2018 IEEE Intelligent Vehicles Symposium (IV)
Changshu, Suzhou, China, June 26-30, 2018

Decision-Theoretic Cooperative Parking for Connected Vehicles: an Investigation

Ali Aliedani[1] and Seng W. Loke[2,1]
[1]Department of Computer Science & Information Technology, La Trobe University, Australia
[2]School of Information Technology, Deakin University, Australia

***Abstract*— In this paper, we investigate a cooperative car parking mechanism called *CoPark-WS*, for finding parking spaces in large areas. It is a decentralised approach which relies on vehicle-to-vehicle communications. There are two parameters we consider here in parking search behavior, namely, searching time and walking distance. The type of strategic cooperation and rational decision-making involved in selecting the search target areas in *CoPark-WS* are demonstrated via extensive simulations, which also show the robust performance of *CoPark-WS* in different circumstances.**

I. Introduction

Vehicles streaming to a car park area at peak hour is commonly associated with significant time consumed due to traffic congestion induced by vehicles circulating, as they search for car park spaces. It is estimated, based on IBM's global survey in 2011, that 20 minutes is the average elapsed time for parking in coveted spots.[1] In general, the lack of car park information and the rising contention levels among searching vehicles are the critical factors in the parking struggle.

Numerous car parking approaches have been proposed. They can be classified based on a car parking infrastructure's role in the allocation task, namely, full-support, partial-support, or no-support.

The decentralized no-supporting infrastructure car park approach aims to coordinate the distribution of vehicles to parking spots relying only or mainly on the vehicles themselves, without (or with little) expensive infrastructure. It can provide scalable performance and can be more economical in contrast to a centralized approach. However, a decentralized approach would need to address the concerns of limited parking information and competition.

In this paper, we introduce a decentralized coordination approach to finding car park spots, applicable to large car park areas, which is a Cooperative car Parking approach that takes into account Walking distance and parking Search time (which we call *CoPark-WS*), and assumes vehicle-to-vehicle communications. A vehicle is assumed to have communication capabilities to extend its observation area by receiving messages from other vehicles, and can make rational decisions based on messages received. The vehicle takes its own responsibility to update its knowledge and to select the target slot based on strategically cooperating with neighbours. Vehicle decision-making takes into account walking distance from where it gets parked to its destination (e.g., a building entrance), as well as car park searching time.

Previously, in [1], we discussed a decentralised car parking allocation mechanism inside a car park by supporting the vehicles with initial information about the position of available vacant slots at the car park gate as given by a RSU at the gate, and the vehicles select their target parking slots and cooperate with others by sharing their targets/intentions about where to park and have competing targets resolved. Here, we take into account walking distance to destinations as well as searching time, and study a different kind of parking area, which is highly spatially distributed and partitioned. We also study different factors that can affect performance including vehicle densities, transmission range and limited participation in cooperation in this context, and consider a different vehicle decision-making model.

II. The *CoPark-WS* Approach

During busy times, a large number of vehicles could be cruising to find car park spaces in the same places, causing a built up of traffic, and with no means of cooperation, cars might compete with one another for similar spaces even if there are good alternatives available. Significant time is then used for parking.

One common way to model the vehicle searching behavior, as in [2][3], is to view the decision to park as being built on the current belief of the vehicle about the availability of vacant parking spots based on what the vehicle itself observs while travelling. The *CoPark-WS* approach builds on this base knowledge and more to make a rational decision. It changes its searching behavior based on vehicle-to-vehicle cooperation. Through analyzing the received messages from neighbours and a new type of cooperative communications including sharing intention (of where to park) and introducing advice (to other vehicles about where to look), the vehicle can decide on an appropriate search target area for itself. The vehicles attempt to reduce the searching time to find parking within acceptable levels of distance to walk to the target building from the parked car, when there is a large number of vehicles entering the car park at nearly the same time (e.g., modelling peak hour traffic).

