

AN AGENT-BASED COMMUNITY TO MANAGE URBAN PARKING

Nesrine Bessghaier^{1,2}, Mahdi Zargayouna² and Flavien Balbo^{1,2}

¹ *Paris-Dauphine University, Lamsade laboratory, Paris, France*

² *UPE, IFSTTAR, GRETTIA, Noisy Le Grand, France*

nesrine.bessghaier@dauphine.fr, hamza-mahdi.zargayouna@ifsttar.fr, balbo@lamsade.dauphine.fr

Keywords: Transport, Communication, Cooperation

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 drivers community shares information about spots availability. The decrease in search time is obtained thanks to agents communication and cooperation. The communication between agents takes place via an inter-vehicular network, not imposing any costly infrastructure. The cooperation model necessitates no prior information and ensures the scalability of the proposed system. Implemented with the Madkit platform, 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 (Lefauconnier and Gantelet, 2005) 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 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. 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 (Yang et al., 2004) and the management of traffic lights (Gradinescu et al., 2007). 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 composed of the agents internal architecture, the communication model and the cooperation 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

In our proposal, drivers in a community keep each other informed of the availability of parking spots. This community is a subset of the drivers in the agglomeration. In this community, each member maintains a mutual awareness on information available.

Our system for the search of places in an urban area is modeled by a type of agent designated by *assistant* agent. It can assist the driver of a vehicle by helping her choose among the available places that

suits her the most. Thus, there is no explicit representation of the environment in our system. The relationship between agents is based solely on their physical proximity.

2.1.1 Assistant agent architecture

The internal architecture of the assistant agent has 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 slots. The itinerary module ensures the calculation of the route to a particular parking slot starting from the driver's current position, but also monitors its movement. Thus, the assistant agent facilitates the travel of the driver to the selected parking slot, following the shortest path. Finally, the decision module performs the decision-making. This module proposes a parking spot to the driver. The proposed spot must meet 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 slots. This knowledge evolves over time with information acquired through the exchange of messages with different assistant agents and to the perception of the agent. The memory of the assistant agent is composed of two disjoint lists:

- LP list (for list of places) contains a set of pairs $\{ \langle place, time \rangle \}$, each pair refers to a specific place: its geographic position and the moment since which it was released.
- BL List (for black list) contains the places that were in LP 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 places is illustrated when an agent chooses a place on its list LP - supposed to be free but, once there, it finds it occupied. In this case, the LP lists shared by the agents contain incorrect information about this place. 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 BL list enables agents to filter the information received and to have

the best information possible as we show it in section 2.3. On the other hand, 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 LP and BL with an associated time that is inferior to the current time minus θ . This parameter should take into account the network activity. Thus, a low value reflects a high volatility as the case may be in rush hour in downtown, while a high value keeps a longer sharing of information and reflects, for instance, the lower volatility in a residential area.

2.1.2 Assistant agent state diagram

The assistant agent passes through four states as indicated by the automata of Figure 1.

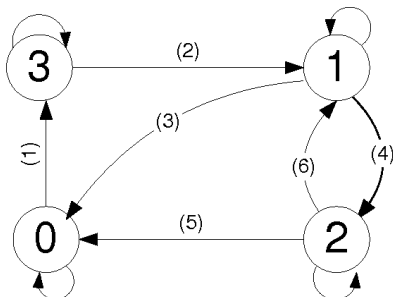


Figure 1: Assistant agent state diagram

- state 0: the vehicle is parked, the agent is stopped.
- state 1: the agent is looking for a parking spot to propose it to the driver of the concerned vehicle.
- state 2: driver moves towards place proposed by assistant agent who stays aware of possible alternatives which are more suitable.
- state 3: the driver is moving to its final destination.

Starting from state 0, the assistant agent goes to state 3 when the driver releases a parking spot (arc (1) in Figure 1). When the driver is near his destination, the assistant agent switches to state 1 (arc (2)). If it finds no place in its list, the driver keeps on driving and looking for a place while the assistant agent keeps on looking (i.e. it remains in state 1). In this case, if the assistant agent cannot offer places before the driver manages to find one on its way, then it returns to state 0 (arc (3)). However, if the agent proposes a place to the driver together with her itinerary, it proceeds to state 2 (arc (4)) and the driver goes to the chosen place. Finally, from state 2, it goes to:

- state 0, if the driver finds a place on her way that suits her better than the one proposed, or when she arrives at the chosen spot and it is free.

