Optimizing vehicle trips using agent negotiation through a traffic matrix

Luis Cruz-Piris,
Diego Rivera,
Ivan Marsa-Maestre
and Enrique de la Hoz
Computer Engineering Department
University of Alcalá
{luis.cruz,diego.rivera,ivan.marsa,enrique.delahoz}@uah.es

ABSTRACT

Multi-Agent Systems (MAS) have been proved to be an effective tool to improve the efficiency of intelligent traffic control systems (using them in traffic lights scheduling, vehicle routing, etc.). Negotiation is one of the most used techniques to improve the global goal of a system while maintaining as much as possible each agent preferences. In this paper, we propose modeling the traffic characteristics produced in a transportation network during a given time interval as a traffic matrix, and then use it as core element in an agent negotiation system. We define how this matrix is generated, and the mechanisms to populate it and update it. Using the traffic matrix data, we propose to use two different selection methods to obtain a subset of agents with voting rights. The first method is based on the average speed of network edges and the second one is based on the traffic jam length for a vehicle. This subset will perform a negotiation process where other agents in the network will be proposed to block their departure or modify their departure time, to mitigate the congestion in the network edges, and as goal, reducing trip duration.

Keywords

Multi-agent systems, Negotiation, Traffic matrix

1. INTRODUCTION

Solving the vehicle traffic congestion in big cities is nowadays one of the biggest challenges for modern societies. In the report "2015 urban mobility scorecard" by Schrank *et al.* [14] it is quantified the annual cost of this problem in the USA: "travel delays due to traffic congestion caused drivers to waste more than 3 billion gallons of fuel and kept travelers stuck in their cars for nearly 7 billion extra hours - 42 hours per rush-hour commuter. The total nationwide price tag: $160 billion, or $960 per commuter".

The intelligent traffic management systems aim to have a global overview of the problem so they can make the right decisions in each case. The Multi-Agent Systems (MAS) help to the decision-making process in distributed systems, such as the traffic scenarios (pedestrians, vehicles, traffic lights or signals, etc.) [3].

Our proposal is modeling a traffic scenario as a three-dimensional data structure we call traffic matrix and use as a tool for the negotiation between the agents. The traffic matrix contains all the necessary information to characterize a traffic scenario. We divide the scenario in time steps and edges, and store the average speed of vehicles located in each step and edge. The operation of our MAS consists of selecting a subset of the agents that fulfill certain characteristics. We have defined two selection methods based on the number of edge-time step tuples that are below a threshold and based on the meters of a route that can be considered as a traffic jam (also using a threshold value) respectively. A negotiation process is started for this subset of agents, where the agents can make two different proposals to other agents: blocking the departure of the trip or changing the departure time.

To validate our proposal, we have used a microscopic traffic simulator and a realistic simulation scenario. For each combination of selection method and proposed actions, we have carried out a set of tests that provide us enough significant results to prove the feasibility of the proposal.

This paper is organized as follows. Section 2 presents the use of multi-agent systems in traffic related environments. Section 3 defines the structure and creation of the traffic matrix, the proposed system operation, the selection methods and the proposed actions for the agents to perform during the negotiation process. Section 4 establishes the simulation scenario to be used in the validation of the proposal. Sections 5 and 6 define the experiments that have been carried out, the obtained results and a discussion about them. Finally, section 7 summarizes our contributions and presents future related research lines.

2. MAS AND NEGOTIATION IN TRAFFIC ENVIRONMENTS

The use of MAS for traffic management is a widely studied area since the 1980s [3, 4, 7, 8].

In the work of Adler and Blue [1], they determine that "vehicles may be modeled as mobile agents that move between regions and whose control is handed off from one agent community to another, akin to an air traffic control system". The authors propose a solution based on the cooperation of agents using principled negotiation.

In some works, the MAS operate by variating the properties of the traffic to affect in an indirect way the traffic related problems. Balaji *et al.* [2] propose obtain data collected by sensors in intersections using agents to variate the green time of semaphores. The authors validate their results with simulations in the PARAMICS software.

Other works model all traffic scenario elements as agents and focus on the coordination methods between them [16, 10, 5].

In [17], Wang reviews some of the agent-based control commercial technologies such as CRONOS, OPAC, SCOOT, SCAT, PRODYN, or RHODES and concludes that, although they prove to be useful and successful for many traffic management problems, costs and difficulties associated with their development, operation, maintenance, expansion, and upgrading are too high and should be improved.