It is supposed that a software agent is installed in each vehicle. And each vehicle is supported with navigation facilities, which consist of static information of a car park map with GPS capability, and sensors to detect the near

[1]https://www-03.ibm.com/press/us/en/pressrelease/35515.wss

vacant parking spots. The vehicle starts the seeking task with an initial belief about the number of free spots at the car park. This initial information can be assumed to be collected from a Road Side Unit (RSU) at the car park entrance which can be estimated from the historical data - this value can be inaccurate to start with.

Figure 1 illustrates the *CoPark-WS* algorithm in a flowchart. The car park area is divided into sub-areas $A = \{a_1, a_2, ..., a_M\}$ based on road segments (a block) and sorted based on the proximity to the destination (e.g., a building entrance). It is supposed that each area has the same number of free spots S. Intuitively, the areas nearer to a destination that has high-interest value to the vehicles would have fewer free spots. This estimate of the number of free spots in a sub-area can be updated when the vehicle visits the sub-area or or via messages from vehicles with experience of parking availability in the sub-area. Firstly, each vehicle will select the sub-area nearest to its destination as a target area, and go towards it, and will broadcast (to other within range vehicles) INFO messages consisting of the selected target area and its own vehicle location information. Accordingly, each vehicle can determine its opportunity to park in a target sub-area based on computing its distance to the target area compared to the distance of other vehicles intending to go to the same area - the vehicle goes to the target area as long as it finds itself to be the nearest, a simple *contention resolution strategy*. A vehicle can lose the chance to park in an area in case in finds that there are S vehicles closer to the area, from the vehicle's perspective - such an area is called a *lost area*. Let M and $O(k)$ represent the total number of areas and the number of lost areas according to a vehicle (or an agent)'s belief at time k, respectively.

Each vehicle weighs the available areas based on our proposed utility function $u(a)$ and selects the area has highest utility value to be the next target area.

$$\forall a_x, a_y \in A, a_x \succeq a_y \iff u(a_x) \geq u(a_y)$$

The utility function assesses an area a from two perspectives: the proximity of the area from the final destination (e.g., a building entry) $u_{dest}(a)$ and the proximity of the area from the vehicle's own location $u_{veh}(a)$. Such assessment of areas can then be used to rank areas and the highest ranked area is the area selected by the vehicle as the target area which it moves towards. The vehicle continually reranks the areas as it receives messages from other vehicles or when it reaches the area itself (and "sees" the conditions in that area):

$$u(a) = u_{dest}(a) + u_{veh}(a)$$

$$u_{dest}(a) = \alpha \times A_a(k) \times V(k) \times \mathbf{I}_a$$

$$u_{veh}(a) = \beta \times (1 - V(k)) \times (\omega_a / (\omega_a - \rho_a(k))) \times \mathbf{J}_a(k)$$

The utility of an area from the perspective of being near to the final destination is represented by $u_{dest}(a)$, which takes into account the availability of free spots in the area as believed by the vehicle at time k, denoted by $A_a(k)$, the fraction of non-lost sub-areas out of all sub-areas as believed by the vehicle at time k, denoted by $V(k)$, and a weight inversely related to the area's ranking with regards to its distance from the final destination $\mathbf{I}_a$ (the nearer the area is to the final destination (e.g., building entrance), the higher the $\mathbf{I}_a$ value). Availability in a sub-area a is given by $A_a(k) = MAX\{0, (\omega_a - \tau_a(k) - \rho_a(k))/\omega_a\}$, where ω_a is the initial number of slots in area a (as obtained by the vehicle at a car park entrance), $\tau_a(k)$ is the number of vehicles looking to park in the area as determined by the agent based on received INFO messages at k, and $\rho_a(k)$ is number of occupied slots in the area based on its own scanning of area a as obtained up to time k. And $V(k) = \frac{M-O(k)}{M}$. Note that our approach is approximate, so that due to partial knowledge or inaccurate information, the values $A_a(k)$ and $V(k)$ are only the vehicle/agent's own belief and may not exactly correspond to the real world situation. Hence, our approach is a heuristics based approach, based on uncertain partial knowledge.

