

# Anticipatory Vehicle Routing using Delegate Multi-Agent Systems

Danny Weyns, Tom Holvoet and Alexander Helleboogh

**Abstract**—This paper presents an agent-based approach, called delegate multi-agent systems, for anticipatory vehicle routing to avoid traffic congestion. In this approach, individual vehicles are represented by agents, which themselves issue light-weight agents that explore alternative routes in the environment on behalf of the vehicles. Based on the evaluation of the alternatives, the vehicles then issue light-weight agents for allocating road segments, spreading the vehicles' intentions and coordinating their behavior. To evaluate the approach, we have developed an initial prototype application. Test results indicate that delegate multi-agent systems are a promising approach for anticipatory vehicle routing.

## I. INTRODUCTION

Monitor and control systems for traffic share two important characteristics with other complex distributed software systems: (1) highly dynamic and changing operation conditions under which the systems have to operate such as a fluctuating amount of traffic, changing behavior of drivers, traffic jams, and road accidents, and (2) the inherent distribution of resources and activity making centralized control hard to achieve; for example: traffic is naturally distributed over the road network, traffic lights act locally (or are coordinated regionally), traffic jams cause local delays, etc. Examples of other classes of systems that share these characteristics are manufacturing control systems, automated transportation systems, and wireless sensor networks.

In our research, we study situated multi-agent systems (situated MAS) for engineering systems with such properties [1], [2], [3]. A situated MAS is a type of MAS which consists of a number of autonomous entities (agents) that are situated (i.e. embedded and localized) in an environment. Control in a situated MAS is decentralized. Situated agents rely on cooperation to achieve the functionality of the system, rather than on individual cognitive capabilities. Situated agents can flexibly adapt their behavior to changing circumstances in the environment making them particularly suitable to cope with dynamic and changing operating conditions.

Over the last five years, our research was mainly driven by applications in the domain of manufacturing control with particular attention for automated transportation systems that use automatic guided vehicles [4]. Recently, the domain of traffic monitoring and control became an important driver for our basic research. The similarities in system characteristics and requirements incite to transfer ideas among domains.

In the near future, vehicles will be equipped with increasingly advanced navigation systems and communication

facilities. This enables interesting applications such as vehicles that collaboratively interpret the local traffic situation and spread useful information to traffic signs that inform drivers about the actual traffic conditions. In this paper, we focus on another interesting application: anticipatory vehicle routing to avoid traffic congestion. Congestion is basically a resource coordination problem where vehicles have conflicting intentions about the use of parts of the road network. One way to avoid (or at least reduce) congestion is by letting vehicles anticipate possible conflicts. In this paper, we propose a MAS-based approach called “delegate MAS” for anticipatory vehicle routing. In this approach, individual vehicles issue light-weight agents that explore alternative paths in the environment on behalf of the vehicles. Based on the evaluation of the alternatives, the vehicles then issue a second type of light-weight agents for allocating road segments, spreading the vehicles' intentions and coordinating their behavior. For experimentally evaluating the approach, we have developed an initial prototype application. The work we present is obviously still in progress, and it is clear that there is still a lot of research to be fulfilled (both conceptually as well as technologically) before we can assess the feasibility of the approach in practice. However, test results show that the approach is quite promising.

*Overview.* The remainder of this paper is structured as follows. In section II, we elaborate on the approach of delegate MAS for anticipatory vehicle routing. Section III explains the prototype application and shows initial test results. In section IV, we compare our approach with related work. Finally, in section V we draw conclusions.

## II. DELEGATE MAS FOR ANTICIPATORY VEHICLE ROUTING

In this section, we explain the approach of delegate MAS for anticipatory vehicle routing. We first explain our starting point on the problem of traffic congestion and we outline requirements for anticipatory vehicle routing. Then we introduce the basic software architecture of delegate MAS and we explain how anticipatory routing is realized with the approach.

### A. Traffic Congestion and Requirements for Anticipatory Vehicle Routing

Traffic congestion is a road condition characterized by slower speeds, increased queueing, and longer trip times. It occurs when roadway demand is greater than its capacity. Traffic congestion has several negative effects such as wasted time for drivers, wastes of fuel, increasing air pollution, etc.

