



Generic model for resource allocation in transportation: Application to urban parking management

Mahdi Zargayouna, Flavien Balbo, Khadim Ndiaye

► To cite this version:

Mahdi Zargayouna, Flavien Balbo, Khadim Ndiaye. Generic model for resource allocation in transportation: Application to urban parking management. Transportation Research Part C: Emerging Technologies, Elsevier, 2016, 71, pp.538-554. <10.1016/j.trc.2016.09.002>. <hal-01366284>

HAL Id: hal-01366284

<https://hal.archives-ouvertes.fr/hal-01366284>

Submitted on 14 Sep 2016

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Generic Model for Resource Allocation in Transportation. Application to Urban Parking Management

Mahdi Zargayouna^a, Flavien Balbo^b, Khadim Ndiaye^b

^a*Université Paris-Est, IFSTTAR, GRETTIA
10-14, boulevard Newton,
77447 Champs sur Marne, France.*

^b*Connected Intelligence - Mines Saint-Étienne,
Hubert Curien Laboratory
UMR CNRS 5516 Mines Saint-tienne,
158 cours Fauriel
42023 Saint-tienne Cedex 2, France*

Abstract

In this paper, we define the online localized resource allocation problem, especially relevant for modeling transportation applications. The problem modeling takes into account simultaneously the geographical location of consumers and resources together with their online nondeterministic appearance. We use urban parking management as an illustration of this problem. In fact, urban parking management is an online localized resource allocation problem, where the question is how to find an efficient allocation of parking spots to drivers, while they all have dynamic geographical positions and appear nondeterministically. We define this problem and propose a multiagent system to solve it. The objective of the system is to decrease, for private vehicles drivers, the parking spots search time. The drivers are organized in communities and share information about spots availability. We have defined two cooperative models and compared them: a fully cooperative model, where agents share all the available information, and a “coopetitive” model, where drivers do not share information about the spot that they have chosen. Results show the superiority of the first model¹.

Keywords: Resource allocation, Urban parking, VANET, Multiagent

¹DOI: 10.1016/j.trc.2016.09.002

1. Introduction

Transport activities have important financial consequences and serious ecological fallout. As a consequence, increasing number of researches optimizing these activities are being conducted in the recent years, especially concerning urban activities. Indeed, investment on infrastructure are limited by space and financial costs, while the fleet of vehicles is continuously increasing. As a consequence, optimizing urban traffic is vital, together with increasing the use of electric vehicles and encouraging vehicle sharing. To this end, intelligent transportation systems are designed to improve the use, safety and efficiency of the transport means and infrastructure with information and communication technologies. Indeed, researchers are more and more interested in problems such as the search of charging stations for electric cars (Acha et al., 2011), the sharing of vehicles (taxi, car, bike, etc.) (Katzev, 2003), urban parking management (Delot et al., 2013), etc.

A part of these problems can be seen as resource allocation problems, where the challenge is to find an optimal allocation of resources to consumers. In transportation applications, the drivers or the travelers are generally the consumers. The resources might be charging stations, parking spots, traffic information or vehicles. However, these resource allocation problems are recurrent in transport applications and should have a generic problem formulation representing them. This formulation would identify the common concepts and constraints of these applications. One of the main characteristics of these problems is that they require the simultaneous consideration of time and space. Indeed, in a transport application, there is always an explicit representation of the environment (i.e. the transport network). The actors (drivers, travelers, etc.) are localized in this environment where they dynamically move. A generic formulation of resource allocation that is specific to these problems is the first objective pursued in this paper.

In transportation applications, the time dimension has to be explicitly represented because the information about resources and/or consumers is not known at the beginning of the allocation. This kind of problem is generally modeled as an online resource allocation (ORA) problem (Tesauro, 2005). The space dimension has to be explicitly modeled because resources and consumers are situated and because the distance between them generally

conditions the allocation: resources and consumers have to be geographically co-located or close enough for the allocation to take place. This kind of problem can be modeled as a localized resource allocation (LRA) problem (Golkar and Sousa, 2011). We present in this paper a generic model for both ORA and LRA problems called OLRA (for Online Localized Resource Allocation) problem. One main contribution is the introduction of a systematic, explicit and dynamic representation of the physical environment in the problem definition. Different instantiations of the problem specify different transport applications. Indeed, consumers might have only access to a sub-part of their physical environment at a certain time, resources might be volatile, especially in a shared environment, and can therefore be taken by any close consumer and resources might be uncontrollable because they are created and released in a nondeterministic way.

A lot of applications can be modeled as an OLRA problem. Actually, a resource allocation problem that involves moving entities (resources or consumers) can be seen as an OLRA problem. For instance, in the search of charging stations for electric cars (Acha et al., 2011), the consumers are mobile (the electric cars), while the charging stations are the resources. In the sharing of vehicles (car, bike, etc.) (Katzev, 2003), both the resources (the vehicles) and the consumers (the drivers or the passengers) are mobile. The scheduling of aircraft landings to multiple run-ways (Beasley et al., 2000) can also be seen as an OLRA problem, where the run-ways are the resources and the planes are the consumers. Among all the applications that can be modeled as an OLRA, we are more particularly interested in the search of urban parking spots, in which spots have to be assigned to drivers. This is one of the most challenging OLRA problems. Indeed, the resources in this problem appear and disappear non-deterministically. In addition, to consume a resource, a driver has to drive to the spot location, with uncertain arrival time due to traffic conditions. The management of urban parking is identified as an important issue to improve the quality of life in urban areas (Bayless and Neelakantan, 2012).

To formally model and analyze the problem of urban parking management, we propose to define it as an OLRA problem. Indeed, the solution to a particular urban configuration implies the assignment of a set of parking spots (the resources) to the drivers (the consumers). In addition to the problem specification, we are interested in proposing an efficient solution to the problem. In this paper, we propose a multiagent transport information system that helps finding parking spots in an urban agglomeration.

The multiagent paradigm facilitates an approach by analogy in the transportation domain where one of the objectives is the coordination of distributed entities. This is why the multiagent approach is often chosen to model, solve 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 multiagent system that facilitates the information sharing related to parking for a community of drivers. In our system, agents communicate to exchange information concerning the parking spots availability. The information come from the users, while remaining anonymous, and their collaboration ensures an information of good quality. Our agent-based approach is totally decentralized and we employ an inter-vehicular communication (using VANET²) to allow vehicles to receive and broadcast information to the other vehicles of the same community. VANETs provide a bottom-up discovery of parking places. One advantage of our proposal is that it can be developed without any additional infrastructure. Our solution is able to function without initial information and ensure drivers to have information that are the most up-to-date possible.

This article is organized as follows. Section 2 defines the OLRA problem as well as the urban parking management problem. Section 3 presents the multiagent system solving the problem. Section 4 details our experimental results. Section 5 discusses the related works. Finally, section 6 concludes this paper and provides some perspectives.

2. Problem definition

In this section, we detail the formal model of Online Localized Resource Allocation (OLRA) (Bessghaier et al., 2012b).

2.1. Description

Consider a set of resources and a set of consumers. Both resources and consumers appear non-deterministically and can subsequently change their position at any moment. Each resource has a state and a set of properties. An allocation changes the state of the resource but the properties that define it remain the same (Topaloglu and Powell, 2005).

²Vehicle Ad Hoc Networks

The problem that we define in this paper involves the assignment of resources to consumers, where both resources and consumers are situated in space and time and are not known in advance. A consumer starts looking for a resource at nondeterministic moments. These moments are not predefined and are discovered during the allocation process. On the other side, the resources are available starting from unknown moments and remain available during an unknown period of time. The compliance of a resource with a consumer's needs is conditioned with their spatial and temporal situation. As for all resource allocation problems, the compliance of the resource with the consumer is also conditioned by the latter's preferences, which concern the state and the properties of the resource. The preference of a consumer for resources is measured with an individual utility function. The local objective of the consumers is to maximize their own utility while the global objective of the allocation system is generally to minimize the total traveled distance and/or the total travel time.

2.2. Problem formulation

In the following, we formulate the Online Localized Resource Allocation (OLRA) problem and we define its various components.

2.2.1. Resources and consumers

An OLRA is a tuple:

$$OLRA = \langle \mathcal{R}, \mathcal{C}, \mathcal{G}, \mathcal{D} \rangle$$

where:

- $\mathcal{R} = \{r\}$ is the set of resources.
- $\mathcal{C} = \{c\}$ is the set of consumers.
- $\mathcal{G} = \langle \mathcal{V}, \mathcal{E} \rangle$ is a directed graph, with \mathcal{V} the set of nodes indexed from 1 to N , and $\mathcal{E} = \{e_{ij} | i, j \in \mathcal{V} \text{ and } i \neq j\}$ the set of edges.
- $\mathcal{D} = \{d_{ij} | i, j \in \mathcal{V} \text{ and } i \neq j, d_{ij} \in \mathbb{R}_+\}$, d_{ij} is the distance between two successive nodes i and j .

Each node of the network can contain one or more resources of \mathcal{R} . Resources may represent, for instance, vehicle seats, parking spots, places to recharge electric cars, etc. The distances between nodes are fixed, while the travel

times may vary according to the dynamics of the environment. If the graph represents a transportation network, the travel times would be impacted by traffic status and congestion.

