travel times even during rush hours and in the presence of collisions.**

***Index Terms*— Congestion Detection, Congestion Avoidance, Traffic Route Guidance.**

I. Introduction

In today's world of unprecedented increase in population and urbanization, traffic congestion has become an inevitably relevant problem [1]. Traffic congestion occurs during rush hour times, due to accidents, road work, or even when the road shifts into fewer lanes. Saving on commute time resulting from traffic congestion cannot be achieved unless real-time congestion information of nearby localities is disseminated to the drivers. The best solution to this problem is Cooperative Traffic Information Systems (CTIS) [2], [3], where each vehicle individually collects traffic information and exchanges it with other users using wireless networks. This provides real-time traffic information which in conjunction with GPS can provide dynamic route guidance. In general, CTIS are either *infrastructure-less* or *infrastructure-based*. The former is based on vehicle-to-vehicle (V2V) communication and follow a peer-to-peer (P2P) model, while the latter mainly relies on on vehicle-to-infrastructure (V2I) communication, and follow a client-server model.

The infrastructure-less model, however, (1) requires relatively high processing capacity at vehicles, (2) aggregation of congestion information at vehicles can lead to inconsistent information leading to false positives [4], [5]. Also, (3) congestion information of areas that are relatively distant are typically not precise. Moreover, (4) distributed route selection decisions can further exacerbate congestion. SOTIS [6] and TrafficView [7] are examples of infrastructure-less systems. On the other hand, the infrastructure-based model (1) may not scale since the server would have to deal with with a large number of updates and queries [8], and (2) centralized servers can violate users' privacy. TraffCon [9] and PeerTIS [10] are examples of infrastructure-based systems.

In order to resolve these problems, we propose a new protocol, COMAD, used for COngestion MAnagement and Deadlock avoidance. In COMAD, infrastructure is installed at traffic lights [11] and used to collect congestion information from nearby vehicles, process congestion information, exchange it with neighboring infrastructure elements, and guide vehicles to their intended destinations through least congested routes. Thus vehicle-to-infrastructure (V2I) communication is used to report vehicle presence, query the infrastructure and attain route information; and infrastructure-to-infrastructure (I2I) communication is used to exchange congestion information. As opposed to pure client-server models, servers at stop lights have limited view of the network and does not track a vehicle throughout the city grid, thus privacy is maintained. Also, as opposed to pure peer-to-peer models, stop lights – instead of vehicles – peer with neighboring stop lights to exchange congestion information, thus high processing capacity at vehicles is not an issue and maintained information at the infrastructure is consistent, and covers a relatively large area of the grid and the system scales. The consistency of the maintained information allows COMAD to effectively identify congested parts of the network and guide vehicles around bottlenecks without inducing secondary bottlenecks or deadlocks. Furthermore, COMAD reacts to arising congestion in case it happens due to unpredictable events such as collisions. Simulation results reveal that vehicles enjoy reduced travel times even during rush hours and in the presence of collisions.

The rest of the paper is organized as follows. In Section II we introduce the idea and the details of COMAD. In Section III we discuss the simulation results. We Finally conclude in Section V.

II. COMAD Details

A. Setup

We assume a city grid in which traffic lights at road intersections are equipped with wireless transceivers having enough computing power and storage capability to process

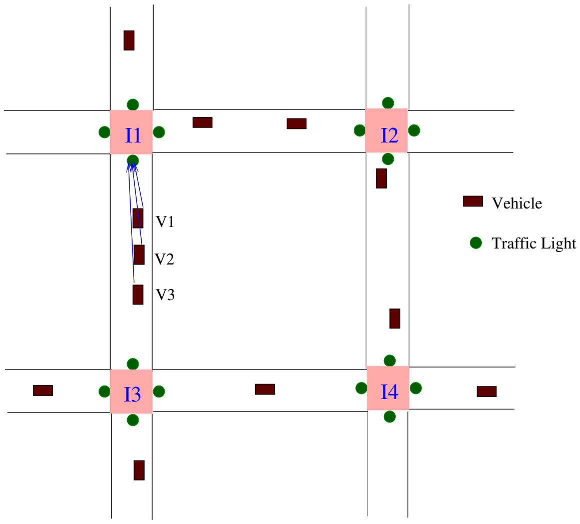


Fig. 1. Collecting traffic information by stop lights.