Negotiation is a widely-used technique in MAS, and it is also used in agent-based traffic management environments. For instance, Desai *et al.* [6] propose the methodology and implementation of CARAVAN, a system to disperse vehicles in a traffic jam situation using virtual negotiation techniques, considering each vehicle an agent and using intervehicular communications (IVC). Also, in the work of Lopes *et al.* [13] a generic model of negotiation for autonomous agents is presented. This work is the description of the implementation and validation of the work previously presented by the same authors in [11] [12], where the negotiation model is described, and experiments to assess the feasibility of the proposal are conducted. The agents described are equipped with a simplified version of the model that handles two-party, single-issue negotiation. The results confirmed a few basic conclusions about human negotiation.

3. AGENTS NEGOTIATION THROUGH A TRAFFIC MATRIX

Our proposal is to apply a MAS to a traffic management environment, where each vehicle is considered an agent. We define and use what we call a “traffic matrix” that will play a paramount role in the agent negotiation task. This matrix will contain the average speed values of the vehicles that circulate through a given edge at a given time interval. Figure 1 shows a representation of this data structure. For each one of the network edges, we store a list of the average speed values that are in that edge in a time interval (by default, the matrix is generated using 180s as time interval).

Additional to the average speed values, the traffic matrix can store the first and second position of each edge list, the length of the edge and the maximum speed allowed in that edge. This information allows to use the matrix as unique input data structure in our proposal.

Our proposal defines two agent selection methods based on this type of information:

1. **Average speed**: For each trip, this method reproduces its route from its depart time. To achieve this, it performs requests to the traffic matrix. If the requested tuple (edge, time step) has a value lower than the one set by a predefined threshold, the weight of the trip is increased in 1. The request returns a numeric value that represents the number of cells from the matrix that fulfill this condition. As the matrix is divided in regular time intervals, a higher result represents a high jam value, and allow us to compare it with the values for other agents in the same conditions.

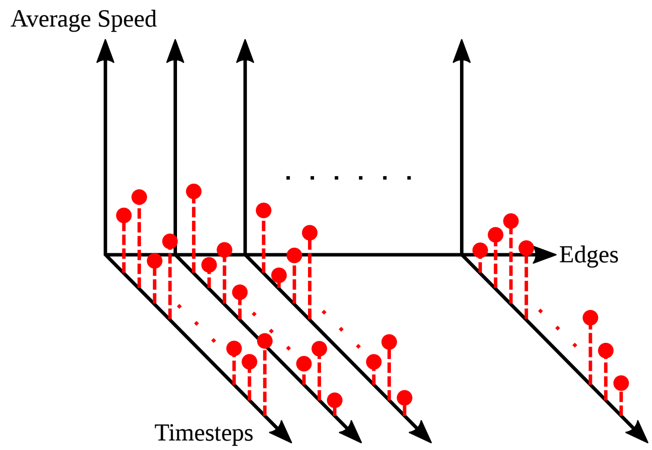

Figure 1: Representation of the proposed Traffic Matrix

2. **Traffic jam length**: For each trip, this method reproduces its route from its depart time. With the obtained average speed value in each request to the matrix and the associated time, the number of meters the vehicle has been in a jam state is returned.

Furthermore, our proposal defines two different actions to be proposed to the agents:

1. **Block departure**: In this action, the agent is proposed to block its departure until it receives a message indicating otherwise. The agent can decide if he accepts or decline the proposal with a certain probability.

2. **Change departure time**: In this action, the agent is proposed to vary its departure time. The agent can decide if she accepts or decline the proposal with a certain probability.

In the algorithm 1 it is shown a pseudo-code of the operation of our proposal.

Our proposal uses as input a set of agents and the traffic matrix (obtained from historic traffic data or a previous simulation). Besides, we need to indicate the algorithm parameters such as the selection mode, the threshold value and the proposed action.

The output of the algorithm (S) is a list of agreements achieved after negotiation. With this list, the trips will be updated.

- In line 1, we define the ACCEPT_PROB system variable, which is used by the agents to accept or decline the proposed actions. In our case, we have set this variable to 0.5, which means that an agent has a 50% probability of accepting or declining an action. In a negotiation scenario, a set of incentives to accept or decline the offers the agents receive must exist. The ACCEPT_PROB value represents the probability for an agent to accept an offer, once the incentives have been evaluated.

- In line 2, the evaluate function is called. It generates a list of agents that will have the right to vote by calling internally to the chosen selection method.

