

A reactive agent based vehicle platoon algorithm with integrated obstacle avoidance ability

Franck Gechter, Jean-Michel Contet, Pablo Gruer and Abderafiâa Koukam

Laboratoire Systèmes et Transports

Université de Technologie de Belfort Montbéliard

Belfort, France

{jean-michel.contet,franck.gechter, pablo.gruer, abder.koukam}@utbm.fr

Abstract—Since a couple of years, multi-agent systems have been used for a wide range of applications such as problem solving, modelling and simulation,... Reactive agent based systems are characterized by their capability to solve complex problems, while maintaining functional and conceptual simplicity of each element. These systems exhibit effectiveness whatever the considering fields (life simulation, robots and cooperation,...). One of the main problems of such method is the difficulty to characterize/measure emergent phenomena. Based on preceding work, this paper presents a reactive agent based vehicle platoon algorithm with an integrated obstacle avoidance ability. The vehicle decision process is then considered as a multi-agent system, the agents of which build collectively the decision, taking into account both platoon and obstacle avoidance constraints. The decision is built through measurement of agency collective state and dynamics.

Keywords-Multi-agent systems, intelligent vehicle, platoon control, obstacle avoidance

I. INTRODUCTION

Since around two decades multi-agent systems and particularly reactive multi-agent systems have been used for a wide range of problem solving, modelling and simulation applications. These approaches are characterized by their capability to solve complex problems, while maintaining functional and conceptual simplicity of involved entities called agents. In many cases, multi-agent based approaches exhibit effectiveness in various fields such as life simulation [1] and robots cooperation [2]. Multi-agent system design generally focuses on agents' definition (internal states, perception and behaviour,...) and/or interactions between agents and their environment using biological [3], [6], [4], [5] or physical inspiration [8], [9], [7]. Another major component of the multi-agent system is the environment and its characteristics (dynamics, topology,...) . As it has been explained in [10], [11], [12], the environment plays a key role in reactive multi-agent systems. From the problem-solving/decision-making point of view, clearly a reactive agent can neither handle a representation of the problem nor compute its solution. Thus, the representation of the problem and parts of the solving process can only be defined through an environment model. Therefore, the environment model can be defined as the place where the system computes, builds and commu-

nicates. Then, two problems emerge: (1) How to define an environment model able to take into account the problem's characteristics and to propose means for the problem solving process? (2) How to measure/evaluate the system's emerging organization in order to check if it corresponds to a valid solution for the initial problem?

This paper deals with these two problems through a real application which integrates an obstacle avoidance ability into a platoon control system. Based on preceding work developed in [13], [14], this paper presents a way to merge, thanks to the environment, platoon control commands and obstacle avoidance decisions.

This paper presents a reactive agent based vehicle platoon algorithm with an integrated obstacle avoidance ability. Platoon systems can be defined as sets of vehicles that apply local or global perception capabilities to adopt and maintain a train-like or other geometric configuration or formation. From the scientific and technological point of view, platoon system's control includes some relevant challenges. The main problem related to platoon systems relates to inter-vehicle distance control and the obstacle avoidance. In literature, the control of platoon's global geometry can be divided into two main approach: global [25] or local vehicle control [13].

The vehicle decision process, expressed as a proposed acceleration vector for the vehicle, is elaborated from the evaluation of a set of indicators characterizing the global state of a system of reactive agents (RMAS). Those agents evolve in a virtual environment produced on the base of vehicles' perceptions of the material environment around them and the perception of preceding vehicle. Agent-to-agent and agent-to-environment interactions are defined in order to produce a distribution of agents over the virtual environment. This distribution, taken as the global state of the system, is analyzed by applying a set of indicators inspired from statistical physics, to calculate a new vehicle's acceleration vector.

The paper is structured as follows: After a discussion about the state of the art concerning the emergence problem in section II, section III describes the multi-agent model used to tackle this issue. Then, section IV exposes experimental

results and section V draws a conclusion of this work bringing future work directions.

II. STATE OF THE ART

Characterizing and evaluating the structure and the properties that are able to emerge from a Multi-Agent System (MAS), is a complex task. This requires both strong theoretical background and well chosen experimental examples. Generally, evaluating the accuracy/efficiency of a system relatively to a specific task requires to take into account task constraints, global and local interaction between elements. Different evaluation functions quantifying the fitness, utility, satisfaction [15], altruism [16], entropy [17], energy [18], to name but a few, have been developed to tackle this problem. Three categories of evaluation functions can be found:

- 1) **Goal based evaluation**, often made through a comparison with other approaches resolving the same problem.

