Toward a reactive agent based parking assistance system

Baudouin DAFFLON, Jean-Michel CONTET, Franck GECHTER, Pablo GRUER
IRTES-SET, UTBM, 90010, Belfort cedex, France
baudouin.dafflon@utbm.fr, jean-michel.contet@utbm.fr, franck.gechter@utbm.fr, pablo.gruer@utbm.fr

Abstract—**Nowadays, automotive equipments tend to offer more and more driving assistance. Among these, car park assist systems are now widespread on both commercial and research levels. This paper proposes a car park assistance based on the use of the multi-agent paradigm. This approach based on the transformation of vehicle perception into a virtual environment for agents and on the evaluation of agency properties. This evaluation is then turned into a decision for vehicle guidance. The proposal is evaluated in simulation by comparing it with industrial solutions.**

I. Introduction

As detailled in [22], trends about equipment and automotive assistance are evolving. Carmakers integrate assistance into vehicles. Major companies such as Google, Toshiba or Toyota propose increasingly attractive innovations. Driver assistance devices, still under development starts today to appear. Car parking assistance is among these functions.
A parking assistance system offers possibility to park a car in a park slot with a autonomous system. Parking environments are area of high density where pedestrians, cars, cycles,... are moving. In addition, car park implies street congestion during maneuver. Several problems could be considerated: decreasing the time spent to park, and increasing number of available slots.
This process can be divided into three tasks: detection of free car park slot, adapted to the car, trajectory definition, taking into account car's characteristics, and trajectory following paying attention to the environment.

The aim of this paper is to present an approach for the car park assist problem. The proposed approach is based on the application of reactive multiagent systems. Multi-agent systems are an efficient approach for problem solving and decision making. They can be applied to a wide range of applications thanks to their intrinsic properties and features such as simplicity, flexibility, reliability, self-organization/emergent phenomena, low cost agent design and adaptation capacity,... It has been shown that reactive multi-agent system are efficient to tackle complex problems [7], cooperation of situated agents/robots [16], data fusion and problem/game-solving [6].
In this context, our proposal consists in decision making by evaluating emergent properties of agent's organization. thus, the car park assistance decision, represented by an acceleration vector for the vehicle, is computed from the evaluation of an indicator that represents the global state of a reactive agents system. In this model, interaction and environment have a preponderant role. As a matter of fact, the agents are immaterial and evolve in a virtual environment which is an abstraction of material environment, which is constructed from vehicle's perceptions. Agent-to-agent and agent-to-environment interactions produce agent dissemination into the virtual environment. This distribution, represents the global system state and is analyzed by an indicator to calculate a new vehicle's acceleration vector.

The paper is structured as follows: After a discussion about the state of the art on the car park assist issue. Section III describes the multi-agent model used to tackle this problem. Then, section IV exposes simulations comparison results and section V draws a conclusion of this work bringing future work directions.

II. State of art

Today, car park assist devices are mainly promoted by cars makers such as Ford or Toyota. These commercial solutions are closed and protected under patent. However, trajectory analysis allows to understand their performance. In the scientific literature, some approaches are presented. In next section, some commercial approaches and techniques are detailed.

A. Commercial solutions

The Parking Assist System (PAS), is the first commercialized automatic parking system developed by Toyota in 2004 initially for the Japanese market and the hybrid Prius models. On vehicles equipped with PAS, the car can steer itself into a parking space with only little action from the driver (speed regulation). The first version of the system was integrated in the Prius sold in Japan in 2003. In 2006, an upgraded version dedicated to the foreign market was proposed on the Lexus LS.

The PAS is based on several sensors : a camera situated in the back and two sonar sensors. Sonar sensors, named "Intuitive Parking Assist", are composed of ultra sonic sensors situated on vehicle's front. These sensors detect obstacles, allowing the vehicle to produce warnings and to compute optimum steering angles. The backup camera system provides park informations to the driver. PAS computes steering angles which are displayed on the navigation/camera touchscreen with obstacle information. The Intelligent Parking Assist System is accessible only when the driver has driven his car near a place with enough space and when the vehicle is shifted to reverse. Then, the embedded system computes the optimum steering angles (cf figure 1). The driver must regulate vehicle speed and the embedded system turns the wheel to park the vehicle.

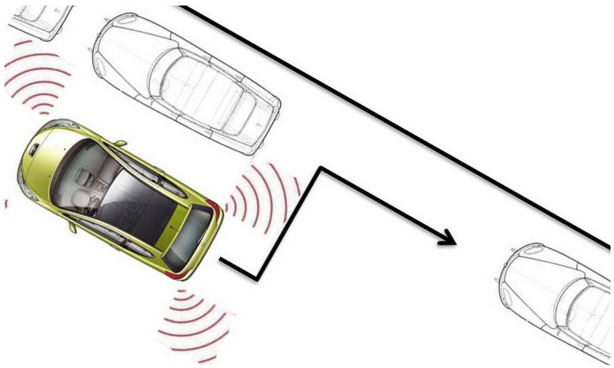

Fig. 1. Parking assist system developed by Toyota

Carmakers such as Ford and Volkswagen improved the performance of the existing Parking Assist System (PAS) by automatic free car park slot detection, taking into account the environnement (road deviation).