D. Weyns, T. Holvoet and A. Helleboogh are with DistriNet labs, Department of Computer Science, Katholieke Universiteit Leuven, B-3001 Leuven, Belgium {danny.weyns, tom.holvoet, alexander.helleboogh}@cs.kuleuven.be

A variety of attempts are used to alleviate traffic congestion; examples are speed regulation, counterflow (sections of highways that operate in opposite directions at different times of the day), lane splitting and filtering, and broadcasting of traffic reports.

One of the objectives of intelligent transportation systems [5], [6] is to exploit advanced information technologies such as floating car data [7], advanced sensing technologies [8], and vehicular communication systems [9] to avoid (or at least reduce) congestion. In this paper, we propose an agent-based approach for anticipatory vehicle routing that makes use of advanced information technologies. The main requirements of the approach are:

- 1) The system should support the coordinated navigation of many cars in a road network trying to avoid congestion in the near future as much as possible.
- 2) The system should provide feedback to individual drivers that allow them to pro-actively anticipate possible bottlenecks.

The approach we propose assumes that the following information technologies are available:

- 1) The road infrastructure is equipped with electronic devices which provide some computation power and are connected through a network. The devices can also communicate with smart modules on board of vehicles.
- 2) Vehicles are equipped with smart devices which can sense that vehicle's local traffic conditions (current position and speed, etc.) and which can communicate with local nodes, i.e. vehicles and electronic devices of the road infrastructure. Furthermore, a smart device is able to calculate all possible routes from the vehicle's current location to its destination with a given maximal distance. We denote this set of routes as the feasible paths of a vehicle to reach its destination.

### *B. Basic Abstractions and Functionalities for Anticipatory Vehicle Routing*

We now introduce the basis abstractions for a software architecture for anticipatory vehicle routing and we outline the required functionalities of the vehicles to anticipate congestions when driving to their destinations.

*1) Basic Abstractions:* The software architecture of our approach is based on three basic abstractions: vehicle agent, virtual environment, and infrastructure agent.

In the software architecture, each vehicle is represented by a vehicle agent that is deployed on the smart device of the vehicle. The vehicle agent has knowledge about that vehicle's start location and destination, and its current state such as position and speed. The vehicle agent provides information to the driver to achieve its goals, i.e. reaching its destination with minimal congestion overhead.

A virtual environment reflects the real traffic environment. The physical road network is mapped onto a graph representation. The nodes of the graph represent discrete road elements, i.e. road segments and crossroads. The virtual

environment is a distributed software entity that is deployed on the electronic devices provided by the road infrastructure.

With each road element an infrastructure agent is associated. Infrastructure agents are also deployed on the electronic devices of the road infrastructure. Vehicle agents can book road elements via the infrastructure agents. Bookings must be refreshed regularly to maintain the reservation. We elaborate on this important issue below.

*2) Functionalities to Support Anticipatory Vehicle Routing:* The main functionalities vehicle agents require for anticipatory routing are (1) functionality to explore relevant paths in the environment, and (2) based on the evaluation of the possible alternatives, functionality to choose one path and record it as an intention.

Vehicle agents need to explore the relevant paths to reach their destination. This results in different alternatives the vehicle agent can select to achieve its goal.

Based on the evaluation of the alternatives, the vehicle agent chooses one path as its intention. Adopting an intention obviously has implications on the possibilities of other vehicles. Therefore, the vehicle agent needs to communicate with the corresponding infrastructure agents and inform them of the time they intend to occupy the corresponding road elements, i.e. the information agents need to book these reservations.

### *C. Delegate MAS for Anticipatory Vehicle Routing*

A typical approach would be to let vehicle agents and infrastructure agents use direct communication protocols to achieve the functionalities for anticipatory vehicle routing. In contrast, we use delegate MASs for obtaining exploration of alternatives and propagation of intentions toward the infrastructure agents. Delegate MAS are introduced in [10]. The approach is closely related to Polyagents introduced by Brueckner and Parunak [11], [12]. Delegate MASs are inspired by food foraging in ant colonies. Food foraging ants execute a simple procedure. In absence of any signs in the environment, ants walk around randomly in search for food. When an ant discovers a food source, it drops a smelling substance - a pheromone - on its way back to the nest while carrying some of the food. This pheromone trail evaporates over time, and disappears if no other ant deposits fresh pheromone. Another ant in search for food will use pheromones in the environment as a source of information to direct its own behavior. Pheromones indicate possible routes to a food source, and ants are urged by instinct to follow this trail to the food source. When the ant finds the food source, it will return with food, while depositing pheromone itself. When the ant discovers that the food source is exhausted, it starts a random search for food again. As the pheromone trail is no longer reinforced, it disappears over time.