Algorithm 1: Vehicle trips optimization

```
Input: A: set of agents (vehicles+trips)
       M: traffic matrix
       SM: Selection mode (speed or length)
       T: Threshold value
       PA: Proposed action (block or change)
Output: S: list of agents agreements
1  ACCEPT_PROB = 0.5
2  list_of_voters = evaluate(M,SM,T)
3  for agent in list_of_voters do
4      candidate = select_candidate(list_of_voters)
5      if PA=="block" then
6          sendProposal(agent,candidate,"block")
7      else if PA=="change" then
8          propTime = obtainDepart()
9          sendProposal(agent,candidate,"change",propTime)
10     else
11         pass
12     end
13     S = update(A,agent)
14 end
```

- For each agent in the list of voters, another agent is selected as candidate (line 4) to receive a proposed action. Both voter and candidate are in the same jam situation. If the proposed action is change departure time, a new departure time proposal is generated by calling the *obtainDepart()* function (this function returns a value ±15% value over the original departure time).
- In line 13, the original set of agents is updated with the results of the negotiation process.

4. PROPOSAL VALIDATION

A traffic scenario is basically composed by a network (a set of edges with a set of specific characteristics) and one or more vehicles (each one with its own characteristics or specific behavior) that make a trip through the network. To validate our proposal, we need to choose the tools that allow us to simulate the behavior of this scenario in the most realistic possible way. We have chosen the "Simulation of Urban MObility" (SUMO) [9] to perform the simulation task.

4.1 Simulation scenario

Using a realistic simulation scenario is essential to perform the validation of the system, as it will provide conditions similar to the real-life scenarios.

In our proposal we have chosen the scenario called "TAPAS Cologne"[15] that is referenced in the SUMO documentation. It is a complete simulation scenario of the German city of Cologne. It was created by the Institute of Transportation Systems at the German Aerospace Center (TIS-DLR), and its goal is to reproduce, with the maximum possible realism, the urban traffic of Cologne. It defines a map of 400 km^2 and 24 hours of traffic.

The original simulation scenario is composed by a road network with 68642 edges, 30354 nodes and 1547333 routes. The size of this scenario causes very high simulation times. Therefore, for the validation of our proposal, we have decided to crop the scenario in a smaller portion (lat-lon: 6.896258, 50.967819, 6.936623, 50.942802) that, still being representative of the original scenario, will yield lower simulation times.

The cropped area is shown in Figure 2. This sub-scenario has 1416 edges and 716 nodes. Equally, the routes of the scenario have been reduced to a more manageable number, using just the routes that are related to the chosen portion of the scenario. The total selected routes are 246374.

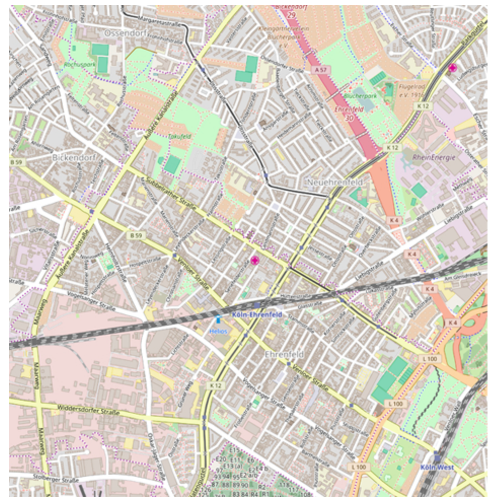

Figure 2: OpenStreetMap capture of the portion of the scenario used in the system validation.

Moreover, in the scenario documentation it is said that for a proper simulation, the parameter *scale* must be set at 0.3. This means that only the 30% of the routes will be inserted during the simulation. For our sub-scenario, this scale value must be calibrated again. We have done this calibration by executing consecutive simulations increasing the value of *scale* in 0.01 for each new simulation. For each simulation, the number of teleports have been measured[1]. The result of these simulations is shown in Figure 3.

Using the results of these tests, we have decided to increase the value of the parameter *scale* to 0.40, because at this value is where we detect that the number of teleports starts to raise (although it is still a reasonable low value: 241), and allows us to use a high volume of traffic. Using this value for *scale* means that the actual number of routes inserted during the simulation is 98550.

SUMO offers multiple output formats for the simulation results. For the sake of clarity of these results we have chosen to group them in two types: the occupation of each edge in each time interval (180s each interval) using the traffic matrix structure, and trip duration at the end of the simulation.