The utility of parking in an area nearer to the vehicle is represented by $u_{veh}(a)$. The area is valued taking into account the area order based on distance to vehicle $\mathbf{J}_a$ (where the nearer the area is to the vehicle, the higher the $\mathbf{J}_a$ value), the fraction of lost areas $(1 - V(k))$, and the inverse of the occupied parking spot ratio at the area based on its own scanning, i.e. $\omega_a/(\omega_a - \rho_a(k))$. The intuition is that when the fraction of non-lost areas is high (i.e., $V(k)$ is high, in a not so busy situation), then $\mathbf{I}_a$ values will play a greater role in computing the overall utilities u for the areas, than $\mathbf{J}_a(k)$ which are then suppressed when computing the overall utility, i.e., ranking of areas is based more on $u_{dest}(a)$ and the vehicle tends to go towards areas nearer the destination. But when busy, then areas nearer to the vehicle are now more important in the rankings as the vehicle seeks parking sooner, rather than try to get near the building, i.e., ranking of areas is based more on $u_{veh}(a)$. The factors α and β are human factors and $\alpha + \beta = 1$, which can be set to capture the pessimism or optimism of the vehicle, but in the following experiments, both values are set to 0.5.

Apart from the simple contention resolution strategy mentioned earlier, *CoPark-WS* uses another type of vehicle cooperation based on the idea of a vehicle giving advice to another vehicle. A vehicle P can send an advice message (ADV) to another vehicle Q in case of realising, via an INFO message from Q, that Q is going to an area that it knows is already fully occupied. ADV is a unicast message and consists of an identifier of the occupied area and a proposed alternative target area (that P issues to Q, say). The proposed target area given by P to Q is the second preferred car park area of P based on its utility function calculations. The receiving vehicle Q's duty, firstly, is to revise its knowledge of parking areas and then can accept or reject the proposed area as a new target area based on its own knowledge so far. The advisee vehicle Q can reject the recommended target area and select other areas based on its own utility function calculations in case it perceives that the proposed area is also occupied. Then, it sends a reply message to the advisor P to inform it that its proposed area is occupied. Through this

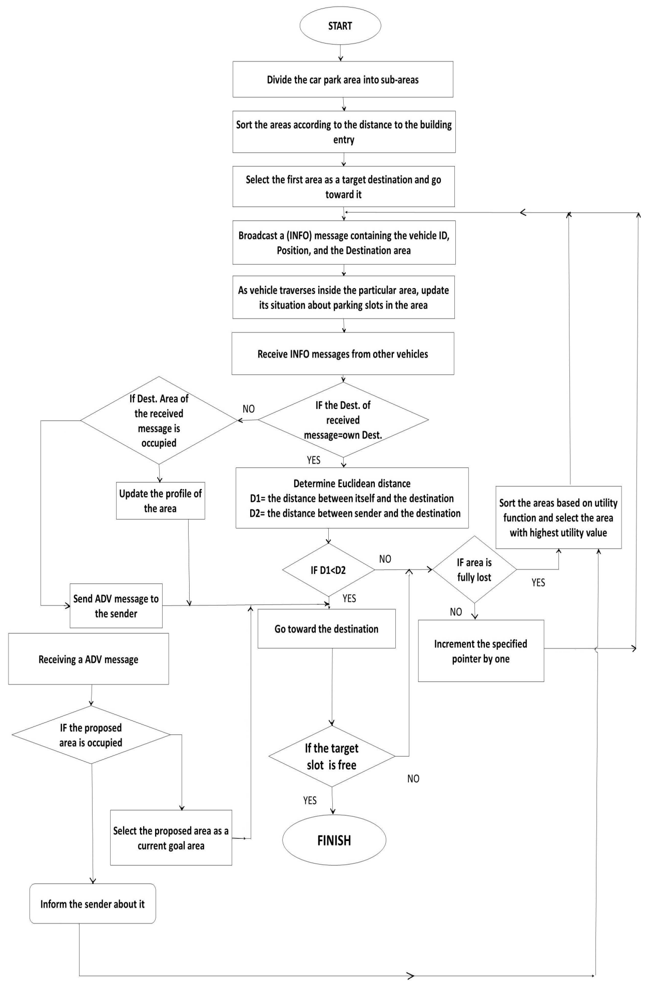

Fig. 1. The *CoPark-WS* flowchart

advising strategy, the vehicles can help each other update and build up their own car park knowledge.

