Delft University of Technology

Delft Center for Systems and Control

Technical report 11-031

A new ant colony routing approach with a trade-off between system and user optimum*

Z. Cong, B. De Schutter, and R. Babuška

If you want to cite this report, please use the following reference instead:
Z. Cong, B. De Schutter, and R. Babuška, "A new ant colony routing approach with a trade-off between system and user optimum," *Proceedings of the 14th International IEEE Conference on Intelligent Transportation Systems (ITSC 2011)*, Washington, DC, pp. 1369–1374, Oct. 2011.

Delft Center for Systems and Control
Delft University of Technology
Mekelweg 2, 2628 CD Delft
The Netherlands
phone: +31-15-278.24.73 (secretary)
URL: https://www.dcsc.tudelft.nl

A New Ant Colony Routing Approach with a Trade-off Between System and User Optimum

Zhe Cong, Bart De Schutter, and Robert Babuška

Abstract— **Dynamic traffic routing (DTR) refers to the process of (re)directing traffic at junctions in a traffic network corresponding to the evolving traffic conditions as time progresses. This paper considers the DTR problem for a traffic network defined as a directed graph, and deals with the mathematical aspects of the resulting optimization problem from the viewpoint of network flow theory. Traffic networks may have thousands of links and nodes, resulting in a sizable and computationally complex nonlinear, non-convex DTR optimization problem. To solve this problem Ant Colony Optimization (ACO) is chosen as the optimization method in this paper because of its powerful optimization heuristic for combinatorial optimization problems. However, the standard ACO algorithm is not capable of solving the routing optimization problem aimed at the system optimum, and therefore a new ACO algorithm is developed to achieve the goal of finding the optimal distribution of traffic flows in the network.**

I. INTRODUCTION

Congestion on traffic roads may have different causes, e.g., accidents, road works, and bottlenecks, but one of the major causes of congestion is the difference between the demand and the capacity of the roads. One promising way of addressing this problem is to improve traffic management and control strategies. Dynamic traffic routing (DTR) [1] is an effective traffic management and control method that guides drivers to their route according to current (and future) traffic conditions when several alternative routes exist to their destination. In dynamic traffic routing, the notion *system optimum* and *user optimum* [2] play an important role. The system optimum is achieved when the vehicles are guided such that the total travel costs of all drivers (typically the total travel time) are minimized, while the user optimum means that on all alternative routes used, the costs are equal and minimal, and higher than those on the routes that are not used [2]. In general, the system optimum and the user optimum are two conflicting objectives, because in the system optimum not every user will optimize his or her individual objectives, and in the user optimum the collective objective will not be optimized. In this paper, we aim at solving the routing problem by considering both the system and the user optimum, finding a trade-off between them to benefit both the collective and the individual objectives.

For this purpose, we introduce a new routing algorithm, derived from the existing class of Ant Colony Optimization (ACO) algorithms [3], [4]. ACO has proven to have strong capabilities for solving hard combinatorial optimization problems, and it has several applications in traffic, including traffic routing [5], [6]. However, most of the algorithms reported in literature have their own limitations, e.g., the algorithm in [5] only focuses on the user optimum and ignores the impact of behavior of individual drivers on the traffic system, while — although it considers the system optimum — the paper [6] only investigates single-origin single-destination networks, and it uses a static traffic model where the traffic conditions are time-invariant, which makes the result less realistic. These problems are the motivation for developing a new algorithm in this paper.

In this paper we use a dynamic traffic network simulation model and include multiple origins and multiple destinations as well as a dynamic travel cost function that is based on the current and future conditions of the traffic network. The main novelty in our approach is that we pay special attention to the trade-off between the system optimum and the user optimum when we optimize the routing. Since the standard ACO algorithm only focuses on the user optimum, we develop a new ant-based optimization method that allows to steer the routing decisions towards the system optimum too, by introducing the *stench pheromone*, which can be used to make links less attractive in the case that there are already too many ants using that link. We use this new ACO algorithm, together with a dynamic traffic flow model, to optimize the routing decisions and to prevent or alleviate traffic congestion.

The rest of this paper is structured as follows. Section II defines the dynamic traffic routing problem. Next, we briefly recapitulate the standard ACO algorithm in Section III. Section IV then introduces the new ACO algorithm with stench pheromone. In Section V this stench-based ACO algorithm is then applied to solve the dynamic traffic routing problem. Section VI illustrates the new method using a study case involving the Singapore Expressway Network. We conclude with a short discussion of open issues and topics for future work in Section VII.