To represent the non-deterministic availability of a consumer or a resource, we define the following function:

$$availability : (\mathcal{R} \cup \mathcal{C}) \times T \rightarrow \{0, 1\}$$

where T is the time horizon. This function returns 0 if the resource is not yet localized, or if the consumer is either not localized or is not interested in any resource.

The two following sets describe the different characteristics of the resources:

- $\mathcal{P} = \{p_1, p_2, \dots, p_m\}$: represents all the possible properties of resources, with m the number of properties.
- $\Delta = \{d_1, d_2, \dots, d_n\}$: contains the description domains of the properties, with $n \leq m$ the number of description domains (two properties might have the same description domain).

A property $p_i \in \mathcal{P} : \mathcal{R} \cup \mathcal{C} \rightarrow d_j$ is a function that gives the value of the property p_i in its description domain $d_j \in \Delta$. The description domain can be quantitative, qualitative or a finite set of data. Each resource is characterized by a set of property-value pairs. The properties that are defined for a resource, together with the corresponding description domain, are given by this function:

$$\varrho : \mathcal{R} \rightarrow (\mathcal{P} \times \Delta)^q$$

where q is the number of properties that are defined for the resource. If \mathcal{R} is homogeneous, resources are defined by the same q properties. Otherwise, the problem considers different types of resources, represented by different properties.

The function $compatibility(c, r, t)$ defines the fact that a consumer c and a resource r are compatible at time t , meaning that the values of the properties correspond to the requirement of the consumer at that moment. For instance, if the property concerns the type of electricity plug, consumers and resources should have the same value for that property all the time.

$$compatibility : \mathcal{C} \times \mathcal{R} \times T \rightarrow \{0, 1\}$$

The three following functions define the dynamic costs and the dynamic positions of resources and consumers.

- $\tau : \mathcal{V} \times \mathcal{V} \times T \rightarrow \mathbb{R}_+$, $\tau(i, j, t)$ returns the travel time between i and j at time $t \in T$ (the time horizon).
- $\rho : \mathcal{R} \cup \mathcal{C} \times T \rightarrow \mathcal{V}$, $\rho(r, t)$ or $\rho(c, t)$ returns the node where the resource r or the consumer c is located at t .

A resource or a consumer on the edge e_{ij} is considered to be positioned on i until it reaches j .

2.2.2. Allocation modeling

The following functions specify the dynamics of the allocation process. The interest of a consumer for a resource varies over time, either following an internal process or following his context. This context may include his current position or his final destination. The usefulness of a resource to a consumer is given by the following utility function:

$$\mu : \mathcal{C} \times \mathcal{R} \times T \rightarrow \mathbb{R}_+$$

$\mu(c, r, t)$ returns the utility of the resource r for the consumer c at time t . However, the consumption of a resource by a consumer is conditioned by their co-location. This can be verified with the following function:

$$\mathbb{1}_{\mathcal{F}} : \mathcal{C} \times \mathcal{R} \times T \rightarrow \{0, 1\}$$

$\mathbb{1}_{\mathcal{F}}(c, r, t)$ is an indicator function that returns 1 if the consumer c could have the same position than the resource r at time t and if they are both available at that time. That means that $\mathbb{1}_{\mathcal{F}}$ returns 1 for all the tuples $\mathcal{F} = \{(c, r, t) \in \mathcal{C} \times \mathcal{R} \times T \mid \rho(c, t) = \rho(r, t) \wedge \text{availability}(c, t) = \text{availability}(r, t) = 1\} \wedge \text{compatibility}(c, r, t)$. $\mathbb{1}_{\mathcal{F}}(c, r, t)$ returns 0 otherwise. Therefore, \mathcal{F} defines the set of potential space-time co-locations of resources and consumers under availability constraints. The availability constraint is necessary to filter the situations where a consumer and a resource are indeed co-located, but not available, for instance if the resource has been taken by another consumer.

The quality of a resource allocation in OLRA is generally related to the distance and the travel time of consumers. Their successive positions throughout the execution are specified with the three following functions.

$$\pi : C \rightarrow (\{1, \dots, |\mathcal{V}| \} \times T)^n, n \in \mathbb{N}$$

π defines the path of a consumer. Applied to a consumer c , π returns the nodes that the consumer has visited while moving towards a resource, together with the times corresponding to his visits. $\pi(c)[i, 1]$ allows to access the index of the i^{th} visited node, while $\pi(c)[i, 2]$ allows to access the corresponding visit time.

For instance, $\pi(c_1)[3, 1] = 10$ indicates that the third node visited by consumer c_1 is v_{10} , while $\pi(c_1)[3, 2] = t_3$ indicates that this visit occurs at time t_3 .

$$\delta(c) = \sum_{i=1 \dots |\pi(c)|-1} d_{\pi(c)[i,1], \pi(c)[i+1,1]}$$

δ determines the total distance traveled by c . The term $d_{\pi(c)[i,1], \pi(c)[i+1,1]}$ represents an element d_{xy} of the D matrix of distances, where $x = \pi(c)[i, 1]$ and $y = \pi(c)[i + 1, 1]$ are, respectively, the i^{th} and the $(i + 1)^{\text{th}}$ node indices returned by $\pi(c)$. $|\pi(c)|$ gives the total number of nodes visited by c .

For instance, if $\pi(c_1) = [(5, t_1), (10, t_3)]$, i.e. consumer c_1 visits node v_5 at time t_1 then node v_{10} at time t_3 ; if the distance $d_{5,10} = 6$, then $\delta(c_1) = 6$.

$$\varphi(c) = \pi(c)[|\pi(c)|, 2] - \pi(c)[1, 2]$$

φ gives the total travel time of a consumer c . The expressions $\pi(c)[|\pi(c)|, 2]$ and $\pi(c)[1, 2]$ are the instants of visits, respectively, of the last node and the first node visited by c . With the same above example, if $\pi(c_1) = [(5, t_1), (10, t_3)]$, then $\varphi(c) = t_3 - t_1$.

2.2.3. Solution constraints

A solution to an OLRA instance is an allocation of resources to consumers. This solution is given by the function γ , which specifies that a customer actually consumes a resource at a certain time:

$$\gamma : \mathcal{C} \times \mathcal{R} \times T \rightarrow \{0, 1\}$$

$\gamma(c, r, t)$ returns 1 if a consumer c takes the resource r at t and 0 if not.

A consumer cannot take a resource if they are not at the same position at the same time. Hence, $\gamma(c, r, t) = 1$ cannot be valid unless $\mathbf{1}_F(c, r, t) = 1$.

OLRA is not bound to specific resource properties. It can model problems where the resources are shareable or not, and where consumers can consume several resources at the same time or not. The considered variant of the problem is specified by two parameters k and l . The solution to the

considered problem has to comply with the two following constraints, which depend on k and l .

$$\sum_{c \in \mathcal{C}} \gamma(c, r, t) \leq k, \forall r \in \mathcal{R}, \forall t \in T \quad (1)$$

$$\sum_{r \in \mathcal{R}} \gamma(c, r, t) \leq l, \forall c \in \mathcal{C}, \forall t \in T \quad (2)$$

The constraint (1) specifies that the resources can be shareable and be taken simultaneously by at most k consumers ($k \in \mathbb{N}$). If the resources are not shareable, k is equal to 1. If several resources are co-located with a consumer, the problem definition may allow him to consume them simultaneously (constraint (2)). The number of resources that can be taken simultaneously is a parameter $l \in \mathbb{N}$. Again, if this is not allowed, l is set to 1. The values of k and l are model parameters and enable to take into account different problem variants and therefore different application types.

2.2.4. Objectives

The social objective of the study of OLRA is generally to minimize the time and/or the distance spent in the search of resources. This social objective can be expressed as:

$$\min \sum_{c \in \mathcal{C}} [\alpha \delta(c) + \beta \varphi(c)]$$

where α and β are positive numbers weighting the relative importance of time and space in the specific problem that is considered. We assume that α and β integrate scaling factors in order for the function terms to be expressed in the same unit and to truly reflect the weights of time and space. Indeed, $\delta(c)$ is usually expressed in meters while $\varphi(c)$ is usually expressed in seconds. The scaling factor could be based, for instance, on the average speed v (expressed in meters per second) of the consumers. In which case, α would be equal to $a \times \frac{1}{v}$ where a is the actual weight of space and $\frac{1}{v}$ the scaling factor.

Besides, every consumer has the local objective of maximizing his own satisfaction by obtaining the resources that best satisfy his preferences and maximize his utility. This personal objective is defined as follow:

$$\max \sum_{r \in \mathcal{R}, t \in T} [\mu(c, r, t) \times \gamma(c, r, t)]$$

A system might behave well w.r.t the local objectives of the consumers while the social objective is not optimized. Or it may exhibit good results for the social objective, while the individual objectives are of poor quality. As usual in this kind of problems, there is a compromise between these two objectives that the proposed solutions to this problem have to find.