III. Experimentation

We evaluated *CoPark-WS* extensively in various car parking circumstances via a simulation tool consisting of JADE (Java Agent Development framework) [4], SUMO (Simulation of Urban Mobility) [5] and TraSMAPI (Traffic Simulation Manager Application Programming Interface) [6]. An agent is an autonomous software entity called a Driver Agent, representing each vehicle, and has been developed using the JADE framework. The agent assumed to be installed in a vehicle updates its strategy based on the collected real-time messages from other vehicles (agents), and data from the environment as provided by the SUMO simulator. The simulation scenario is described in the next section followed by the simulation results.

A. Simulation scenario configuration

The large car parking area employed in experiments is shown in Figure 2. The car park map captured is based on

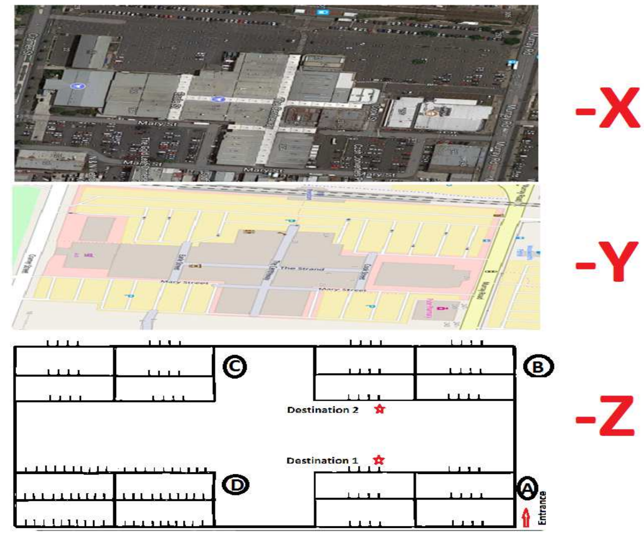

Fig. 2. Car parking lots scene

the area around the Preston market in Melbourne, Australia, which is often busy. The figure shows the car park map viewed in Google maps, OpenStreet Map, and the SUMO representation we built, respectively. For simplicity, it is supposed that there is one entrance gate for incoming vehicles, two final destinations (building entries) marked with red stars, and there are four car parking sections, as in Figure 2-Z. The car park sections close to the destinations have limited free spots. The car parking spots are distributed among sections A, B, C, D with 12, 12, 12, 60 spots, respectively. The parking spots nearer to a destination has higher priority for vehicles as they all want to park near the building entry. The vehicles are labeled based on their order of arrival. The vehicles with even IDs are set to go to destination 1 and the others go to destination 2.

For comparison, an algorithm CoPark-GD (or Cooperative car parking with Greedy behavior) is designed, where each vehicle is provided with initial information of the locations of free parking spots and is assumed to have wireless communication capability that is used to share intentions (i.e., target areas/spots) among vehicles via broadcasting INFO messages. Here, in contrast to *CoPark-WS*, each vehicle's objective is simply to park as near to the destination as possible, regardless. It uses the first-come-first-park rule to resolve contentions to determine who parks in a given parking spot.

B. Searching time

Searching time can be defined as time consumed from the time the vehicle enters the parking area via the entrance to find a parking spot. The searching time of *CoPark-WS*, CoPark-GD, and GD (which is the same as the CoPark-GD strategy but with the message transmission range set to 50m, i.e. modelling minimal or no cooperation) for 100 vehicles are determined from simulations. The average, maximum, minimum, median, and standard deviation values are illustrated in Figure 3. The dominance of CoPark-GD's performance over GD demonstrates the advantage of cooperation. However, *CoPark-WS* achieves better performance than CoPark-GD. For example, the maximum values of searching time of vehicles in *CoPark-WS* and CoPark-GD are 4753, and 10252 simulation time units (stu), respectively.