- state 1, that is to say that the search cycle starts again. This happens when the driver arrives at the place and find out that it is taken (for instance, a driver from outside the community would have found it)

2.2 Communication Model

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 places, however remote. For instance, in Figure 2, agents As_4 and As_6 share information via agent As_5 . The second vector concerns the movement of vehicles that mechanically move their information. For example, in Figure 2, agents As_1 and As_2 do not share information yet but will do so shortly following movement of vehicles.

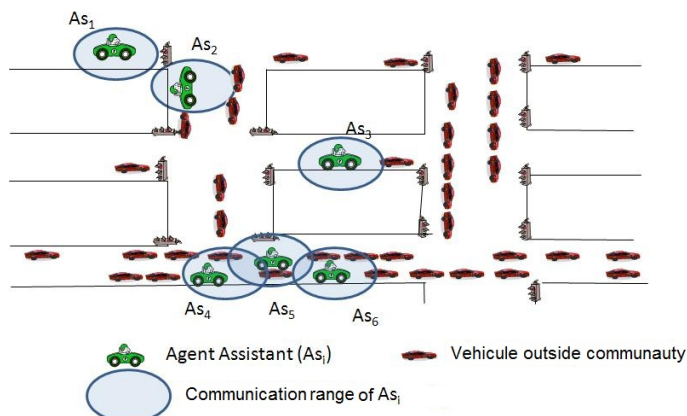


Figure 2: Information dissemination in the community through an inter-vehicular network

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. It is the case for agent As_3 in the figure. But also if many agents choose the same place as may be the case for agents As_4 and As_6 . 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. For instance, in Figure 2, communication between As_4 , As_5 and As_6 implies the exchange of four messages. On the scale of the entire transportation network, the number of messages at a time

t is the sum of communication between all adjacent agents. Depending on the density of the network, this can represent a large number of messages. However, the communications take place very locally between vehicles and the total number of messages by agent is less important than in a centralized architecture. In section 3, we will measure the average number of exchanged messages per agent in our system. Furthermore, in order to maintain a satisfactory quality, we limit the validity time of the information as explained in section 2.1.1, and we've taken into account event information as explained in Section section 2.3 which follows.

2.3 Cooperation model

To manage parking in an urban environment, our MAS is based on the cooperation of its agents to share information regarding the availability of places. This cooperation uses two types of broadcast. The first type concerns all the information the agent has when not looking for a parking space. Otherwise, it only broadcasts information that doesn't interest him (i.e. the places to which it won't head).

2.3.1 Ensure the update

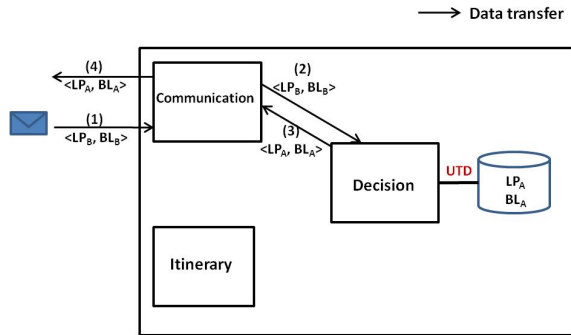


Figure 3: Maintaining information exchanged coherence

The messages exchanged between assistant agents from the same community contain their two lists LP and BL. They contain for each agent, respectively, the places it knows to be possibly free and those it knows to be probably taken.

The update of the lists is illustrated in Figure 3. The communication module of the assistant agent extracts the lists LP_B and BL_B from each received message and forwards it to the decision module. These correspond respectively to arcs (1) and (2) in Figure 3. The decision module updates both lists by aggregating the various received lists (LP_B and BL_B) with its own (LP_A and BL_A) using the following algorithm :