Applying the traffic matrix to this simulation scenario, we have obtained the results shown in Figure 4. In the horizontal plane, the figure represents the simulation steps in seconds and the network edges. To make the figure clearer, the edges have been sorted from higher to lower occupation values. In the vertical axis, the occupation values of

[1]A teleport is an event that happens in SUMO simulations when vehicles are blocked for a given time. It consists on the automatic disappearing and appearing of the vehicle, in order to unlock the simulation

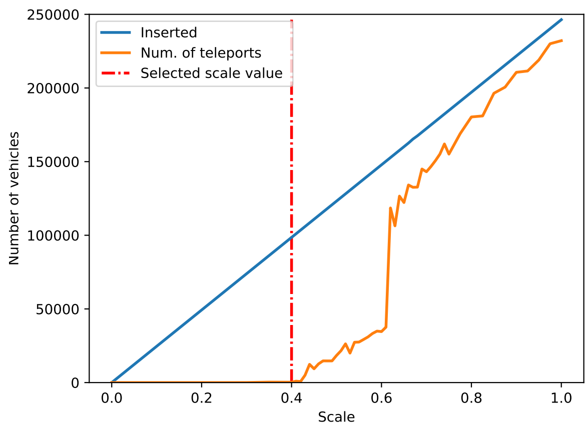

Figure 3: Evolution of the teleports number per the scale parameter.

each step and edge are shown. This representation of the results shows in an intuitive way that there are two time steps around which the occupation is especially higher than the average (around 25000 and 60000 seconds).

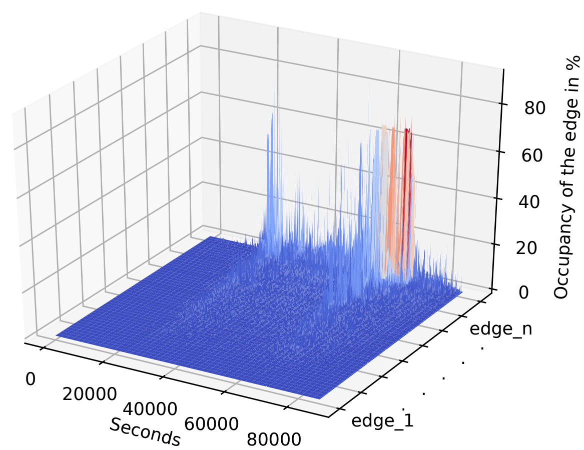

Figure 4: Traffic matrix simulation values

The trip duration at the end of simulations will be used to quantify the possible improvement of the simulation results once we apply our proposal and compared with the simulation scenario.

5. EXPERIMENT SETTINGS

To validate our proposal, we have defined two sets of tests: the first one will allow us to find a threshold value used in the *evaluate()* function. The second one is designed to test intensively the different variations of the proposed algorithm once the threshold value is set. These experiments are:

- **"Threshold" experiment**: Through the traffic matrix we can obtain information such as average speed in an edge in each time step. Using the functions defined in section 3, we must obtain the threshold value that, in the worst case scenario (using "block departure" proposed action), won't modify significantly the original simulation scenario. We have set the maximum number of blocked trips to the 10% of the total trips.

  For each one of the selection methods of voting right, we have carried out an extensive series of simulations, varying the threshold value from 0.1% to 30% with a 0.1 step between values.

- **"Evaluation" experiment**: With the threshold value obtained in the previous experiment, we have carried out another set of tests. For each possible combination of the selection methods and proposed actions, we have performed 100 simulations, measuring in each one the variations of the trip durations.

  Given that the original scenario has been cropped for this paper, there are a small number of trips which duration is extremely low. These trips can produce false positive or false negative results during the evaluation process. This is because any small variation in the scenario provokes a very high variation in these types of trips. To minimize this issue, in the result evaluation we don't have used the values of the trips which duration is lower than 200 seconds. Similarly, the vehicles that have suffered the teleport phenomena during the simulation have also been discarded in the evaluation process.

6. EXPERIMENT RESULTS

In this section we analyze the obtained results after carrying out the experiments defined in section 5.

In figure 5, we show the results of the "threshold" experiment. In the horizontal axis, the threshold values from 0 and 0.25 are shown. The number of vehicles inserted in the simulation are represented in the vertical axis. We can see how the values obtained for the selection method based on the average speed decrease softer than the values obtained for the selection method based on traffic jam length. This is because high-length edges with higher occupation are counted as 1 in the first method, but are quantified as its total length value in the second method.

