Multiagent-based Cooperative Vehicle Routing Using Node Pressure and Auctions

Madhavan Seshadri
School of Computer Science and Engineering
Nanyang Technological University
Singapore
madhavan001@e.ntu.edu.sg

Zhiguang Cao
Energy Research Institute @NTU (ERI@N)
Nanyang Technological University
Singapore
zhiguangcao@ntu.edu.sg

Hongliang Guo
University of Electronic Science and Technology of China
Chengdu, China
guohl1983@uestc.edu.cn

Jie Zhang
School of Computer Science and Engineering
Nanyang Technological University
Singapore
zhangj@ntu.edu.sg

Ulrich Fastenrath
Traffic Information Management and Routing Optimization
BMW Group
Munich, Germany
Ulrich.Fastenrath@bmw.de

***Abstract*—Traffic congestion is a menace in the society with serious economic implications. Pressure based routing systems are known to be effective in wireless sensor networks, however rarely applied to transportation systems due to the nature of interactions between vehicles. This paper aims to alleviate the traffic congestion by adapting the node pressure concept into the traffic management system. In particular, we propose a Multi-Agent System (MAS) with vehicle agents and infrastructure agents, which can collaborate to provide static and dynamic solutions for reducing the node pressure. Pertaining to the static solution, the infrastructure agents rely on a reinforcement learning method to calculate the optimal routes for vehicle agents. Pertaining to the dynamic solution, the infrastructure agents dynamically adjust and re-route vehicle agents based on a novel multi-unit combinatorial auctioning system proposed in this paper. Extensive experiments on realistic traffic simulation platform have proven our methods, especially the dynamic solution, to achieve significant improvement in the reduction of node pressure and travel-times for the vehicle agents in comparison to others.**

***Index Terms*—Cooperative Vehicle Routing, Multi-agent Systems, Q-learning, Auctioning for Path allocation**

I. INTRODUCTION

Traffic congestion is a plaguing problem in our urban society with serious economy, health and other implications [1]. The problem has been attracting attention from researchers, Government Agencies, and industry. Singapore, for instance has formulated the Electronic Road Pricing Scheme to regulate the usage of roads within the network by pricing for their usage. Several studies such as Road Breakdown Minimization [2] and traffic light control [3] have been persormed. Multi-agent systems have been effectively used to alleviate the condition, with [4] and [5] being some prominent examples of their application. A transportation system can be modeled as a large, distributed and dynamic multi-agent system (MAS) in which the vehicles are represented as agents that move on the road network following their own route, which are determined by themselves or by the infrastructure agents at the road intersections [6].

However, all the multi-agent approaches fail to take into account the node-pressure and the throughput associated with the nodes in the network. Back-pressure routing has been extensively used in wireless sensor networks [7], [8] for routing data packets. It is a method of routing in queuing based networks for increasing the network throughput [9]. Though, it has been successfully applied in other domains, direct application of the concept to vehicle routing problem is infeasible as the constraints involved vary widely. Vehicles in a transportation network are independent decision making entities deciding their own routes and cannot be routed like data packets. Additionally, finite cost is incurred by the vehicles for using different road links and vehicles cannot be placed in infinitely long queues.

Most of the current multi-agent approaches can be categorized into reactive and proactive systems. Reactive Multi-agent system models get information from the environment which help the agents devise their strategies [10]–[12]. The underlying principle behind this is that traffic congestion is unpredictable and that the objective of the agents can be best reached based on the traffic information already available. In [11], the authors formulate a Reactive Traffic-Aware Routing Strategy (ReTARS) for real-time urban vehicular environments. ReTARS leverages prior global knowledge and real-time vehicular traffic information to calculate dynamic routes and avoiding unnecessary routes.

In contrast to reactive agents, proactive agents [4], [13], [14] forecast the future traffic conditions by combining the intended routes from different driving agents. Then, agents will reroute themselves beforehand when traffic congestion is predicted based on certain traffic congestion models. In [13], the authors have proposed several heuristics as re-routing methods to prevent future potential traffic congestions. Some re-routing heuristics discussed are: (1) randomly choosing a path from k shortest paths; (2) balancing the flow of vehicles along different routes; (3) maximizing the entropy of the routes

978-1-5386-1526-3/17/$31.00 ©2017 IEEE

(which is equivalent to reaching a uniform distribution of vehicles over the road networks). Additionally, apart from the reactive and proactive strategies used in Multi-agent cooperative vehicle routing problem, several other approaches have been used for the task. One such approach [15], is the usage of auction strategy to allocate certain resources in road network.

The approach being discussed in this paper is proactive. The major contributions of this paper are as follows: (1) To the best of our knowledge, this paper is the first to align the intentions of the infrastructure agents and the vehicle agents and improve the flow in the network by considering node pressure; (2) The reinforcement learning method used for price update of a node considers the traffic conditions at the upstream and downstream nodes which has an indirect implication on the network flow; (3) The multi-unit auctioning scheme ensures that the resources are allocated only to the agent that obtains the most improvement in its utility.