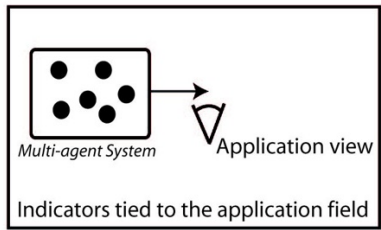


Figure 1. Application based indicator

The first category is the most widely used and is the simplest to put into practice. Indeed, it does not take into account the agent properties of the model used and is totally based on an application driven measurement. For instance, if a system aims at finding the shortest path in a graph, it is easy to evaluate its results compared to paths found by other methods. Thus, for this category, building an indicator only requires a good knowledge of the application field.

- 2) **Global emerging structures and properties analysis.**

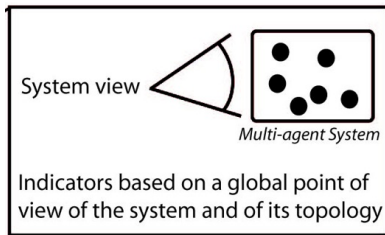


Figure 2. Global indicator

The second category is also wide spread in physics inspired approach literature. For example, entropy based indicators can be classified into this category. However, entropy [17], [19] has got particular properties. For instance, it is not a regular state indicator since it is a state function which no depends on the past transformations of the system (i.e. two identical systems can be in a the same state but with two different values for their entropy depending on their previous states). Moreover, contrary to what general entropy's physical definition implies (i.e. measure of organization/disorder taking into account microscopic events between individual components), computer science means of computing it are not taking into account the local phenomenon of the system. Finally, in physics, entropy is supposed to grow with time, which is not necessarily the case in computer science applications where it can decrease locally [20].

- 3) **Global measurement based on local phenomenon properties integration.**

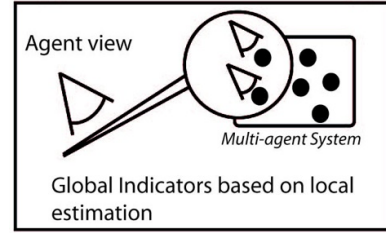


Figure 3. Global indicator built from local behavior estimation

Third category indicators tend to overcome these drawbacks. Energy is a good candidate because (1) it is a real state function, (2) it can be computed on both agent and system level [18]. However, it is not always easy to compute energy. Generally, energy is well adapted to force-based or physics inspired behavioural models. Another good approach to build indicators is based on statistical physics/thermodynamics duality. This approach, increasingly used, is based on the partition function, denoted by Z , is the source of all thermodynamic potential functions such as Gibbs function, enthalpy, free energy... , which is computed taking into account statistically each individual component state and interaction between them. The main difficulty of Z -based methods is the restrictive condition that needs to be fulfilled. For instance, has the number of agents an important role in the partition function computation? In other words, is the number of agents big enough to be statistically significant? Actually, statistic physics literature [21] exposes that the validity of this approach is not tied to the number

of elements involved but on the number of accessible states in the system. Then, in the case of a continuous environment, the infinite number of states permits to legitimate the use of the partition function.

III. SYSTEM DESCRIPTION

A. Global description

Figure 4 presents the proposed functional organization for an integrated obstacle avoidance local platoon control, intended for a follower vehicle agent.

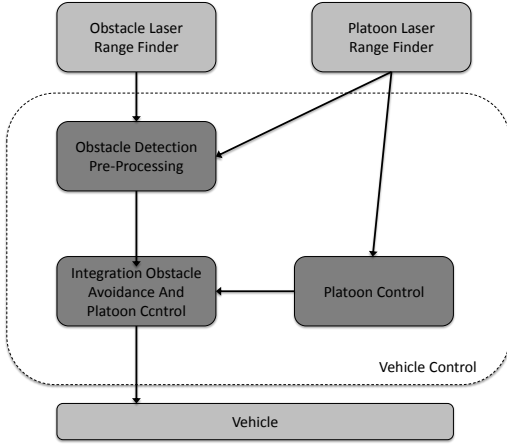


Figure 4. Functional diagram of the vehicle multi-agent control algorithm

Platoon Control is fed with data furnished by a laser range finder. From this information the position (vectorial distance) and the orientation of the preceding vehicle is extracted. Then, *Platoon Control* computes an ideal vehicle command order (speed and direction) to follow the preceding vehicle. The output can be considered as a goal position to reach. *Integrated Obstacle Avoidance and Platoon Control* is fed by the output of the *Platoon Control* and by pre-processed information obtain from *Obstacle Laser Range Finder* and *Platoon Laser Range Finder*. The goal of *Obstacle Detection Pre-Processing* is to remove the preceding vehicle from the set of obstacles since it has already been taken into account by *Platoon Control*. *Integrated Obstacle Avoidance and Platoon Control* is the core of the multi-agent system proposed in this paper. From obstacles' data and platoon control output, this system computes a global command order (speed and direction) to the vehicle aimed at accomplishing the platooning while avoiding detected obstacles. Next paragraphs III-B and III-C expose in further details *Platoon Control* and *Integrated Obstacle Avoidance*.

