Management of Urban Parking: An Agent-Based Approach

Nesrine Bessghaier[1,2], Mahdi Zargayouna[1], and Flavien Balbo[1,2]

[1] Université Paris-Est, IFSTTAR, GRETTIA, Noisy le Grand, France
hamza-mahdi.zargayouna@ifsttar.fr
[2] Université Paris Dauphine, Laboratoire Lamsade-CNRS, Paris, France
{nesrine.bessghaier,balbo}@lamsade.dauphine.fr

Abstract. In the context of road urban traffic management, the problem of parking spots search is a major issue because of its serious economic and ecological fallout. In this paper, we propose a multi-agent system that aims to decrease, for private vehicles drivers, the parking spots search time. In the system that we propose, a community of drivers shares information about spots availability. Our solution has been tested following different configurations. The first results show a decrease in parking spots search time.

1 Introduction

The growing density of urban populations and the complexity of cities infrastructure are challenges that our societies must face. In the domain of transportation, the management of urban traffic growth is one of the important issues. For instance, inside the European union, the congestion cost represents several billions of Euros, and 1% of the Union GDP. Besides the financial consequences, transport has serious ecological fallouts since it is responsible for a large part of carbonic gas emission. The pollution - including noise - generated by traffic has become a serious obstacle to the quality of life and to the health of urban populations. An important part of these emissions is due to “search traffic”, that is the traffic generated by drivers looking for parking spots. A study [8] performed in the French cities of Grenoble, Lyon and Paris in 2005 reveals that, in average 10 % of the moving vehicles at a moment are looking for a parking and until 60% in the small parisian streets. This search traffic leads to 10 millions hours lost per year in France. The design of an intelligent transport system (ITS) dedicated to the management of parking spots should allow to limit the cost of this useless activity. An ITS for urban parking is different from those designed for the management of parking in reserved spots (car park, road infrastructure, etc.). The issue here is to adapt to a complex problem necessitating the consideration of a dynamic and open environment. The solution provided to this problem has to use minimal information on a shared, volatile and uncontrollable resource. The available information is minimal because, without a dedicated infrastructure for parking, there is no information resource about the availability of spots. In addition, the size of the vehicles and the space between them are characteristics that condition locally the number of available spots in a given space. Since the context is urban parking, the resource is shared and accessible to everyone

A. Ramsay and G. Agre (Eds.): AIMSA 2012, LNAI 7557, pp. 276–285, 2012.

in an uncontrolled manner. The availability of a spot is volatile and depends totally of the activity of the transport network. It is then necessary to provide a solution in which the information management adapts to the characteristics of this resource. Thus, an ITS has to be able to function without initial information and ensure to its users to have an information that is the most up-to-date possible. Finally, in an important agglomeration, the management of the volatility of this information might need important information flows. Indeed, in order to have up-to-date information of the parking evolution, we have to update the data for each vehicle that leaves or takes a spot.

In this paper, we propose an agent-based transport information system that helps to find parking spots in an urban agglomeration. The multi-agent paradigm facilitates an approach by analogy in the transportation domain, one of the objectives of which is the coordination of distributed entities to insure the safety and efficiency of transport systems. This is why the multi-agent approach is often chosen to model, solve and/or simulate transportation problems. This approach is particularly relevant for the management of parking spots, since the problem is indeed to take into account human behaviors that interact in a complex, dynamic and open environment. We propose a MAS that facilitates the information sharing relative to parking for a community of drivers. In our system, agents communicate to exchange information concerning the parking spots availability. The information comes from the users and their collaboration has to ensure an information of good quality.

There exists commercial transport applications that use an information sharing between users, for the localization of mobile speed cameras for instance. These systems use information which volatility is not very high and for which a centralized architecture could be adequate. In our case, we choose a totally decentralized architecture with an inter-vehicular communication (V2V) to allow vehicles to receive and broadcast information to the other vehicles of the same community. This communication support opens new perspectives in terms of services and has been used for instance to improve road safety [15] and the management of traffic lights [5]. Our solution supports a collaborative process grounded on mutual awareness. Each agent receives the information within its reach and participates to its update. The local information processing avoids centralization, which is costly and makes the scalability more difficult.