B. Scientific and open solutions

Carmakers solutions are classified and no public scientific description exist. Nevertheless in the literature, some car park solutions are presented. Three principal issues are raised : the free slot detection, the car park trajectory computation and the trajectory following:

1) Free slot detection : free parking slot detection is possible through a wide range of methods:
   - Ultrasonic sensor-based method : this method is described by [23], [5], [20] and bases on different sound pulse to determine the obstacles position (cf figure 2).
   - Lidar-based method presented in [17], [19] uses laser range finder to determine the presence of obstacles. Unlike methods based on ultrasonic sensor, lacks precision on small distance and is often assisted by camera to ensure accuracy of measurements of near by obstacle.
   - Method based in mono or stereo camera [24].

Fig. 2. Radar sensors spot detection

2) Car park trajectory computation: the main method to find the best path [18], [21] uses physical properties. Planned path of garage parking basically consists of two lines: a line extending current vehicles rear direction, and a line extending the central line of target position. A circular segment whose radius is the smallest rotating radius of the installed steering system connects the transition part of two lines as shown in figure 3. Current semiautomatic parking assist systems do not consider the multi turn parking trial situation. System checks whether it can track the planned path before starting parking operation.

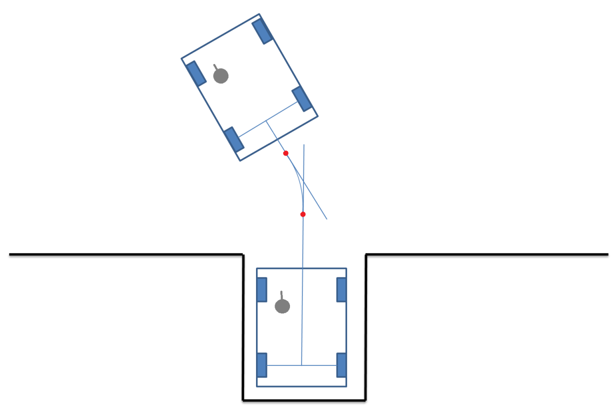

Fig. 3. Path Planning Method

The main drawback of these methods is a specific parametric for each vehicle type (length, turning radius, space ...) required by the use of Bézier curve in order to keep the class C^2 property [24]. The second drawback is the lack of trajectory adaptation when the environment changes(obstacle, pedestrian,...).

3) Trajectory following : To drive a car into a free slot, one way is to follow a trajectory calculated with previ-

ous methods. In autonomous system, main approaches are based on local and decentrallised behavior. Indeed vehicle computes acceleration and wheel direction commands with its local perceptions [3] (cd figure 4). Approaches based on a physics and vehicle dynamics [1] [8], [9] ... use local control strategies generally including PID controllers or other regulation-loop control.

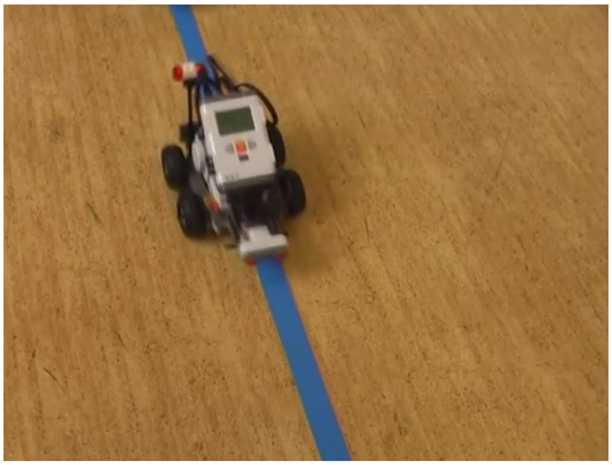

Fig. 4. Robots following trajectory

In the next section, is detailed a method based on merging between computation and followed trajectory methods.

III. System description

As detailled in [14], literature contains several exemple of MAS (Swarm algorithms, particule filtering,...). In this part, we will propose a new RMAS system based on physical interaction between agents and environment.

A. Global overview

The proposed approach is based on the application of reactive multi-agent system [10], [1]. In this model virtual environment plays a key role and is created from the vehicle perception. This virtual area represents the car neighborhood where, through real environment, obstacles provided by sensor are placed. In this virtual environment, agents are randomly spawned. A set of agent-agent, agent-obstacle and agent-goal interactions based on Newtonian laws are applied to agents. This results in agent dissemination. Decision is made by analyzing the emergent dissemination patterns. From this decision, command are calculated and applied to the vehicle.