$$LP_A = \{\dots \langle pl_{i_1}, t_{i_1} \rangle, \dots\}$$

```

 $LP_B = \{\dots \langle pl_{i_2}, t_{i_2} \rangle, \dots\}$ 
 $BL_A = \{\dots \langle pl_{i_3}, t_{i_3} \rangle, \dots\}$ 
 $BL_B = \{\dots \langle pl_{i_4}, t_{i_4} \rangle, \dots\}$ 
for all  $\langle pl_i, t_i \rangle \in LP_B$  do
  if  $(pl_i \in BL_A)$  then
    if  $(t_i^B \geq t_k^A)$  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
        mise-a-jour( $(pl_i, t_i^B), LP_A$ )
      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
      up-to-date( $(pl_i, t_i^B), BL_A$ )
    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 (LP_B and BL_B) and update the local list (LP_A and BL_A) with the date associated with the places. Thus, if there are two conflicting information, then the newest information is kept, since as the last driver who has visited this place has the information that is most probably correct about its availability.

After updating the two lists LP_A and BL_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 it is off, in which he does not send nor receive messages.

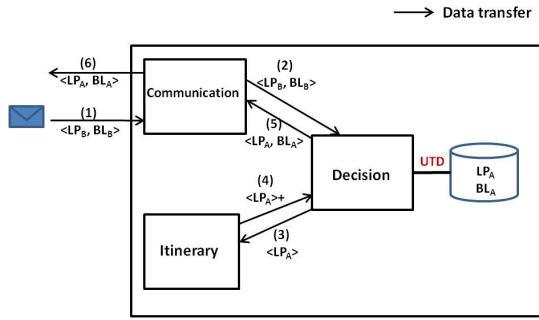


Figure 4: Computation of the chosen place

2.3.2 Manage the chosen place

When the driver is looking for a parking spot, she may request help from the system. The corresponding assistant agent updates its lists LP and BL from the received messages, as described above. Then, to choose the place to propose to the driver (Figure 4), the decision module sends to the itinerary module the entire LP containing the list of places known to be free. This corresponds to the arc (3) in Figure 4. The itinerary module calculates the routes for each place on this list and forwards the result to the decision module (arc (4)).

Based on the selection criteria set by the driver (which can cover the distance or the safety of the location for instance), the decision module proposes a place that meets the needs of the driver. Then it clears of its LP list the information corresponding to the proposed spot. Finally, it sends the rest of the list and the BL 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 chances of finding the place free. In addition, during the movement of the driver to the chosen spot, the assistant agent can suggest an alternative place that best meets her needs, launching at each time step the search procedure.

Figure 5 represents the global behavior of the *assistant agent* taking into account its communication.

3 Experiments

3.1 Used platform

The application and the chosen model do not dictate a particular multi-agent platform. For our simulations, we used the platform MadKit (Multi-Agent Development Kit) (Gutknecht and Ferber, 2000) for its ease of

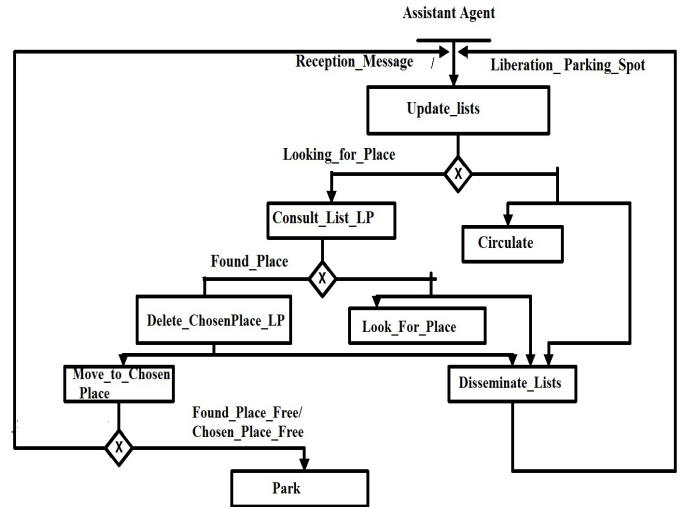


Figure 5: Agent assistant behavior

use and implementation. It is a platform for development and execution of multi-agent systems, designed according to the organizational model AGR (Agent / Group / Role). In order to simulate the behavior of situated agents, we used the TurtleKit. Thus, we ensure that each cycle all agents perform their actions relatively to their states and manage their communications.

3.2 Configuration

The objective of the community management of parking spots is to allow the sharing of information about available places in a decentralized manner. For the validation, it is necessary to compare the effectiveness of the process of finding places for drivers without using the system with that of drivers using the community system. To do this, we keep the same number of agents simultaneously existing in the network (300) and we change the number of agents that belong to the community from zero to 300. There is no explicit entry and exit of agents in the system, but in fact each agent leaving a place can be considered as a new agent who begins to look for a place after a certain travel time. The travel times associated to the edges of the network are fixed, i.e. we don't consider the impact of possible congestion on the system behavior.