The remainder of this paper is organized as follows. In section 2, we describe our multi-agent model. We describe our simulations setup and report our results in section 3. We discuss related work in section 4 before to conclude and describe the perspectives of this work.

2 System Design

2.1 Agents Model

Our system for the search of spots in an urban area is modeled by a type of agent designated by *assistant* agent. An assistant agent assists one driver of a vehicle by helping him choosing among the available spots that suits him the most. The internal architecture of the assistant agent is composed of three modules: a *Communication* module, an *Itinerary* module and a *Decision* module. The first module enables the agent to communicate with its neighbors in the community. This communication is based on messages and allows to exchange information about the availability of parking spots. The itinerary module ensures the calculation of the route to a particular parking spot starting from the driver's current position, but also monitors its movement. Finally, the decision module takes care of the decision making. This module proposes a parking spot to the driver. The latter must meet the criteria specified by the driver, which may concern for instance its distance, the time since its release, or the safety of its location. In addition to proposing parking spots, the decision module manages a memory containing information related to the spots. This knowledge evolves over time with information acquired through the exchange of messages with different assistant agents and to the perception of the agent. This memory is composed of two disjoint lists:

- The FS list (for Free Spot): It is a set of pairs $\{< spot, time >\}$, where each one refers to a specific spot: its geographic position and the moment since which it was released.
- The OS List (for Occupied Spots) : It contains the spots that were in FS but which turned out to be occupied with the moment since which this information was known.

Both lists are exchanged by the assistant agents and are updated gradually by the knowledge of each one. Their combined use provides a dynamic update of the system information. Indeed, one consequence of the volatility of information regarding the availability of spots is illustrated when an agent chooses a spot on its list FS - supposed to be free - but, once there, it finds it occupied. In this case, the FS lists contain incorrect information about this spot. In the absence of a centralized component that would filter this information from the memory of agents, we need a mechanism to limit the spread of information. Thus, the OS list enables agents to filter the information received and to have the best information possible. Besides, to allow an update of the list without specific information, the decision module of each assistant agent shall filter outdated information after a time θ, i.e the spots in FS and OS with an associated time that is inferior to the current time minus θ. This parameter takes into account the network activity. Thus, a low value reflects a high volatility as the case may be in rush hours in downtown, while a high value keeps a longer sharing of information and reflects, for instance, the lower volatility in a residential area.

The assistant agent passes through four states as indicated by the automata of Fig 1:

- state 0: The agent is inactive, he does not participate in the community (that means that the vehicle is parked).

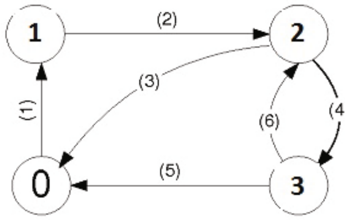

Fig. 1. Assistant agent state diagram

- state 1: The agent is active, he participates in the community, while the driver is moving (without looking for a spot).
- state 2: The agent looks for a parking spot to propose it to the driver of the concerned vehicle. Meanwhile, the driver looks for a spot on its own.
- state 3: The agent has proposed a spot and the corresponding itinerary to the driver. He stays aware of possible spots alternatives which would be more suitable than the proposed one. Meanwhile, the driver moves towards the proposed spot.

Starting from state 0, the assistant agent goes to state 1 when the driver releases a parking spot(arc (1) in Fig. 1). When the driver is near his destination, the assistant agent switches to state 2 (arc (2)) and starts looking for an available spot. If the agent cannot propose spot (the FS list is empty for example), the driver keeps on driving and looking by himself for a place while the assistant agent tries to update its knowledge in order to find a free parking spot(i.e. it remains in state 2). In this case, if the assistant agent cannot offer places but the driver finds one on its way, then agent returns to state 0(arc (3)). However, if the agent proposes a place to the driver, together with its itinerary, it proceeds to state 3(arc (4)) and the driver goes to the chosen spot. Finally, from state 3, it can go to:

- state 0, if the driver finds a spot on his way that suits him better than the one proposed or when he arrives at the chosen spot that stills free. So the driver parks his vehicle and the agent will be stopped.
- state 2 and the search cycle starts again. This happens when the driver arrives at the place and find it already taken (for instance, a driver from outside the community would have found it)

We have chosen to make the agents communicate via an inter-vehicular network. This choice allows the information exchanged to move following two vectors. The first is specific to the communication. Indeed, the messages exchange takes between each two neighboring vehicles in the same community, and by transitivity agents can be informed of the availability of spots. The second vector concerns the movement of vehicles that mechanically move their information. However, the broadcast of information within the community can lead to a deterioration in the quality and in the effectiveness of the system. There is quality degradation if an isolated agent cannot access or share its information. The effectiveness of the system can be challenged by a very large number of communications. Indeed, the information update is based on a restricted broadcast that depends on the location, but this communication is systematic. The density of the network can generate a large number of messages. However, the communications take place very locally between vehicles.

2.2 Cooperation Model

Our MAS is based on the cooperation of agents to share information regarding the availability of spots. This cooperation uses two types of broadcast. The first type concerns all the information that the agent has when not looking for a parking spot. Otherwise, it only broadcasts information that are not interesting.The messages exchanged between assistant agents from the same community include their lists (FS and OS) which contain, respectively, the spots that are possibly free and those probably taken.

The communication module of the assistant agent extracts the lists FS_B and OS_B from each received message and forwards it to the decision module. The decision module updates both lists by aggregating the various received lists $(FS_B$ and $OS_B)$ with its own $(FS_A$ and $OS_A)$ by using the algorithm following bellow.

```
$LP_A = \{\ldots\langle pl_{i_1}, t_{i_1}\rangle, \ldots\}$
$LP_B = \{\ldots\langle pl_{i_2}, t_{i_2}\rangle, \ldots\}$
$BL_A = \{\ldots\langle pl_{i_3}, t_{i_3}\rangle, \ldots\}$
$BL_B = \{\ldots\langle pl_{i_4}, t_{i_4}\rangle, \ldots\}$
for all $\langle pl_i, t_i\rangle \in LP_B$ do
  if $(pl_i \in BL_A)$ then
    if $(t_i^B >= t_k^A)$ then
      eliminate($pl_i$, $BL_A$)
      add( $pl_i$, $t_i^B$, $LP_A$)
    end if
  else
    if $(pl_i \in LP_A)$ then
      if $(t_k^A < t_i^B)$ then
        update(($pl_i$, $t_i^B$, $LP_A$)
      end if
    else
      add( $pl_i$, $t_i^B$, $LP_A$)
    end if
  end if
end for
for all $\langle pl_i, t_i\rangle \in BL_B$ do
  if $(pl_i \in BL_A)$ then
    if $(t_j^A < t_i^B)$ then
      update(($pl_i$, $t_i^B$, $BL_A$)
    end if
  else
    if $(pl_i \in LP_A)$ then
      if $(t_j^A < t_i^B)$ then
        eliminate($pl_i$, $LP_A$)
        add( $pl_i$, $t_i^B$, $BL_A$)
      end if
    end if
  else
    add( $pl_i$, $t_i^B$, $BL_A$)
  end if
end for
```

The idea is to browse each received list $(FS_B$ and $OS_B)$ and update the local list $(FS_A$ and $OS_A)$ with the date associated with the spots. If there are two conflicting informations, then the newest one is kept, since the last driver who has visited this spot has the information that is most probably correct about its availability. After updating the two lists FS_A and OS_A, the decision module refers to the communication module, which is responsible for its dissemination to other neighbors in the community. This message processing is done in every state of the agent assistant, with the exception of the state 0.

When the driver is looking for a parking spot, he may request help from the system. The corresponding assistant agent updates its lists FS and OS from the received messages. Then, the decision module sends to the itinerary module the entire FS containing the list of spots known to be free. The itinerary module calculates the routes for each spot on this list and forwards the result to the decision module. Based on the selection criteria set by the driver, the decision module proposes a spot that meets the needs of the driver. Then it deletes the information corresponding to the proposed spot from its FS list. Finally, it sends the rest of the list and the OS list to the communication module which takes care of their distribution to the neighbors. The removal of the information about this spot will reduce its spread within the community. Thus, the assistant agent increases the driver's chances of finding the spot free. In addition, during the movement of the driver to the chosen spot, the assistant agent can suggest an alternative spot that best meets her needs. If the driver finds the spot the proposed by the system is occupied, then the corresponding agent updates the list BL and begins to look him for another spot.