As shown in Figure 5, the process can be summarized by the following main steps:

1) Perception : the vehicle perceives the environment. The laser range finder perception is projected into the virtual environment (obstacles, car park location, ...).
2) Decision : based on RMAS observation. Decision is computed in the form of an acceleration vector.
3) Command : the acceleration vector computed is translated in command for the vehicle. This command is composed of a speed and a steering angle.

[1] Stanford's Autonomous Car Parallel Parks By Sliding Sideways

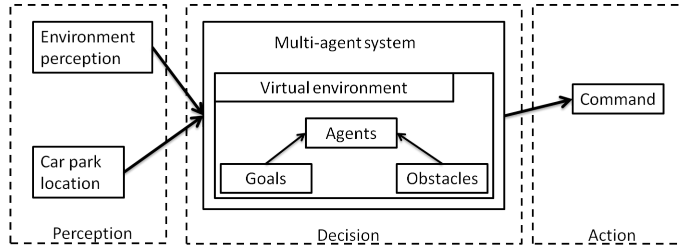

Fig. 5. Park assist process

B. Perception

System percepts are provided by vehicle's sensors. Free car park detection is not addressed in this paper. To detect a free car park slot, we use specific beacons in the environment. These beacons can be 3D points in simulation, signpost in experimentation,... Theses beacons materialize the limits of free slot.

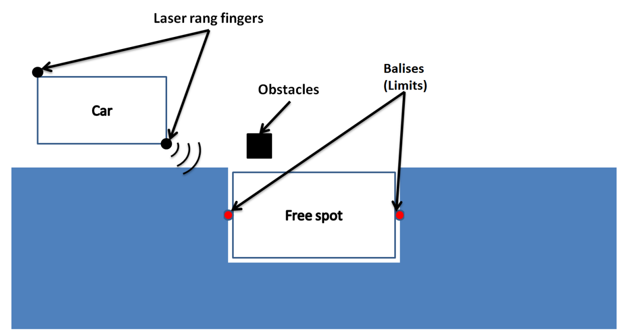

Fig. 6. Vehicle perception

Environment perception is realized by two LMS 120 lidar (Figure 6). Their opening angle of 270 degrees can cover a zone around the vehicle.

C. Reactive multi-agent system

1) Environment: As stated before, agents environment is the corner stone of the approach. It links vehicle's world with the decision process mechanism. The virtual environment is composed of :

- Obstacles perceived by the vehicle sensors are transmitted and placed in the virtual environment. Each obstacle is represented by an aggregate of repulsive spots. For each lidar ray a spot is representing. These entities have not specific behavior. In addition a spot belt mark the environment boundary corresponding to the sensor perception limits.
- The two goals represent free spot limit in virtual environment. They are represented by an attractive spot and are provided by the sensor.

- Agent are generated by system and have specific behavior detailled in following part. Agents are separated into two independents populations attracted by one of the two goals. System need two population to determined a point of balance in car park process and smooth transitions between forward and backward.

2) Agents: can be considered as small mass particles evolving in a force field. They are generated by obstacles, agents themselves and goals. Initially, the agents are created at random position. Environment perception is realized through a circular frustum[2]. Circular frustum is used to limit agent interaction. Agent's parameters are shown in table I.

Parameter	Description
m	mass
(p_x, p_y)	position
I_R	radius of fustrum

TABLE I
DECISION AGENT PARAMETERS

Only near neighbors agents in the environment are taken into account to compute RMAS interaction. This allows to take into account the only entities present in the perception zone in order to compute the forces on the agent's body. A projection of goal position is made to keep it on frustum borders when it's out it (cf figure 7). This projection allows agents to know the goal direction in the environment.

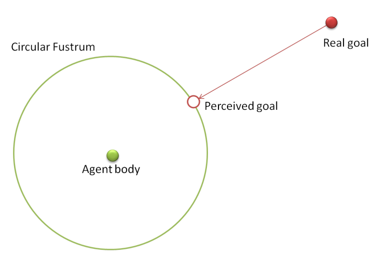

Fig. 7. Projection of the goal in the border of field of view

3) Interaction models: In this part, we detail the agent-agent and agent-environment interactions, all inspired by physics.

- **Interaction between agents (Repulsion) :** This interaction is made through a classical Newtonian repulsion force in $1/d^2$. Considering two agents i and j situated at positions A_i and A_j, this force can be expressed as follow : $\overrightarrow{Fr}_{ij} = m_i.m_j \frac{\vec{A_iA_j}}{\left\|\vec{A_iA_j}\right\|^3}$ for each agent i such as $\left\|\vec{A_iA_j}\right\| < I_R$. In this equation, mass of agents i and j are respectively denoted m_i and m_j. Since agency is homogeneous, masses are the same (i.e. $m_i = m_j = m$).