For reliability, we repeated simulation runs in the experiments and results were consistent across runs. Table 1 shows the superior performance of *CoPark-WS* over CoPark-GD in four runs, using the percentage of the average values computed with:

$$\frac{AVG_{CoPark-GD} - AVG_{CoPark-WS}}{AVG_{CoPark-GD}}$$

Attributes	Run-1	Run-2	Run-3	Run-4
AVG	0.55	0.58	0.54	0.57
Max	0.64	0.67	0.41	0.52
Min	0.36	0.38	0.31	0.39
Median	0.55	0.6	0.54	0.51
Std	0.69	0.7	0.63	0.7

TABLE I

COMPARISON OF SEARCHING TIME BETWEEN *CoPark-WS* AND COPARK-GD IN FOUR ITERATIONS.

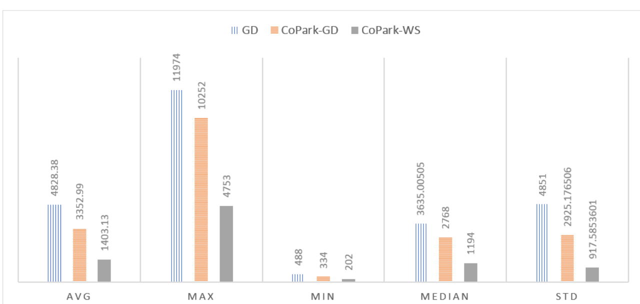

Fig. 3. The searching time comparison among CoP-WS, CoPark-GD, and GD approaches with 100 vehicles.

C. The effect of vehicle density

Increasing the number of vehicles in the car park area can increase the rate of vehicle competition and alter the park spot occupancy situation. In this experiment, *CoPark-WS* and CoPark-GD approaches are evaluated with different vehicle densities. The transmission range of the vehicle is set to 300m. The average searching time is computed as shown in Figure 4. When the number of vehicles is less than the number of free spots nearer to the destination, the greedy behaviour obtained a slight benefit over cooperation, i.e. with 25 vehicles, the average searching time in CoPark-GD is less than at *CoPark-WS*. Note it is supposed that in the run with vehicles implementing CoPark-GD, vehicles have accurate initial information, while vehicles in the *CoPark-WS* run have uncertain initial belief. However, there is degradation in CoPark-GD's performance with increasing vehicle density. At the same time, *CoPark-WS* shows quasi-stability and superior performance. For example, when the number of vehicles is 100, the average searching times were 1000 and 2800 stu for *CoPark-WS* and CoPark-GD, respectively.

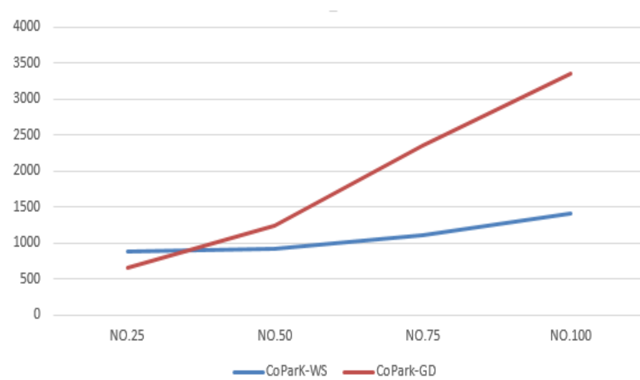

Fig. 4. The average searching times for different vehicle densities (25, 50, 75, 100) with vehicles implementing CoP-WS and CoPark-GD approaches.

With regards to walking distance, Figure 5 illustrates the difference in walking distances of individual vehicles comparing *CoPark-WS* and CoPark-GD with 50 vehicles. The walking distance is the distance between the parking slot where the vehicle is parked to a final destination. The difference is computed as the walking distance of a vehicle in *CoPark-WS* minus the walking distance of the same vehicle when implementing CoParkGD, the outcome of the same ID-ed vehicle in different simulation runs. A negative value means that *CoPark-WS* achieves less walking distance and vice versa. From the figure, there are 12 vehicles parked at same distance parking slots in both approaches and the remainder divides equally among positives and negatives. However, the average searching time in *CoPark-WS* is 919 stu, whereas in CoPark-GD, it is 1245 stu.