In section 5 we established a 10% value for the maximum of blocked trips (green horizontal line in the figure). Using this limitation, we have selected as threshold value the intersection between this line and the line representing the traffic jam length selection method. This value is 0.04 (red vertical line).

The second set of tests ("evaluation" experiment), has offered the results shown in Table 1. The results have been grouped in three categories depending on the trip duration compared with the original scenario: smaller, equal or greater. For each selection method and proposed action, the table shows their mean (μ) and standard deviation (σ).

The results show that a high percentage of trips have seen its duration after applying our proposed algorithm. Although the algorithm presents a certain variation in the decisions taken by the agents in each simulation, the low standard deviation values suggest that the result is consistent.

Comparing the obtained values for each selection method, the average speed method presents the best results. This is because the metric based on the average speed method is

Table 1: Percentage of vehicles whose trip duration is smaller, equal and greater than its original trip duration

		Smaller		***Equal***		***Greater***	
Selection method (voting rights)	***Proposed action***	μ	σ	μ	σ	μ	σ
Average Speed	Block departure	68.67	1.12	19.38	0.51	11.95	0.67
	Change departure time	70.51	1.61	18.80	0.70	10.69	0.93
Traffic jam Length	Block departure	48.50	0.99	28.19	0.44	23.31	0.72
	Change departure time	46.68	0.89	29.05	0.35	24.27	0.79

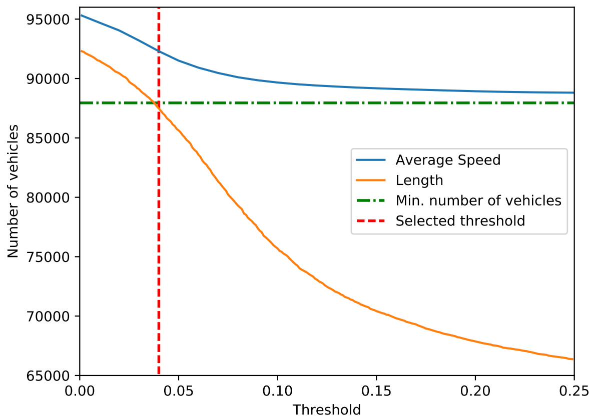

Figure 5: Results of the “threshold” experiment

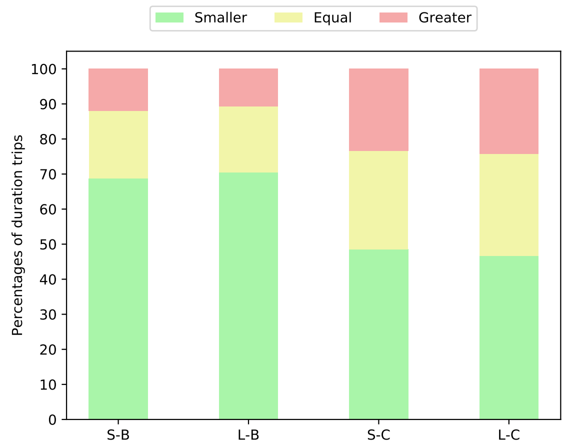

Figure 6: Representation of the average values for smaller, equal and greater trip durations

more realistic than the one based on the traffic jam length method (a long jammed edge penalizes too much the results of the second case).

If we focus on the proposed actions, it is possible to see how the difference between both types of action is lower than the 2%. It is important to note that the block departure proposed action erase from the simulation almost the 10% of the total vehicles. Therefore, the action of changing the departure time is a better option, because it achieves similar results without altering the simulation scenario.

The average values from Table 1 are graphically shown in Figure 6. For each combination of selection method (S: average speed, L: traffic jam length) and proposed action (B: block departure, C: change departure time) we show a bar in the figure. Green bars with slashes represent the percentage (y axis) of trips that reduce its duration, yellow bars represent those that stay equal and red bars with backslashes represent the trips that have an increased duration.

7. CONCLUSIONS

In this paper, we have proposed using a voting-based negotiation agent system to optimize the vehicle trip durations. The main component of our proposal is modeling the traffic data as a matrix and use it as evaluation method in the negotiation process. After selecting a realistic traffic simulation scenario, we have designed, implemented and validated the proposal using it.

After analyzing the results of the validation tests, we can conclude that the proposed solution is feasible and it significantly improves the duration of the most part of vehicles from the simulation scenario, where more than the 75% of the trip durations are lower (almost the 50% in the worst case) or equal to the original durations.