II. PROBLEM STATEMENT

A traffic network can be modeled as a directed graph with nodes and links as shown in Figure 1. The set of origins of the network is denoted by $\mathscr{O}$, the set of destinations by $\mathscr{D}$, the set internal nodes by $\mathscr{N}$, while $\mathscr{L}$ is the set of all links in the network. For each OD pair $(o,d) \in \mathscr{O} \times \mathscr{D}$, $q_{\text{in},o,d}(k)$ is the traffic inflow with destination node d entering at origin node o at time step k, where k indicates the time instant $t = kT$ with T the simulation time step of the traffic flow model. For each link $l \in \mathscr{L}$, the flow of vehicles at time

The authors are with the Delft Center for Systems and Control, Delft University of Technology, Delft, The Netherlands, email: {z.cong,b.deschutter,r.babuska}@tudelft.nl

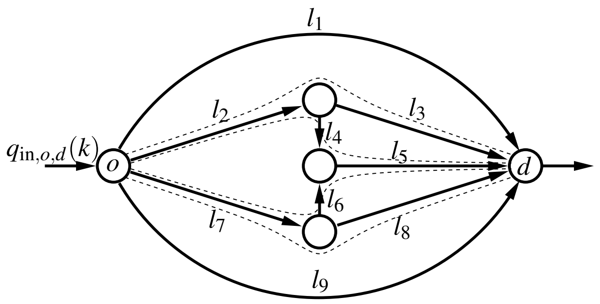

Fig. 1. Illustration of how the inflow $q_{\text{in},o,d}(k)$ of the OD pair (o,d) is distributed along multiple routes. The dashed lines correspond to the 4 routes with the highest flows (see the network pruning step of Section V-B for details).

step k that are traveling towards destination d is denoted by $q_{l,d}(k)$.

The total inflow $Q_{n,d}(k)$ at time step k of node $n \in \mathcal{N}$ with destination node d is given by:

$$Q_{n,d}(k) = \sum_{\ell \in I(n)} q_{\ell,d}(k),$$

where $I(n)$ is the set of incoming links for node n. Each outgoing link $l \in O(n)$ (with $O(n)$ the set of outgoing links for node n) is then characterized by the traffic flow $q_{l,d}(k)$:

$$q_{l,d}(k) = \beta_{n,l,d}(k) Q_{n,d}(k),$$

where $\beta_{n,l,d}(k)$ is the splitting rate for link l at node n with the destination d. The total flow $q_l(k)$ on link l is then given by:

$$q_l(k) = \sum_{d \in \mathcal{D}} q_{l,d}(k) \quad .$$

Each link has a dynamic cost $\varphi_l(k)$ per unit traffic flow, so the resulting flow pattern can be characterized by the cost function

$$J(k) = \sum_{j=1}^{N_{\text{p}}} \sum_{l \in \mathcal{L}} \varphi_l(k+j) q_l(k+j), \tag{1}$$

where N_{p} is the prediction horizon, which is introduced so that we can optimize the routing based not only on the current traffic conditions, but also including the future traffic conditions. The problem of minimizing $J(k)$ can be considered as a dynamic version of minimum cost flow routing problem.

A traffic network normally contains a large number of links and nodes, so optimizing all the splitting rates $\beta_{n,l,d}(k)$ dynamically results in a huge nonlinear, non-convex optimization problem, which is computationally very complex, making it almost intractable to solve the problem on-line. To address this issue, we introduce a two-step approach consisting of a network pruning step, followed by a flow optimization step using the new stench-based ACO algorithm that will be proposed in Section IV. The resulting DTR approach will then be elaborated in Section V. But first we briefly introduce the standard ACO algorithm.

III. Standard Ant Colony Optimization

Essentially, the Ant Colony Optimization (ACO) algorithm aims to find optimal (e.g., shortest) routes in a network by assigning and dynamically updating pheromone levels to the links in the network, where in the end the links with the highest pheromone levels correspond to the optimal routes.