These simple behavioral patterns result in an emergent behavior of the ant colony that is highly ordered and effective at foraging food while being robust against dynamics in the environment. An important property of this type of collective behavior is that global information is made available locally. The pheromone trails denote in which direction the ants

have to move to find food at a remote location. This leads to the following engineering principles: (1) place relevant information as signs in the environment (locally available data informs about global properties); (2) limit the lifetime of this information - evaporation - and refresh the information as long as it remains valid (allows the system to cope with changes and disturbances).

We apply these principles in our approach and define two types of light-weight agents, which represent two different delegate MASs and which share a common virtual environment for indirect communication. To distinguish them from vehicle and infrastructure agents, we call the light-weight agents “ant agents” or ants further on. Delegate MASs consist of ant agents that reside in a virtual software environment which reflects the traffic environment, and in which ant agents can navigate. It is important to notice that the ants of the delegate MASs can explore alternative paths orders of magnitudes faster as vehicles can drive.

1) *Exploration Ants:* A vehicle agent generates exploration ants at a certain frequency which explore the feasible paths from the vehicle’s current position to the destination. These exploration ants are scouts that each explore a feasible route through the underlying system and evaluate this route. To evaluate a path, an exploration ant follows the path through the virtual environment, and interacts with the infrastructure agents at the different nodes by asking the infrastructure agent *what* the travel time would be *if* a vehicle would arrive at a particular time in the near future. The exploration ant collects this information, and then proceeds to the next node in the path, in which this behavior is repeated. When arriving at the destination, the exploration ant returns and reports back to its base, i.e. the vehicle agent that issued the exploration ant. To illustrate this, consider the scenario in Fig. 1 that shows the Leonard crossroad, a well known Belgian congestion point between the Brussels Ring and the E411 motorway. A vehicle represented by the dark rectangle

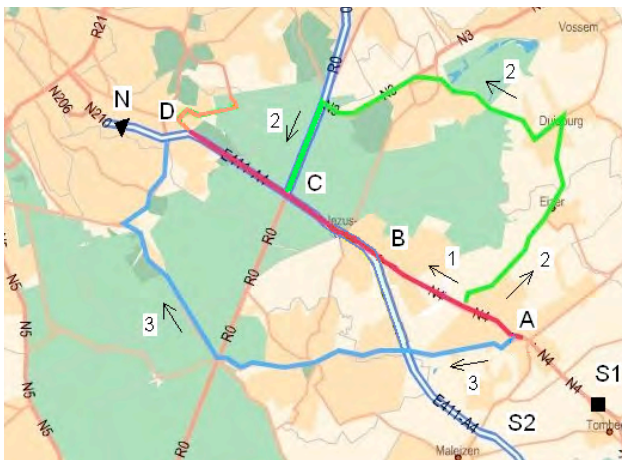


Fig. 1. Scenario of anticipatory vehicle routing. The dark rectangle represents the vehicle at its current location; the dark triangle represents the vehicle’s destination.

located at point  $S_1$ , intends to drive to a location indicated by the dark triangle located at point  $N$ . From the crossroad  $A$ , the vehicle has three possible trajectories to reach its destination: the motorway (indicated by arrow 1), the short detour (arrow 2), and the long detour (arrow 3).

Fig. 2 illustrates the exploration process for a simplified scenario. The vehicle agent on the right hand side creates three exploration ants for scouting feasible paths on behalf of the vehicle agent. These ants bring back information on the current alternatives to reach the vehicle’s destination. The list of alternatives is refreshed regularly as exploration ants are sent out regularly.

2) *Intention Ants:* Exploration requires the infrastructure agents to possess an adequate estimate of their future load. To serve this purpose, vehicle agents generate intention ants, which propagate the intentions of vehicle agents through the virtual environment.

The process goes as follows. When a vehicle agent has constructed a set of valid paths to follow, the vehicle agent selects one candidate path to become its intention. The criteria used for this selection depend on quality factors such as minimizing traveling time and minimizing standstill. Each vehicle agent can use its own driver-specific combination of factors. Then, the task agent creates intention ants, at a certain frequency, to inform the infrastructure agents that are involved in this intended path.