II. Problem Formulation for the Node Pressure Oriented Traffic Management

In this section, we focus on the formulating the vehicle routing problem using the node-pressure minimization approach.

A. Assumptions

There are two reasonable assumptions we make for the problem formulation: (a) Traffic jams as explained in the Introduction section is the result of the bottleneck of road intersections. Thus if node pressure can be defined as a function of the rate of inflow and the rate of outflow, then traffic occurs when the inflow is greater. (b) Free flow velocity: When the traffic state is in free flow (meaning the vehicles do not experience any traffic congestion) then the vehicle velocity is approximately equal to that of the speed limit of the road segment the vehicle is on $v_f = v_{lim}$; where v_f is the free flow velocity and v_{lim} is the speed limit on the road segment.

B. Mathematical Problem Formulation

The objective of the problem is to minimize the average node pressure cost of all the vehicles in the network. The main constraints for this are: (1) Every vehicle route must completely lead the vehicle from Origin to the Destination; (2) The vehicle inflow to any particular node in the network should not exceed its throughput capacity to minimize the traffic jam occurrence. Based on the Assumptions and constraints, the problem can be formulated as follows:

$$\min_{\vec{\mathbf{x}}_{\mathbf{i}}} \frac{1}{N} \sum_{i=1}^{N} \boldsymbol{\beta}^\top \boldsymbol{A}^+ \boldsymbol{x}_i \;\left|\; \begin{array}{l} \boldsymbol{A}\boldsymbol{x}_i = \boldsymbol{b}_i, \boldsymbol{x}_i \in \{0,1\}^n; \\ \boldsymbol{A}^+ \sum_{i=1}^{N} \boldsymbol{x}_i \preceq \boldsymbol{c}, \end{array}\right. \tag{1}$$

where N is the total number of vehicles in the network, β is a column vector of length n, where β_i represents travel time along road segment i. n is the total number of intersections in the network. Each element x_i has a binary value from set $\{0, 1\}$ where 0 represents the road segment being not taken and 1 represents vice-versa. $\mathbf{A}$ is the node-incidence matrix of the network and $\mathbf{A+}$ is a transformed representation of the incidence matrix for calculating node inflow. The term $\mathbf{A+}\sum_{i=1}^{N} x_i$ represents the node inflow vector and $\mathbf{c}$ is the capacity vector. Thus $\mathbf{A+}\sum_{i=1}^{N} x_i \preceq \mathbf{c}$ is the condition to avoid exceeding throughput.

C. Problem Analysis

The problem is analyzed on the following parameters: (1) Scale: Modern metropolitan cities are huge networks and individual vehicles are autonomous decision making entities, making the problem infeasible to be solved on a central system. (2) Dynamic: The vehicles can join and leave the system at any point of time making it a truly dynamic case. (3) Real-Time: The drivers will not wait for a long time for the route to be calculated. The number of rerouting has to be kept minimum. (4) Non-Convex: the problem needs to return routes to the drivers, which confines the solution to either choose or not to choose a road segment, i.e, zero-one problem, which belongs to the scope of combinatorial optimization [16]. Combinatorial optimization problem is notoriously difficult to solve in optimization domain.

To solve the node pressure minimization problem, we formulate our own MAS in the following sections.

III. Multi-agent Static-Route calculation Via Reinforcement Learning

MAS aid in the realistic representation of the dynamic nature involved in a transportation network. To achieve the same, we propose a MAS consisting of two unique types of agents, namely the vehicle agents and the infrastructure agents each with their own separate objectives. The vehicle agents are associated with objectives as seen in a real driver and infrastructure agents' objective is to collect, monitor and update network node pressure.

A. Vehicle Agent

The vehicle agents are meant to represent the interest's of the driver. In this paper, we consider three different types of vehicle agents, with one of our own, **Minimizes the price (Type I)** is a new type of vehicle agent which takes into account node-pressure as one of the criterion for selecting routes. Agents of this type aim to minimize the total price of the route which is formulated as a function of the distance and that of the node pressure.

$$\min_{\vec{\mathbf{x}}} \;\; \boldsymbol{\lambda}^\top \boldsymbol{x} \;\middle|\; \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \boldsymbol{x} \in \{0,1\}^n, \tag{2}$$

where $\boldsymbol{\lambda}$ is the price vector of length m and the value of λ_i is calculated by the following equation:

$$\boldsymbol{\lambda} = P(\boldsymbol{d}) + Q(\boldsymbol{\beta}^\top \boldsymbol{A}^+), \tag{3}$$

where $P(x)$ is the function converting distance into price and $Q(x)$ is the function converting node pressure as price.