3 Experiments

For the validation, it is necessary to compare the effectiveness of the process of finding spots of drivers who use the system with drivers who don't. We have chosen the proximity to the current position as the decision criterion to choose a spot. The parameters that are selected for the system are as follows. First, the number of agents within and outside the community. Then, the time spent by an agent on a spot (OT, for occupation time). The third and final parameter is the lifetime of the information on the availability of a spot. We wish to verify the impact of this parameter of information dissemination on the results and to find the thresholds beyond which there is no point in maintaining this information.

To evaluate the different scenarios, we choose the following criteria. The first one is the success rate (SR, or effective use rate of the system) by the agents of the community which represents the ratio between the number of drivers who have found a parking via

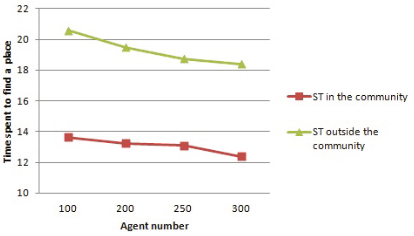

Fig. 2. Relevance of the system

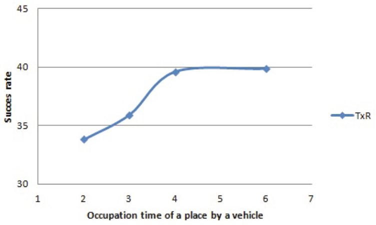

Fig. 3. Impact of spot rareness

the system by the total number drivers. The second criterion is the average spent time to find a spot per agent (ST). All time variables are expressed in number of execution cycles.

In the first series, we vary the number of agents in the community (NbA). This allows us to see the impact of system use on search time. In the second one, we study the impact of the rareness of spots on the success rate (SR). Finally, we compare the average number of exchanged messages per agent in our system and in centralized approach. In the graph shown in Fig. 2, we represent the average time spent to find a spot in and outside of the community according to the number of agents. The abscissa axis gives the number of agents that are taken into account. For instance, the value 100 means that 100 agents into the community spend on average 13.62 cycles to find a free spot whereas 100 agents outside community spend 20.57 cycles on average.

We can notice that, the more agents in the community we have, the less time they spend searching spots. This result is due to the fact that spots availability information is better propagated in the community, when the number of its members is important. Therefore, vehicles using the system spend less time to find a parking spot. Moreover, we can also note that the average time to find a spot (ST) for an agent of the community is much lower than that of an agent outside the community. The difference varies from one to seven cycles when all agents are in the community. For example, if there are 100 agents in the community, the average ST is equal to 13.62 cycles, whereas the average ST of an agent outside the community (i.e. 200 agents) is 19.48 cycles. According to these results, we can conclude that our proposal is useful and effective, especially when the community size is large enough. In a centralized approach, when the number of requests increases, the cost of communication rises and the quality of the service degrades because the response time increases while the information is volatile and can be obsolete meanwhile.

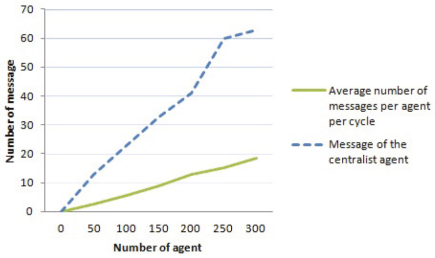

Fig. 4. Comparison of the number of messages per agent

In the next series of simulations, we fixed the number of spots in the network and varied the occupation time of a spot by a vehicle (OT). The Figure 3 illustrates the variation of the success rate according to the rareness of spots. For example the SR is 33.82 % when the OT is equal to 2 cycles. However this rate increases to 39,85 % when OT is 4 cycles. At the end, it stabilizes as the number of spots is limited. These results prove that more spots are rare more the system is useful, until a certain limit due in the limitation of the resources. Indeed, when a driver perceives several free spots, he does not really need help to find one. However, if they are rare the proposed system turns out to be very useful.