[2]Agent's field of view

$$\begin{cases} Fr_i^X = \sum_{i \neq j} \left(m^2 \frac{(x_j - x_i)}{((y_j-y_i)^2+(x_j-x_i)^2)^{\frac{3}{2}}} \right) \\ Fr_i^Y = \sum_{i \neq j} \left(m^2 \frac{(y_j - y_i)}{((y_j-y_i)^2+(x_j-x_i)^2)^{\frac{3}{2}}} \right) \end{cases} \quad (1)$$

- **Interaction between agents and goal (Attraction):** This interaction is modeled as a simple linear attraction force defined, for agent i, as follows:

$$\overrightarrow{Fd}_i = \beta_d.m.\vec{A_iD_c} \quad (2)$$

with $\vec{A_iD_c}$ the distance from agent i to the goal position.

- **Interaction between agents and obstacles (Repulsion):** This interaction has the same formulation as the interaction between agents. Thus, interaction between agent i and an obstacle o is given by: $\overrightarrow{Fo}_i = m.m_o \frac{\vec{OA_i}}{\left\|\vec{OA_i}\right\|^3}$ such as $\left\|\vec{A_iO}\right\| < I_R$, where m and m_o are respectively the mass of the agent i and the mass of the obstacle o. $\vec{A_iO}$ is the distance between i and o. Then, if all obstacles in the perception range of agent i are taken into account, we obtain this equation:

$$\begin{cases} Fo_i^X = \sum_{o} \left(m \cdot m_o \frac{(x_i - x_o)}{((y_i-y_o)^2+(x_i-x_o)^2)^{\frac{3}{2}}} \right) \\ Fo_i^Y = \sum_{o} \left(m \cdot m_o \frac{(y_i - y_o)}{((y_i-y_o)^2+(x_i-x_o)^2)^{\frac{3}{2}}} \right) \end{cases} \quad (3)$$

Theses interactions allow to place agents in virtual environment. Indeed each agent state can be computed using Newton's law of motion and taking into account all influences (agents, obstacles, goal) in the agent's frustum. For this computation, environment model is considered to be continuous and time to be discrete each time step being triggered by the system. If $\vec{\gamma}_i$ denotes acceleration, m the mass of agent i then: $\vec{\gamma}_i = \frac{1}{m}(\overrightarrow{Fr}_i + \overrightarrow{Fo}_i + \overrightarrow{Fd}_i)$. By substituting all the forces by their expressions and integrating twice, the following equation is obtained:

$$\vec{Z}_i(t) = \vec{Z}_i(t-1) + ... \quad (4)$$

$$... \left(\vec{V}_i(t-1)\delta t + \frac{(\delta t)2}{2m} \left(\overrightarrow{Fr}_i + \overrightarrow{Fo}_i + \overrightarrow{Fd}_i \right) \right)$$

with $\vec{Z}_i(t) = \begin{pmatrix} x_i(t) \\ y_i(t) \end{pmatrix}$ and, $\vec{V}$ the velocity vector.

Sum of forces applied on agent body (attraction and repulsion) is used to move it in virtual environment. Each step time, new position was computed according the vector $\vec{Z}_i(t)$.

4) Decision making: The decision process is based on the evaluation of the distribution of agents in the virtual environment. Two agent populations were created, each agent interacts with only one of two goals (cf figure 8).

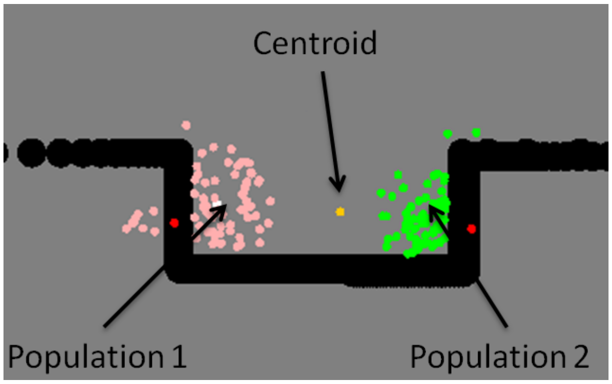

Fig. 8. Final composition between centers

Agents attracted by a front goal will guide a forward gear while agents attracted by the rear goal will order to reverse. To keep homogeneity, agents population 1 are attracting by one goal and agent population 2 by the second. The system evaluates the repartition of the two populations and defines two centers of mass. Final decision is based on results of the composition of theses centers.

$$\overrightarrow{P_{dir}(x,y)} = \frac{1}{nb_a}\sum_{i=0}^{nb_a}\overrightarrow{P_i(x,y)} \qquad (5)$$

The result of this composition is noted $\overrightarrow{P_{dir}(x,y)}$ and correspond to a vector from vehicle and center of mass of agent's bodies $P_{mean}(x,y)$ (Figure 9). $\overrightarrow{P_{dir}(x,y)}$ result of sum $\overrightarrow{P_i(x,y)}$ which corresponding to $i-th$ body vector defined by $i-th$ body location and car pivot.

$\overrightarrow{P_{dir}(x,y)}$ is a new command to control vehicle. The angle between $\overrightarrow{P_{dir}(x,y)}$ and local X axis is a steer angle adopted by the vehicle. The length of $\overrightarrow{P_{dir}(x,y)}$ corresponds to the speed command reference.