The most basic ACO algorithm is called the Ant System [3] and works as follows. A set of M ants are randomly distributed over the network to search the best route to one or more destinations in the solution space $\mathcal{S}$. Initially for each ant c, a partial solution $s_{\text{p},c} \in \mathcal{S}$ is empty and all pheromone variables are set to a value $\tau_0 > 0$. In each iteration step, each ant c moves from node i to node j in the network, based on a probability $p_c\{j|i\}$ as below:

$$p_c\{j|i\} = \frac{\tau_{ij}^{\alpha}}{\sum_{l \in \mathcal{N}_{i,c}} \tau_{il}^{\alpha}}, \ \forall j \in \mathcal{N}_{i,c}, \tag{2}$$

with τ_{ij} the pheromone level previously deposited by the ants on link (i,j) and the parameter $\alpha \geq 1$ determining the relative importance of τ_{ij}. The feasible neighborhood $\mathcal{N}_{i,c}$ of ant c at node i is the set of nodes that are connected to i and that have not yet been visited by ant c in the current iteration. Afterwards, it adds link (i,j) to its partial solution $s_{\text{p},c}$ until it reaches the destination node. This is the inner loop in the ACO algorithm. Within the inner loop, each ant repeatedly applies (2) to construct a solution $s_{\text{p},c}$. Note that a higher pheromone value τ_{ij} in (2) increases the probability that ants at node i choose node j as their next node.

When all ants have reached a destination,(or when the maximum number of steps K_s has been reached), all the candidate solutions $s_{\text{p},c}$ are evaluated using a fitness function $F(s) : \mathcal{S} \to \mathbb{R}^+$. The fitness function $F(s)$ is accordingly used to calculate the pheromone deposit $\Delta\tau_{ij}(s)$ for the next iteration:

$$\Delta\tau_{ij}(s) = \begin{cases} F(s) \ , & \text{if } (i,j) \in s \\ 0 \ , & \text{otherwise.} \end{cases} \tag{3}$$

The pheromone level τ_{ij} on link (i,j) is updated by:

$$\tau_{ij} \leftarrow (1 - \rho_{\text{evap}})\tau_{ij} + \sum_{s \in S_{\text{upd}}} \Delta\tau_{ij}(s), \tag{4}$$

with $\rho_{\text{evap}} \in (0,1)$ the evaporation rate and S_{upd} the set of solutions that are eligible to be used for the pheromone update. This is the outer loop in the ACO algorithm. Note that there exist various rules to construct S_{upd}, of which the most standard one belongs to the Ant System and uses all the candidate solutions found in the current iteration.

IV. Ant Colony Optimization with Stench Pheromone

The main reason for introducing the stench pheromone is to prevent too many ants from converging to only one optimal solution s^*. Ants always have the ability to find the best route in the network, but sometimes one does not only want them to find the best route only, but rather an optimal distribution of ants in the network is desired. Unfortunately, standard

ACO has no capability to achieve this goal. However, with the stench pheromone ants can be pushed away when there are already enough ants on the best solution, and since no more ants can be attracted by the best route, they start to search the second, third, etc. best solutions in the solution space.

So the stench pheromone is used to keep ants away from a given link. This stench pheromone, when deposited on a link (i,j), will result in a decrease of the pheromone level τ_{ij}, and therefore also in a decrease of the probability $p_c\{j|i\}$. As a result fewer ants will choose link (i,j). So two types of pheromone are used in the new ACO algorithm proposed in this paper. The regular pheromone is deposited by ants while the stench pheromone is deposited by the global pheromone deposit mechanism. The ants will then choose links under the combined effect of the two types of pheromone. In the extreme case, when the stench pheromone is strong enough to cover the regular pheromone on the link (i,j), the total pheromone level τ_{ij} may become negative, which means that ants try to avoid traveling to this link, and this link is not attractive for the ants anymore.

Compared to (3), a new pheromone deposit function is given by:

$$\Delta\tau_{ij}(s) = \begin{cases} F(s) - G_{(i,j)} \ , & \text{if } (i,j) \in s \\ 0 \ , & \text{otherwise.} \end{cases} \quad (5)$$

where $G_{(i,j)}$ is a fitness function assigning a stench value to the link (i,j); this value will in general depend on the number of ants that have selected the link (i,j) in their final solution (see Section V.V-C for an example).

The resulting value $\Delta\tau_{ij}(s)$ is substituted into the same update function (4) to calculate the pheromone level τ_{ij} for the next iteration. But note that the stench pheromone may also make τ_{ij} negative (see (5)). Correspondingly, $p_c\{j|i\}$ will have no practical meaning if τ_{ij} is negative, and therefore, (2) needs to be some modified as follows:

$$p_c\{j|i\} = \frac{(\max\{\tau_{\min}, \tau_{ij}\})^{\alpha}}{\sum_{l\in\mathcal{N}_{i,c}} (\max\{\tau_{\min}, \tau_{il}\})^{\alpha}}, \ \forall j \in \mathcal{N}_{i,c}, \quad (6)$$

where $\tau_{\min}$ is the minimum pheromone level on each link, which guarantees a lower bound of pheromone level on each link, and thus prevents the denominator of (6) becoming zero.