B. Platoon Control

The platoon control model used in this paper is based on a local approach (i.e. each vehicle is only able to perceive its preceding). Thus, each vehicle determines its

trajectory relatively to a preceding vehicle. In the case of a lateral platoon¹, as presented in this paper, the preceding vehicle is in front (in the direction of the movement) with a longitudinal distance l_o and aside with a lateral distance l_a . To perform this lateral platoon, we adapt a linear platoon control² (i.e. $l_a = 0$) model based on an virtual spring damper model [13]. The adaptation relies on using the concept of virtual robot tracking (VRT), introduced in [22] for mobile robot formations. In this approach, each real robot creates a virtual robot to follow, whose movements rely on the real preceding robot. In our case, the virtual preceding vehicle is tracked by applying the local mechanism applied to the linear platoon system.

The virtual vehicle is placed at l_o and l_a distance from the preceding vehicle, with the same orientation (Cf. figure 5). Thus, l_o represents the longitudinal distance and l_a the defines lateral distance along the wheel axis.

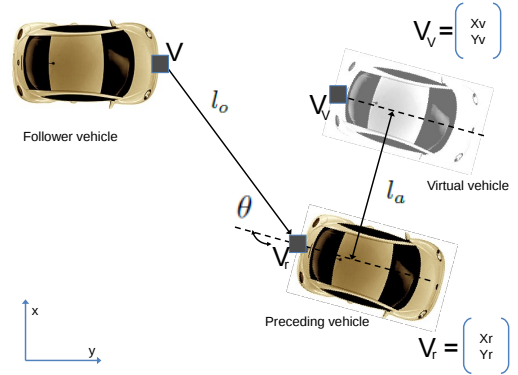


Figure 5. Virtual Vehicle model

Then, the virtual vehicle's position (x_v, y_v) and the preceding vehicle position (x_r, y_r) can be linked as follows:

$$\begin{cases} x_v = x_r + l_a \cos(\theta) + l_o \sin(\theta) \\ y_v = y_r - l_a \sin(\theta) + l_o \cos(\theta) \end{cases} \quad (1)$$

The follower vehicle tracks its virtual reference using the linear platoon control developed in [13]. This model relies on vehicle to vehicle interaction forces, which are computed considering a virtual physical link, made by a bending spring-damper model as shown in figure 6.

¹ Echelon formation : column formation with an offset from the preceding by a lateral distance.

² Column formation: vehicle behind each other

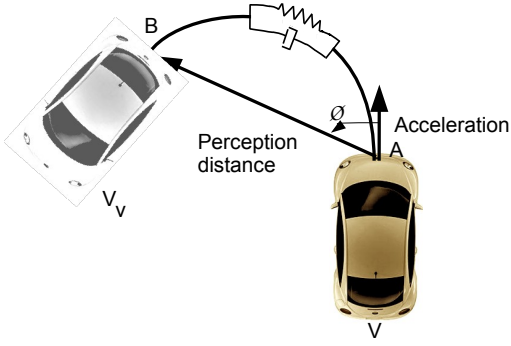


Figure 6. Virtual link between vehicles

By contrast with a classical physical link which is bidirectional, the virtual spring damper only affects the follower vehicle. The damper is used to avoid longitudinal oscillation, and the bending is aimed at compensating the lateral deviation error. Algorithms' details are explained in [13].

By discrete integration of Newton's law of motion, vehicle's state (Acceleration, Speed, Position and orientation) can be determined. This state is then translated into a goal (i.e. an attractive destination point in the agents' virtual environment) to reach in the *Integrated Obstacle Avoidance and Platoon Control* algorithm. This goal is translated into an attractive destination point in the agents' virtual environment.

C. Obstacle avoidance

The proposed approach is based on a collective decision process. This decision relies on the interpretation of agency indicators, which takes into account both local and global phenomena. The process is defined as follows:

- 1) Virtual environment properties are modified by perceptions (platoon control output is translated into an attractive spot, obstacle detection pre-processing output is translated into repulsive areas).
- 2) Agents interact with their environment and with each other following a physics inspired interaction model.
- 3) Agents behaviors lead to a spatio-temporal organization adapted to a virtual environment configuration.
- 4) Agency organization is measured by indicators the output of which are interpreted as a command law for the real vehicle.