We have chosen proximity as the decision criterion to choose a place (the place in LP that is the closest). This test has the advantage not to rely on qualitative information such as the safety of the place and limit the effects of bias. Besides, an absolute temporal criterion like choosing the last released place would have the consequence to direct all searching agents to

the same places.

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 place (OT). This last parameter is used to check the impact of the turnover of available spots on the relevance of using the system. The third and final parameter is the lifetime of the information on the availability of a place. 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 (or effective use rate of the system) by the agents of the community which represents the ratio between the number of times that drivers have found a parking through the system by the total number of times that they have parked. The second criterion for evaluating our proposal is the average time spent to find a place by agents type. The relevance of this test is to verify that a driver earns to use the system rather than finding a place on its own, and to assess this gain. All times are expressed in number of execution cycles.

For each simulation we have performed 15 executions, each one running on 50 cycles. The tables given in the next section contain the means of these different versions. The network of our simulation contains 17 edges and it takes 10 cycles to traverse an edge. At the start of execution, we set the agents randomly on the network in which we set a fixed number of places in each edge. Half of them are traveling and the other half is stopped. Each arc is bidirectional, and an agent can turn around if needed.

3.3 Results

In order to demonstrate effectiveness and utility of our proposal, we conducted many series of simulations in which the criteria are the average time spent to find a place (ST, to minimize) and the success rate of the system (SR, to maximize). In the first series, we set θ which represents the number of cycles beyond which an information about a place becomes obsolete, to 13 and we have varied the number of agents in the community (NbA). This allowed us to see the impact of system use on search time. In the second, we tried to study the information distribution duration impact on ST and SR. To do this, we set NbA and occupation time of a place by a vehicle (OT) and varied θ . Finally, we studied the impact of the rarity of places on the success rate (SR).

3.3.1 System use impact

In the graph shown in Figure 6, we represented the average time spent to find a place in and outside of the community according to the number of agents. The total number of agents in the simulation is 300, i.e the number of agents outside the community is the complement to 300 of those in the community. The abscissa axis gives the number of agents that are taken into account. For example, the value 100 means that 100 agents into the community spend on average 13.62 cycles to find a free place whereas 100 agents outside community spend 20.57 cycles on average.

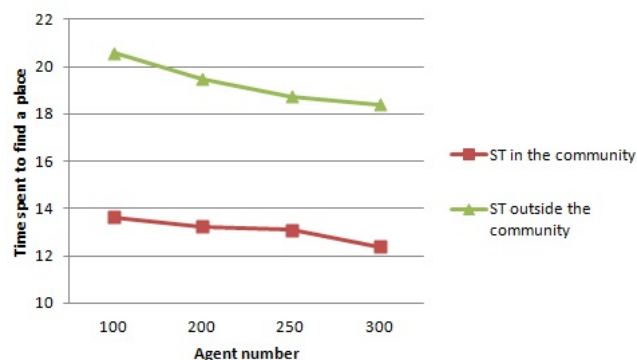


Figure 6: Profit of the system

We can notice that more the number of agents in the community increases, more the time of search decreases. This result is due to the fact that places availability information is better propagated in the community, when the number of its members is quite important. Therefore, vehicles using the system will spend less time to find a parking spot. Moreover, we can also note that the average time to find a place (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 proposition is useful and effective, especially when the number of members community is enough important.

3.3.2 Impact of information lifetime

In the second series, we study the impact of information dissemination fields on the availability of a place. Thus, θ gives the number of cycles beyond which an agent considers the information as outdated. The less θ , the lower the information is disseminated among

agents. This way, we can, based on the simulation for which we obtained the best results, consider the importance of the volatility of the information. Therefore, we fixed the number of members community to 300 and vary θ . Table 1 contains the results of this second series.