2.3. Urban parking allocation model

2.3.1. The problem of urban parking

The management of urban traffic growth is an important issue, since the number of drivers that are cruising for parking can exceed 33% of all traffic in large crowded city centers (Shoup et al., 2005). The situation is getting worse, because the usage of cars is increasing, the cost of energy is getting higher and parking spaces are getting scarcer and more coveted. Indeed, several studies such as (Bayless and Neelakantan, 2012) have identified the importance of better parking systems to improve the quality of life, and an increasing number of smart parking solutions are being proposed to help optimizing the search for parking spots.

There is a growing conscience that cities are unable to cope with the continuous increases in car traffic, the management of parking spaces moved from being “important but rather narrow concerns about safety and the obstruction of traffic flow on the streets” into a consideration of the degree to which parking policy contributes to the wider economic, environmental and social policies of towns and cities (Valleley et al., 1997). Parking policies, if they are well designed, contribute to more efficient use of the transport network, higher densities, lower emissions, and better urban design (Shoup et al., 2005). If not, they can act in the opposite direction.

Three main objectives have been identified (Marsden, 2006):

- to use urban parking management as a means of regenerating specific parts of the urban area (e.g. town centre: providing more parking attracts business);
- to use parking controls to restrain traffic and improve environmental quality, or to encourage the use sustainable transports;
- to secure sufficient income from the parking operation to cover costs or to fund other activities.

These are the reasons that lead us to consider this problem in particular. We first propose an OLRA modeling of the urban parking problem, then we propose a multiagent solution to it.

2.3.2. An OLRA modeling of urban parking

In the urban parking application, the set of resources \mathcal{R} is homogeneous, and composed of the parking spots. At the start of execution, \mathcal{R} might be equal to \emptyset , and is enriched by the parking spots made available. The set of consumers \mathcal{C} is composed of the drivers. \mathcal{G} is the transport network of the considered town, region or neighborhood. The nodes of the network represent either a crossroad or a parking spot on an edge. The time horizon T is the considered timeframe for the execution, typically 24 hours. The function $availability(r, t)$ returns 1 from the moment when a spot r is free until the moment when it is occupied. Similarly, $availability(c, t)$ returns 1 from the moment when a driver c is seeking a parking spot.

The possible properties of a parking spot are its size in centimeters ($d_{size} = \mathbb{R}_+$), the rating of the neighborhood ($d_{neighborhood} = [0, 1]$) and its safety ($d_{safety} = [0, 1]$). A parking spot r_1 can have the following values: $size(r_1) = 200$, $neighborhood(r_1) = 0.9$ and $safety(r_1) = 0.9$. The *compatibility* function in the model allows to define the minimal conditions a driver has for a spot. For instance, a driver can be interested in the only safe spots ($safety \geq 0.7$), which size is longer than 2 meters and that are not further from his final destination than 500 meters. Among the resources matching these conditions, the drivers uses his utility μ to sort them following a preference criterion, for instance from the nearest to the furthest.

In this problem, the spots can be taken by anyone, but not more than one driver can take a spot. As a consequence, the parameter k is equal to 1. In addition, not more than one spot can be co-located with a driver, and one driver cannot take more than one spot at the same time. The parameter l is then also equal to 1.

In order to assess our proposal with an objective decision criteria, we have used a utility function that takes into account the time to reach the resources:

$$\mu(c, r, t) = \frac{1}{\tau(\rho(r, t), \rho(c, t), t)}$$

Following the value of l and k , each spot will be taken by at most one driver. If the driver leaves the spot and starts looking for another, they both will be considered as a new consumer and a new resource. Each driver c is

looking for one resource r and his problem is to find among the potential pairs $\{(r, t) \mid (c, r, t) \in \mathcal{F}\}$ the one maximizing $\mu(c, r, t)$.

Here follows a model of urban parking as an instance of OLRA.

node : a spot or crossroad

parking spot : a spot where one can park

$\mathcal{R} = \{\textit{parking spot}\}$

$\mathcal{C} = \{\textit{driver}\}$

\mathcal{G} = transport network

\mathcal{V} = the set of nodes in \mathcal{G}

\mathcal{D} = the distances between nodes in \mathcal{G}

T : execution timeframe

- $$\textit{availability}(r,t) = \begin{cases} 0 & \text{if } r \text{ is occupied} \\ 1 & \text{otherwise} \end{cases}, r \in \mathcal{R}$$
$$\textit{availability}(c,t) = \begin{cases} 1 & \text{if } c \in \mathcal{G} \text{ and wants to park} \\ 0 & \text{otherwise} \end{cases}, c \in \mathcal{C}$$

- $\rho : (\mathcal{R} \cup \mathcal{C}) \times T \mapsto \mathcal{V}$, returns the node where the designated resource or consumer is located

- $\tau : \mathcal{V} \times \mathcal{V} \times T \mapsto \mathbb{R}_+$, returns the travel time between two nodes

- μ : utility function that measures the usefulness of a spot towards a driver at time t

$$\mu(c,r,t) = \frac{1}{\tau(\rho(r,t), \rho(c,t), t)}$$

- $\mathcal{P} = \{\textit{size}, \textit{safety}, \textit{rating}\}$, the set of possible properties of spots in \mathcal{R}

- $\Delta = \{\mathbb{R}_+, [0, 1]\}$, set of possible description domains for the properties

- $\varrho(r) = [(\textit{size}, \mathbb{R}_+), (\textit{safety}, [0, 1]), (\textit{rating}, [0, 1])]$, $\forall r \in \mathcal{R}$, the property-domain pairs for spots.

a driver can only park on one spot at a time, and a spot is occupied by at most one driver, thus

- Parameters $k = l = 1$

- \mathcal{F} : set of tuples (*driver, parking spot, time*) such that *driver* and *parking spot* are co-located and available.

In the following, we propose a solution to the problem of urban parking that is based on a distributed architecture. In our configuration, there is no central information system nor an infrastructure that would list the available spots. Each driver has to expand his knowledge about the available spots by locally interacting with the other drivers. That means that we don't have a single accurate and up-to-date representation of \mathcal{F} , but each driver must build his own approximation of \mathcal{F} based on the knowledge that he has, which continuously evolve over time.

3. Multiagent solution

We propose a multiagent solution that helps finding parking spots in an urban agglomeration. This solution uses minimal information on shared, volatile and uncontrollable resources. The multiagent system works without initial information, without infrastructure to collect information about spots availability, and without a central information system (Bessghaier et al., 2012a).

3.1. Agents Model

The multiagent system proposed here is fully decentralized. Agents employ inter-vehicular communication to exchange information with the other vehicles, which have to belong to the same community of equipped vehicles. The choice of a distributed approach allows, among other advantages, to minimize the infrastructure needed to implement this solution and to limit investment.

The proposed system for the search of spots in an urban area is made of a type of agent designated by *assistant* agent, embedded in the driver's vehicle and supports him during his spot search. The assistant agent passes through four states as indicated by the automata of Figure 1.

- state 0: the vehicle is parked, the assistant agent is stopped.
- state 1: the driver has left his origin and is moving toward his destination.
- state 2: the assistant agent is looking for a parking spot to propose it to the driver of the concerned vehicle.

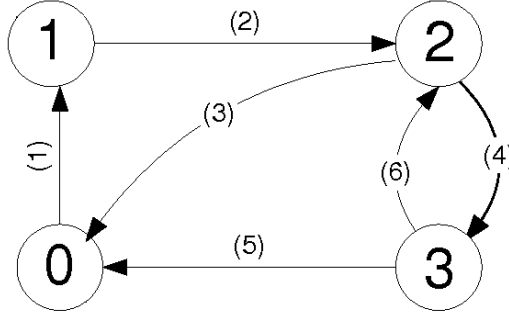


Figure 1: Assistant agent state diagram

- state 3: the driver moves toward the spot proposed by the assistant agent. The latter stays aware of possible alternatives which would be more suitable for the driver.

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

- state 0 (arc (5)), 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 and it is free.
- state 2 (arc (6)), the search cycle starts again. This happens when the driver arrives at the spot and finds out that it is taken (for instance, a driver from outside the community would have found and taken it)

The internal architecture of the assistant agent is composed of three modules: an *Itinerary* module, a *Communication* module and a *Decision* module. These modules are detailed in the next sections.

3.1.1. Itinerary module

The itinerary module calculates the route to the chosen parking spot starting from the driver's current position and then guides him to the spot. The choice of a parking spot is based on:

- the interaction with other assistant agents (*Communication module*)
- the choice of a spot maximizing μ and maximizing the chances to find the spot available (*Decision module*)

3.1.2. Communication module

This module enables the agent to communicate with his neighbors, which have to belong to the community as well. This communication is based on messages and allows to exchange information about the availability of parking spots.

Our choice to make the agents communicate via an inter-vehicular network allows the information exchanged to move following two vectors. The first is specific to the communication. Indeed, the messages exchange takes place between each two neighboring vehicles in the 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 their movements.

However, the broadcast of information within the community can lead to a deterioration of the quality and the effectiveness of the system. There is quality degradation if an isolated agent cannot access or share his information. It is the case for agent As_3 in the figure. But also if many agents choose the same spot as it may be the case for agents As_4 and As_6 . The effectiveness of the system can also 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 per agent should be less important than in a centralized architecture.