The intention ants follow the selected path, and virtually travel the route of their selected candidate solution. On their virtual journey, the intention ants acquire travel and queuing times from the infrastructure agents on their path. Changes which occurred since the exploration, immediately become visible when these infrastructure agents provide the information. In contrast to the exploration ants, intention ants inform the infrastructure agent that their vehicle agent is likely to visit them at the estimated time. In this way, intention ants make a booking on the road segment, and the infrastructure agent adjusts its local schedule to account for this visit. As a consequence, infrastructure agents are able to predict their load more accurately to their visiting exploration and intention ants. Similar as exploration ants, intention ants report back to their vehicle agent to inform the agent about the quality and schedule of the bookings.

Fig. 3 illustrates this for our example scenario. Based on the information that was gathered by the three exploration ants, the vehicle agent decides that the path explored by the last exploration ant (which explored candidate path 3) fits the goal best. To confirm this and align this decision with the involved road elements, an intention ant walks path 3 and makes bookings on its way.

The intention information at the vehicle agent - the booking - evaporates. Evaporation of bookings reflects truth maintenance of the intention information in the environment. Vehicle agents must create intention ants to refresh their intention at a frequency that is sufficiently high to maintain their bookings at the road elements. While refreshing, a vehicle agent observes the evolution of the expected traffic

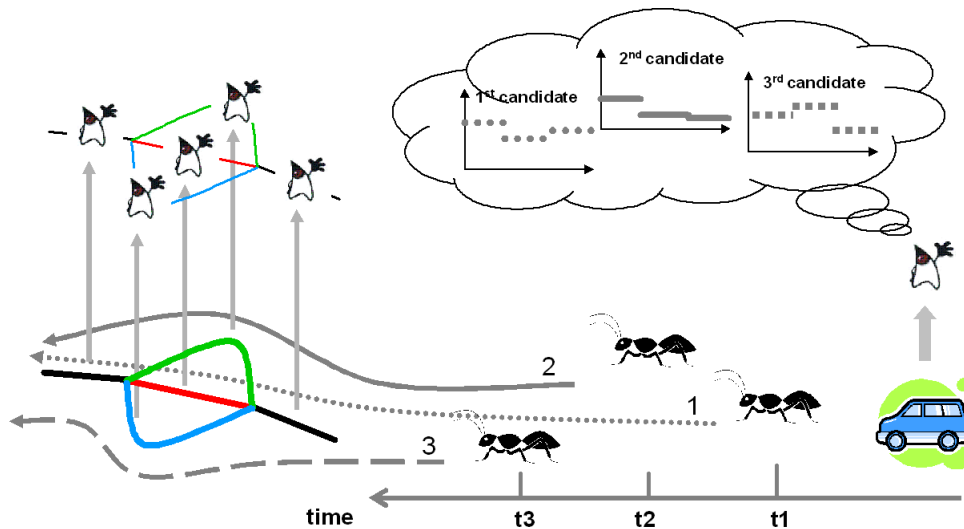


Fig. 2. Exploration ants, issued by a vehicle agent, scout feasible paths by roaming the graph environment (simplification of the scenario in Fig. 1)

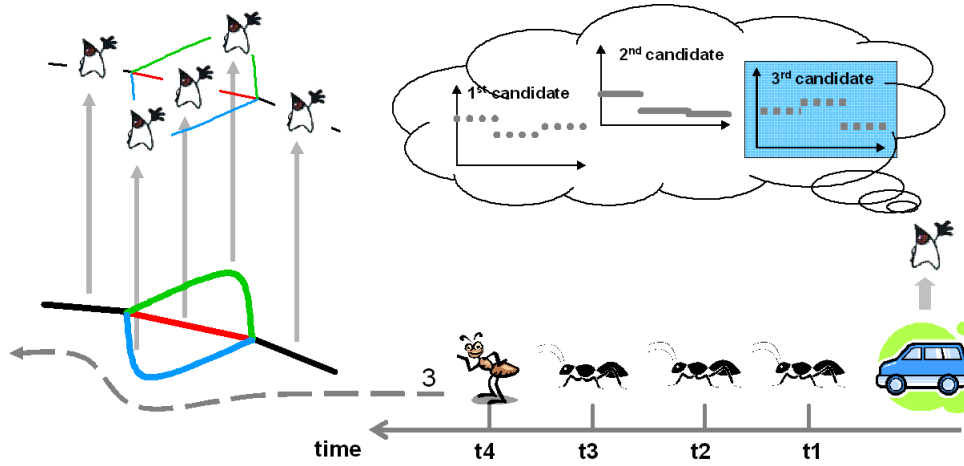


Fig. 3. Intention ant, issued by a vehicle agent at t4, communicate the intention of the vehicle agent through the environment