Since interaction mechanisms are based on physics, agents can be considered as small mass particles evolving in a force field. This force field is generated by representatives in the virtual environment of real obstacles (repulsion), goal to reach (attraction), perception boundaries (repulsion) and agents themselves (repulsion)³. Initially, agents are spread

³In this application, each laser range finder measurement is considered as an obstacle

randomly in the virtual environment. Agents behaviors consist in finding a stable state under the influence of all these forces. This state can be generally considered as force field extremum. The number of agents is taken to be the square root of the perception surface. This number is generally taken in similar approaches. Studies, made in [14], show experimentally that this number gives best results. Agents' parameters are the following:

Parameter	Description
m	mass of the decision agent
(p_x, p_y)	agent position in x and y
(v_x, v_y)	agent speed in x and y
(a_x, a_y)	agent acceleration in x and y
I_R	agent interaction range

Table I
DECISION AGENT PARAMETERS

Interactions:

Interactions are separated into three categories:

- 1) Interaction between agents. (Repulsion)
- 2) Interaction between agents and obstacles representatives including virtual environment boundaries. (Repulsion)
- 3) Interaction between agents and goal to reach. (Attraction)

- *Agent-Agent Interaction*: 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 : $\vec{F}_{A_i A_j} = \Gamma_a \cdot m_i \cdot m_j \frac{A_i \vec{A}_j}{\|A_i \vec{A}_j\|^3}$ for each agent i such as $\|A_i \vec{A}_j\| < I_R$. 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. $m_i = m_j = m$).

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

- *Agent-Obstacle Interaction*: This interaction has the same formulation as the previous. Thus, interaction between agent i and an obstacle o is given by: $\vec{F}_{O_i} = \Gamma_o \cdot m \cdot m_o \frac{O \vec{A}_i}{\|O \vec{A}_i\|^3}$ such as $\|A_i \vec{O}\| < I_R$, where m and m_o are respectively the mass of the agent i and the mass of the obstacle o . Obstacle mass is a mean to represents obstacle's real size measured by the sensor. $A_i \vec{O}$ 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:

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

- **Agent-Goal Interaction:** This interaction is modeled as a simple linear attraction force defined, for agent i , as follows: $\vec{F}\vec{d}_i = \Gamma_d \cdot m \cdot A_i \vec{D}_c$ with $A_i \vec{D}_c$ the distance from agent i to the destination point (translation of the result of the platoon control developed in [13]).

- **Decision agent behavior computation:**

Following classical Physics principles, each agent state (i.e. Position, Speed and Acceleration) can be computed using Newton's law of motion and taking into account all influences (agents, obstacles, goal) in the agent's perception range. 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: $\vec{\gamma}_i = \frac{1}{m}(\vec{F}\vec{a}_i + \vec{F}\vec{o}_i + \vec{F}\vec{d}_i)$. By substituting all the forces by their expressions and integrating twice, the following equation is obtained:

$$\begin{aligned} \vec{Z}_i(t) &= \vec{Z}_i(t-1) + \dots \\ &\dots \left(\vec{V}_i(t-1)\delta t + \frac{(\delta t)^2}{2m} \left(\vec{F}\vec{a}_i + \vec{F}\vec{o}_i + \vec{F}\vec{d}_i \right) \right) \end{aligned} \quad (4)$$

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

Agency evaluation and decision process:

As a result of agent interactions, taking into account obstacles and goal position, an evolving pattern of spatial distribution of decision agents emerges. The selection of the adapted control command for the vehicle is then based on the evaluation, through appropriate indicators, of agency emerging patterns. The indicators used are first based on agency mass distribution analysis on both static and dynamical points of view. Then, evaluation is made using the statistical physics partition function. This function can take into account constraints induced by obstacles on the agency.

- **Mass distribution analysis:**

Mass distribution analysis is based on regular global indicators widely in use in multi-agent systems and called *Global emerging structures and properties analysis* in section II. This analysis takes as input data the spatial and speed distributions of agents. From these two vectorial indicators are extracted. The first of them is based on the computation of the mean position of agents (i.e. the barycenter, since all agents have the same mass). The result of this computation is then denoted by \vec{P}_a (see figure 7) and corresponds to a vector which is oriented from the vehicle position to the mean decision agents' position. The second \vec{S}_a introduces dynamic information and is proportional to mean agents speed. This vector allows to anticipate direction change taken by agents as a result of the presence of obstacles. Both vectors are then combined to furnish control speed vector \vec{M} (cf. equation 5).

$$\vec{M} = \left(\vec{P}_a + \frac{\vec{S}_a}{\sqrt{nb_A}} \right) \frac{\|\vec{P}_a\|}{\left\| \left(\vec{P}_a + \frac{\vec{S}_a}{\sqrt{nb_A}} \right) \right\|} \quad (5)$$

with nb_A the number of agents.