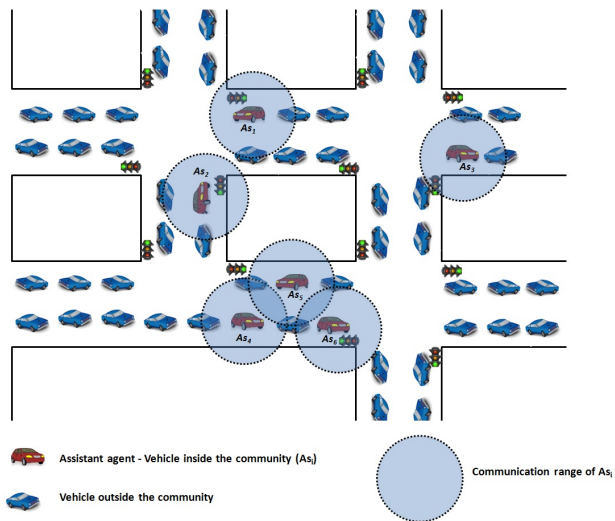


Figure 2: Information dissemination in the community through an inter-vehicular network

3.1.3. Decision module

Finally, the decision module takes care of two types of decision. With this module, the agent decides on which information it shares. It also chooses which parking spots are the most relevant for the driver. The main issues of the information management about parking spots concern, on the one hand, the definition and the use of the information and on the other hand the update of this information. These two elements influence the quality of the knowledge of the assistant agent and therefore the quality of the decision process. The decision is based on information that is given by the members of the community via communication and is related to two events: 1) a parking spot is released by a member of the community; 2) a parking spot is occupied by a driver belonging or not to the community.

The choice of a parking spot for the driver must meet his criteria, which may concern for instance its distance, the time since its release, or the safety of its location. In other words, the decision module of a driver c implements the utility function μ and computes his own approximation of \mathcal{F} (which defines the set of potential space-time co-locations of resources and consumers), both defined in the OLRA model. By abuse of notation, we use \mathcal{F} to denote the knowledge of the assistant agent about the available spots.

3.2. Cooperation model

The cooperation model depends on the nature of the information that is shared and how this information is used by the assistant agents. Section 3.2.2 presents a cooperation model where agents share information about parking spots while in section 3.2.3, the agents exchange in addition their intention about the parking spots. The shared information are:

- \mathcal{F} which contains information about the free spots and their associated release time.
- \mathcal{O} which contains the spots that were in \mathcal{F} but which turned out to be occupied with the moment that this information was known.

The minimal definition of a parking spot is a pair $\langle spot, time_{discovery} \rangle$: the geographic position of the spot and the moment when it was released. \mathcal{F} and \mathcal{O} contain each a set of such pairs that are exchanged between assistant agents. The combined use of the two sets 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 his \mathcal{F} set assumed to be free but, once there, he finds it occupied. In this case, the \mathcal{F} sets contain incorrect information about this spot. In this case, the concerned spot is moved from \mathcal{F} to \mathcal{O} and this information will spread over the community. Both sets are exchanged by the assistant agents and are updated gradually by the knowledge of each one, following two alternative cooperation models. The cooperative model influences the choice by the assistant agent of which spots in his \mathcal{F} to broadcast and which to hide. In the following paragraph, we describe the common behavior of the agents, regardless of the chosen cooperation model before to give the specific behavior to each model.

3.2.1. Generic behavior

The environment being dynamic, there is a “social update process” with information acquired in real time by communication, and also a “temporal update process” because the agent’s knowledge evolves over time. The social update process begins in the communication module of the assistant agent where the sets \mathcal{F} and \mathcal{O} from each received message (denoted \mathcal{F}_B and \mathcal{O}_B respectively) are extracted and forwarded to the decision module. This corresponds respectively to edges (1) and (2) in Figure 3. The decision module updates both sets by aggregating the received \mathcal{F}_B and \mathcal{O}_B sets with its own

\mathcal{F} and \mathcal{O} sets (denoted \mathcal{F}_A and \mathcal{O}_A respectively). The idea of the update process is to browse each received set (\mathcal{F}_B and \mathcal{O}_B) and update the local sets (\mathcal{F}_A and \mathcal{O}_A) in consequence, using the \oplus operator.

$$\begin{aligned}\mathcal{F}_A &\leftarrow \mathcal{F}_A \oplus \mathcal{F}_B \\ \mathcal{O}_A &\leftarrow \mathcal{O}_A \oplus \mathcal{O}_B\end{aligned}$$

The \oplus operator is defined as follows :

$$S_1 \oplus S_2 = S_1 \cup S_2 - \{ \langle spot, time_{discovery} \rangle \in (S_1 \cup S_2) \mid \langle spot, time_{discovery2} \rangle \in (S_1 \cup S_2) \wedge time_{discovery} < time_{discovery2} \}$$

where S_1 and S_2 are sets of $\langle spot, time_{discovery} \rangle$ pairs. The \oplus operator merges the two sets ($S_1 \cup S_2$), and if there are two information about the same spot ($\langle spot, time_{discovery} \rangle$ and $\langle spot, time_{discovery2} \rangle$), only the one with the newest $time_{discovery}$ is kept. Note that $time_{discovery}$ associated with a spot in \mathcal{O} is the time when the occupancy of the spot has been discovered by a driver.

Then an update of \mathcal{F}_A or \mathcal{O}_A is launched, using the \ominus operator.

$$\begin{aligned}\mathcal{F}_A &\leftarrow \mathcal{F}_A \ominus \mathcal{O}_A \\ \mathcal{O}_A &\leftarrow \mathcal{O}_A \ominus \mathcal{F}_A\end{aligned}$$

The \ominus operator is defined as follows :

$$S_1 \ominus S_2 = S_1 \setminus \{ \langle spot, time_{discovery} \rangle \in S_1 \mid \langle spot, time_{discovery2} \rangle \in S_2 \wedge time_{discovery} < time_{discovery2} \}$$

$S_1 \ominus S_2$ subtracts from S_1 the $\langle spot, time_{discovery2} \rangle$ pairs $\in S_2$ with an information about a spot in S_1 ($\langle spot, time_{discovery} \rangle$ and $\langle spot, time_{discovery2} \rangle$) with a newer $time_{discovery}$. This way, the information in \mathcal{F} and \mathcal{O} continuously concern different spots with the newest information available.

The temporal update process of the agents is a filtering of outdated information after a time θ from \mathcal{F} and \mathcal{O} . The spots that were discovered θ time ago are deleted. The value of the parameter θ has to be chosen taking into account the transport network activity. Thus, a low value reflects a high volatility (e.g. 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. This is done using the \odot operator.

$$\mathcal{F}_A \leftarrow \mathcal{F}_A \odot \theta$$

The \odot operator is defined as follows :

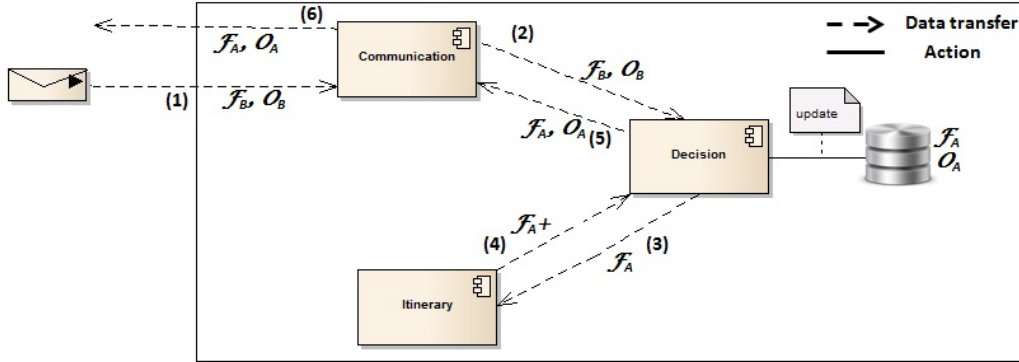


Figure 3: Assistant agent internal data flows

$S \odot \theta = S - \{\langle spot, time_{discovery} \rangle \in S \mid time_{discovery} < time_{current} - \theta\}$, with $time_{current}$ the current time.

The result of the application of $S \odot \theta$ will be the the $(\langle spot, time_{discovery} \rangle)$ pairs in S , where all information older then θ is filtered (all the pairs where $time_{discovery} < time_{current} - \theta$ will be absent from $S \odot \theta$).

When the driver is looking for a parking spot, \mathcal{F}_A is sent to the itinerary module. This corresponds to the edge (3) in Figure 3. The itinerary module computes the routes for each spot on this set and forwards the result back to the decision module (edge (4) in Figure 3). Based on the utility function μ , the decision module proposes a spot that best meets the needs of the driver. Finally, he sends \mathcal{F}_A and \mathcal{O}_A to the communication module (edge (5) in Figure 3) which takes care of their distribution to the neighbors (edge (6) in Figure 3). Now, with the chosen spot and the lists \mathcal{F}_A and \mathcal{O}_A , the assistant agent has to choose which information to broadcast to the neighboring vehicles. We have identified two possibilities for this information broadcast, following two cooperation models: “coopetitive” model and cooperative model, both described in the following paragraphs. Note that during the movement of the driver to the chosen spot, the assistant agent can receive new information about spots. That might make him suggest an alternative spot to the driver that would better meet his needs, following the generic behavior that we have just described.