V. Ant Colony Routing Algorithm

A. Main Algorithm

Ant Colony Routing (ACR) is developed to solve the dynamic traffic routing problem by using the ACO algorithm with the stench pheromone. It is important to note that there are several differences between an ant network and a traffic network. First of all, traffic network management strives for the system optimum, different from ants which strive for the user optimum. Second, in a traffic network each vehicle has a given destination associated to it, while the ants in ACO do not have individually pre-assigned destinations. Third, a traffic network is constrained by link capacity, but an ant network is not. Last but not least, link costs in a traffic network dynamically depend on the traffic conditions, while links costs in an ant network are fixed and static. Solutions for each these problems are proposed in detail below.

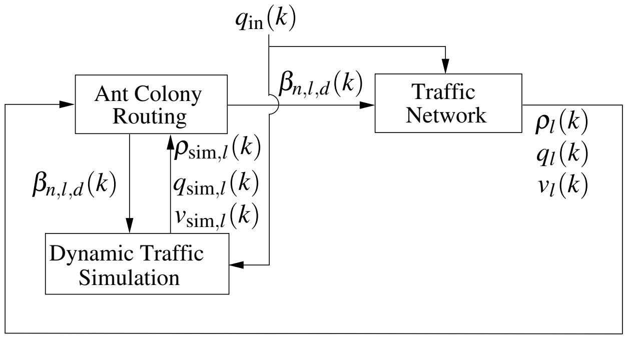

Fig. 2. Closed-loop control of traffic system with the ACR algorithm and the dynamic traffic prediction model.

In order to make ants strive for the system optimum as well, the stench pheromone is used to prevent ants from all converging to the same route (or links). Recall that the stench deposit mechanism introduced in Section IV can push ants away from the given link. Therefore, the stench pheromone makes ants distribute themselves over several best routes, and hence steers the traffic routing towards the system optimum.

Furthermore, ants in a network with multiple destinations may be guided to a destination that they are not going to if the pheromone levels are not distinguished by different destinations. To solve this problem, we use *colored* ants, where we assign one color per destination. Therefore, traffic flows can follow the trajectory of the corresponding colored ants to the desired destinations. Colored ants produce colored pheromone, and they only interact with that given pheromone. Note that the stench pheromone is uncolored and thus affects all ants.

Moreover, we can add a maximum ant capacity to the links of the ant network to constrain the number of ants (this can be done though a barrier function approach by increasing the stench pheromone drastically as the total number of ants on a link approaches the maximum link capacity).

Finally, we use a dynamic traffic simulation to define the link cost. The closed-loop control strategy of using ACR to determine the correct traffic flow distribution in a traffic network is illustrated in Figure 2. The dynamic traffic model is used to predict and simulate the evolution of the traffic network for the period $[kT, (k+N_{\text{p}})T]$. This provides the simulated variables — density $\rho_{\text{sim},l}(k+j)$, flow $q_{\text{sim},l}(k+j)$, and space-mean speed $v_{\text{sim},l}(k+j)$ on each link l for $j = 1,\ldots,N_{\text{p}}$ — to the ACO optimization algorithm. All these variables are used to determine the dynamic costs $\varphi_l(k+j)$, and pheromone is deposited accordingly. The corresponding ants distribution is then used to decide the splitting rate $\beta_{n,l,d}(k)$ (see (14) below), which is the control signal applied to the real traffic network during the period $[kT, (k+1)T]$. At time step $k+1$, the whole process is repeated again in a rolling horizon fashion.

The control strategy in Figure 2 represents the upper control layer, while in the lower control layer, the splitting rates β are translated into specific (hard or soft) route guidance mechanisms, such as dynamic matrix panels with route information, dynamic tolling, interaction with route guidance devices, and so on. However, the exact implementation of the route guidance mechanisms is outside the scope in this paper (see, e.g., [7] for more information).

As indicated before the new ACR method we propose consists of two steps: network pruning and dynamic flow optimization. These steps are explained in more detail in the next subsections.

B. Network Pruning

The network pruning part aims to remove the unnecessary links and nodes in a traffic network and to obtain the best links for each OD-pair so that flows only have to be distributed over those links rather than over the entire network. This is illustrated in Figure 1, where only the routes indicated by dashed lines could be chosen. Note that the flows of different OD pairs may share the same links.