conditions of its current intentions through the reports on the estimated waiting times that intention ants bring back. This estimate is compared to the estimates of the candidate solutions that are found and refreshed by the exploration agents. When the quality of the estimation of the current intention drops significantly below the qualities of other candidate solutions, the vehicle agent may revise its intention.

3) *Divide and Conquer*: As a consequence of the intention propagation process, the actual intention of a vehicle agent, as it is distributed to the different infrastructure agents, is only the path that the vehicle agent intends to follow. This intention is then aligned with the schedules of the involved road elements. The vehicle agent decides on the path to follow in an intention, the infrastructure agents and the virtual environment decide on the resulting schedule, which may or may not correspond with the beliefs of the vehicle agent based on the information from the exploration

ants. As such, this process relieves the vehicle agent from massive communication using complex protocols to ensure e.g. a two phase commit for reserving all road elements.

### III. PROTOTYPICAL APPLICATION

To evaluate the approach of delegate MAS for anticipatory vehicle routing, we have applied the approach in a simple prototypical application. We simulated the traffic scenario of the Leonard crossroad near Brussels, shown in Fig. 1. The traffic network consists of 195 road segments and 4 main crossroads, i.e. *A*, *B*, *C* and *D*. We simulated a morning peak scenario where vehicles travel from south to north using the central motorway. We focus on the flow of vehicles that travel from point  $S_1$  in the south to point  $N$  in the north. These vehicles typically enter the central motorway at intersection *B*. However, in a morning peak scenario, this typically results in congestion near intersection *B*, due to a peak load of traffic on the central motorway that originates from point  $S_2$ . During a simulation run, 500 vehicles are

injected into the road network at the main entry points  $S_1$  and  $S_2$ .

Here is a high-level description of the behavior of the vehicle agents:

---

```
//BEHAVIOR VEHICLE AGENT

getAlternatives():
  //Gather feasible routes from the vehicle's
  //current location to the destination
  routes = getFeasibleRoutes(currentLoc, dest);

explore():
  //Explore the set of possible routes
  foreach(route : routes)
    ants[i] = new ExplorationAnt(route);
    ants[i].explore()
  end
  //Each ant records the cost of each route
  //The cost is defined as:
  //cost = w_dist*distance + w_time*timeWaited
  bestRoute = minimalCost(ants);

decide():
  //Change intention if best route is 'delta'
  //better as current intention; i.e. delta
  //expresses the sensitivity of intention
  //revision
  if(reviseIntention
    (bestRoute, currentRoute, delta))
    currentRoute = bestRoute;
  end
  //Book the road segments
  ant = new IntentionAnt(currentRoute);
  ant.bookRoute();

act():
  //Move the vehicle
  vehicle.move(currentLoc, dest, currentRoute);
```

---

In each simulation step, the route reservations of each road element evaporate according to a predefined evaporation rate:

---

```
//EVAPORATION RESERVATIONS ROAD ELEMENT

decayReservations():
  //evaporate_rate: determines how quickly
  //reservations decay
  foreach(reservation : reservations)
    reservation[i].evaporate(evaporate_rate);
  end
```

---

The parameters  $w\_dist$  and  $w\_time$  allow to tune the preferences of the driver, i.e. the ration  $w\_dist$  to  $w\_time$  expresses the relative importance of the distance traveled versus the time spent waiting due to congestion. We consider 5 different settings:

- Setting 1  $w\_dist/w\_time = 1/0$ : The driver minimizes the distance traveled
- Setting 2  $w\_dist/w\_time = 1/0.5$ : The driver has a stronger preference for minimizing the distance traveled than for reducing the time waited due to congestion
- Setting 3  $w\_dist/w\_time = 1/1$ : The driver balances the importance of distance traveled and time waited due to

congestion

- Setting 4  $w\_dist/w\_time = 0.5/1$ : The driver has a stronger preference for reducing the time waited due to congestion than for minimizing the distance traveled
- Setting 5  $w\_dist/w\_time = 1/0$ : The driver minimizes the time waited due to congestion