In Fig. 4, we report the number of messages handled by each agent in each cycle in our proposal, that we compare with a centralized solution. In the centralized approach, there are two messages exchanged with the central agent for each parking spot search (request and response). There is also a message informing that the chosen spot is taken. Finally, when the driver leaves a spot, there is a message informing about the spot release. This agent is unique, which limits the total number of messages but is a bottleneck. Fig. 4 shows that, even when the number of agents increases, the average number of messages that everyone should process remains reasonable (60 messages).

4 Related Work

In this section, we compare our proposal with the approaches of the literature. In the first subsection, we present previous solutions to the problem of parking spots, and in the second, we compare our cooperation model with the state of the art.

4.1 Application

Several studies such as [10] have identified the importance of better parking systems to improve the quality of life. Several works have proposed solutions to help drivers find a place as soon as possible either in car parks or in urban areas. In the first case, the e-car presented in [6] offers the driver to book online a parking space. SPARK [11] is an application of smart car that offers a new way of parking for large parking lots using communication in a VANET[1]. Another work in this context is presented in [4]. The authors employ inter-vehicular communication where a driver releasing a parking place disseminates information to her neighbors and assigns the resource to one of them. Thus, this solution assumes that the driver remains in the vicinity of the place and nearby vehicles that are interested until the allocation is made.

Other solutions have also been proposed for parking in an urban environment. San Francisco is the first city to adopt this type of intelligent system. Indeed, it has deployed SFpark [2]. This is an innovative project management of parking places in real time. The collection of real-time information is done through ground sensors. They are connected to a wireless network and thus allow to indicate at all times the presence or absence of cars on a place and relay this information to a centralized database. Drivers can be informed of the availability through various means such as electronic road signs placed along the streets, dynamic maps on the web site dedicated to mobile applications. In France, the first experiment took place in Lyon in early 2009, under the Predit[3]. This is a system for analyzing the presence of vehicles and calculating the length of the parking lot with sensors under the roadway every 2.5 meters that communicate via RFID. Information is updated every 10 seconds and disseminated to clients on their PDAs or via SMS. Another work in this context is presented in [3] where the authors propose an algorithm using a VANET and based on the dissemination of information regarding the state of urban

[1] Vehicle Ad Hoc Networks.

[2] `http://sfpark.org/`

[3] French Research Program and innovation in land transport `http://www.predit.fr`

parking fee. They produce such information and disseminate them to nearby vehicles. Then, the vehicles exchange information through inter-vehicular communication.

We note that the majority of solutions in the literature are centralized and use sensors to collect information, they require significant investments in contrast to the solution we propose in this paper. A less expensive option is explained in [14], where the authors have proposed to use an architecture called ParkNet based on a network of mobile sensors positioned in the vehicle that collects information about the availability of parking places in an urban area, along with the movement of the vehicle. They have proposed two architectures: a centralized architecture and a distributed architecture. In the distributed architecture, another vehicle-related sensors act as detectors of places and disseminators of information to other sensors.

4.2 Cooperation by Mutual Awareness

There are two main solutions to support mutual awareness in multi-agent systems: the creation of a dedicated channel and the distribution. The first solution is based on an architecture that intermediates between agents. This architecture might be such the MAS environment [12] or thematic channels [2]. As part of our application, the use of inter-vehicular communication to ensure the locality of interactions does not allow this type of approaches as there is no common channel to all agents.

The broadcast solution is the easiest to handle mutual awareness [7,9]. It has been used in [9] as support for a dynamic organization of autonomous vehicles and is similar to our approach. Mutual awareness is limited by the scope of communication and allows dynamic update of the representation of the world (groups, agents) through broadcasted messages. We have extended this result to a large number of agents and to the management of very volatile information thanks to the implementation of an epidemic spread of information [1]. Indeed, information from a vehicle is spreading in the network as it is valid (valid and not temporally questioned) by the movement of vehicles and their spontaneous interactions. The management of information quality that we propose has implications for the efficiency of the solution by mechanically limiting the dissemination of information to where it is useful.