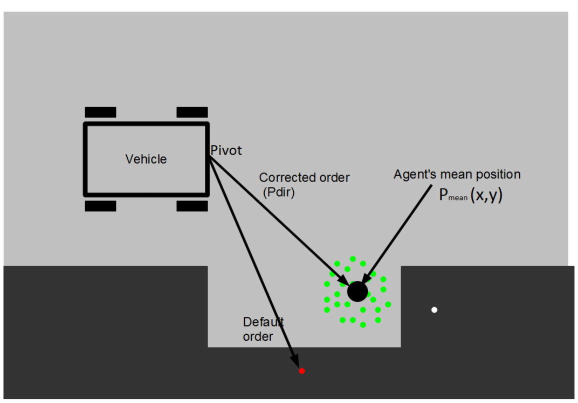

Fig. 9. Decision vector computation

The end of parking maneuver is detected when population of agents is in balance between the attraction of different goals. It is to say that the evaluation of the population of the corresponding agent forward is in the same order of evaluation length as the population participating in the control of reverse.

IV. Simulation and Results

The implementation of these systems has been made in simulation to test and validate our approach. We use a simulator designed for simulating vehicle in an urban environment [3] and JANUS a multi-agent platform developed by the IRTES-SET laboratory.

A. Simulation tool

1) VIVUS Simulator: To assess the quality of our approach, realistic simulations have been done using VIVUS simulator [15], a vehicle simulator developed by the IRTES-SET[4] laboratory. VIVUS is based on PhysX for real physical behavior, and Unity3D for good 3D performance.

This software can simulates behaviors for each vehicle such as perceptions with laser range finder or cameras, physical reaction between elements (wheels, car's parts,...),... Physical reaction are computed using the same physical law as real world (collision, gravity,...) and considering the properties of the environment (friction with road, materials of ground and walls,...). VIVUS has already been used to test various intelligent vehicle algorithms such as linear platoon control [2], [7], obstacle avoidance and driving assistance [13], and intelligent crossroads simulations in [4].

2) Janus: Janus[5] is a new multi-agent platform that was specifically designed to deal with the implementation and deployment of holonic and multi-agent systems [11], [12]. It is based on an organizational approach and its key focus is that it supports the implementation of the concepts of role and organization as first-class entities. This consideration has a significant impact on agent implementation and allows an agent to easily and dynamically change its behaviour. The platform also natively manages the concept of holon to facilitate the deployment of holonic multi-agent systems and thus contributes to fill the gap between conception and implementation phases in this domain.

B. Simulation protocol

Simulations are performed on a 3D geo-localized model of the city of Belfort (France). Two different situations have been chosen. Figure 10 shows a 3D representation of these simulations. As area is geo-localized, ground has same properties as reality (scale, tilt,...) (cf Table II).The first simulation is angle parking and the second one is parallel parking.

angle pattern	
place width	3.8 m
place length	4 m
parallel pattern	
place width	5 m
place length	4 m

TABLE II
PLACE PROPERTIES

[3]www.vivus-simulator.org
[4]http://set.utbm.fr/
[5]http://www.janus-project.org/

Vehicles used in the simulation represent (graphically and physically) of experimental laboratory's vehicles, they satisfy the constraints and share the same characteristics. They are equipped with two 270 degrees virtual laser range finder, replica of LMS SICK 200 and GPS-RTK simulation required to follow and study trajectories (Table III).

car width	1.8 m
car length	3.05 m
max steer angle	30 degrees
max speed	2 m/s

TABLE III
CAR PROPERTIES

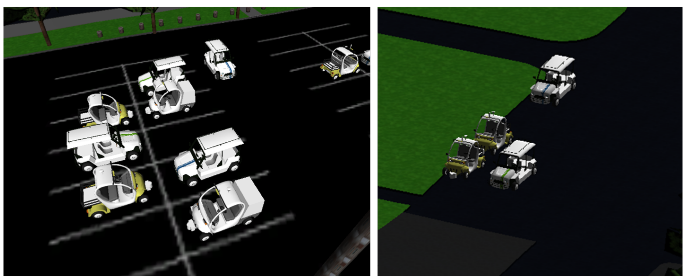

Fig. 10. Car park simulations : angle pattern (left) and parallel pattern (right)

The simulations presented were examined according to the following criteria: the length of the parking space required and the position variation between a vehicle optimally aligned position and the position result with the algorithm.

Bézier curve: Pattern shown in figure 11 have been generated by Matlab based on characteristics of our vehicle by following the approach described in [24] and depending on the turning radius (T_r) and the length of the vehicle (cf figure 11).

$$T_r = \sqrt{\frac{E^2}{\sin^2 a - E^2}} - \frac{L}{E} \tag{6}$$

with E = wheel base L = width a = steering angle. simulation used a following path algorithm developed by IRTES-SET laboratory and tested in various projects [7]. Cars are placed on curve tangent, this figure (figure 11) shows control point needed to build curve (7 for parallel park, 4 for angle park), curve and insignificant error concerning alignment with spot. In following tests, this simulation will be the benchmark.