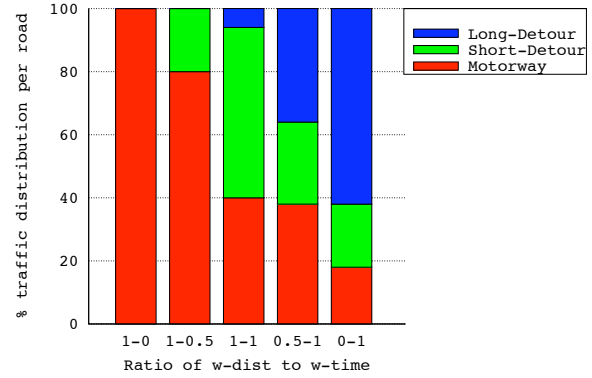


Fig. 4. Traffic distribution on alternative roads for different ratios of  $w\_dist$  to  $w\_time$  (traffic flow from  $S_1$  in scenario Fig. 1);  $\delta = 1$  and  $evaporation\_rate = 1$

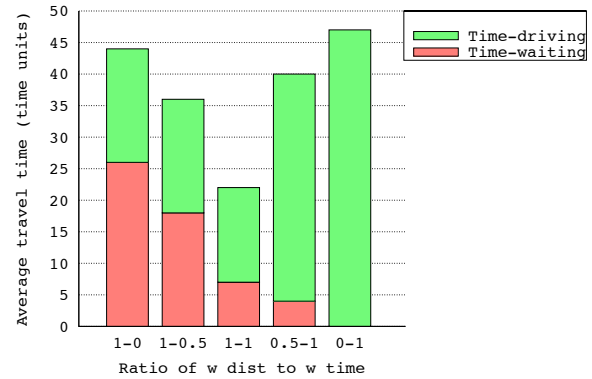


Fig. 5. Average travel time for different ratios of  $w\_dist$  to  $w\_time$  (traffic flow from  $S_1$  in scenario Fig. 1);  $\delta = 1$  and  $evaporation\_rate = 1$

We performed experiments with each of the 5 settings. The results are shown in Fig. 4 and Fig. 5.

Fig. 4 shows for each of the five settings how the traffic from  $S_1$  is distributed across the three alternative routes, i.e. the central motorway, the short detour and the long detour. The figure shows that the more a vehicle prefers minimizing the distance traveled over minimizing the time spent in congestion, the more the central motorway is used. On the other hand, the more a vehicle prefers minimizing the time spent in congestion over minimizing the distance traveled, the more (longer) detours are used.

Fig. 5 shows the average travel time for vehicles from  $S_1$  for each of the five settings, as well as the average proportion of the travel time the vehicles spent waiting in congestion.

The figure shows that setting 3, which balances minimizing distance and waiting time, yields better results than the other settings. Settings 1 and 5 perform worst with respect to the travel time. Setting 1 neglects waiting time in congestion and consequently, a large proportion of its travel time comprises time waiting in congestion. On the other hand, setting 5 neglects travel distance and as a consequence, long detours result in a large travel time.

#### IV. RELATED WORK

Dynamic vehicle routing is an extensively studied field of research [13]. In this paper, we focus the discussion of related work on a number of representative agent-based approaches used for vehicle routing.

[14] proposes a MAS-based transportation simulation where travelers are represented by agents that make independent decisions about their actions. The architecture is composed of a route planner that generates travel plans, a micro-simulation that executes plans simultaneously and evaluates congestion, and a feedback module for resolving inter-dependencies. Strategy evaluation is based on event-based communication. Example events are an agent that changes an activity, a vehicle that enters or leaves a link, etc. The main differences between [14] and anticipatory routing with delegate MAS are: (1) the interaction between agents in [14] is based on events while in the delegate MAS approach vehicle agents coordinate by means of delegate MAS that interact via the environment; and (2) the behavior of agents in [14] is based on explicit plans while vehicle agents with delegate MAS adapt their behavior dynamically with changing circumstances in the environment.