Every agent makes a decision individually, and based on his own knowledge, regarding the spot choice. If agents A and B cannot exchange their information, because their communication fields did not overlap directly or

via other vehicles, they might head towards the same spots and there might be a conflict on the same spot. The cooperation models defined in the following sections will try to limit the occurrences of such a situation. In our experiments, we verify if this situation happens, with respect to the cooperation model in the one hand and to defined parameters in the other.

The itinerary module calculates the travel times based on the latest known status of the network. The chosen spot is then the best possible for the driver when the assistant take his decision. At each reception of information about available spots, the itinerary module recalculates the shortest paths with the new status of the traffic. There is a continuous replanning for the drivers until they reach the chosen spot, that makes these choices the best possible, provided the knowledge of the agent and the absence of a central planner.

3.2.2. “Coopetition” model

The “Coopetition” model is a combination of cooperation and competition (Luo, 2007). In our implementation of this model, agents are indeed cooperative, because they altruistically share information about spots. But they are also competitive, because they do not share an information about a spot if they are interested in it and planning to take it. When the decision module chooses a spot, it deletes the information corresponding to the proposed spot from \mathcal{F} . The removal of the information about this spot reduces its spread within the community. Thus, the assistant agent increases the driver’s chances of finding the spot free.

3.2.3. Fully cooperative model

In the fully cooperative model, the agents exchange the information about the spots, but they also exchange their preferences about the parking spots as well. Indeed, they broadcast their intentions to the other agents. In this model, \mathcal{F} contains tuples $\{ \langle spot, time_{discovery}, time_{reach}, time_{intention} \rangle \}$. The $time_{intention}$ and $time_{reach}$ define the moment when the agent took a decision to head toward a spot and the time needed for him at that moment to reach the spot. This way, every agent would know if he can be at a spot before another has reached it. If not, he would choose another spot with more chances for him to get it. The fields $spot$ and $time_{discovery}$ together with the θ parameter are still used to filter the \mathcal{O} and \mathcal{F} sets with the \oplus , \ominus and \odot operators.

4. Experiments

In a dynamic and online problem such as the urban parking problem, evaluating a new proposal is an issue. Indeed, optimal solutions are for static problems where the problem data do not change during execution. Optimal solutions also assume the presence of a central system that optimizes every vehicle’s route. To find optimal solutions for this problem, we would have to solve a static allocation problem with each modification in the problem data (traffic times, new driver, new available spot, etc.). Provided this difficulty, and since we propose fully distributed solutions, our objective in this experiments section is to compare the system behavior with the two cooperation models that we propose for the drivers (coopetitive and cooperative). In all the simulations, the two models are compared with a default model, where the drivers are not informed at all (called the reference simulation).

4.1. Configuration

To demonstrate the effectiveness and utility of our proposal, we have conducted many series of simulations. Our objective is to assess three hypothesis. The first hypothesis is that inter-vehicular communication has positive impact on urban parking. To do so, we compare simulations results with inter-vehicular communication and driving assistance, with simulations where drivers look for spots on their own. The second hypothesis is that the quality of the result depends on the quality of \mathcal{F} , thus declaring drivers intentions about spots is better for urban parking. To assess this hypothesis, we compare results with the coopetitive model versus results with the cooperative model. The third hypothesis is related to communication costs, and states that the number of messages per agent is less in a system with inter-vehicular communication than the number of messages treated by a central server guiding the drivers.

In all of these simulations, the criterion to minimize is the average time spent to find a spot. We use the road network of the city of Saint Etienne, France. We place 124 spots on this network. We have 300 agents in all the simulations, and we vary the number of agents in the community from 100 to 300 with a step of 20 agents. As a consequence, we vary the number of agents outside the community from 200 to zero.

We define two system parameters. The first parameter is θ , the information lifetime in the agent’s knowledge base. The second parameter is r , the radius of information broadcast around a vehicle. θ is expressed in simulation

time cycles while a value of $r = 1$ is proportional to 4.26 meters. To tune the two system parameters θ and r , we define mean values and vary the two parameters around these values. For r , the mean value is chosen so that the communication range of 200 agents (the mean number of considered agents) cover the whole network if placed optimally. The computed value is 12. We vary the values of r in $\{5, 10, 15, 20\}$, then in $\{5, \dots, 60\}$ for specific tests. For θ , we run a first set of experiments to find a mean search time, which we use as a mean value for θ . The computed value is 14. We choose the values of θ in $\{5, 10, 15, 20\}$, then in $\{5, 10, \dots, 35\}$.

4.2. Results

In the following four paragraphs, we test our three hypothesis and we assess the impact of the parameters θ and r .

4.2.1. Cooperative Vs. cooperative model

The first set of experiments is related to the comparison between three simulations:

1. a simulation with no intervehicular communication
2. a simulation with intervehicular communication and a cooperative model
3. a simulation with intervehicular communication and a cooperative model

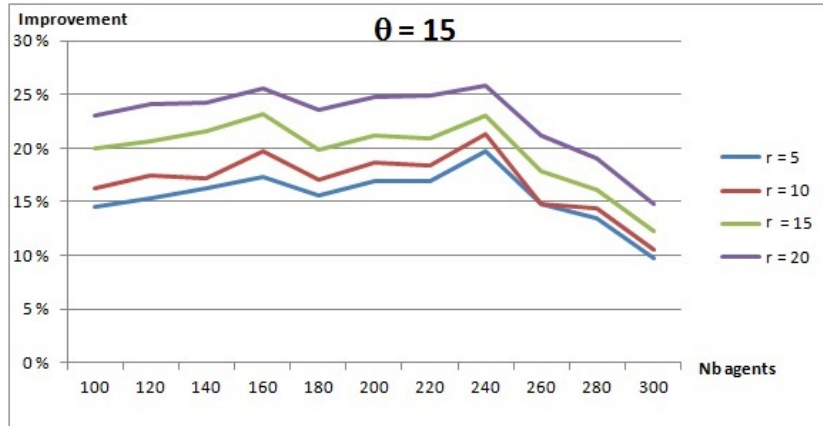


Figure 4: Cooperative model (with $\theta = 15$)

Figure 4 provides the results related to the cooperative model with $\theta = 15$. The results with $\theta \in \{5, 10, 20\}$ will be reported in the next subsection. The

x-axis reports the number of considered agents in each simulation and the y-axis reports the improvements w.r.t a simulation with no intervehicular communication. Each point in the different curves represents an average of 20 simulation runs. Results show that the cooperative model outperforms the reference simulation. However, starting from 240 agents, the difference between the two models becomes less important. This result validates the first hypothesis: using intervehicular communication and driving assistance is indeed beneficial for urban parking.

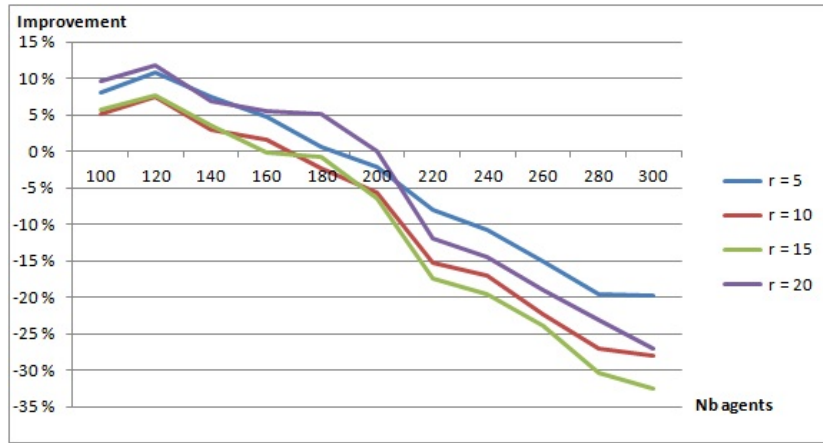


Figure 5: Coepetitive model (with $\theta = 15$)

An example of results related to the coepetitive model is reported in Figure 5, with $\theta = 15$. The results with $\theta \in \{5, 10, 20\}$ have the same trend, and suggest the same interpretation. The results show that the coepetitive model outperforms less and less the reference simulation when we consider from 100 to 180 agents. Starting from 200 agents, the reference simulation gives better results. That means that when the number of agents becomes widely greater than the number of available spots (3 agents per 2 spots), it becomes useless to use urban parking assistance using a coepetitive model. This is due to the fact that hiding the chosen spot by the agent might lead several agents to choose the same spot, which is called “multiple-car-chasing-single-space” (Shi et al., 2004), especially when the number of available spots becomes limited. The figure 6 shows the effect of the chosen model on the concentration of vehicles around spots. We see clearly that the cooperative model enables to limit this concentration. Hypothesis 2 is also valid.