Other agent types: A system with all vehicle agents aiming to minimize the price is not a realistic representation of the transportation network, hence we propose an additional two types of agents with different intentions. (1) Shortest Distance

(SD, Type II): Agents of this type aim to reduce the total travel distance along the path of the vehicle from Origin to Destination. Mathematically, the objective is represented as

$$\min_{\vec{x}} \quad \boldsymbol{d}^\top \boldsymbol{x} \mid \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \boldsymbol{x} \in \{0,1\}^n, \tag{4}$$

where d is a column vector of length m and (d_i) of $\mathbf{d}$ represents the length of road segment i. A, x and b have the same meaning as the previous equation. (2) Apriori Least Expected Travel Time (Type III): Agents of this type choose the path that minimizes their expected travel time. The computation of the route is done at the beginning of the journey. Mathematically, the decision is represented as value of λ_i is calculated by

$$\min_{\vec{x}} \quad \boldsymbol{t}^\top \boldsymbol{x} \mid \boldsymbol{A}\boldsymbol{x} = \boldsymbol{b}, \boldsymbol{x} \in \{0,1\}^n, \tag{5}$$

where $\boldsymbol{t}$ is the column vector of length m and (t_i) of $\boldsymbol{t}$ represents the travel time along the road i. A, x and b have the same meaning as in the Eq.(1).

B. Infrastructure agents

Infrastructure agents are imperative to assist in the communication and coordination among the vehicle agents in a structured manner. In this subsection, we shall discuss the different tasks of the infrastructure agents

1) Intention Collection: Since, all the vehicles in the network influence each other, it is necessary to collect the intentions from the vehicle agents [17]. The infrastructure agents present in the road intersections are used for the intention collection from the vehicle agents beginning their trip. The following information is collected: (1) O-D pair; (2) Type. Additionally, these intentions are only collected by the first infrastructure agent to which the vehicle is approaching.

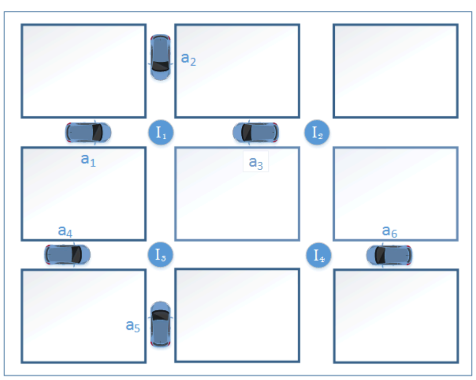


Fig. 1: Intention Collection from Agents

Fig. 1 best explains this process. Infrastructure agent I_1 collects intention from all the vehicle agents approaching it, which have just begun their trip, namely $\{a_1, a_2, a_3\}$. Similarly, Infrastructure agent I_3 collects from $\{a_4, a_5\}$, I_4 collects from $\{a_6\}$ while I_2 does not collect any.

2) Node Pressure Update: The infrastructure agents after gathering intentions, periodically update the Node pressure price based on the 'then' traffic scenario in its vicinity. In this case, the detection of traffic jam can be done by a simple check of inflow rate against the node capacity. If the traffic is grater than the threshold capacity in any node, the node will be congested in the next time step, otherwise it will not be.

In a linear update of the node pressure,

$$\beta_{t+1} = \beta_t + \alpha_t \left(\boldsymbol{A}^{+} \Sigma x_i - \boldsymbol{c} \right), \tag{6}$$

where β_t is the node pressure at time t and α_t is the update parameter for the node pressure.

However, this simple, linear price update of node pressure fails to account the congestion at the upstream and downstream nodes, thereby not being an optimal method to update the node pressure value. To improve the same, we propose a reinforcement learning method, namely the Q-learning approach for node pressure update.

C. Node-Pressure Update using Reinforcement Learning Solution

Q-learning is a model-free reinforcement learning technique. Specifically, Q-learning can be used to find an optimal action-selection policy for any given (finite) Markov decision process (MDP). It works by learning an action-value function that ultimately gives the expected utility of taking a given action in a given state and following the optimal policy after.

1) Q-learning Approach: A Markov decision process is a 5-tuple (S, A,$P_{sa}^{s'}$, $R(s', a)$, $\gamma \in [0, 1]$), where 1) S is a finite set of states; 2) A is a finite set of actions; 3) $P_{sa}^{s'} = Pr(s_{t+1} = s' | s_t = s, a_t = a)$ is the probability that action a in state s at time t will lead to state s' at time $t + 1$; 4)$R(s', a)$ is the immediate reward received after transition to state s' from state s; 5)$\gamma \in [0, 1]$ is the discount factor, which represents the difference in importance between future rewards and present rewards. To instantiate the Q-learning algorithm for the infrastructure agent so as to update the price information, we need to map the 5-tuple elements in MDP to the specific application domain.