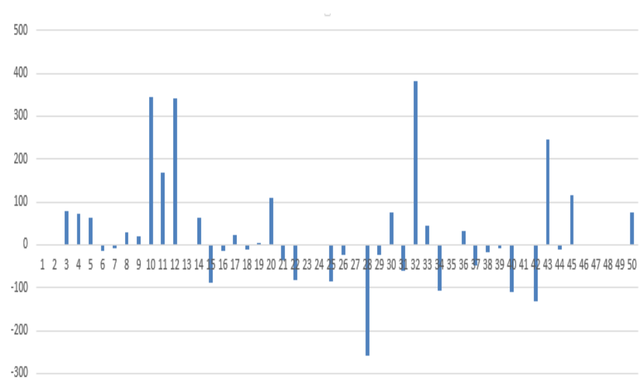

Fig. 5. The walk distance differences of individual vehicles comparing *CoPark-WS* to CoPark-GD.

D. The effect of transmission range

The transmission range is one communication parameter that can be adapted to expand the knowledge level about the surrounding environments. In this experiment, we study the

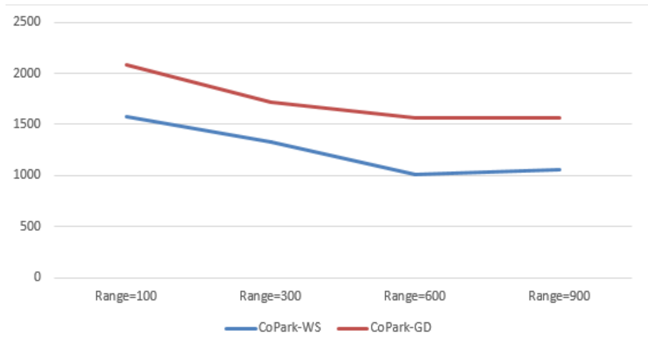

Fig. 6. The average searching time of 60 vehicles implementing CoP-WS and CoPark-GD approaches at various transmission ranges (100, 300, 600, 900m).

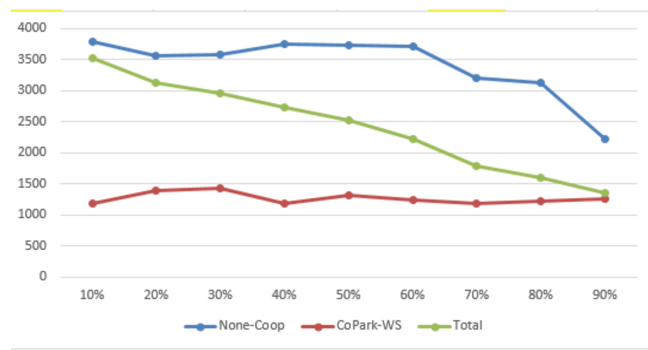

Fig. 7. The average searching time of total (80), non-cooperative, and cooperative (implementing *CoPark-WS*) vehicles with various penetration rates.

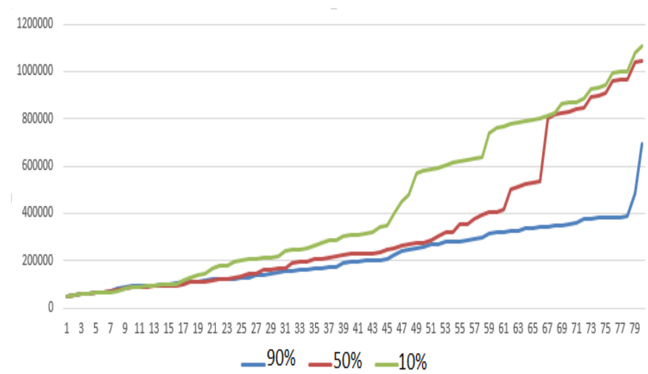

Fig. 8. Wall clock time of when a vehicle gets parked for each of the 80 vehicles with 10%, 50%, and 90% penetration rates.