Although the conducted experiments yield satisfactory results, there are some avenues for further research in this topic. For instance, it could offer interesting results to define a method for selecting a depart time based on the global system considerations and not only in the agent point of view.

To demonstrate the validity of our proposal, the traffic matrix has been generated using data from a simulation in off-line mode. If the traffic matrix is generated using historical traffic data, it could be updated in real-time by the agents.

Another future work line would be to research about the possible incentives that would allow to obtain a proposal acceptance rate close to the probability value desired.

It would be also interesting to research in the departure time proposal mechanism when using the “change departure time proposed action”, so it would be possible to consider other factors to find the time instants that suppose a lower impact in the agent set, instead of changing the departure time in a $\pm 15\%$.

Finally, other combinations of selection methods and proposed actions could be proposed and tested to find possible better results.

Acknowledgments

This work has been supported by the Spanish Ministry of Economy and Competitiveness grants TIN2016-80622-P, TIN 2014-61627-EXP and TEC2013-45183-R, and by the University of Alcalá through CCG2016/EXP-048.

REFERENCES

[1] J. L. Adler and V. J. Blue. A cooperative multi-agent transportation management and route guidance system. *Transportation Research Part C: Emerging Technologies*, 10(5-6):433 – 454, 2002.

[2] P. G. Balaji, X. German, and D. Srinivasan. Urban traffic signal control using reinforcement learning agents. *IET Intelligent Transport Systems*, 4(3):177–188, September 2010.

[3] A. L. C. Bazzan and F. Klugl. A review on agent-based technology for traffic and transportation. *The Knowledge Engineering Review*, 29(3):375–403, 006 2014.

[4] B. Chen and H. H. Cheng. A review of the applications of agent technology in traffic and transportation systems. *IEEE Transactions on Intelligent Transportation Systems*, 11(2):485–497, 2010.

[5] Z. Dan, L. Cai, and L. Zheng. Improved multi-agent system for the vehicle routing problem with time windows. *Tsinghua Science & Technology*, 14(3):407 – 412, 2009.

[6] P. Desai, S. W. Loke, A. Desai, and J. Singh. Caravan: Congestion avoidance and route allocation using virtual agent negotiation. *IEEE Transactions on Intelligent Transportation Systems*, 14(3):1197–1207, 2013.

[7] J. France and A. A. Ghorbani. A multiagent system for optimizing urban traffic. In *Intelligent Agent Technology, 2003. IAT 2003. IEEE/WIC International Conference on*, pages 411–414. IEEE, 2003.

[8] J. Z. Hernández, S. Ossowski, and A. García-Serrano. Multiagent architectures for intelligent traffic management systems. *Transportation Research Part C: Emerging Technologies*, 10(5-6):473 – 506, 2002.

[9] 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*, 5(3&4):128–138, December 2012.

[10] F. Logi and S. G. Ritchie. A multi-agent architecture for cooperative inter-jurisdictional traffic congestion management. *Transportation Research Part C: Emerging Technologies*, 10(5):507–527, 2002.

[11] F. Lopes, N. Mamede, A. Q. Novais, and H. Coelho. A negotiation model for autonomous computational agents: Formal description and empirical evaluation. *Journal of Intelligent & Fuzzy Systems*, 12(3, 4):195–212, 2002.

[12] F. Lopes, N. Mamede, A. Q. Novais, and H. Coelho. Negotiation strategies for autonomous computational agents. In *Proceedings of the 16th European Conference on Artificial Intelligence*, pages 33–37. IOS Press, 2004.

[13] F. Lopes, A. Q. Novais, N. Mamede, and H. Coelho. Negotiation among autonomous agents: experimental evaluation of integrative strategies. In *Artificial intelligence, 2005. epia 2005. portuguese conference on*, pages 280–288. IEEE, 2005.

[14] D. Schrank, B. Eisele, T. Lomax, and J. Bak. 2015 urban mobility scorecard. 2015.

[15] S. Uppoor and M. Fiore. A large-scale vehicular mobility dataset of the Cologne urban area. In *14èmes Rencontres Francophones sur les Aspects Algorithmiques des Télécommunications (AlgoTel)*, pages 1–4, Hérault, France, France, May 2012.

[16] R. van Katwijk and P. van Koningsbruggen. Coordination of traffic management instruments using agent technology. *Transportation Research Part C: Emerging Technologies*, 10(5):455–471, 2002.

[17] F.-Y. Wang. Agent-based control for networked traffic management systems. *IEEE Intelligent Systems*, 20(5):92–96, 2005.