For the network pruning step a quasi-static approach is considered, where a day is divided into several time slots (e.g., a morning rush hour, a non-busy midday period, and the evening rush hour) where for each time slot we determine the best routes considering the average traffic conditions (in particular, average link speeds) for the given time slot. The network pruning can be easily solved by using linear programming, as will be shown next.

The aim is to determine the traffic flows $q_{l,d}$ on each link l in the network for each destination $d \in \mathscr{D}$ such that the total travel cost is minimized. We assume here that the travel time is the travel cost to be optimized. For each link $l \in \mathscr{L}$ we define the average travel time t_l as L_l/v_l with L_l the length of link l and v_l the average speed on link l.

Now we can define the problem of minimizing the total travel time as:

$$\min_{q_{l,d}} \sum_{d\in\mathscr{D}} \sum_{l\in\mathscr{L}} T \cdot t_l \cdot q_{l,d} \tag{7}$$

with T the simulation time interval. Note that (7) aims to minimize the total travel time, because $T \cdot q_{l,d}$ expresses the number of vehicles on link l per simulation step, and thus $T \cdot t_l \cdot q_{l,d}$ corresponds to the total travel time on link l.

We also have to define some additional constraints:

$$q_{l_{\text{out}}(o),d} = q_{\text{in},o,d}, \quad \forall (o,d) \in \mathscr{O} \times \mathscr{D} \tag{8}$$

$$\sum_{d\in\mathscr{D}} \sum_{l\in I(n)} q_{l,d} = \sum_{d\in\mathscr{D}} \sum_{l\in O(n)} q_{l,d}, \quad \forall n \in \mathscr{N} \tag{9}$$

$$\sum_{d\in\mathscr{D}} q_{l,d} \leqslant q_{\text{cap},l}, \quad \forall l \in \mathscr{L} \tag{10}$$

$$q_{l,d} \geqslant 0, \quad \forall l \in \mathscr{L}, \forall d \in \mathscr{D}, \tag{11}$$

where $q_{\text{cap},l}$ is the capacity of link l, $l_{\text{out}}(o)$ is the single outgoing link of node o (otherwise we can introduce a virtual link with zero length and zero travel time), and $I(n)$ and $O(n)$ are respectively the sets of incoming and outgoing links for node n. Note that (9) states that the inflow of node n equals the outflow of node n (conservation of vehicles). It is easy to verify that (7)–(11) is a linear programming problem, which can be solved very efficiently using, e.g., a simplex method or an interior-point algorithm [8], [9].

Based on the solution of the above linear programming problem, we select for each OD-pair $(o,d) \in \mathscr{O} \times \mathscr{D}$ the N_{best} best links with the highest link flows. The non-selected links and links that do not belong to any path from o to d are then discarded. The resulting reduced network will then be used in the dynamic flow optimization step.

C. Dynamic Flow Optimization

The dynamic flow optimization part is based on ACO with the stench pheromone and colored ants, and works as follows. The ant network corresponds to the real traffic network graph, but with only the links selected in the network pruning step included. At each step k (corresponding to the time instant $t = kT$), we put the ants on the origins in proportion to the total demand for each OD-pair in the prediction period $[kT, (k+N_{\text{p}})T]$ and let them find their way to their destination, all according to their color. In order to compute the travel times of the ants on the links, we now use the dynamic traffic flow model as follows.

At time step k the current state of the traffic network is measured at estimated. Next, a simulation is run for the period $[kT, (k+N_{\text{p}})T]$ using the current state, the expected demand for each OD-pair, and the current splitting rates β. From the simulation results the average speed $v_{\text{av},l}$ is determined for each link l in the network, and next the cost for each ant traveling on link l can be computed as $\frac{L_l}{v_{\text{av},l}}$, where L_l is the length of link l.

For the ACO algorithm we define the fitness function F for a given solution s as the travel time on that route. The stench function $G_{(i,j)}$ for each link (i,j) could be a monotonously increasing function of the number of ants N_{ij}^{ant} on the link (i,j), which is initially low, and then increases gradually as the threshold number of ants $N_{\text{thresh},ij}^{\text{ant}}$ is reached. This threshold number of ants $N_{\text{thresh},ij}^{\text{ant}}$ corresponds to a threshold traffic density $\rho_{\text{thresh},ij}$ in the traffic network, which in its turn could be equal to the critical traffic density $\rho_{\text{crit},ij}$, or in some cases, like links near schools or hospitals, a much lower value. The general formulation of $G_{(i,j)}$ is:

$$G_{(i,j)} = \max\left(0, P_{ij}(N_{ij}^{\text{ant}} - N_{\text{thresh},ij}^{\text{ant}})\right) \tag{12}$$

where P_{ij} is the slope of the stench pheromone function, and $N_{\text{thresh},ij}^{\text{ant}}$ is the threshold number of ants on link (i,j). The threshold number of ants can be calculated by:

$$N_{\text{thresh},ij}^{\text{ant}} = \rho_{\text{thresh},ij} \cdot \lambda_{ij} \cdot L_{ij} \cdot \alpha \tag{13}$$

where λ_{ij} is the number of lanes on link (i,j), L_{ij} is the length of the link (i,j), and α is a coefficient of the number of vehicles that one ant represents.