State: $\boldsymbol{S}$ is a set of all possible average upstream node congestion, average downstream node congestion and current node congestion information, the congestion level can be categorized into three states, namely no congestion (sum of vehicle intentions less than threshold), light congestion (sum of vehicle intentions greater than threshold, but less than two times the threshold), and heavy congestion (sum of vehicle intentions greater than two times of the threshold). **Action:** $\boldsymbol{A}$ is the set of prices that the node can take; making the action set discrete for Q-learning algorithm to execute. The lowest price (β) is zero, and the highest price is 100. $P_{sa}^{s'}$ is given by the environment (vehicle agents), since once the action is taken (price is updated), vehicle agents sensitive to node pressure price will adapt its route and communicate the changed intention to the infrastructure nodes. **Reward:** The reward function $\boldsymbol{R}$ is defined as the average traffic severity of the upstream nodes, downstream nodes and the current node. The traffic severity is expressed as $e^{(\text{intention}-\text{threshold}-\text{default-pressure})}$. γ is set to be 0.99.

The Q-values are updated in an iterative way as in Eq.(7):

$$Q_{t+1}(s,a) = (1-\alpha_k)Q_t(s,a) + \alpha_k R_{t+1}(s',a) + \gamma \max_a Q_t(s',a), \tag{7}$$

where $Q_t(s,a)$ is the value of the state-action pair at the t^{th} time unit, $\alpha_t \in (0,1]$ is the learning rate.

D. Algorithm Summary

In this approach of problem solving, all the vehicle and infrastructure agents participate in a global negotiation process to achieve a cooperative route within the network. This process ensures that the routes obtained minimize the summation of node pressure within the network.

The Algo.1 provides the approach to calculate an efficient route for all the Type I agents in the network. Type III agents do not change the routes based on this algorithm, as the routes are calculated based on the intended traffic from all the agents, however the travel-time on the road link remains constant during the calculation process.

However, the simple approach has several issues associated with it. The amount of communication between the vehicle and infrastructure agents is huge, as in every iteration, the vehicle agents and infrastructure agents communicate with each other resulting in new traffic scenario resulting in a new Q value. This iterative negotiation process may never reach termination resulting in sub-optimal routes decided by the agents. Thus the approach is clearly unsuitable for real-time applications. However, it can be used for the theoretical study of the application of pressure-based routing. The traffic conditions in the network change dynamically during the trip due to the addition of new vehicles in the network, making the calculated routes sub-optimal. To mitigate the effects of the above disadvantages, we propose an auction based re-routing mechanism.

IV. Multi-agent Dynamic re-Routing Via Multi-unit Auctions

MAS Static Routing mechanism fails to account the transmuting traffic scenario combined with the inefficient negotiation process. To mitigate these effects, we propose a simple route calculation mechanism with dynamic re-routing based on Multi-Unit auctions.

In this method, vehicles calculate their current route based on the current β values which is periodically updated using the Q- learning approach. Instead of communicating their route to the infrastructure agents, vehicle agents Type I can apply for route changing at the nodes. To reduce the number of re-routing for the vehicle agents of Type I, we propose the process of segregating the nodes along a route into two categories, namely major nodes and minor nodes: Major nodes are the most important nodes along the current path of the vehicle where the re-routing decisions can be taken. They are usually the nodes with greater number of path options towards a particular destination node. All the nodes along the path which are not classified as a major node can be classified as a minor node.

input : $V = \{v_1, ..., v_Q\}$, a set of vehicle agents;
$I = \{i_1, ..., i_n\}$, a set of infrastructure agents;
$V_{sp}= \{v_1, ..., v_l\}$, vehicle agents that need guidance at i_i; l is the total number of vehicles
$\beta^0= \{\beta_i^0 = 0; \forall i \leq n\}$, calculated node pressure values;
$k \leftarrow 0$; Initialize Q_0 value and state

while $Q_{k-1}(s,a)\,!= Q_k(s,a)$ **do**
  foreach $v_j \in V$ **do**
    if $v_j \in V_{sp}$ **then**
      Recomputes its route based on β
    communicates its route to nearest $i_k \in I$
  calculate $Q_{k+1}(s,a)$ based on Eq.(7)
  $\beta^{k+1} \leftarrow$ Action value of $Q_{k+1}(s,a)$
  $k \leftarrow k+1$

Algorithm 1: Multiagent-based Route Guidance

The idea of this classification is to limit the number of re-routing decisions to minimize the inconvenience to the vehicle agents. Additionally, the major nodes are fixed for an Origin-Destination node pair and can be pre-computed for the whole network. The actual computation process is not in the scope of this paper and has been omitted.

1) Auction Problem Formulation: A set of bidders (Agents of Type III) can be represented as $A = \{a_1, a_2, \ldots, a_n\}$ compete for a set of goods (spots on the road) $R = \{R_1, R_2, \ldots, R_n\}$ which is controlled by the auctioning system (a set of infrastructure agents). Each good has multiple units and has a limited supply capacity $c_j \in \mathbb{N}$.The demand of the bidder is described as a multiset $S = (\rho_1, \rho_2 \ldots, \rho_m)$ where the demand for a particular good is $\rho_j \in \{0, 1\}$.