In this equation, vector \vec{M} is calculated from vector \vec{P}_a and \vec{S}_a . Vector \vec{M} is set following equation 5, so that it has the same norm than vector \vec{P}_a , with an angular displacement (cf. Figure 7):

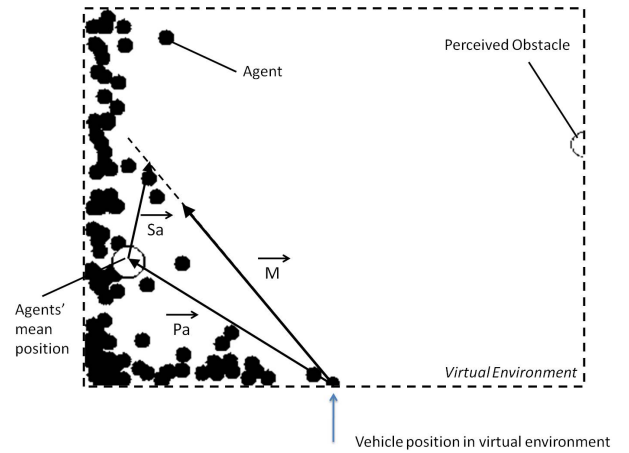


Figure 7. Decision vector computation

- **Obstacles constraints evaluation**

A third indicator is then used. This can be classified in the *Global measurement based on local phenomenon properties integration* category as explained in section II. It is inspired by statistical physics and thermodynamics and has initially been developed in [23]. This approach can be considered as a way to link the microscopic and a macroscopic points

of view in the agents' system. The statistical physics based method gives an important role to the partition function Z , which is computed thanks to energetic measurement. From this mathematical function indicators can be extracted that represent the global evaluation of the system state based on local phenomena and constraints evaluation. According to the interaction model, the energy measurement can be detailed as follows :

- **Kinetic energy:** . In the following equation, the agent i is represented by its mass m_i and its speed \vec{V}_i : $E_K = \frac{1}{2}m_i\vec{V}_i\vec{V}_i$
- **Potential energy:** for agent i , potential energy U is computed using the classical expression: $U = \delta W + \delta Q$ where δW is the work flow on the system point of view and δQ the heat flow (in the case of computer systems, $\delta Q = 0$). The work flow δW computation is made using only conservative forces (Cf. equation III-C).

$$E_p = \delta W = \vec{F}_{total}\vec{du} = \dots \quad (6)$$

$$\dots = \sum_i \vec{F}_{d_i}\vec{du} + \sum_i \vec{F}_{a_i}\vec{du} + \vec{F}_{d_0}\vec{du}$$

with \vec{du} a unit vector in the direction of agent speed and \vec{F}_{d_i} , \vec{F}_{a_i} , \vec{F}_{d_0} the forces exposed in III-C.

Agent's energy corresponds to the sum of the kinetic and the potential energy : $E = E_k + E_p$. Free energy can then be computed as follows:

$$\begin{cases} A(T,V,N_i) = -\ln(Z) \\ Z = e^{-\beta E} \end{cases} \quad (7)$$

With A the free energy potential. T the temperature, V the volume and N_i number of element supposed to be constant. The assumption about having constant volume and a fix number of element are realistic for RMAS. The system volume is the RMAS environment volume. Note that this environment where agent evolve is naturally limited, so V can be considered as constant. In the same manner, the number of elements is the number of agents which can be constant too. Constant temperature means that the system evolves with a constant thermal contact. So in simulation, we suppose that we have a constant temperature in the environment. This assumption has already been applied in *Theory of communication* [24] and to explain *statistical entropy* [21]. From now, system indicators can be computed [21].

Finally, the global decision vector \vec{D} is computed according to free energy (A) value using this equation:

$$\vec{D} = (1 + \kappa.A) \cdot \vec{M} \quad (8)$$

with κ a scalar factor and A the thermodynamic potential. This vector is then applied as a command law to the vehicle.

IV. EXPERIMENTAL RESULTS

A. Global overview

The goal of experiments presented in this paper is to check the reliability and the performance of the proposed model with real vehicles. Scenarios have been defined using a side by side platoon frame. The goal is then to study the lateral stability of follower vehicles when obstacles disturb ideal conditions. Two experiments are reported:

- **Single obstacle avoidance:** This consists in studying the follower vehicle behavior when an obstacle is appears on its nominal path.
- **Multiple obstacles avoidance:** This consists in studying follower vehicle behavior with a multiple-obstacle configuration.

B. Vehicles description

Experiments were performed with commercial electric vehicles (GEM cars, figure 8) modified in our laboratory to be controllable by a standard computer (direction, speed and brake are controlled by an embedded computer).