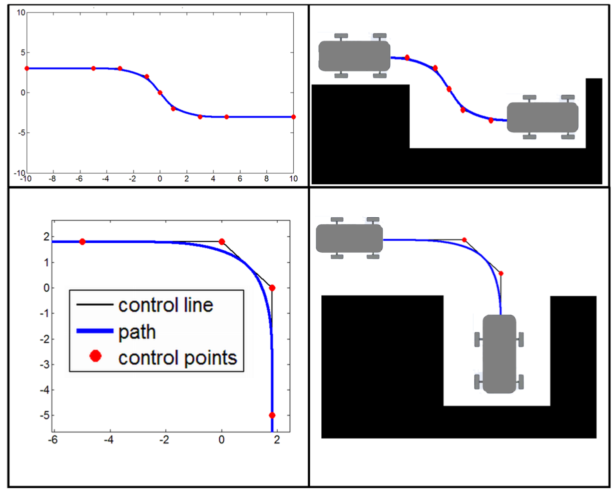

Fig. 11. Vehicle parked in parallel (top) - Vehicle parked in an angle pattern (bottom) in meters

C. Simulation results

1) Park in angle pattern simulation: The park in angle pattern simulation allows to compare theses two approaches in a basic situation. Indeed, vehicle must only run in back, without transition or external disturbances. In this situation, RMAS system is not better with the first criterion (cf figure 12) however it is faster.

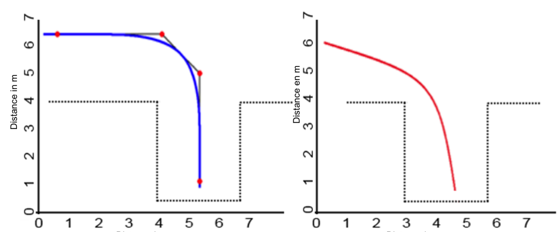

Fig. 12. Vehicle parked in an angle pattern following Bézier curve (left) and RMAS (right)

Although both systems are successful, the vehicle following the bezier curve produces no error. To reach the same result with RMAS, a larger place will be need but with same place, there is an error around 13 to 15 degrees.

2) Park in parallel simulation: Park in parallel situation is the most difficult parking operation. For human, it's required to look at dead angles, dynamic obstacles and forward/backward transition. In simulation, we follow Bézier and RMAS algorithm and find some differences. Unlike park in angle, our approach is more attractive than Bézier. Indeed, Bézier curve need more space, around twice the vehicle length and cannot use reverse without driver intervention (cf Figure 13). At the opposite, RMAS manages transitions between forward and backward. The result is a smaller place but a slower maneuver and a most important angle with road. The main advantage is a self adaptation to spot and environment.

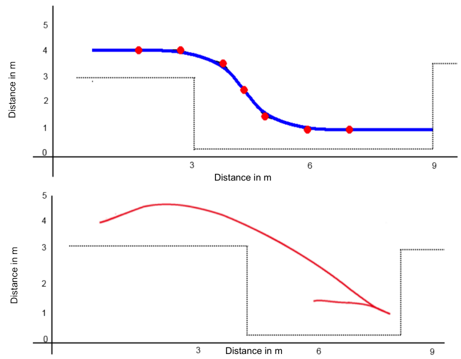

Fig. 13. Vehicle parked in parallel following Bézier curve (top) and RMAS (bottom)

As figure 14 shows, Bézier (black) cannot be evaluated before 1,5 vehicle length while the RMAS system (gray) offered a path emerging that can be considered valid around 1,5 vehicle length. Within the assisted parking, space and time constraints, the approach proposed in this paper provides several advantages. Indeed more cars are parked and will be less crowded. After 1,8 vehicle length meters, the two system allows the vehicle to park. between 2 vehicle length to 2.2 vehicle length, the error of the vehicle following Bezier curve is negligible, around 3 degrees, while the RMAS on this same interval produces an error ranging from 7 to 4 degrees. On a smaller place which size is between 4 and 5, only the RMAS vehicle achieves to park. The RMAS produced a larger error : between 19 and 10 degrees. Below 1,2 vehicle length neither of the two systems proposed allows to say than car is parked.