The above process consisting of a simulation followed by the ACO algorithm is repeated several times for each time k until the splitting rates β converge or until a stopping criterion (e.g., maximum number of iterations, or the changes in the splitting rates dropping below a given threshold) is reached.

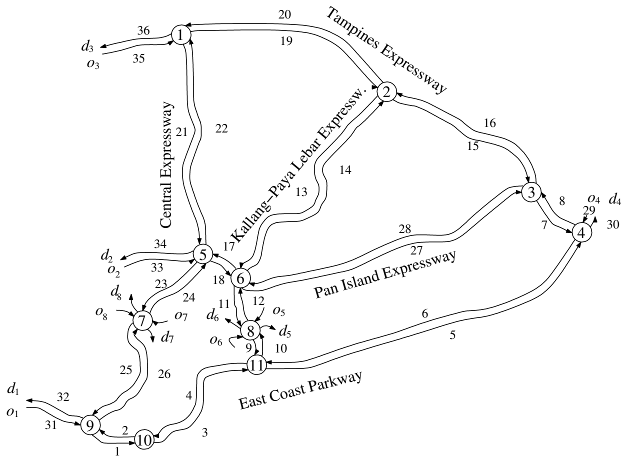

Fig. 3. The central and eastern parts of the Singapore expressway network.

Once the search is finished, we translate the number of ants back into the splitting rates as follows. Let $n_{\text{ants},l,d}(k)$ be the number of ants going to destination d via link l in the optimal solution produced by the stench-based ACO-algorithm with colored ants for time step k. For each node n in the ant network, each destination d, and each outgoing link l of node n in the ant network, we can obtain the splitting rate $\beta_{n,l,d}(k)$ according to the ratio of the number of ants $n_{\text{ants},l,d}(k)$ with destination d on the link l to the total number of ants with destination d on all the outgoing links of node n:

$$\beta_{n,l,d}(k) = \frac{n_{\text{ants},l,d}(k)}{\sum_{\ell \in O_{\text{ant}}(n)} n_{\text{ants},\ell,d}(k)} \tag{14}$$

where $O_{\text{ant}}(n)$ is the set of outgoing links of node n in the ant network. If the node n or the link l are present in the real traffic network graph but not in the ant network (due to the network pruning step), we set $\beta_{n,l,d}(k) = 0$. Next, the splitting rates β are imposed on the real traffic flows via route guidance measures.

VI. Case Study

A. Singapore Expressway Network

In order to illustrate the new ACR approach we apply it to the same case study that was considered in [6], viz. the Singapore expressway network [10]. Instead of investigating the whole network, only the central and eastern parts are chosen (see Figure 3), which contain 18 highway stretches (36 if we consider both directions), 8 origins, and 8 destinations. This area includes the central business district, connected to origins and destinations 5, 6, 7, and 8, as well as the connection with the airport through origin and destination 4.

In this case study, the ACR algorithm will be applied at origin o_4 (i.e. for traffic from the airport and the east region of the island), and all traffic is assumed to have its destination in the business district (destinations d_5, d_6, d_7, d_8). We set the simulation period as 2 hours, representing a morning rush hours from 7 : 30 am to 9 : 30 am. The inflow of the network increases from 0 veh/h to 6000 veh/h during the first half hour, stays at 6000 veh/h for an hour, and then decreases from 6000 veh/h to 0 veh/h during the last half hour.

B. Simulation Settings

We use the METANET traffic model [11] parameters as found in [12]: simulation sample time $T = 10$ s, critical density $\rho_{\text{crit},m} = 27$ veh/km/lane, free-flow speed $v_{\text{free},m} = 110$ km/h, speed-flow relationship parameter $a_m = 2.34$, and speed equation parameters $\eta = 30$ km^2/h, $\tau = 10$ s, $\kappa = 20$ veh/km.