Figure 8. SeT-car: SeT laboratory modified electrical vehicle

- Vehicle Control Structure

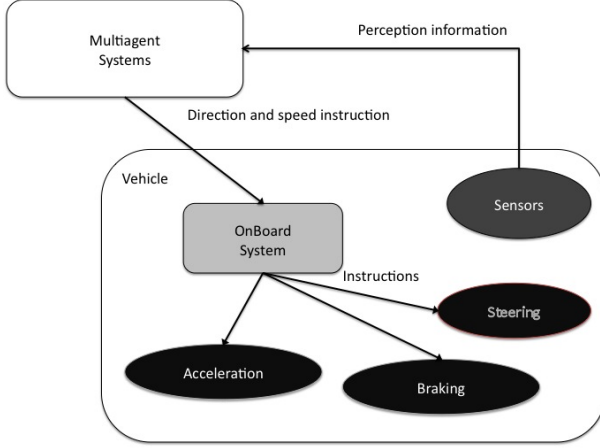


Figure 9. SeT-Car architecture

Figure 9 illustrates the organization of SeT-Cars' embedded system. The vehicle's multi-agent system receives perception data and sends direction and speed commands to the vehicle onboard computer. The onboard control device translates these informations into steering, braking and acceleration commands to each specific operation module.

- Vehicle Perception

Vehicle's perception is performed by two laser range finders (LRF). One of them is devoted to the obstacle avoidance and the other to the platoon as itself (cf. figure 4).⁴

Distance and the angle between vehicles are computed by the platoon LRF. Reflective beacons have been used to improve perception quality.

Each vehicle is also equipped with a centimetre-precise RTK GPS. These sensors are used for vehicles position capture during experiments. The retrieved information is then used off-line to draw the vehicles' real trajectories.

- Platoon parameters.

Vehicle's intrinsic properties (mass, engine power,...) and regular public transportation characteristics (maximum acceleration/deceleration,...) have been used to define platoon parameters. These are exposed in table 10:

⁴Two sensors have been used in order to obtain functional redundancy and to facilitate experiment control (failure detection,...). However, the system can also work with only one sensor for both functions.

Mass	500 kg
Max speed	8 km/h (2 m/s)
Acceleration	0.1 m/s ²
Deceleration	-0.1 m/s ²
Max deceleration	-3 m/s ²
Regular distance (l_0)	180 cm + 300 cm (vehicle length)
Lateral distance (l_a)	120 cm

Figure 10. Parameters of vehicle model and platoon control

The dimensions provided and measured in experiments are measured from center to center of vehicle or obstacle.

C. Test zone

Experiments were performed on the Belfort's technological park.

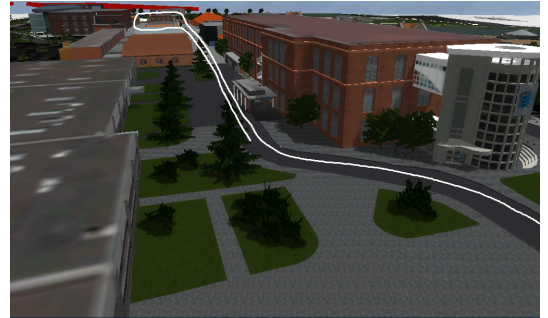


Figure 11. Experimentation environment

Figure 11 shows a 3D representation of the experiment's area. This has been chosen because of its properties (curves, topology,...) which resembles to urban territory usual characteristics.

D. Experimental results

1) *Single obstacle avoidance:* The goal of the experiment is to observe follower vehicle behavior when an obstacle is on its nominal trajectory. Figure 12 shows the experimental configuration, i.e a single obstacle is placed on the road on the follower vehicle trajectory.

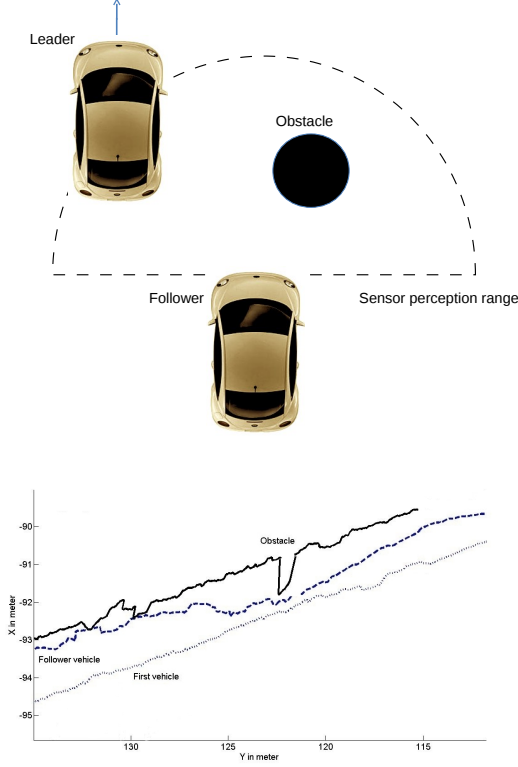


Figure 12. Path results from the lateral obstacle avoidance

Figure 12 shows vehicles trajectories during the experiment (with X Y the distance in meter from the start vehicle position). Follower vehicle avoids the box by producing an escape trajectory and recovers afterwards the nominal side by side platoon configuration. When obstacles enter into the perception range, the follower vehicle decreases the lateral distance to on average 0.54 meter (lateral distance to an obstacle equals 2 m and vehicle width equals 1.2 m).