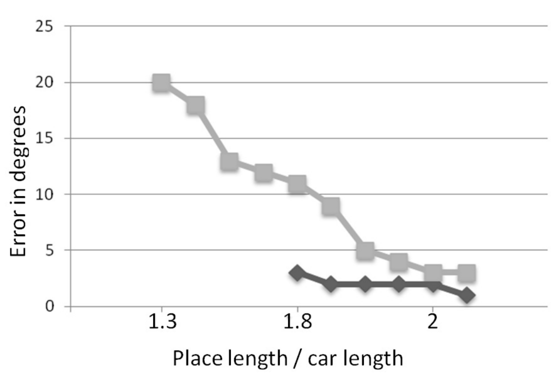

Fig. 14. Evaluation of park assistance system

3) Park in parallel simulation with obstacles: This simulation is an evolution of previous one. We use the same protocol and the same algorithms. In addition, we add an obstacle near the place. This part shows the importance of the self adaptable system. Whereas the analytical solution needs a new computation, and is totally useless in dynamic environment, RMAS can adapt trajectory in this situation and in run time (cf figure 15).

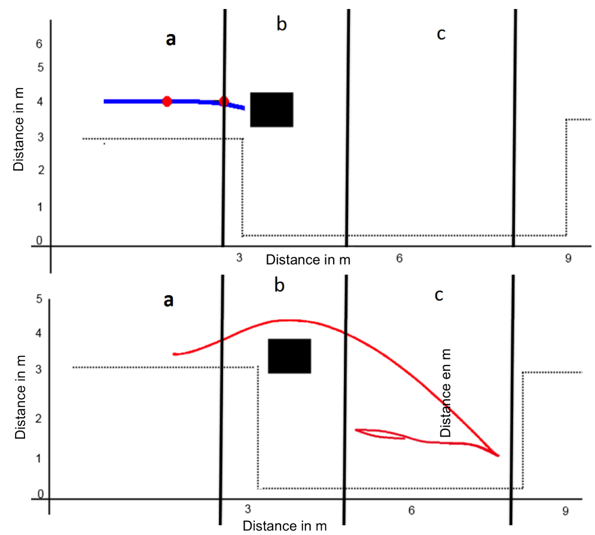

Fig. 15. Vehicle parked in parallel with obstacle following Bézier curve (top) and RMAS (bottom)

The following graph (cf figure 16) shows the distance between the vehicle and the obstacle for each evaluation system. Step (a) represents the approach. We can note a first anticipation by the RMAS. Phase (b) corresponds to the obstacle avoidance. The vehicle following the curve stops while the other system based on RMAS bypasses obstacle. In phase (c), while the car follow the curve is always stopped, the RMAS allows to continue the maneuver.

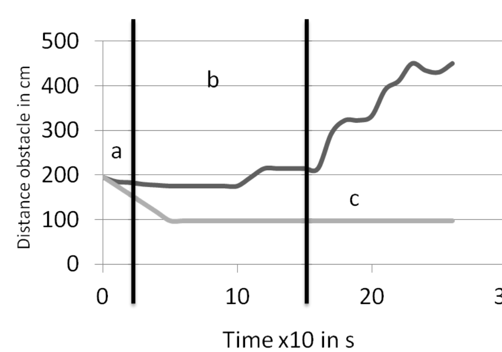

Fig. 16. Evaluation of park assistance system with obstacle (black : bezier, grey: rmas

This simulation shows analytic's limits solution in dynamic environment. Solution based on emerging solution like multi-agents system is an interesting key for motion in urban environment. Indeed, urban environment has strong constraint needed a run time adaptation. To conclude these simulations we can summarize by following table IV. We can see two informations about simulations, car park length and angle between road and parked vehicle. It's interesting to notice that car park with obstacle simulation could not give numerical results because of simulation does not reach to the end.

Simulation	place length / car length	angle error
Angle Bézier	1	3-4 degrees
Angle RMAS	1	13-15 degrees
Parallel Bézier	2	2-7 degrees
Parallel RMAS	1.5	15-24 degrees
Parallel with obstacles Bézier	none	none
Parallel with obstacles RMAS	1.5	17-21 degrees

TABLE IV
SIMULATION SUMMARIZE

V. CONCLUSION

The paper presents a generic and emerging park assist application based on multi-agents system. In this model agents virtual environment plays a key role by merging perceptions data. Agents are reacting to virtual environment variations through attraction/repulsion behaviors based on physics lows where evaluation of trajectory is done by observation of agents populations spread. The main advantages proposed by this system is a run time and self adaptation to road configurations.

This solution was successfully tested in simulation and results obtained are encouraging to test using real laboratory vehicles and real sensors.

As creation and updating virtual environment is a main step, it could be supplemented by other sensors such as camera or ultrasonic system with an adapted fusion function. More sensors could be adding to obtain a robust and redundant virtual environment.

In order to continue this research, we are now working on generic and emerging solution to adapt free place detection to any cars and increase goals composition fonction for a better and smoother transition in car's motion. Those works are done with the support of the French ANR (National Research Agency) through the ANR-VTT *SafePlatoon* [6] project (ANR-10-VPTT-011).

REFERENCES

[1] Jean-Michel Contet, Franck Gechter, Pablo Gruer, and Abderrafiaa Koukam. Evaluation of global system state thanks to local phenomenona. In *Biennial European Conference on Artificial Intelligence (ECAI)*, jun 2008. Grèce, Patras, june 21-27.