real-time traffic data. Vehicles also are equipped with wireless transceivers. Vehicles communicate with traffic lights (V2I) and traffic lights communicate with each other (I2I) through DSRC (Dedicated short-range communications), which operates within 75 MHz of spectrum in the 5.9 GHz band and has the maximum transmission range of 1 KM. We thus assume that traffic lights are within 1 KM distance from each other. Traffic lights that are not direct communication rage can still exchange information through intermediate traffic lights acting as relays.

B. Traffic Information Collection and Dissemination

Vehicles broadcast ARRIVAL messages periodically to signal their presence to traffic lights within communication range – Refer to Figure 1 . Traffic lights collect and process ARRIVAL messages and maintain information about the number of vehicles currently waiting for service by this light signal. Basically, a traffic light is acting as a first-in-first-out M/G/1 [12], [13] queue serving vehicles. The mean queue length and mean waiting time in the queue is given by the Pollaczek-Khinchine formula [14]. The mean queue length is used to identify whether the number of vehicles in a road segment is approaching the capacity of this segment, and the mean waiting time is useful, when combined with the mean waiting time in other segments, in computing the most efficient path to reach a destination.

To disseminate the traffic information (mean queue length and mean waiting time in a road segment) in an UPDATE message throughout the network, we rely on an efficient flooding mechanism. The initiator of the UPDATE message (a stop light infrastructure) sends the message to all its neighboring stop lights. Nodes receiving this UPDATE message, which are in the same grid *row* as the initiator, forward the message towards all neighbors except the one from which the message was received. Nodes receiving this UPDATE message, which are in the same grid *colum* as the initiator, forward the message only

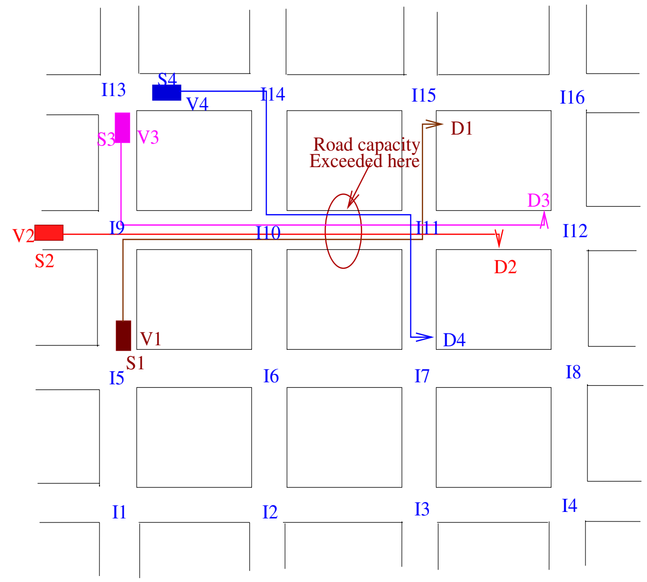


Fig. 2. Congestion Avoidance in COMAD.

towards neighbors on their same grid column except the the one from which the message was received. In this way, when an UPDATE message is flooded, each stop light intercepts its only once. The flooding process is repeated periodically. Collecting information periodically about the state of all road segments allows each traffic light to construct and update a global cost matrix, COST-MATRIX, that can be used to guide vehicles to their intended destinations as follows.