[15] proposes a reservation-based system for alleviating traffic congestion, specifically at intersections, in which vehicles are fully controlled by agents. The approach is based on vehicle agents requesting and receiving time slots from the intersection agents during which they may pass. The intersection is divided into a grid of reservation tiles, that can only be occupied by one vehicle at a time. When a vehicle agent requests a reservation, the intersection agent simulates the journey of the vehicle through the intersection with the parameters provided. If none of the cells of the requested trajectory is reserved, the intersection agent accepts the request, otherwise, it rejects. In the latter case, the vehicle agent decelerates and tries again at the next time step. A vehicle agent can cancel its reservation if it determines that it can not meet the reservation. The authors demonstrate a drastic improvement of their approach in comparison with static traffic lights. In the approach of [15], vehicle agents and intersection agents interact directly. Contrary, in anticipatory routing with delegate MAS vehicle agents and infrastructure agents coordinate indirectly via delegate MAS. To deal with dynamics in the environment, the evaporating bookings have to be reinforced regularly. Although [15] considers a crossroad as a grid of cells, there is only one agent in charge to manage the reservation of all the cells. In vehicle routing with delegate MAS control is distributed over the different segments of the road network.

[16] studies the effects of traffic forecasts on drivers' decision making. To select routes, drivers use simple heuristics. The traffic forecast system perceives drivers' decisions and returns a forecast. This allows drivers to adapt their behavior and learn over time. The authors show that the approach yields realistic results. In the approach of [16], drivers coordinate indirectly via the aggregated traffic information. Differentiation of drivers' behavior is based on an individual probability for the selection of routes which is an abstraction of drivers preferences. In vehicle routing with delegate MAS, vehicle agents and infrastructure agents coordinate indirectly as well. However, in our case the coordination happens locally in the environment. Furthermore, our approach easily supports vehicle agents to select concrete individual preferences such as minimal distance, minimal standstill, etc.

[17] proposes a pheromone-based model for predicting traffic congestion. In this approach, vehicles are provided with an intelligent module that allows to use them as moving sensors to detect actual traffic conditions. Vehicles use three types of digital pheromones based on the sensed traffic conditions, i.e. a pheromone that indicates the recent presence of a vehicle, a breaking pheromone, and a pheromone that indicates distance between two neighboring vehicles. The pheromones are combined and integrated on a map that serves as a guidance for drivers. [17] focuses on congestion prediction, while our approach focuses on anticipatory vehicle routing to avoid congestion.

[18] introduces the hierarchical routing system (HRS). In HRS, a traffic network is split into several smaller networks (sectors) by introducing a hierarchy between the roads. At each intersection in the network (locally and at sector level), a routing table is used for guiding the cars. An ant-based algorithm is applied for dynamic routing. Ants are injected in the network according to the vehicles requests. The ants maintain the probability tables in the network. Three types of ants are used: (1) local ants maintain and evaluate local routes (i.e., within the same sector), (2) backward ants take the opposite path as a corresponding local ants and update the routing tables, (3) exploration ants are generated by routing nodes of each sector to find and maintain routes between different sectors. This way, routes are trained in advance for each car in the traffic. The approach shows high adaptability in relatively complex networks. HRS differs from anticipatory routing with delegate MAS in several manners. The ant routing algorithm with timetable updating and the route finding system are centralized whereas routing with delegate MAS is conceptually fully distributed. On the other hand, HRS applies a hierarchical approach fitting the natural structure of road networks and dealing with complexity. Anticipatory routing with delegate MAS uses a flat approach. Introducing a hierarchy may be interesting for more complex road networks.

#### V. CONCLUSION

In this paper, we presented delegate MAS for anticipatory vehicle routing. The approach associates coarse-grained

agents with entities in the domain (vehicles and road elements). These agents interact through light-weight agents that explore alternative routes and book road segments on behalf of vehicles. Two main principles underly the approach. First, control is decentralized, avoiding the bottleneck of a centralized traffic monitoring and control center. Moreover, decentralization of control reflects the inherent distribution of resources and activities in traffic. Second, we use the environment to support indirect coordination and truth maintenance, which enables the system to flexibly adapt to dynamic and changing operating conditions in traffic. We have applied the approach in a simple traffic scenario. The results show that delegate MAS are a quite promising approach for dynamic vehicle routing. Next steps in our research include: (1) applying the delegate MAS approach in advanced traffic scenarios to study scalability, different preferences, etc., (2) comparing traffic assignment of the delegate MAS with more theoretical work (e.g. dynamic user optimal traffic assignment models); (3) studying and developing a formal foundation of the delegate MAS approach.