N	θ	ST	SR
(1)	6	18	0%
(2)	7	15.2	40.2%
(3)	8	15.2	42.2%
(4)	9	15.2	40.3%
(5)	10	15.5	39.5%
(6)	12	15.2	42.1%
(7)	13	15.2	40.5%
(8)	16	15.7	37.5%

Table 1: Information lifetime impact

In terms of search time, the results improve when we increase the field dissemination (7) to remain stable (8-13) before degrading. The success rate of the system in turn improves and wins 40 points when θ varies from 6 to 7. This shows the impact of the volatility of information in such a system on its effectiveness because the agents acted as agents outside the community. Then, the rate remains more or less stable in the other simulations until it begins to degrade from the simulation (6). This shows that a balance must be found for the validity of such information and that there is a threshold beyond which it is useless to keep this information. When the field of dissemination is too large, then agents may choose places that have been taken, due to the oldness of information.

3.3.3 Places rareness Impact

In this series of simulations, we fixed the number of places in the network and varied the occupation time of a place by a vehicle (OT). The Figure 7 illustrates the variation of the success rate according to the rareness of places. 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 places is limited. These results prove that more places 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 places, he/she does not really need a help to find one. However, if they are rare the proposed system turns out very useful.

3.3.4 Communication cost

In Figure 8, we report the number of messages handled by each agent in each cycle in our proposal, that

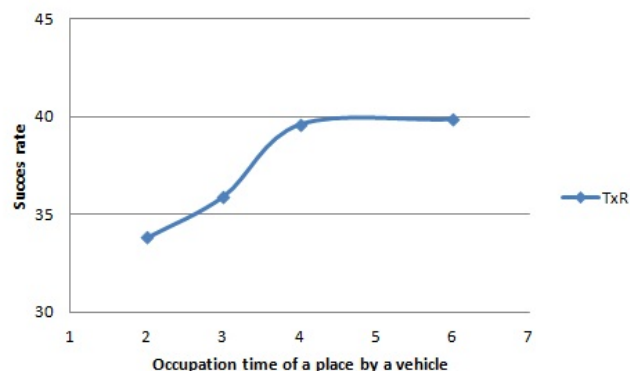


Figure 7: Impact of place rareness

we compare with a centralized solution. In the latter, each agent assistant that frees up a place sends a message to a central system. There are two messages for each parking spot search (request and response) and a message informing that the chosen place is taken. In our approach each agent is listening to both client and server information. In a centralized approach, this agent is unique, which limits the total number of messages but is a bottleneck. Figure 8 shows that even when the number of agents increases, the average number of message that everyone should process remains reasonable (60 messages).

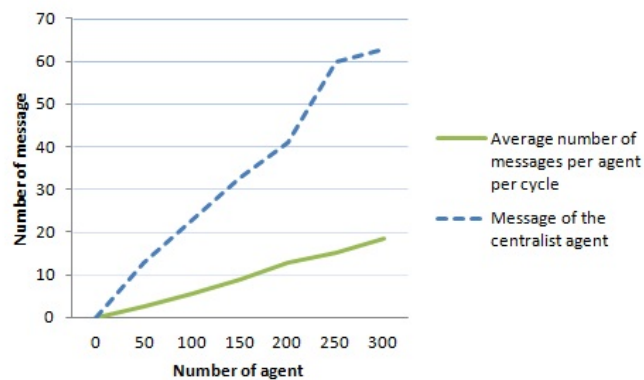


Figure 8: Comparison of the number of messages per agent

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 (Nangeroni, 2010) 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 (Hodel-Widmer and Cong, 2004) offers the driver to book online a parking space. SPARK (Rongxing et al., 2009) is an application of smart car that offers a new way of parking for large parking lots using communication in a VANET¹. Another work in this context is presented in (Delot et al., 2009). 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. 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 website dedicated to mobile applications. In France, the first experiment took place in Lyon in early 2009, under the PreditThis 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 (Caliskan et al., 2006) where the authors propose an algorithm using a VANET and based on the dissemination of information regarding the state of urban 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 (Suh et al., 2009), where the authors have proposed to use an architec-

¹Vehicle Ad Hoc Networks

ture 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 (Saunier and Balbo, 2009) or thematic channels (Busetta et al., 2003). 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 (Kaminka et al., 2002; Legras and Tessier, 2004). It has been used in (Legras and Tessier, 2004) 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 (Becker et al., 2002). 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 places in an urban area. Our solution 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 places. 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 places 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.