2) Basic Auction: In the basic auction, each bidder submits bid individually for each item ρ_i in the multiset where $\rho_i = 1$ and the valuation of all the individual items has to follow the condition that $\sum_{i=1}^{n} \rho_i v(R_i) \leq$ *Price of Current Path*. In this methodology, each vehicle agent submits bids for at-most one path which is better than the current path.

The bids are passed to the nearest infrastructure agents, which then communicate the bids for the respective edges to the appropriate infrastructure agents. The infrastructure agents auction the resources on the edges controlled by it. The winner decisions are then communicated back to the vehicle agents. To successfully be able to change to the new path, the bidder needs to win all the edges along the new path towards the destination. If the vehicle agent does not receive a response by time τ, it continues on the current path, where τ is the travel time to reach the next major node.

3) Multi-Unit Combinatorial Auction: Multi-Unit Combinatorial Auctions [18]–[20] are an extensively studied auctioning scheme able to represent and solve complex multi-set bid problems. In this subsection, we shall apply Multi-Unit Combinatorial Auction for solving the vehicle re-routing problem. Each bidder i has a valuation function, $v_i(S) \leq$ *Price of Current Path* to denote the valuation of the multiset

S. To participate in the auction, each bidder will submit a set of bids $B_i = \{b_{i1}, b_{i2} \ldots b_{ik}\}$ where $|B_i| = k$ and $b_{il} = (S_{il}, v_i(S_{il}))$ to the auctioneer. B_i is submitted using the XOR bidding language, which implicitly constraints that bidder i can win at-most one bid in B_i. This way it is ensured that the vehicle agents can submit bids for multiple paths which have better cost than the current path.

The auctioneer collects all these bids from the bidders and allocate each unit of the good to at-most one bidder . The auction should maximize the revenue of the auctioneer, meaning that the Infrastructure agents will try to allocate to the maximum gain in the utility values for the collection of vehicle agents of Type III. To achieve this goal, a winner determination problem is solved by the collection of infrastructure agents as formulated below.

$$\max \sum_{i=1,}^{n} \sum_{b_{il} \in B_i} v_i\left(S_{il}\right)\phi_{il} \left| \begin{array}{l} \forall j \in R, \sum_{i=1}^{n} \sum_{b_{il} \in B_i} s_i^j \phi_{il} \leq c_j; \\ \forall i \in A, \sum_{b_{il} \in B_i} \phi_{il} \leq 1, \phi_{il} \in \{0,1\}, \end{array} \right. \tag{8}$$

where ϕ_{il} is the combinatorial denotion of the bid $b_{il} \in B_i$ being chosen. The other symbols are as explained in the problem statement. The auctioneer aims to maximize the utility by solving Eq.(8). This WDP problem is a Multi-Integer Linear Programming problem by nature.

V. Experiments and Analysis

We experiment our approaches compared against existing methodologies to verify its effective nature in reducing the congestion and node pressure.

(a) San Francisco City Centre (b) Munich City Centre

Fig. 2: Road Networks used

A. Environment Settings

Simulation was performed in SUMO [21]. To simulate a real traffic scenario, two dense networks were used, namely San Francisco and Munich city centers. The area chosen is represented in Fig. 2. The number of lanes and the capacity of the nodes, links are as per the real network. Following are the properties used: (1) Network areas are: $520000m^2$ and $468000m^2$; (2) The number of OSM links are 1913 and 8209; (3) The number of OSM nodes are 960 and 3334. Vehicle configuration are: Length is $5m$ and the minimal gap to be followed between the vehicles is $2.5m$; The system uses the Krauss car-following model [22]; Origin-Destination pairs are randomly generated and are same for all the simulations; The vehicles will not occupy the road resources after reaching the destination; Traffic light duration is $20s$. Major and minor node computation is done prior to the simulation and the results are stored. All experiments have been performed in a PC with configuration: Processor is Intel-Core i7-6500 running with RAM of 20.00 GB at a maximum clock-speed of 2.60 GHz.