effect of transmission range on *CoPark-WS*'s performance. The average searching time of 60 vehicles is shown in Figure 6 with implementing *CoPark-WS* and CoPark-GD runs. The simulation scenario consists of four car parking sections located in separate areas, and the car park size is $1000m \times 500m$. *CoPark-WS* consumes less time-to-park than CoPark-GD in all communication ranges. For example, the average searching time (or time-to-park) in case of transmission range 100m is 1570 stu and 2080 stu in *CoPark-WS* and CoPark-GD, respectively. In both approaches, there is reduction in time-to-park with increasing the range from 100m to 300m and 600m. However, the performance seems to be stable when increasing the range from 600m to 900m. One possible reason is that the vehicle with transmission range up to 600m; the vehicles already observes most of car park areas so that increasing the range does not result in increasing knowlegde.

E. The effect of penetration rate - a fraction of vehicles cooperating

In this experiment, we evaluate the impact of having only a fraction of vehicles cooperating, (i.e.. not all are equipped with wireless communications) and applied the *CoPark-WS* approach, which we call penetration rate. The average searching time (or time-to-park) of 80 vehicles, non-cooperative vehicles, and cooperative vehicles (that implement *CoPark-WS*) with different percentages of equipping (i.e., 10% of vehicles are cooperative vehicles, and then 20%, 30%, 40%, 50%, 60%, 70%, 80%, 90%) is shown in Figure 7. The non-cooperating vehicles implement the GD approach with supported initial information of free parking spot locations. The non-cooperative vehicles are interleaved with cooperative vehicles when entering the car park area, distributed as a fraction of every ten vehicles. For example, in 50% percentage, the first five vehicles of each ten vehicles entering the car park is non-cooperative.

The average searching time of all vehicles (i.e., total) is reduced, gradually, with increasing penetration rates. In addition, the average searching time of the non-cooperative vehicles reduces with the increase in the number of cooperative vehicles. That can be due to the cooperating vehicles, reducing the concentration of the vehicles nearer to a destination and distributing them to over a larger broader area. For *CoPark-WS*, there is robustness in working with different penetration rates as the performance of cooperating vehicles are just as good when fewer or more vehicles cooperate. Introducing advice messages among vehicles plays a key role to handle the knowledge gap. Moreover, it showed superiority in performance compared to non-cooperative vehicles across different penetration rates.

For more analysis, Figure 8 illustrates the walk clock time sequence for when vehicles get parked with penetration rates 10%, 50%, 90%. It is clear that the value of *CoPark-WS* on reducing time-to-park and traffic congestion in areas, especially with the late coming vehicles. Late vehicles tend to struggle from the knowledge gap between the initial information they get at the entrance (which is now more out of date) and the current free spots because of the earlier parked vehicles - this is despite the fact that the last vehicles in every ten coming in are selected to be cooperative vehicles. In our simulations, we also noted that the last 20 parked vehicles at 50% penetration rate are non-cooperative vehicles, while in 10% penetration rate, the last 40 parked vehicles are non-cooperative vehicles.

IV. Related Work

Self-coordinated car parking methods have been classified into two categories based on the method used for collecting parking information.

In the first category, it is supposed that vehicles are supported with perfect car parking information, there are different knowledge of competitor vehicle profiles, and game theory is used to to resolve the contention. The work in [7] considers assigning parking slots to the vehicles in an ad-hoc game theory framework with Nash equilibrium. In [8] is proposed that the driver's decision of choosing to park either on-street or off-street considers different levels of knowledge about the preferences of other drivers. In [9] is an investigation of the situation of selecting to park at a particular parking lot based on a stochastic policy. It is assumed that vehicles receive broadcast information from a parking lot infrastructure about arrival and departure rates. Although, the aforementioned work present a stable decentralized parking allocation system theoretically, it can be hard in a real life implementation to capture accurate real-time information of the number and location of parking spots and the searching vehicles. In addition, they neglect the cost of traveling to a target slot. The traveling time, which can be affected by road network topology and traffic flow, can diminish the opportunity to park. Also, the vehicle can, while driving, potentially detect a better target slot.