2) *Multiple obstacle avoidance:* This experiment is aimed at studying the follower vehicle trajectory with a multiple obstacles configuration.

Figure 13 shows the obstacle configuration: a barrier made of trees on the left and a box wall on the right. Leader's vehicle path goes between those obstacles. Follower trajectory needs several corrections to avoid obstacles while respecting side by side platoon configuration.

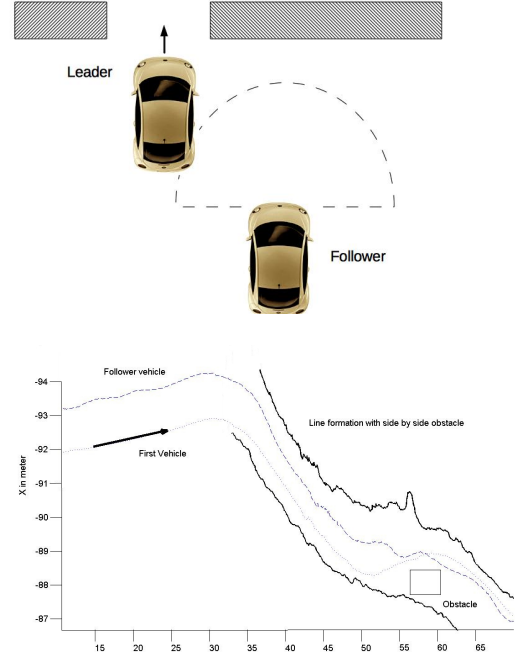


Figure 13. Path results from the multiples obstacles avoidance

Figure 13 shows the follower vehicle trajectory. Obstacles positions are also shown (measures are made with RTK-GPS). Follower vehicle reduces lateral distance proportionally to inter-obstacle distance. When obstacle constraints are too high, the follower vehicle decreases the lateral distance up to 0.14 m. In this experiment, the lateral distance between the head vehicle and the obstacle is less than 1.5 m.

V. CONCLUSION

This paper presents a reactive multi-agent system based on an active environment and a local-global evaluation of agency emerging pattern for the decision making. In this model, agents' virtual environment plays a key role by merging and translating data from sensors, which observe the real vehicle surrounded environment. Agents are reacting to environment variations through attraction/repulsion behaviors based on Physics. Agency emerging pattern is then characterized thanks to two kinds of indicators. The first type of indicators is based on global evaluation of pattern. The second indicator is based on a global evaluation which takes into account the local phenomenon. These indicators are based on a partition function used in statistical physics. This system has been successfully used on the real vehicle with different obstacle configurations. Results obtained are encouraging since platoon lateral and longitudinal stability is preserved in normal use (i.e. without any obstacle). Besides,

the system shows relevant adaptive abilities when the platoon is faced with obstacles.

In order to continue this research, we are now working on several key points. First, development is made to integrate data from various sensors in order to be able to deal with different points of view (i.e. taking into account information on sidewalk given by a specifically oriented sensor). Finally, we also plan to modify this application to be able to use it with other geometrical configurations. On the other hand, experimentation with more elaborated measurements and data analysis have to be performed to precisely evaluate a set of quality indicators. Those works are done with the support of the French ANR (National Research Agency) through the ANR-VTT *SafePlatoon*⁵ project (ANR-10-VPTT-011).