B. Comparative Experiments

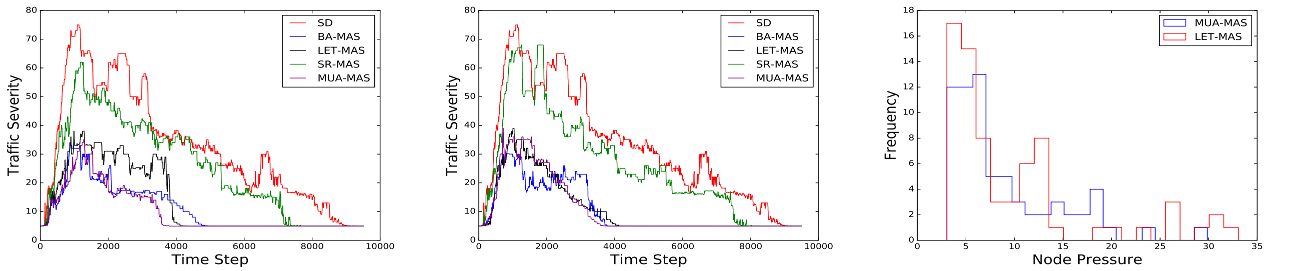
For the purpose of comprehensively studying the proposed algorithm in comparison to the conventional approaches, the following algorithms were used: (1) Distance Optimization: Each agent in the system tries to optimize its route based on the distance to reach the destination; (2) LET MAS:Least Expected Travel Time Path planning with Intersection rerouting, where all the vehicle agents calculates their initial path based on LET metric and at each intersection reroutes itself towards the destination based on the same metric [23]; (3) Static Route Calculation MAS (SR-MAS): This is a node-pressure minimization approach as discussed in the Algorithmic Table 1.(4) Basic Auction MAS (BA-MAS) as described in the previous section; (5) Multi-Unit Combinatorial Auction MAS (MUA-MAS): The auctioning mechanism proposed in this paper with the Winner Determination problem solved by the equation proposed in Eq.(8). Comprehensive comparison was done varying the percentage of different agents in the network. For the San Francisco Network, 4000 vehicles were used in simulation and for the Munich Network, 2000 vehicles were used. Vehicles were steadily introduced into network at a rate of 8 vehicles per simulation step in both networks.

Traffic Severity Comparison In the Multi-Agent system, two different distribution patterns used were [I: 40%, II: 30%, III: 30%] represented as $\gamma = 0.4$, proportion of Type I agents and [I: 60%, II: 20%, III: 20%] represented as $\gamma = 0.6$. This distribution pattern was used for the approaches 3,4 & 5.

The metric used for the comparison is defined as in Eq.(9). This representation gives increasing severity to the nodes with greater node pressure. In road networks, such nodes form the bottleneck which reduces the throughput. The symbol x^t represents the road being used at time instance t

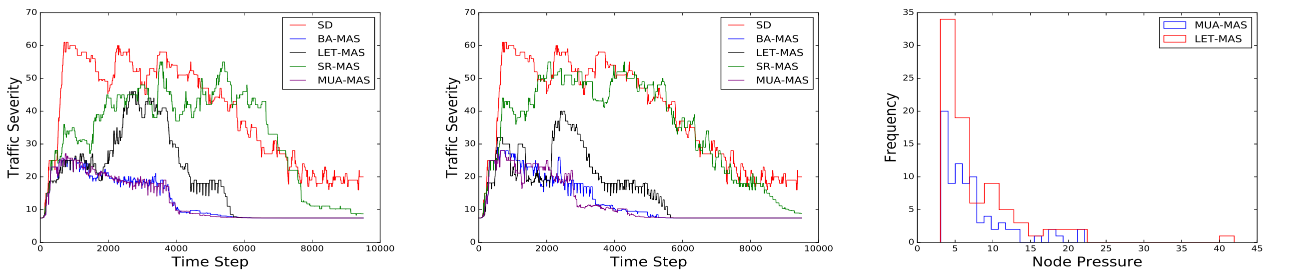
$$\textit{Traffic Severity}^t = \log\left(\sum \exp\left(\mathbf{A} + \sum x_i^t - \mathbf{c}\right)\right) \tag{9}$$

Fig. 3 represents the results of the above experiments performed. Fig. 3a and 3b are the results of the experiments performed on the San Francisco City Center Map with $\gamma = 0.4$ and $\gamma = 0.6$ respectively, while Fig. 4a and 4b are corresponding results for Munich Map. All the experiments prove that the static approaches of route computation, namely the Distance Optimization and Static Node Pressure Solution is significantly worse than the dynamic solutions in reducing the node pressure. Among the dynamic approaches, the proposed MUA-MAS outperforms the BA-MAS and LET-MAS in all the experiments, reducing the node pressure in the network. The auction based MAS systems seem to implicitly adjust the traffic based on the future predicted traffic. In the Fig. 3a, the LET-MAS outperforms the BA-MAS due to the high traffic density in the grid based San-Francisco Network. Also, while determining the winners in the auction, the BA-MAS does not



(a) Traffic Severity Comparison[$\gamma = 0.4$] (b) Traffic Severity Comparison[$\gamma = 0.6$] (c) Frequency Comparison of Node Pressures

Fig. 3: Comprehensive Performance Comparison for San Francisco Map



(a) Traffic Severity Comparison[$\gamma = 0.4$] (b) Traffic Severity Comparison[$\gamma = 0.6$] (c) Frequency Comparison of Node Pressures

Fig. 4: Comprehensive Performance Comparison for Munich Map

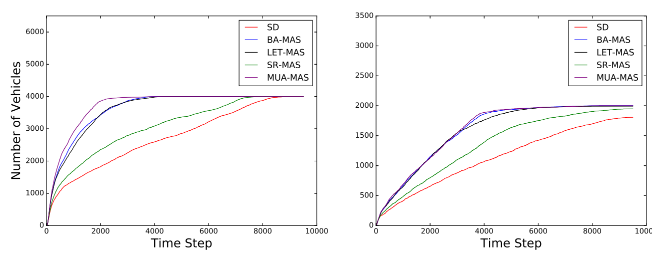
account for the greatest multiset bid which has been accounted for in MUA-MAS.