C. Congestion Avoidance and Dissolution

A traffic light can identify the least cost path to any destination by applying Dijkstra's [15] shortest path algorithm to the COST-MATRIX. However, this approach is naive and has shortcomings. Consider a scenario in which vehicles at different locations in the grid requesting guidance are independently suggested to take same road segment at the same time. There is a good chance that all these vehicles will reach that particular road segment at the same time and thereby experience congestion. In such a case, the affected vehicles will look out for alternate best routes to their respective destinations by consulting the common road infrastructure. Taking measures after congestion has already been caused, increases commute time and also causes protocol overhead in finding alternate routes. Still it is useful to have a way to react to congestion as it happens since there are case in which congestion is hard to avoid.

Furthermore, a reservation protocol can be used to *proactively* avoid congestion, ensuring that the capacity of a road segment is not exceeded as vehicles traverse it. By consulting with a traffic light, a vehicle identifies its best path to its destination based on the current COST-MATRIX information. The vehicle then tries to reserve the road segments that belong to this computed path before it physically reaches the segments. Using this reservation strategy, each road segment keeps track of the vehicles that are expected to come within a period of time and checks whether the expected number of vehicles exceed the road's capacity and deny reservation requests if congestion is expected to happen. If any traffic light denies a reservation request along a road segment of the path, the whole path reservation is denied, in a way similar to the popular RSVP protocol. Reservations are maintained as *Soft* state in traffic lights; i.e. reservations expire if not refreshed, which makes it possible for subsequent reservations from other vehicles to be accepted.

Figure 2 depicts a city grid example to illustrate how congestion is avoided if at all possible and how to react to congestion in case it happens. Intersections are enumerated from I_1 to I_{16}. Assume that each road segment has a capacity of three vehicles at any point in time. Consider a case in which 4 vehicles are traveling from sources S_1, S_2, S_3, and S_4 to D_1, D_2, D_3, and D_4, respectively. Each vehicle, consulting its current road segment traffic light, identifies a *best* path to its destination. Now assume that the four best paths include the road segment $I_{10} - I_{11}$. In this case, congestion is likely to happen at this segment. However, relying on path reservations, this congestion can be avoided as path reservation requests will be granted on first-come-first-serve basis to at most three vehicles to avoid causing congestion at road segment $I_{10} - I_{11}$. Reservation requests for the remaining vehicles will be denied and the COST-MATRIX will reflect the fact that road segment $I_{10} - I_{11}$ is reserved causing new suggestions for best paths from traffic lights to reflect different uncongested paths.

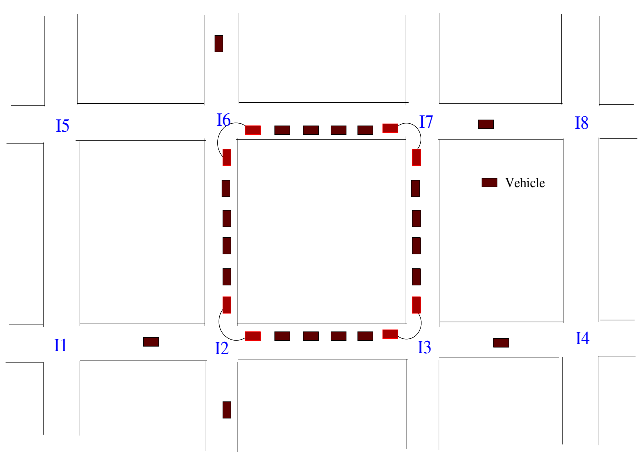


Fig. 3. Deadlock Avoidance Mechanism

D. Deadlock Avoidance

In general, deadlocks can arise from circular dependencies – Figure 3 – in which vehicles are waiting for each other to move forward. COMAD is not susceptible to this situation whether due to the advance reservation mechanism, which will only reserve a road segment for a vehicle if it is not congested, or due to the fact that stop lights are aware of congestion in the grid and will suggest uncongested paths to vehicles. To illustrate, let us consider the scenario depicted in Figure 3. Assume that the road segment $I_6 - I_7$ has already granted the maximum number of reservations for vehicles that it can accommodate. Hence, this road segment will not be suggested by stop lights to other vehicles. As a result, vehicles on road segment $I_2 - I_7$, $I_3 - I_2$ and others will be suggested to take alternate road segments than $I_6 - I_7$ thus alleviating circular dependencies. Even in the presence of a deadlock due to an unexpected situation traffic scenario, traffic lights in the vicinity of the deadlock will propose alternate least congested routes to vehicles, which do not create circular dependency.