In order to take the future traffic conditions into consideration, we put the prediction horizon $N_{\text{p}} = 30$, which means the prediction time length is $N_{\text{p}}T = 30 * 10$ s $= 300$ s $= 5$ minutes. Since the maximal inflow of the network is 6000 veh/h, at most 500 vehicles can enter the network during this period. Therefore, we set the number of ants in the trial to 500, using one ant to represent one vehicle. Moreover, for a given route s, the travel time cost $\varphi(s)$ is the sum of the costs on all of links of route s, $\varphi(s) = \sum_{l \in s} \frac{L_l}{v_{\text{av},l}}$. The fitness function $F(s)$ is the inverse of the travel cost $\varphi(s)$:

$$F(s) = \frac{1}{\varphi(s)} \tag{15}$$

In the Singapore express network, the slope P_{ij} in (12) is set at 1 in most of links. However, link 1, 2, 3, 4, 9, 10, 11, 12, 17, 18, 23, 24, 25, and 26 are in the central business district, so traffic flows should be more limited on these links, and hence the corresponding slope P_{ij} is set at 2 so as to more heavily penalize too many ants converging on those links. Besides, the threshold density per lane is 25 veh/km/lane. Both parameters are substituted into (13) to calculate the threshold number of ants on each link.

C. Simulation Results

Through applying network pruning, we find out two best routes, $\mathscr{R}_1 = \{29, 6, 10\}$ (15 km) and $\mathscr{R}_2 = \{29, 8, 28, 11\}$ (16 km). Link 29 is the link connected to the origin 4, and link 6 and 8 are two outgoing links of link 29. For the dynamic flow optimization part, we first run ACR with and without the stench pheromone for just one time step to compare the different results. In Figure 4, we can see the number of ants on link 29 is always 500, because every ant will pass it. At the beginning of the iteration cycle, the numbers of ants choosing link 6 and 8 are nearly equal. However, at the end of the iteration cycle, almost every ant chooses link 6 if there is no stench pheromone deposited, while with the stench pheromone, almost half of the ants choose link 8. This is because the stench pheromone prevents too many ants from converging to one link, and thus pushes some ants to search for a second best link. The splitting rate at each node in the traffic network is then calculated by the distribution of ants under the effect of the stench pheromone.

In Figure 5, the blue solid lines represent the flow on each link in route $\mathscr{R}_1 = \{29, 6, 10\}$, with destination node d_5, and the red dash lines represent the flow on each link in route $\mathscr{R}_2 = \{29, 8, 28, 11\}$, with destination node d_6. If we use ACO without the stench pheromone, all the vehicles

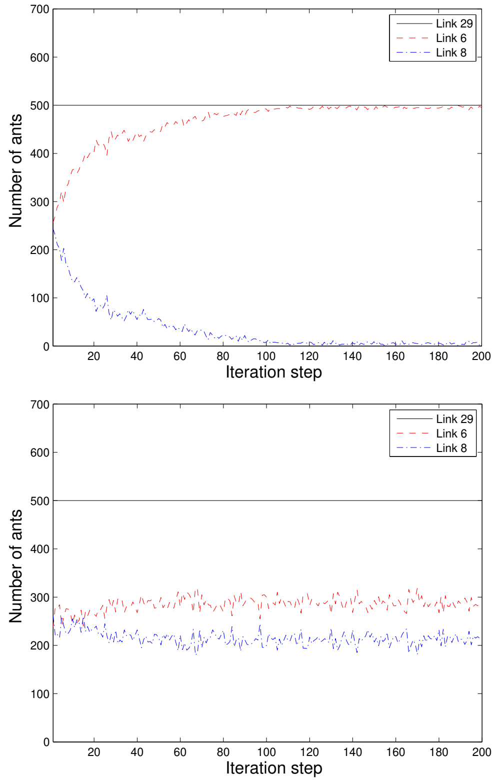

Fig. 4. Without the stench pheromone, all ants tend to choose link 6, while with the stench pheromone, ants choose both links. Link 6 attracts a little more ants.

are guided to the best route $\mathscr{R}_1 = \{29, 6, 10\}$. However, the capacity of link 6 is 6340 veh/h, while the capacity of link 10 is 4755 veh/h, because there is the bottleneck from link 6 to link 10. Hence, we need introduce the stench pheromone to disperse the vehicles. We can see that the flows are divided over both routes in Figure 5, with a little more vehicles choosing $\mathscr{R}_1 = \{29, 6, 10\}$. The stench pheromone decreases the pheromone amount on the best route, and successfully directs part of the traffic flows to route $\mathscr{R}_2$. Under this traffic assignment, neither of traffic flows exceeds the capacity.