Node Pressure Severity To facilitate free flow traffic, besides the overall Traffic severity as defined in Eq.(9), the number of nodes with high node pressure values is important as fewer such nodes imply increased traffic flow rate. A comparison was made between the best proposed approach (MUA-MAS) and the best conventional approach (LET-MAS). The nodes with low node pressure values (< 3.0) were discarded and the average over all the time-steps was considered with all other parameters remaining the same.

The Fig. 3c and Fig. 4c shows the advantage of the proposed MUA-MAS approach over the LET-MAS in reducing the number of nodes with high node pressure values. Overall, number of high-pressure nodes is lower for the MUA-MAS and also most of the high-pressure nodes have node-pressure values (< 5.0) which is significantly better than the LET MAS which has several nodes in Fig. 3c with very-high node pressure values of approximately > 30.0. Another interesting observation is the fact that the total number of high-pressure nodes (> 3.0) are very limited ($< 10\%$) of the total number of nodes in both the networks, thus proving Assumption (a).

Agent based Metric Evaluation This sub-section focuses on highlighting the agent level advantages of the auction based MAS strategies. Three different parameters were considered for this purpose, namely the system termination state, Average Travel-Time and Distance incurred by the vehicles. For the System Termination State, all the approaches were compared while for the Average Travel-Time and Average Distance, only the dynamic approaches were compared.

Fig. 5a and Fig. 5b show that in both the networks, the number of completed trips quickly reaches the peak value for



(a) San Francisco (b) Munich

Fig. 5: Completed Trips vs. Simulation Step

Map	MUA-MAS	BA-MAS	LET-MAS
SFO	880	911	968
Munich	1873	1909	1873

TABLE I: Average time travelled by all the vehicles (s)

Map	MUA-MAS	BA-MAS	LET-MAS
SFO	1985.38	2659.52	2157.61
Munich	2321.3	3886.75	2348.1

TABLE II: Average distance travelled by All Vehicles (m)

the Auction based MAS, in which the MUA-MAS outperforms BA-MAS. This shows that the approaches are successful in completing all the vehicle trips faster than the other approaches, even faster than the LET-MAS whose primary objective is to minimize the travel duration. On a close perusal of Table I, shows that the average Travel Time for the trips under MUA-MAS is less than the other dynamic approaches thus in line with the Fig. The Table II is also favorable towards the MUA-MAS as the average trip distance is lesser than both

BA-MAS and LET-MAS as it is able to maximize the cost benefit in the WDP in Eq.(8).

C. Computation Complexity

In the proposed MUA-MAS, the Winner Determination Problem formulated in Eq.(8) is MILP in nature. As such, the LP problem can be solved in polynomial time using IPM or Karmakar's Algorithm. However the IP problem is NP-hard thus making the MILP problem of exponential complexity in the worst case. However the heuristic based approaches can solve the average problem in polynomial time. As such, since only the vehicles approaching Major Nodes participate in the auction, the complexity in the average case is polynomial. Experiments were performed to measure the time required to solve the MILP problem and the result is tabulated in the Table. III. The WDP problem takes less than 0.25 seconds to solve for upto 4000 vehicles for San Francisco Map.

Map	$\gamma = 0.4$		$\gamma = 0.6$	
	Mean	Max	Mean	Max
SFO	0.02	0.21	0.01	0.08
Munich	0.049	0.23	0.041	0.14

TABLE III: WDP Computation Time (s)

VI. Conclusion

In this paper, we have successfully identified road intersections as the bottleneck for the free flow of traffic and have formulated the Node-Pressure minimization problem to reduce this effect. To solve the node pressure minimization problem, we have formulated three different types of Multi-Agent Systems with the ability to handle the stochastic nature of the problem. The MAS, consisting of two unique types of agents, namely the vehicle agents and infrastructure agents is able to represent the intentions of the same while solving the problem through a negotiation based approach. The SR-MAS is able to achieve an improvement, by calculating the routes over a negotiation approach updating the road link price using a Q-learning approach. However, to handle the dynamic nature of traffic, we proposed two unique auction based re-routing approaches, namely BA-MAS and MUA-MAS. The MUA-MAS Winner Determination Problem is Mixed Integer Linear Programming (MILP) in nature and is able to maximize the utility for the set of Infrastructure agents. Our experiments proved that MUA-MAS outperforms all the other competitive MAS systems in reducing the network node pressure.

VII. Acknowledgement

This work is supported by BMW Group, also partially supported by the MOE AcRF Tier 1 funding (M4011261.020) awarded to Dr. Jie Zhang. We also wish to acknowledge the funding for this project from Nanyang Technological University under the Undergraduate Research Experience on CAmpus (URECA) programme.