III. Performance Evaluation

A. Simulation Setup

We next evaluate through simulations the performance of COMAD and compare it to the traditional shortest path routing protocol (TSPRP). To this end, we build a simulator with ideal MAC layer using Visual Studio 2010. The simulator generates exponentially distributed random traffic to simulate traffic scenarios in a city grid. Each vehicle has a pre-defined destination to which the simulator identifies best routes. We run the simulation multiple times, with each run consuming 180 seconds, and report average values for interesting metrics. Table I lists the simulation parameters.

TABLE I
Simulation Parameters.

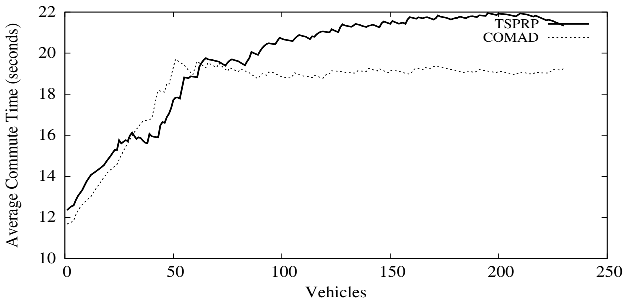
Parameters	Values
Size of the network	$6KM * 6KM$
Size of city grid network	6 x 6
Total number of intersections	36
Length of a road segment	$1KM$
Duration of simulation	180 seconds
Time required by a vehicle to travel a road segment under no congestion	3 seconds
Capacity of a road segment	10 cars
Communication/transmission Range	DSRC antenna /$1KM$

B. Performance Metrics

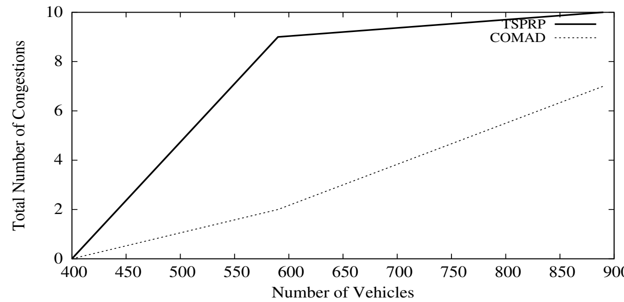
We analyze the performance of our proposed protocol using the following performance metrics:
(a) *Average commute time* is the average time required for a vehicle to reach its destination from its original location.
(b) *Total number of congestions* refers to the total number of congested segments in the grid.
(c) *Road Utilization* refers to percentage of a road segment's capacity that is occupied by vehicles.

C. Simulation Results

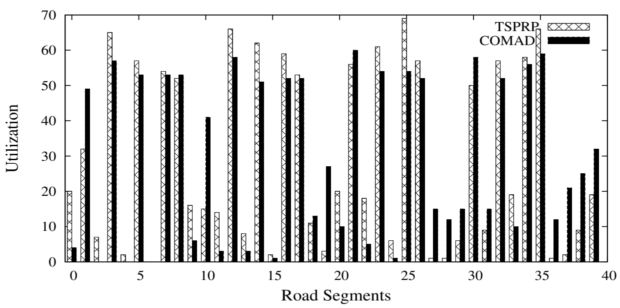
Vehicles are instantiated at random locations with a rate of 5 vehicles/second. Figure 4(a) compares the average commute time consumed by vehicles for TSPRP and the COMAD protocols. The figure demonstrates that commute times for both protocols are close to each other when the number of vehicles in the network is limited. The probability of a congestion in this case is small. However, as the number of vehicles increases, commute time when using COMAD becomes smaller than when using TSPRP. Since the COMAD