[2] J.M Contet, F. Gechter, F. Gruer, and A. Koukam. Reactive multi-agent approach to local platoon control: stability analysis and experimentations. *International Journal of Intelligent Systems Technologies And Application*, 2010.

[3] Lionel Cordesses, Benoit Thuilot, Philippe Martinet, and Christophe Cariou. Curved path following of a farm tractor using a cp-dgps. *6th IFAC Symposium on Robot Control*, 2000.

[4] B. Daffon, F. Gechter, J.M. Contet, A.J. Abbas-Turki, and P. Gruer. Intelligent crossroads for vehicle platoons reconfiguration. *International Conference on Adaptive and Intelligent Systems*, 2011.

[5] Sethson M. Degerman P., Pohl J. Hough transform for parking space estimation using long range ultrasonic sensors. *SAE Paper*, 2006.

[6] G. DiMarzo-Serugendo, A. Karageorgos, O.F. Rana, and F. Zambonelli. "engineering self-organising systems: Nature-inspired approaches to software engineering". *Lecture notes in Atificial intelligence*, vol. 2977:299, 2004.

[7] M. El-Zaher, F. Gechter, P. Gruer, and M. Hajjar. A new linear platoon model based on reactive multi-agent systems. *The 23rd IEEE International Conference on Tools with Artificial Intelligence ICTAI, IEEE Computer Society*, 2011.

[8] Gabriel H. Elkaim and Robert J. Kelbley. Control architecture for segmented trajector following of a wind-propelled autonomous catamaran. *IEEE Conference on Decision & Control*, 2006.

[9] P. Encarnacao and A. Pascoal. Combined trajectory tracking and path following: an application to the coordinated control of autonomous marine craft. *IEEE Decision and Control*, 2001.

[10] Pablo Gruer Franck Gechter, Jean-Michel Contet and Abderafiaa Koukam. A reactive agent based vehicle platoon algorithm with integrated obstacle avoidance ability. *Self-Adaptive and Self-Organizing Systems*, 2010.

[11] Stéphane Galland, Nicolas Gaud, Sebastian Rodriguez, and Vincent HILAIRE. Janus: Another yet general-purpose multiagent platform. In *the 7th Agent-Oriented Software Engineering Technical Forum (TFGAOSE-10)*, Paris, France, dec 2010. Agent Technical Fora, Agent Technical Fora.

[12] Nicolas Gaud, Stéphane Falland, Vincent Hilare, and Abderrafiaa KOUKAM. An organizational platform for holonic and multiagent systems. In *the Sixth International Workshop on Programming Multi-Agent Systems (ProMAS 08), of the Seventh International Conference on Autonomous agents and Multiagent Systems (AAMAS)*, pages 111–126, Estoril, Portugal, 2008.

[13] F. Gechter, J.M. Contet, P. Gruer, and A. Koukam. Car driving assistance using organization measurement of reactive multi-agent system. *Procedia CS1(1): 317-325.*

[14] Franck Gechter, Vincent Chevrier, and Francois Charpillet. A reactive agent-based problem-solving model :application to lo calization and tracking. *ACM Transactions on Autonomous and Adaptive Systems*, V:1–35, 2006.

[15] Franck Gechter, Jean-Michel Contet, Olivier Lamotte, Stéphane Galland, and Abderrafiaa Koukam. Virtual intelligent vehicle urban simulator: Application to vehicle platoon evaluation. *Simulation Modelling Practice and Theory (SIMPAT)*, feb 2012. (To be published).

[16] Jean-Pierre Georg, Marie Pierre Gleises, Francisco J. Garijo, Victor Nol, and Jean-Paul Arcangeli. Self-adaptive coordination for robot teams accomplishing critical activities. *PAAMS*, 2010.

[17] Jung HG. and al. Integrated side/rear safety system. *The 11th European Automotive Congress*.

[18] Dong Suk Kim Ho Gi Jung, Chi Gun Choi and Pal Joo Yoon. System configuration of intelligent parking assistant system. *13 th World Congress on Intelligent Transport Systems and Services*, 2006.

[19] Pal Joo Yoon Ho Gi Jung, Dong Suk Kim and Jaihie Kim. Parking slot markings recognition for automatic parking assist system. *Intelligent Vehicles Symposium*, 2006.

[20] HG. Jung and al. Parking space detection.

[21] Fintzel K and al. 3d parking as sistant system. *IEEE Intelligent Vehicles Symposium*, 2004.

[22] Bishop Richard. *Intelligent vehicle technology and trends*. ITS, 2008.

[23] H. Satonaka and al. Development of parking space detection using an ultra-sonic sensor. *The 13th World Congress on Intelligent Transportation Systems and Services*, 2006.

[24] E. Wahl and RD. Therburg. Developing a motion-stereo parking assistant at bmw. *MATLAB Digest - November 2008*, 2008.

[6] http://safeplatoon.utbm.fr