In the second category, collecting parking information is part of the searching process. In other words, the car parking infrastructure's role in assistance is restricted. The vehicle has a tradeoff between cooperating to collect information and competing to find a suitable parking slot. The work in [10] aimed to reduce contention among vehicles by restricting sharing to only information that the vehicle itself is not interested in. The work in [11] addressed the competition issue from a different angle by considering a leaving vehicle as the coordinator that announces about its free slot, gathering the requests from interested vehicles and then reserving the parking space for one. The work in [12] addresses the use of sensor-to-vehicle communication, where sensors send the occupancy information to closer vehicles, and via vehicle-to-vehicle communication, a leaving vehicle disseminates the leaving event locally. It found that the search time can be reduced but sometimes increased, while there is reduction in walking distance.

V. Conclusion

Much time can be wasted by drivers in finding parking spaces at crowded areas. We aim to facilitate the searching process by enabling cooperation among agents/vehicles. A cooperative car parking approach considering the searching time and walking distance (*CoPark-WS*) is introduced. It has been extensively evaluated with various car parking conditions. Our results show promise not only for cooperative parking in general but *CoPark-WS*-type of decision-making behaviour when it comes to car parking without centralised coordination. We showed that *CoPark-WS* can significantly reduce time-to-park with an acceptable level of walking distance. Also, our results show that even if not all vehicles are implementing our approach (with different penetration rates), there is still reduction in time-to-park, on average, for all.

References

[1] A. Aliedani, S. W. Loke, A. Desia, and P. Desai, "Investigating Vehicle-to-Vehicle Communication for Cooperative Car Parking: the CoPark Approach" *Proceeding of IEEE International Smart Cities Conference (ISC2)*, pp. 7-13, 2016.

[2] I. Benenson, K. Matens and S. Birfir, "PARKAGENT: An agent-based model of parking in the city" *Computers, Environment and Urban Systems*, vol. 32, no. 6, pp. 431-439, 2008.

[3] R. G. Thompson, and A. J. Richardson, "A parking search model" *Transportation Research Part A: Policy and Practice*, vol. 32, no. 3, pp. 159-170, 1998.

[4] F. Bellifemine, A. Poggi and G. Rimassa, "JADE: a FIPA2000 compliant agent development environment" *Proceedings of the fifth international conference on Autonomous agents*, pp. 216-217, 2001.

[5] D. Krajzewicz, "Traffic simulation with SUMO–simulation of urban mobility" *Fundamentals of traffic simulation*, pp. 269-293, 2010.

[6] T. Azevedo, P. Araujo and R. Rossetti, "JADE, TraSMAPI and SUMO: a tool-chain for simulating traffic light control" *arXiv preprint arXiv:1601.08154*, 2016.

[7] D. Ayala, O. Wolfson, B. Xu B. Dasgupta and J. Lin, "Parking slot assignment games" *Proceedings of the 19th ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems*, pp. 299-308, 2011.

[8] E. Kokolaki, M. Karaliopoulos, and I. Stavrakakis, "Leveraging information in parking assistance systems" *IEEE Transactions on Vehicular Technology*, vol. 62, no. 9, pp. 4309-4317, 2013.

[9] A. Schlote, C. King, E. Crisostomi, and R. Shorten, "Delay-tolerant stochastic algorithms for parking space assignment" *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 5, pp. 1922-1935, 2014.

[10] N. Bessghaier, M. Zargayouna, and F. Balbo, "Management of urban parking: an agent-based approach" *International Conference on Artificial Intelligence: Methodology, Systems, and Applications*,pp. 276-285, 2012.

[11] T. Delot, N. Cenerario, S. Ilarri, and S. Lecomte, "A cooperative reservation protocol for parking spaces in vehicular ad hoc networks" *Proceedings of the 6th International Conference on Mobile Technology, Application Systems*,pp.30-37, 2009.

[12] G. Tasseron, K. Matens and R. van der Hejden, "The Potential Impact of Vehicle-to-Vehicle and Sensor-to-Vehicle Communication in Urban Parking" *Intelligent Transportation Systems Magazine, IEEE*, vol. 7, no. 2, pp. 22-33, 2015.