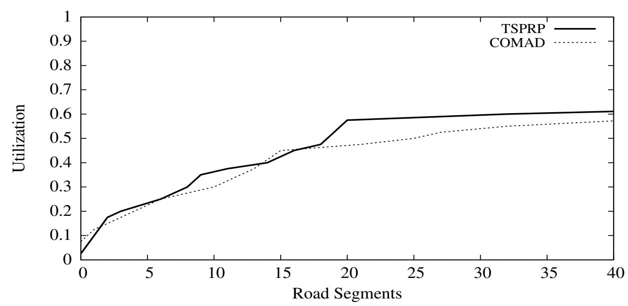
(a) Average Commute Time of Vehicles



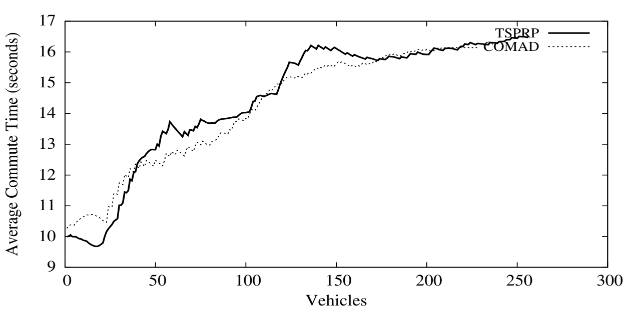
(b) Average Number of Congested Roads



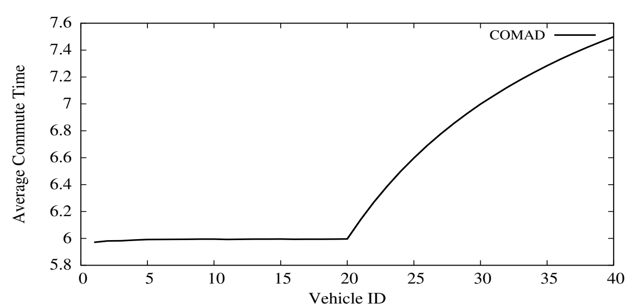
(c) Road Utilization



(d) Cumulative Distribution Function (CDF) of Road Utilization



(e) Average Commute Time of vehicle under Unexpected Traffic Event



(f) Average Commute Time of vehicle under Deadlock Situation

Fig. 4. Simulation results comparing COMAD to TSPRP.

estimates and avoids road congestion, it certainly outperforms the TSPRP for networks with a large number of vehicles.

Figure 4(b) depicts the total number of congested roads among the 60 road segments of the 6x6 city grid. As TSPRP chooses shortest path randomly from all the available shortest paths and does not care about congestion on other road segments while choosing a path to destination, therefore, TSPRP exhibits a higher number of congested segments than COMAD. While the number of congestions increases for both protocols with the increase in the number of vehicles, the number of congestions for COMAD is still much less than in the TSPRP case.

Figure 4(c) and Figure 4(d) show the total and cumulative road utilization by all the vehicles in the city grid, respectively. The figures highlight the fact that some under-utilized road segments for TSPRP are moderately utilized for COMAD. This clearly shows that COMAD protocol is distributing the traffic load across the network, which basically avoids road-bottlenecks.

Figure 4(e) shows the average commute times for vehicles when random road blockage occurs. While it may seem that TSPRP outperforms the COMAD, this is not the case. COMAD has a greater number of vehicles served which in-turn causes the average commute time to increase slightly. The relatively low commute time for TSPRP is due to that fact that fully congested roads prevent more vehicles from being considered in the average commute calculation. Thus, the COMAD outperforms TSPRP when road blockages occur since more vehicles make it to their destinations.

Figure 4(f) depicts deadlock scenario. 40 vehicles are forced into a deadlock situation identical to the one in Figure 3. Using COMAD, all vehicles emerge from the deadlock. The first few cars emerge from the deadlock rather quickly. This is shown by the low average commute time prior to the 20 vehicle mark. Slowly, the rest of the vehicles follow suit, which in turn causes the spike seen in the figure after the 20 vehicle mark.