## VI. ACKNOWLEDGMENTS

The research described in this paper is funded by the Foundation for Scientific Research in Flanders (FWO) and the IWT Strategic Basic Research Project SBO-60823.

## REFERENCES

- [1] D. Weyns and T. Holvoet, "A Reference Architecture for Situated Multiagent Systems," in *Environments for Multiagent Systems III*, ser. Lecture Notes in Computer Science, vol. 4389. Springer, 2006.
- [2] D. Weyns, A. Omicini, and J. Odell, "Environment as a First-Class Abstraction in Multiagent Systems," *Autonomous Agents and Multi-Agent Systems*, vol. 14, no. 1, pp. 5–29, 2007.
- [3] A. Helleboogh, G. Vizzari, A. Uhrmacher, and F. Michel, "Modeling Dynamic Environments in Multi-Agent Simulation," *Autonomous Agents and Multi-Agent Systems*, vol. 14, no. 1, pp. 87–116, 2007.
- [4] D. Weyns, K. Schelfhout, T. Holvoet, and T. Lefever, "Decentralized control of E'GV transportation systems," in *4th Joint Conference on Autonomous Agents and Multiagent Systems, Industry Track*. Utrecht, The Netherlands: ACM Press, 2005.
- [5] ITS America: Intelligent Transportation Society of America, <http://www.itsa.org/>.
- [6] ERTICO: Intelligent Transportation Systems for Europe, <http://www.ertico.com/>.
- [7] B. Kerner, C. Demir, R. Herrtwich, S. Klenov, H. Rehborn, M. Aleksic, and A. Haug, "Traffic state detection with floating car data in road networks," in *IEEE International Conference on Intelligent Transportation Systems*. IEEE, 2005.
- [8] The Sensor, Intelligent Transportation System Institute, <http://www.its.umn.edu/sensor/index.html>.
- [9] L. Wolf, M. Bechler, and S. Jaap, "Potentials and Technologies of Mobile Internet for Transportation," in *5th International Conference on Intelligent Transportation Systems Telecommunications, France*, 2005.
- [10] T. Holvoet and P. Valckenaers, "Exploiting the Environment for Coordinating Agent Intentions," in *Environments for Multiagent Systems III*, ser. Lecture Notes in Computer Science, vol. 4389. Springer, 2006.
- [11] H. V. D. Parunak and S. Brueckner, "Concurrent Modeling of Alternative Worlds with Polyagents," in *7th International Workshop on Multi-Agent-Based Simulation*, Hakodate, Japan, 2006.
- [12] H. V. D. Parunak, S. Brueckner, D. Weyns, T. Holvoet, P. Verstraete, and P. Valckenaers, "E Pluribus Unum: Polyagent and Delegate MAS Architectures," in *8th International Workshop on Multi-Agent-Based Simulation*, Honolulu, Hawaii, 2007.
- [13] E. Schmitt and H. Julia, "Vehicle Routing Guidance Systems: Classification and Comparison," in *6th International Conference on Intelligent Transportation Systems Telecommunications, Toronto, Canada*, 2006.
- [14] B. Raney and K. Nagel, "Iterative route planning for large-scale modular transportation simulations," *Future Gener. Comput. Syst.*, vol. 20, no. 7, pp. 1101–1118, 2004.
- [15] K. Dresner and P. Stone, "Multiagent traffic management: A reservation-based intersection control mechanism," in *3rd International Joint Conference on Autonomous Agents and Multiagent Systems, New York*. IEEE Computer Society, 2004.
- [16] F. Kluegl and A. Bazzan, "Route Decision Behaviour in a Commuting Scenario: Simple Heuristics Adaptation and Effect of Traffic Forecast," *Journal of Artificial Societies and Social Simulation*, vol. 7, no. 1, 2004, <http://jasss.soc.surrey.ac.uk/7/1/1.html>.
- [17] Y. Ando, Y. Fukazawa, O. Masutani, H. Iwasaki, and S. Honiden, "Performance of pheromone model for predicting traffic congestion," in *5th International Joint Conference on Autonomous agents and Multiagent Systems*. New York: ACM Press, 2006.
- [18] B. Tatpmir and L. Rothkrantz, "Hierarchical Routing in Traffic Using Swarm-Intelligence," in *6th International Conference on Intelligent Transportation Systems Telecommunications, Toronto, Canada*, 2006.