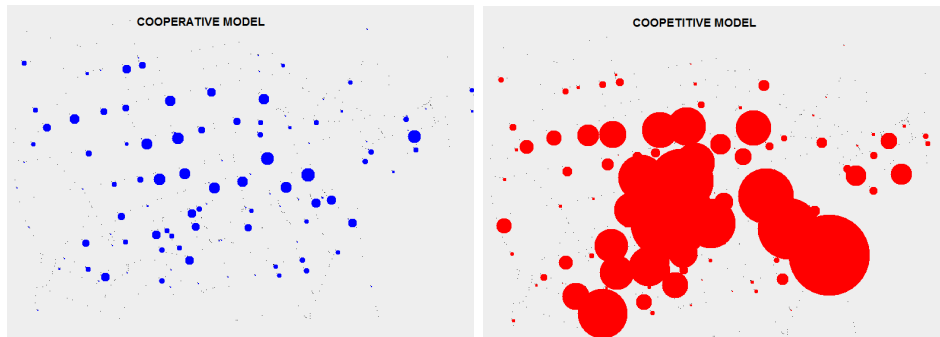


Figure 6: Density of the agents searching for a spot (300 agents and $\theta = 15$)

In the remainder of this experiments section, we use the simulation based on the cooperative model for investigating other aspects of the applications.

4.2.2. Impact of information lifetime

In the following series of simulations, we investigate the impact of the parameter θ , expressing the information lifetime, on the average search time. The results are reported in Figure 7. As we can see, a greater value of θ is beneficial for urban parking, since the improvements w.r.t the reference simulation are higher with higher values of θ . However, we observe a stagnation of the improvement beyond the mean value of $\theta = 15$. Indeed, the results are almost the same with $\theta = 15$ and $\theta = 20$. This result suggests that there is an optimum value of θ to be found, beyond which it is useless to keep information about spots. Indeed, high values of θ incur large data to be stored and exchanged between vehicles. If the marginal benefit of increasing θ becomes negligible, it is better not to increase it. This value has to be found and tuned for every considered region and each considered timeframe. Moreover, these results confirm that the use of θ reflects the volatility of the parking spots (see section 3.2.1). Indeed, Figure 7 shows that the increase of θ influences mostly the result with a low volatility, i.e. when the number of agents is less than 240.

4.2.3. Impact of communication field

In the following series of simulations, we investigate the impact of the parameter r , expressing the communication field of the vehicles. In an urban area, vehicles communication fields can be limited due to obstacles, the objective of our investigation is to check whether lower communication field

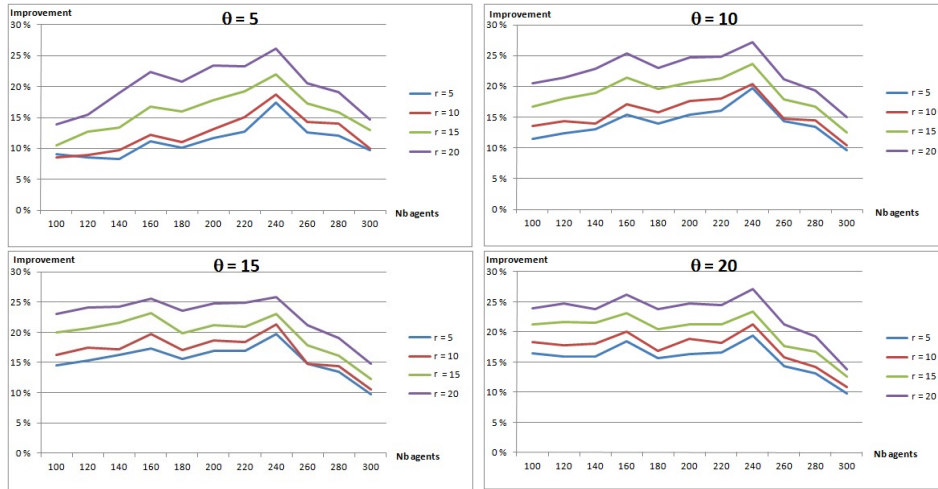


Figure 7: Impact of information lifetime, with $\theta = \{5, 10, 15, 20\}$

would significantly impact quality of service. Besides, it would be interesting to artificially limit that field since a higher value of r incurs higher number of exchanged messages and also more information to store and manage. Besides, the value of r cannot be increased indefinitely. Indeed, as the communication range r increases, it may cause communication interference and worsen the communication efficiency as well as the application built upon. In (Schmidt et al., 2011), it is specified that the degradation in VANETs starts to become significant starting from a range of 300 meters. In our simulation, $r = 60$ is equivalent to 255.6 meters and remains lower than the 300 meters threshold.

The results are reported in Figure 8. We have varied the value of r from 5 to 60 with a step of 5. As we can see it, a greater value of r is always beneficial, since the improvements w.r.t the reference simulation are higher with higher values of r . We also observe that the best results are with $\theta = 10$ and $\theta = 15$, higher values of θ provide worse results. However, whatever the value of θ , we observe less and less improvement starting from $r = 25$. There is a balance to find between the value of r and the number of incurred exchanged messages. This will be the object of the following paragraph.

4.2.4. Number of messages

In the centralized version, each agent informs a central server when he releases a parking spot or if he finds a proposed spot occupied. The central server performs the choice of the spots, based on the vehicle knowledge and

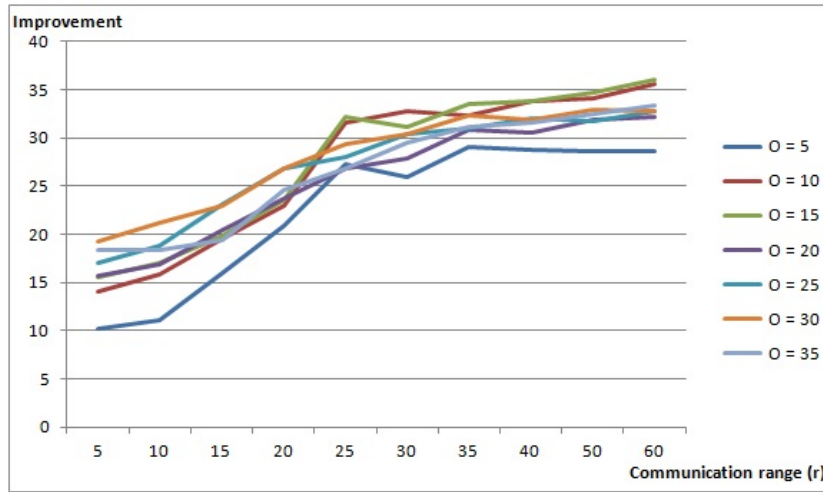


Figure 8: Impact of the variation of r

following the cooperative model. Thus, the central server mimics the same behavior of the cooperative model, and there is no difference in terms of travel times between the cooperative and central solutions. The objective is only to compare the number of messages per agent in the cooperative model with the number of messages manipulated by the central server.

We vary r from from 5 to 60 with a step of 5. The results are reported in Figure 9. Each point in the curve is the average of the simulation results with $\theta = \{0, 5, 10, \dots, 35\}$. As expected, a greater value of r incurs higher number of exchanged messages in the cooperative model (r has obviously no impact on the centralized version since vehicles don't communicate directly). However, this number remains inferior to the number of messages handled by a central server until $r = 45$. Beyond this value, the cooperative model generates more messages per agent than the centralized version.

For this series of experiments, it seems than a value of $r = 25$ is a good compromise between search time optimization (around 27% better than the reference simulation) and number of exchanged messages (around 2000, which is 5 times less than the results with $r = 60$). These results validate hypothesis 3.

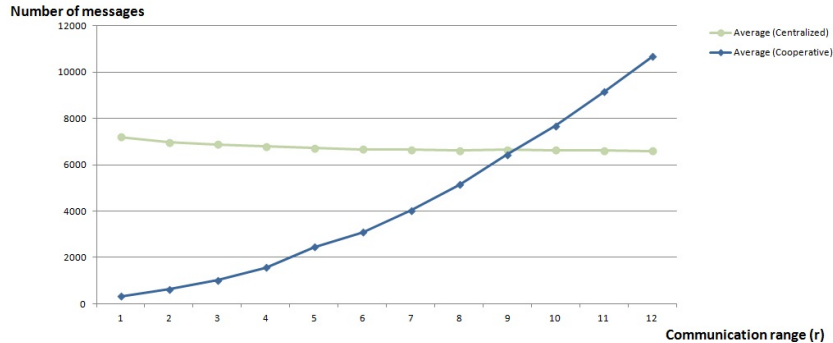


Figure 9: Number of messages (centralized vs. cooperative model)

5. Related works

In this section, we compare our proposal with the approaches of the literature. Section 5.1 discusses alternatives to our modeling of OLRA and section 5.2 discusses other solutions to urban parking management.

5.1. Alternatives in resource allocation problem modeling under space and time constraints

Several allocation problems face the constraints related to space or time constraints. The Online Resource Allocation (ORA) problems deal with the time constraint while the Localized Resource Allocation (LRA) deals with the space constraint. The knowledge about resources and/or consumers, but also the allocation process itself, can be constrained by the time or space properties of the problem. In this section, these two constraints are discussed, either when the space and time properties are considered separately (ORA and LRA) or together.