IV. Conclusions and Future Work

The proposed COMAD is clearly capable of reducing congestion levels in city grids, leading to better commute times for vehicles, while avoiding deadlock situations. Further research is needed to identify optimal protocol parameters such as the frequency of flooding UPDATE messages and the frequency of refreshing soft state during path reservations. Moreover, we would like to explore the feasibility of integrating this technology with other interactive applications such as helping people choose alternate destinations for shopping malls, restaurants, or places like post offices. Implementing and testing COMAD in a testbed or even in a city grid is part of our future endeavors.

References

[1] D. Lewis, "America's traffic congestion problem: Toward a framework for national reform," *Discussion Paper, The Hamilton Project, Advancing Opportunity, Prosperity and Growth*, 2008.

[2] M. Seredynski and P. Bouvry, "A survey of vehicular-based cooperative traffic information systems," in *The 14th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, 2011, pp. 163–168.

[3] R. Von Tomkewitsch, "Dynamic route guidance and interactive transport management with ali-scout," *IEEE Transactions on Vehicular Technology*, vol. 40, no. 1, pp. 45–50, 1991.

[4] L. Wischhof, A. Ebner, and H. Rohling, "Information dissemination in self-organizing intervehicle networks," *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 1, pp. 90–101, 2005.

[5] F. Li and Y. Wang, "Routing in vehicular ad hoc networks: A survey," *Vehicular Technology Magazine, IEEE*, vol. 2, no. 2, pp. 12–22, 2007.

[6] L. Wischoff, A. Ebner, H. Rohling, M. Lott, and R. Halfmann, "Sotis-a self-organizing traffic information system," in *The 57th IEEE Semiannual Vehicular Technology Conference, 2003. VTC 2003-Spring.*, vol. 4, 2003, pp. 2442–2446.

[7] T. Nadeem, S. Dashtinezhad, C. Liao, and L. Iftode, "Trafficview: traffic data dissemination using car-to-car communication," *ACM SIGMOBILE Mobile Computing and Communications Review*, vol. 8, no. 3, pp. 6–19, 2004.

[8] C. Sommer, A. Schmidt, Y. Chen, R. German, W. Koch, and F. Dressler, "On the feasibility of umts-based traffic information systems," *Ad Hoc Networks*, vol. 8, no. 5, pp. 506–517, 2010.

[9] K. Collins and G.-M. Muntean, "A vehicle route management solution enabled by wireless vehicular networks," in *IEEE INFOCOM Workshops 2008,*, 2008, pp. 1–6.

[10] J. Rybicki, B. Scheuermann, M. Koegel, and M. Mauve, "Peertis: a peer-to-peer traffic information system," in *Proceedings of the 6th ACM international workshop on Vehicular Internetworking*, 2009, pp. 23–32.

[11] V. Gradinescu, C. Gorgorin, R. Diaconescu, V. Cristea, and L. Iftode, "Adaptive traffic lights using car-to-car communication," in *IEEE 65th Vehicular Technology Conference, 2007. VTC2007-Spring.*, 2007, pp. 21–25.

[12] J. Akhter and M. Islam, "Cost based performance evaluation of m/g/1/k traffic model," in *The 14th International Conference onComputer and Information Technology (ICCIT)*, 2011, pp. 303–308.

[13] J. Smith, "Properties and performance modelling of finite buffer m/g/1/k networks," *Computers & Operations Research*, vol. 38, no. 4, pp. 740–754, 2011.

[14] W. Chan, T.-C. Lu, and R.-J. Chen, "Pollaczek-khinchin formula for the m/g/1 queue in discrete time with vacations," *IEEE Proceedings of Computers and Digital Techniques*, vol. 144, no. 4, pp. 222–226, 1997.

[15] "Dijkstra's algorithm," *http://en.wikipedia.org/wiki/Dijkstra's_algorithm*.