REFERENCES

- [1] H.V.D. Parunak, R. Bisson, S. A. Brueckner, "Agent interaction, multiple perspectives, and swarming simulation", *AAMAS '10 Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems, Volume 1, International Foundation for Autonomous Agents and Multiagent Systems*, Richland, SC, 549-556.
- [2] A. Glad, O. Simonin, O. Buffet, F. Charpillet, "Influence of different execution models on patrolling ant behaviors: from agents to robots", In *Proceedings of the 9th International Conference on Autonomous Agents and Multiagent Systems, Volume 3 (AAMAS '10)*, Vol. 3. International Foundation for Autonomous Agents and Multiagent Systems, Richland, SC, 1173-1180.
- [3] E. Bonabeau, M. Dorigo, G. Theraulaz, *Swarm Intelligence: From Natural to Artificial Systems* M. Dorigo and M. Birattari, *Swarm intelligence*, Scholarpedia (1999)
- [4] S. Brueckner. *Return from the ant : Synthetic eco-systems for manufacturing control*. PhD thesis, Humboldt University Berlin, Department of Computer Science, 2000.
- [5] J. Bajo, D. Tapia, S. Rodriguez, A. Luis, and J. Corchado. "Nature inspired planner agent for health care", *International Work-Conference on Artificial and Natural Neural Networks IWANN 2007*: 1090-1097.
- [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 Artificial intelligence*, 2004, vol. 2977, 299 p.
- [7] F. Gechter, V. Chevrier, F. Charpillet, "A Reactive Agent-Based Problem-Solving Model : Application to Localization and Tracking", *ACM Transactions on Autonomous and Adaptive Systems (ACM TAAS)*. volume 1 issue 2 . p 189-222 (2006)
- [8] C.W. Reynolds. *Flocks, herds, and schools: A distributed behavioral model*, in *computer graphics. SIGGRAPH Conference Proceedings*, pages 25-34, 1987.
- [9] K. Zeghal and J. Ferber. "A reactive approach for distributed air traffic control", In *international conference on artificial intelligence and expert systems* (1994), pp. 381-390.
- [10] H.V.D. Parunak, "Go to the Ant: Engineering Principles from Natural Agent Systems", *Annals of Operations Research*, Vol. 75, No. 1. (1997), pp. 69-101
- [11] J-P. Müller. and H.V.D. Parunak, "Multi-Agent systems and manufacturing". *Proceedings of INCOM 1998*, Nancy/Metz, France, June 24-26, 65-170.
- [12] D. Weyns, H.V.D. Parunak, F. Michel, T. Holvoet, J. Ferber: "Environments for Multiagent Systems State-of-the-Art and Research Challenges", *E4MAS 2004*: 1-47.
- [13] J-M. Contet, F. Gechter, P. Gruer, A. Koukam. "Reactive Multi-agent approach to local platoon control: stability analysis and experimentations", *International Journal of Intelligent Systems Technologies And Application*. 2010. 231-249
- [14] F. Gechter, J-M. Contet, P. Gruer, A. Koukam, "Car-driving assistance using organization measurement of reactive multi-agent system", *Procedia Computer Science* 1(1): 317-325., 2010
- [15] A. Drogoul, J. Ferber, E. Jacopin, "Pengi: Applying Eco-problem Solving for Behavior Modeling in an Abstract Eco-System", *European Simulation Multiconference, ESM'91*, Mosekilde (Ed.), 1991. 189-222.
- [16] O. Simonin, and J. Ferber, "Modeling Self Satisfaction and Altruism to handle Action Selection and Reactive Cooperation", *Simulation of Adaptive Behavior (SAB2000)*, From Animals to Animats, Paris, Int. Soc. For Adaptive Behavior, pp. 314-323, 2000.
- [17] H.V.D. Parunak, S. Brueckner, "Entropy and self-organization in multi-agent systems", In *Proceedings of the fifth international conference on Autonomous agents (AGENTS '01)*. ACM, New York, NY, USA: 124-130
- [18] N. Gaud, F. Gechter, S. Galland, A. Koukam, "Holonc Multiagent Multilevel Simulation: Application to Real-Time Pedestrian Simulation in Urban Environment", *International Joint Conference on Artificial Intelligence IJCAI* p 1275-1280, Hyderabad, India - January 6-12 (2007).
- [19] T.R. Balch, "Hierarchic Social Entropy: An Information Theoretic Measure of Robot Group Diversity", *Autonomous Robots* 8(3): 209-238 (2000).
- [20] Y. Kanada, and M. Hirokawa, "Stochastic Problem Solving by Local Computation based on Self-organization Paradigm", *27th Hawaii International Conference on System Sciences (HICSS-27)*, 82-91, 1994
- [21] R. Balian, *From Microphysics to Macrophysics*, Springer, 2007
- [22] J.P. Desai, J. Ostrowski, and V. Kumar. "Controlling formations of multiple mobile robots", In *Proceedings of the IEEE International Conference on Robotics and Automation*, vol.4, no., pp.2864-2869 vol.4, 16-20 May 1998
- [23] J-M. Contet, F. Gechter, J-P. Gruer, A. Koukam, Evaluation of global system state thanks to local phenomena, *Biennial European Conference on Artificial Intelligence (ECAI)*, Greece, Patras, June 21-27 (2008).

⁵<http://web.utbm.fr/safeplatoon>

- [24] C.E. Shannon, *The Mathematical Theory of Communication*, Bell System Technical Journal, vol. 27, pp. 379-423 and 623-656, July and October, 1948.
- [25] Martinet P., Thuilot B., Bom J.: "Autonomous Navigation and Platooning using a Sensory Memory", *International IEEE Conference on Intelligent Robots and Systems, IROS'06, Beijing, China, (2006)*.