We are considering different perspectives to this work. First, we are expanding 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 (SCEMAMA and CARLES, 2004). On the other hand, we 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. In addition, we have integrated all the criteria that come into play when choosing a parking place and associate that information with each place in the lists *BL* and *LP*. From this perspective, it is also relevant to associate a confidence factor weighting the information provided by a particular driver, and an incentive for drivers to participate in the form of bonus or exclusive places.

REFERENCES

- Becker, C., Bauer, M., and Hahner, J. (2002). Usenet-on-the-fly: supporting locality of information in spontaneous networking environments. In *Workshop on Ad Hoc Communications and Collaboration in Ubiquitous Computing Environments*. ACM Press.
- Busetta, P., Merzi, M., Rossi, S., and Legras, F. (2003). Intra-role coordination using group communication: A preliminary report. In Dignum, F., editor, *Advances in Agent Communication: International Workshop on Agent Communication Languages and Conversation Policies, ACL2003*, pages 231–253. Springer Verlag.
- Caliskan, M., Graupner, D., and Mauve, M. (2006). Decentralized discovery of free parking places. In *Proceedings of the 3rd international workshop on Vehicular ad hoc networks, VANET'06*, pages 30–39. ACM.
- Delot, T., Cntrario, N., Ilarri, S., and Lecomte, S. (2009). A cooperative reservation protocol for parking spaces in vehicular ad hoc networks. In *Mobility'09 Proceedings of the 6th International Conference on Mobile Technology, Application and Systems*. ACM.
- europenne, C. (2005). Livre vert "face aux changements démographiques, une nouvelle solidarité entre générations". Communication de la Commission COM(2005), Bruxelles.
- Gradinescu, V., Gorgorin, C., Diaconescu, R., Cristea, V., and Iftode, L. (2007). Adaptive traffic lights using car-to-car communication. In *IEEE Vehicular Technology Conference*, pages 21–25.
- Gutknecht, O. and Ferber, J. (2000). Madkit: a generic multi-agent platform. In *Proceedings of the fourth international conference on Autonomous agents, AGENTS '00*, pages 78–79, New York, NY, USA. ACM.
- Hodel-Widmer, T. B. and Cong, S. (2004). Psos, parking space optimization service. In *The 4th Swiss Transport Research Conference*.
- Kaminka, G., Pynadath, C., and Tambe, M. (2002). Monitoring teams by overhearing: A multi-agent plan-recognition approach. *Journal of Artificial Intelligence Research*, 17:83–135.
- Lefaucconnier, A. and Gantelet, E. (2005). *La recherche d'une place de stationnement: Stratgies, Nuisances associes, Enjeux pour la gestion du stationnement en France*. Sareco : L'ingnierie du stationnement, bureau d'études et de conseil.
- Legras, F. and Tessier, C. (2004). Lotto: Group formation by overhearing in large teams. In *Advances in Agent Communication*, volume 2922 of *Lecture Notes in Artificial Intelligence*, pages 254–270. Springer Verlag.
- Nangeroni, C. (2010). Stationnement intelligent : la bonne place au bon moment. *Ville, rail et transport*, (507):30–33.
- Rongxing, L., Xiaodong, L., Haojin, Z., and Xuemin, S. (2009). Spark: A new vanet-based smart parking scheme for large parking lots. In *IEEE INFOCOM*, pages 1413–1421.
- Saunier, J. and Balbo, F. (2009). Regulated multi-party communications and context awareness through the

environment. *International Journal on Multi-Agent and Grid Systems*, 5(1):75–91.

- SCEMAMA, G. and CARLES, O. (2004). Claire-siti, public and road transport network management control: A unified approach. In *IEE Road Transport Information and Control Conference*, pages 11–18. Institution of Electrical Engineers.
- Suhas, M., Sanjit, K., Gruteser, M., and Trappe, W. (2009). Parknet: A mobile sensor network for harvesting real time vehicular parking information. In *MobiHoc S3'09: Proceedings of the 2009 MobiHoc S3 workshop on MobiHoc S3*, pages 25–28.
- Yang, X., Liu, J., and Zhao, F. (2004). A vehicle-to-vehicle communication protocol for cooperative collision warning. In *Mobile and Ubiquitous Systems: Networking and Services*, volume MOBOQUITUS The First Annual International Conference, pages 114–123.