References

[1] K. Zhang and S. Batterman, "Air pollution and health risks due to vehicle traffic," *Science of the total Environment*, vol. 450, 2013.
[2] H. Guo, Z. Cao, M. Seshadri, J. Zhang, D. Niyato, and U. Fastenrath, "Routing multiple vehicles cooperatively: Minimizing road network breakdown probability," *IEEE Transactions on Emerging Topics in Computational Intelligence*, vol. 1, pp. 112–124, April 2017.
[3] Z. Cao, H. Guo, J. Zhang, and U. Fastenrath, "Multiagent-based route guidance for increasing the chance of arrival on time.," in *AAAI*, pp. 3814–3820, 2016.
[4] R. Claes, T. Holvoet, and D. Weyns, "A decentralized approach for anticipatory vehicle routing using delegate multiagent systems," *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, pp. 364–373, 2011.
[5] J. France and A. A. Ghorbani, "A multiagent system for optimizing urban traffic," in *IEEE/WIC International Conference on Intelligent Agent Technology.*, pp. 411–414, 2003.
[6] S. Jiang, J. Zhang, and Y.-S. Ong, "A pheromone-based traffic management model for vehicle re-routing and traffic light control," in *International conference on Autonomous agents and multi-agent systems (AAMAS)*, pp. 1479–1480, 2014.
[7] L. Ying, S. Shakkottai, A. Reddy, and S. Liu, "On combining shortest-path and back-pressure routing over multihop wireless networks," *IEEE/ACM Transactions on Networking (TON)*, vol. 19, no. 3, pp. 841–854, 2011.
[8] L. Tassiulas, "Adaptive back-pressure congestion control based on local information," *IEEE Transactions on Automatic Control*, vol. 40, pp. 236–250, 1995.
[9] M. J. Neely and R. Urgaonkar, "Optimal backpressure routing for wireless networks with multi-receiver diversity," *Ad Hoc Networks*, vol. 7, no. 5, pp. 862–881, 2009.
[10] P. Toth and D. Vigo, *The vehicle routing problem.* Society for Industrial and Applied Mathematics, 2001.
[11] R. H. Khokhar, M. A. Ngadi, M. S. Latiff, and M. Amin, "Reactive traffic–aware routing strategy for urban vehicular environments," *International Journal of Ad Hoc and Ubiquitous Computing*, vol. 10, no. 3, pp. 149–163, 2012.
[12] R. Kohout and K. Erol, "In-time agent-based vehicle routing with a stochastic improvement heuristic," in *AAAI*, pp. 99–120, 1999.
[13] J.-S. Pan, I. S. Popa, K. Zeitouni, and C. Borcea, "Proactive vehicular traffic rerouting for lower travel time," *IEEE Transactions on Vehicular Technology*, vol. 62, pp. 3551–3568, 2013.
[14] Z. Cao, S. Jiang, J. Zhang, and H. Guo, "A unified framework for vehicle rerouting and traffic light control to reduce traffic congestion," *IEEE Transactions on Intelligent Transportation Systems (T-ITS)*, vol. 18, no. 7, pp. 1958–1973, 2017.
[15] S. R. Thangiah, O. Shmygelska, and W. Mennell, "An agent architecture for vehicle routing problems," in *ACM symposium on Applied computing*, pp. 517–521, 2001.
[16] M. Jiinger, G. Reinelt, and S. Thienel, "Practical problem solving with cutting plane algorithms in combinatorial optimization," 1995.
[17] J. Li, Q. Wu, and D. Zhu, "Route guidance mechanism with centralized information control in large-scale crowd's activities," in *International Joint Conference on Artificial Intelligence (IJCAI)*, pp. 7–11, 2009.
[18] K. Leyton-Brown, Y. Shoham, and M. Tennenholtz, "An algorithm for multi-unit combinatorial auctions," in *AAAI*, pp. 56–61, 2000.
[19] R. Gonen and D. Lehmann, "Optimal solutions for multi-unit combinatorial auctions: Branch and bound heuristics," in *2nd ACM conference on Electronic commerce*, ACM, 2000.
[20] R.-L. Leskelä, J. Teich, H. Wallenius, and J. Wallenius, "Decision support for multi-unit combinatorial bundle auctions," *Decision Support Systems*, vol. 43, no. 2, pp. 420–434, 2007.
[21] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent development and applications of sumo-simulation of urban mobility," *International Journal On Advances in Systems and Measurements*, vol. 5, no. 3&4, 2012.
[22] M. Behrisch, L. Bieker, J. Erdmann, and D. Krajzewicz, "Sumo–simulation of urban mobility: an overview," in *3rd International Conference on Advances in System Simulation*, ThinkMind, 2011.
[23] S. Wang, S. Djahel, and J. McManis, "A multi-agent based vehicles re-routing system for unexpected traffic congestion avoidance," in *17th International IEEE Conference on Intelligent Transportation Systems (ITSC)*, pp. 2541–2548, 2014.