VII. Conclusions and Future Work

We have proposed a new method for solving the Dynamic Traffic Routing (DTR) problem using a two-step approach: network pruning and network flow optimization. This approach significantly reduces the computational burden of solving the complex DTR optimization problem. Besides, we have also developed a novel ant-based algorithm with the stench pheromone, which can be used to prevent ants converging to one route and hence it can steer the DTR distribution towards the system optimum rather than the user optimum, as is the case in standard ant colony optimization.

Further work includes more detailed case studies, extensive assessment of the performance and efficiency of the proposed ant colony routing method compared to other DTR methods, inclusion of and integration with other traffic control measures, analysis of the theoretical properties (e.g., convergence) of the new ant colony optimization algorithm

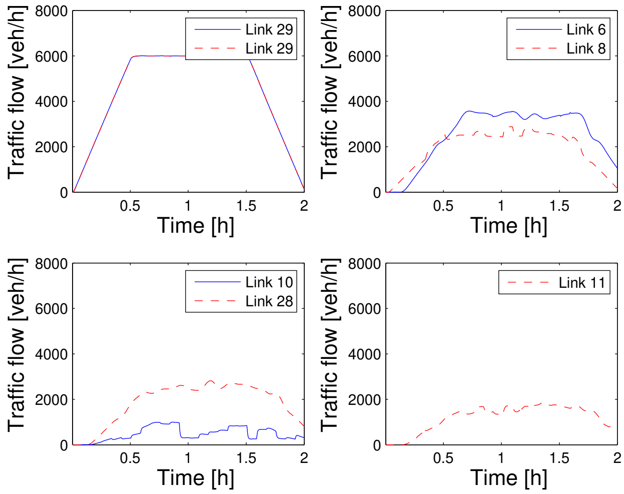

Fig. 5. Using ACO with the stench pheromone, the traffic flow are divided over $\mathscr{R}_1$ and $\mathscr{R}_2$.

with stench pheromone, and investigation on how to tune the trade-off between user optimum and system optimum through the selection of the stench function G.

Acknowledgments

Research supported by the BSIK project "Next Generation Infrastructures (NGI)", the European COST Action TU0702, the European 7th Framework Network of Excellence "Highly-complex and networked control systems (HYCON2)", and the Transport Research Centre Delft.

References

[1] P. Kachroo and K. Özbay, *Feedback Control Theory for Dynamic Traffic Assignment*. London, UK: Springer, 1999.

[2] J. G. Wardrop, "Some theoretical aspects of road traffic research," *Proceedings, Institute of Civil Engineers, Part II*, vol. 1, pp. 325–378, 1952.

[3] M. Dorigo, V. Maniezzo, and A. Colorni, "Ant system: optimization by a colony of cooperating agents," *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 26, no. 1, pp. 29–41, 1996.

[4] M. Dorigo and L. Gambardella, "Ant Colony System: a cooperative learning approach to the traveling salesman problem," *IEEE Transactions on Evolutionary Computation*, vol. 1, no. 1, pp. 53–66, 1997.

[5] B. Tatomir and L. Rothkrantz, "Hierarchical routing in traffic using swarm-intelligence," *Intelligent Transportation Systems Conference, 2006. ITSC '06. IEEE*, pp. 230–235, 2006.

[6] D. Alves, J. van Ast, Z. Cong, B. De Schutter, and R. Babuška, "Ant colony optimization for traffic dispersion routing," in *Proceedings of the 13th International IEEE Conference on Intelligent Transportation Systems*, Madeira Island, Portugal, Sept. 2010, pp. 683–688.

[7] R. Bishop, *Intelligent Vehicle Technology and Trends*, ser. Artech House ITS Library. Boston, Massachusetts: Artech House, 2005.

[8] Y. Nesterov and A. Nemirovskii, *Interior-Point Polynomial Algorithms in Convex Programming*. Philadelphia, Pennsylvania: SIAM, 1994.

[9] S. J. Wright, *Primal-Dual Interior Point Methods*. Philadelphia, Pennsylvania: SIAM, 1997.

[10] C. Willoughby, "Singapore's motorization policies," *Transport Policy*, vol. 8, pp. 125–139, 2001.

[11] A. Messmer and M. Papageorgiou, "Metanet: A macroscopic simulation program for motorway networks," *Traffic Engineering and Control*, vol. 31, pp. 466–470, 1990.

[12] D. Alves, "Ant dispersion routing for traffic optimization," Master's thesis, Delft University of Technology, Delft, The Netherlands, 2009.