A first constraint in the problem considered in this paper is related to the availability of the information about resources and/or consumers during the allocation process. In ORA, the information is not known at the beginning of the allocation. The satellite exploitation (Narula-Tam et al., 2004) is an example of such problems, where image requests are submitted dynamically. The work in (Lardieri et al., 2007) is another example where the authors propose an architecture supporting dynamic resource management with a reallocation process to optimize and reconfigure system resources at runtime and to adapt to changing mission needs and resource status. In LRA, the information is known in a specific area. The domain that faces frequently this

kind of allocation problems is telecommunication. In Madan et al. (2010), the objective is to find the optimal spectrum and power allocation between users in a telecommunication cell. In Ta et al. (2005), the authors propose a real-time truck allocation process to enhance the efficiency of the hauling operation.

The second constraint concerns space. In smart grids for instance (e.g. (Hom-Chaudhuri and Kumar, 2011)), the allocation problems are generally LRA problems, since the objective of the allocation is to reduce the transactions price which depends strongly on the distance between resources and consumers. Another instance of space constraints of the allocation process is Dohler et al. (2004), since the objective in a multi-hop communication system is to optimize the bandwidth and power allocation to each relaying hop over spatially separated relaying mobile terminals to drastically increase end-to-end capacity.

When the allocation problem has space-time characteristics, one of them is often dominant. For instance, the space dimension is implicit if the modeling considers the time to reach the resource. Resource allocation in wireless cellular networks (e.g. in (Madan et al., 2010)) tackles the problem of wireless channels allocation in an online fashion because of the time-varying nature of the resources. The resource is localized but the main issue is related to its online nature.

This partial modeling does not enable to take into consideration simultaneously and explicitly the constraints related to space and to time. For instance, if the space is not modeled explicitly, we cannot express the fact that the consumer knowledge about a resource depends on the distance between them. This was the choice of the authors in (Topaloglu and Powell, 2005), where they model the problem of resource allocation related to fleet management. They reduce the problem to a dynamic allocation problem by decomposing it into sub-problems since the resource requests are known in well-determined time and area.

Our contribution is to introduce an explicit representation of the physical environment in the problem, and to consider both time and space constraints. This notably allows us to consider that consumers do not have complete knowledge about resources and their states following the space and time dimensions. To the best of our knowledge, such a modeling has not been proposed so far.

5.2. Urban parking management

Several studies 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. In the following paragraphs, we present the current research on this subject, from the proposals for closed parking, to outdoor and urban areas. These works rely mostly on sensors in the environment, some propose reservation methods, others rely on vehicle-to-vehicle interaction and multiagent approaches. This section is structured as follows: First, we present the algorithms that consider controlled environment such as closed parkings. Then we present the extension of these algorithms to outdoor environment. The next paragraph presents the reservation-based approaches. Intervehicular communication and agent based solutions for urban parkings are presented in the next two paragraphs.

A great number of the existing solutions for parking management detect parking spots in controlled environments such as closed parking. For instance, ParkSens³ is a parking spots locators system that senses the occupancy status of a parking spot with a magnetic sensor. Commercial systems for the indoor parking management have also been proposed. For instance, SIPARK is a parking management developed by Siemens that guides drivers to empty parking spaces in an indoor parking. The system monitors every single parking spot with ultrasonic detectors. Airports also provide more and more real-time information on available parking spaces to drivers, using electronic signs about parking availability (Shin and Jun, 2014). Even though these systems are very useful, they are however circumscribed to single protected areas or buildings.

In (Yoo et al., 2008), the authors have developed a wireless sensor network for outdoor environment to extend monitoring from indoor to outdoor. San Francisco is the first city to adopt this type of intelligent system. Indeed, it has deployed SFpark⁴ for the 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 transfers 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

³<http://www.parksens.net/>

⁴sfpark.org/

maps on the website dedicated to mobile applications. The authors in (Shin and Jun, 2014) also propose a guidance algorithm, in which information about spots is discovered via sensors. In (Giuffrè et al., 2012), the authors have proposed an architecture to provide parking management solutions for street parking, using sensor network. Proposals using central information and sensing infrastructure include (Mackowski et al., 2015) and (He et al., 2015). In (Samaras et al., 2014), the authors use a method for estimating the free size of parking spots using wireless sensors. In France, the first experiment took place in Lyon in early 2009 with the OptiSur project (Giraudon et al., 2013). The system analyzes the presence of vehicles and calculates the length of the parking spot 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 mobile devices. To consider real-time environment, the authors in (Shi et al., 2004) used fuzzy clustering of drivers, and assigned spots considering the preferences of each group of drivers. With these works, it is possible to manage parking spots for both indoor and outdoor environments in real-time. However, since they use sensors to collect information, they require significant investments, in contrast to the solution we propose in this paper.

To improve parking guidance, new processes such as online reservation have been adopted in parking management systems. The bay area rapid transit (BART), in San Francisco/Oakland metropolitan area, has been developed as a centralized intelligent reservation and real time availability system which provides parking availability via telephone and internet. The authors in (Teodorovic and Lucic, 2006) have also introduced a reservation function for urban parking. Geng and Cassandras (2012) have proposed a reservation system and dealt with an optimal allocation problem of parking spot considering parking cost and distance to destination. The e-car concept presented in (Hodel and Cong, 2003) offers the driver to book online a parking space. In (Chen et al., 2015), the authors consider a smartphone-based parking reservation system. In these solutions, it is assumed that information is perfect and some authors propose a program to achieve an optimum allocation of parking spaces. In our system, the spots are uncontrollable and such central reservation and optimization is neither desirable nor possible.

Intervehicular communication has also been used for parking guidance, in several ways. For instance, SPARK (Rongxing et al., 2009) is an application of smart car that offers a new way of parking for large parking spots using communication in a VANET. 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. In (Caliskan et al., 2006), 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. Readers interested in VANETs solutions for parking assistance are invited to refer to (Baldev and Ghosh, 2015) for a recent review. As we can see it, some studies have analyzed the contribution of VANETs to parking context (Szcurek et al., 2010; Vaghela and Shah, 2011), but very few researches explore the impact of VANETs on the optimization of the parking process (Tasserion and Martens, 2014).

Besides, several agent-based solutions have been proposed for urban parking management. For instance, the authors in (Benenson et al., 2008) have proposed an agent-based model simulating parking in a city. The purpose is to assess the impact of additional parking supply. The authors in (Chou et al., 2008) have also proposed an agent-based solution where the parking guidance is performed by negotiation between drivers and the parking based on parking price. Some other researchers have investigated driver behavior in urban parking environment. For instance, in (Lan and Shih, 2014), the authors have designed a phone-based system to track a driver’s trajectory to detect when they are about to leave their parking spot. This approach is complementary to ours, in which case, the information about spots availability would be created sooner. Some works have proposed parking spot search algorithms to give drivers the highest chances to find a free parking spot. For instance, in (Thompson and Richardson, 1998), the authors have proposed a model for estimating the utility of a car park and the choice model of a driver, taking into account access, waiting, direct and egress cost components. In (Thompsona et al., 2001), they have proposed a behavioral model of parking choice when faced with parking signs. Again, these works are complementary with ours, since they provide an extended choice model that could be integrated in our agents behaviors.

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 (Suhas et al., 2009), 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. Our proposal in this paper is even cheaper, since vehicles do not scan the environment looking for parking spots, this information being generated through actual releases of spots.

6. Conclusion

In this paper, we propose a modeling for the resources allocation problem taking into account simultaneously the location and the moment when the resources are available. Our modeling is well adapted to the transportation domain where many applications are characterized by the difficulty to take into account their space-time dimension. Our modeling is able to take into account several kinds of constraints: i) space constraints: the resources or consumers have static or dynamic positions?; ii) time constraints: do the availability of the resources and needs of consumers change overtime?; iii) space-time constraints: do the resources and the consumers have to be at the same location?

We have used our modeling to specify the management of parking spots in an urban area and proposed a multiagent solution. The system is based on 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 places and no central storage of information. The distributed solution also allows to provide quick spots proposals to the drivers, since assistant agents do not perform complex allocation, but only computes and compares shortest paths.

Our experiments have mainly validated three hypothesis: inter-vehicular communication has positive impact on urban parking (with a cooperative model), declaring drivers intention about spots is better for urban parking and the number of messages per agent is less important than for a central server (until a threshold of communication range).

We are considering different perspectives to this work. Among them, we want to improve the decision function of the agents in order to take into account other criteria than the time and space and to evaluate how the aggregation of their preference impacts the quality of the solutions.