5 Conclusion and Perspectives

In this paper, we propose a solution for the management of parking spots in an urban area. It is based on a multi-agent approach for the design of a community of drivers that interact to keep up to date information regarding the availability of parking spots. Communication between agents is supported by an inter-vehicular network with a radius of restricted broadcast, ensuring the consideration of local information. Our system works without prior information on the spots and no central storage of information. We have focused our validation on the average search time and showed a decrease regardless of the density of the vehicular network. The first perspective is to expand our testing protocol to take into account the particular hazards of data transmission inherent to this type of network and data traffic using the system Claire-Siti [13]. The second, is to study the definition of two architectures that re-centralize some of the processing. The objective is to compare the two architectures with the one presented here.

References

1. Becker, C., Bauer, M., Hahner, J.: Usenet-on-the-fly: supporting locality of information in spontaneous networking environments. In: Proceedings of CSCW 2002 (2002)
2. Busetta, P., Merzi, M., Rossi, S., Legras, F.: Intra-role Coordination Using Group Communication: A Preliminary Report. In: Dignum, F.P.M. (ed.) ACL 2003. LNCS (LNAI), vol. 2922, pp. 231–253. Springer, Heidelberg (2004)
3. Caliskan, M., Graupner, D., Mauve, M.: Decentralized discovery of free parking places. In: Proceedings of the 3rd International Workshop on Vehicular Ad Hoc Networks, VANET 2006, pp. 30–39. ACM (2006)
4. Delot, T., Cénérario, N., Ilarri, S., Lecomte, S.: A cooperative reservation protocol for parking spaces in vehicular ad hoc networks. In: Mobility 2009 Proceedings of the 6th International Conference on Mobile Technology, Application and Systems, pp. 1–8. ACM (2009)
5. Gradinescu, V., Gorgorin, C., Diaconescu, R., Cristea, V., Iftode, L.: Adaptive traffic lights using car-to-car communication. In: IEEE Vehicular Technology Conference, pp. 21–25 (2007)
6. Hodel-Widmer, T.B., Cong, S.: Psos, parking space optimization service. In: The 4th Swiss Transport Research Conference (2004)
7. Kaminka, G., Pynadath, C., Tambe, M.: Monitoring teams by overhearing: A multi-agent plan-recognition approach. Journal of Artificial Intelligence Research 17, 83–135 (2002)
8. Lefauconnier, A., Gantelet, E.: La recherche d'une place de stationnement: Stratégies, Nuisances associées, Enjeux pour la gestion du stationnement en France. L'ingénierie du stationnement, bureau d'études et de conseil, Sareco (2005)
9. Legras, F., Tessier, C.: LOTTO: Group Formation by Overhearing in Large Teams. In: Dignum, F.P.M. (ed.) ACL 2003. LNCS (LNAI), vol. 2922, pp. 254–270. Springer, Heidelberg (2004)
10. Nangeroni, C.: Stationnement intelligent: à la bonne place au bon moment. Ville, Rail et Transport 507, 30–33 (2010)
11. Rongxing, L., Xiaodong, L., Haojin, Z., Xuemin, S.: Spark: A new VANET-based smart parking scheme for large parking lots. In: IEEE INFOCOM, pp. 1413–1421 (2009)
12. Saunier, J., Balbo, F.: Regulated multi-party communications and context awareness through the environment. International Journal on Multi-Agent and Grid Systems 5, 75–91 (2009)
13. Scemama, G., Carles, O.: Claire-siti, public and road transport network management control: A unified approach. In: IEE Road Transport Information and Control Conference, pp. 11–18 (2004)
14. Suhas, M., Sanjit, K., Gruteser, M., Trappe, W.: Parknet: A mobile sensor network for harvesting real time vehicular parking information. In: Proceedings of the 2009 MobiHoc S3, pp. 25–28 (2009)
15. Yang, X., Liu, J., Zhao, F.: A vehicle-to-vehicle communication protocol for cooperative collision warning. In: Mobile and Ubiquitous Systems: Networking and Services, pp. 114–123 (2004)