References

- Acha, S., Green, T. C., Shah, N., 2011. Optimal charging strategies of electric vehicles in the uk power market. Innovative Smart Grid Technologies (ISGT), 2011 IEEE PES, 1 – 8.
- Baldev, D. S., Ghosh, P. A., 2015. Review of parking spaces assistance schemes in vehicular networks. ETCEE–2015, 17.
- Bayless, S. H., Neelakantan, R., December 2012. Smart parking and the connected consumer: Opportunities for facility operators and municipalities. Tech. rep., ITS America, Washington, DC 20036 USA.
- Beasley, J. E., Krishnamoorthy, M., Sharaiha, Y. M., Abramson, D., 2000. Scheduling aircraft landings the static case. *Transportation science* 34 (2), 180–197.
- Benenson, I., Martens, K., Birfir, S., 2008. Parkagent: An agent-based model of parking in the city. *Computers, Environment and Urban Systems* 32 (6), 431 – 439, *geoComputation: Modeling with spatial agents*.
- Bessghaier, N., Zargayouna, M., Balbo, F., 2012a. Management of urban parking: An agent-based approach. In: Ramsay, A., Agre, G. (Eds.), *Artificial Intelligence: Methodology, Systems, and Applications*. Vol. 7557 of *Lecture Notes in Computer Science*. Springer Berlin Heidelberg, pp. 276–285.
- Bessghaier, N., Zargayouna, M., Balbo, F., 2012b. Online localized resource allocation application to urban parking management. In: *Proceedings of the The 2012 IEEE/WIC/ACM International Joint Conferences on Web Intelligence and Intelligent Agent Technology - Volume 02. WI-IAT '12*. IEEE Computer Society, Washington, DC, USA, pp. 67–74.
- Caliskan, M., Graupner, D., Mauve, M., 2006. Decentralized discovery of free parking places. In: *Proceedings of the 3rd international workshop on Vehicular ad hoc networks. VANET'06*. ACM, pp. 30–39.

- Chen, Z., Yin, Y., He, F., Lin, J. L., 2015. Parking reservation for managing downtown curbside parking. *Transportation Research Record* 2 (2498), 1218.
- Chou, S.-Y., Lin, S.-W., Li, C.-C., 2008. Dynamic parking negotiation and guidance using an agent-based platform. *Expert Systems with Applications* 35 (3), 805 – 817.
- Delot, T., Cenerario, N., Ilarri, S., 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, pp. 30:1–30:8.
- Delot, T., Ilarri, S., Lecomte, S., Cenerario, N., 2013. Sharing with caution: Managing parking spaces in vehicular networks. *Mobile Information Systems* 9 (1), 69–98.
- Dohler, M., Gkelias, A., Aghvami, H., 2004. A resource allocation strategy for distributed mimo multi-hop communication systems. *Communications Letters, IEEE* 8 (2), 99–101.
- Geng, Y., Cassandras, C. G., 2012. A new smart parking system infrastructure and implementation. *Procedia-Social and Behavioral Sciences* 54, 1278–1287.
- Giraudon, C., Ruchet, S., Thébert, M., Aug. 2013. *Projet Optisur. Optimisation de la surveillance du stationnement sur voirie, rapport final de contrat, Predit, juin*. Tech. rep., Predit, In French.
- Giuffrè, T., Siniscalchi, S. M., Tesoriere, G., 2012. A novel architecture of parking management for smart cities. *Procedia-Social and Behavioral Sciences* 53, 16–28.
- Golkar, B., Sousa, E., 2011. Adaptive localized resource allocation with access point coordination in cellular networks. In: *IEEE International Conference on Communications (ICC)*. pp. 1–5.
- He, F., Yin, Y., Chen, Z., Zhou, J., 2015. Pricing of parking games with atomic players. *Transportation Research Part B: Methodological* 73, 1 – 12.

- Hodel, T. B., Cong, S., 2003. Parking space optimization services, a uniformed web application architecture. In: ITS World Congress Proceedings. pp. 16–20.
- HomChaudhuri, B., Kumar, M., 2011. Market based allocation of power in smart grid. In: American Control Conference (ACC), 2011. IEEE, pp. 3251–3256.
- Katzev, R., 2003. Car sharing: A new approach to urban transportation problems. *Analyses of Social Issues and Public Policy* 3 (1), 65–86.
- Lan, K.-C., Shih, W.-Y., 2014. An intelligent driver location system for smart parking. *Expert Systems with Applications* 41 (5), 2443 – 2456.
- Lardieri, P., Balasubramanian, J., Schmidt, D. C., Thaker, G., Gokhale, A., Damiano, T., July 2007. A multi-layered resource management framework for dynamic resource management in enterprise dre systems. *Journal of Systems and Software* 80, 984–996.
- Luo, Y., 2007. A coopetition perspective of global competition. *Journal of World Business* 42 (2), 129 – 144.
- Mackowski, D., Bai, Y., Ouyang, Y., 2015. Parking space management via dynamic performance-based pricing. *Transportation Research Part C: Emerging Technologies* 59, 66 – 91, special Issue on International Symposium on Transportation and Traffic Theory.
- Madan, R., Boyd, S. P., Lall, S., Jun. 2010. Fast algorithms for resource allocation in wireless cellular networks. *IEEE/ACM Trans. Netw.* 18, 973–984.
- Marsden, G., 2006. The evidence base for parking policiesa review. *Transport Policy* 13 (6), 447 – 457, parking.
- Narula-Tam, A., Macdonald, T., Modiano, E., Servi, L., 2004. A dynamic resource allocation strategy for satellite communications. In: In proceeding of: Military Communications Conference. Vol. 3. IEEE, pp. 1415 – 1421.
- Rongxing, L., Xiaodong, L., Haojin, Z., Xuemin, S., 2009. Spark: A new vanet-based smart parking scheme for large parking lots. In: *IEEE INFOCOM*. pp. 1413–1421.

- Samaras, I. K., Arvanitopoulos, A., Evangelidou, N., Gialelis, J. V., Koubias, S. A., 2014. A fuzzy rule-based and energy-efficient method for estimating the free size of parking places in smart cities by using wireless sensor networks. In: Proceedings of the 2014 IEEE Emerging Technology and Factory Automation, ETFA 2014, Barcelona, Spain, September 16-19, 2014. pp. 1–8.
- Schmidt, R. K., Kloiber, B., Schüttler, F., Strang, T., 2011. Degradation of communication range in vanets caused by interference 2.0-real-world experiment. In: Communication Technologies for Vehicles. Springer, pp. 176–188.
- Shi, Han, B., Wang, J., 2004. Study of the mode of real-time and dynamic parking guidance and information systems based on fuzzy clustering analysis. In: Proceedings of 2004 International Conference on Machine Learning and Cybernetics, 2004. IEEE, pp. 2790 – 2794.
- Shin, J.-H., Jun, H.-B., 2014. A study on smart parking guidance algorithm. Transportation Research Part C: Emerging Technologies 44, 299 – 317.
- Shoup, D. C., Association, A. P., et al., 2005. The high cost of free parking. Vol. 206. Planners Press Chicago.
- Suhas, M., Sanjit, K., Gruteser, M., 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. pp. 25–28.
- Szczurek, P., Xu, B., Wolfson, O., Lin, J., Rishe, N., 2010. Learning the relevance of parking information in vanets. In: Proceedings of the seventh ACM international workshop on VehiculAr InterNETworking. ACM, pp. 81–82.
- Ta, C. H., Kresta, J. V., Forbes, J. F., Marquez, H. J., 2005. A stochastic optimization approach to mine truck allocation. International journal of surface mining, reclamation and environment 19 (3), 162–175.
- Tasseron, G., Martens, K. ans Van der Heijden, R., November 2014. The potential impact of in-car information on urban parking. the case of spatial heterogeneity and heterogeneous driver behavior. Tech. rep., TRAIL Research School, Radboud University Nijmegen, The Netherlands.

- Teodorovic, D., Lucic, P., 2006. Intelligent parking systems. *European Journal of Operational Research* 175 (3), 1666 – 1681.
- Tesauro, G., 2005. Online resource allocation using decompositional reinforcement learning. In: *Proceedings of the 20th national conference on Artificial intelligence*. Vol. 2. AAAI Press, pp. 886–891.
- Thompson, R. G., Richardson, A. J., 1998. A parking search model. *Transportation Research Part A: Policy and Practice* 32 (3), 159 – 170.
- Thompsona, R. G., Takadab, K., Kobayakawab, S., 2001. Optimisation of parking guidance and information systems display configurations. *Transportation Research Part C: Emerging Technologies* 9 (1), 69 – 85.
- Topaloglu, H., Powell, W. B., March 2005. A Distributed Decision-Making Structure for Dynamic Resource Allocation Using Nonlinear Functional Approximations. *Operations Research* 53, 281–297.
- Vaghela, V. B., Shah, D., 2011. Vehicular parking space discovery with agent approach. In: *Proceedings of the International Conference & Workshop on Emerging Trends in Technology*. ACM, pp. 613–617.
- Valleley, M., Jones, P., Garland, R., Macmillan, A., 1997. *Parking perspectives: A sourcebook for the development of parking policy*. Landor Publishing.
- Yoo, S.-E., Chong, P. K., Kim, T., Kang, J., Kim, D., Shin, C., Sung, K., Jang, B., 2008. Pgs: Parking guidance system based on wireless sensor network. In: *3rd International Symposium on Wireless Pervasive Computing, 2008. ISWPC 2008*. IEEE, pp. 218 – 222.