



**HAL**  
open science

# Multi-agent simulation of autonomous industrial vehicle fleets: Towards dynamic task allocation in V2X cooperation mode

Juliette Grosset, Aj Fougères, M Djoko-Kouam, Jean-Marie Bonnin

## ► To cite this version:

Juliette Grosset, Aj Fougères, M Djoko-Kouam, Jean-Marie Bonnin. Multi-agent simulation of autonomous industrial vehicle fleets: Towards dynamic task allocation in V2X cooperation mode. *Integrated Computer-Aided Engineering*, 2024, 31 (3), pp.249-266. 10.3233/ICA-240735 . hal-04527026

**HAL Id: hal-04527026**

**<https://hal.science/hal-04527026>**

Submitted on 9 May 2024

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

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

# Multi-agent simulation of autonomous industrial vehicle fleets – towards dynamic task allocation in V2X cooperation mode

J. GROSSET<sup>a,b</sup>, A.-J. FOUGERES<sup>a,\*</sup>, M. DJOKO-KOUAM<sup>a,c</sup>, and J.-M. BONNIN<sup>b</sup>

<sup>a</sup> ECAM Rennes Louis de Broglie, Bruz, France

<sup>b</sup> IMT Atlantique, IRISA, UMR 6074, Rennes, France

<sup>c</sup> IETR, UMR CNRS 6164, CentraleSupélec, Rennes, France

**Abstract.** The smart factory leads to a strong digitalization of industrial processes and continuous communication between the systems integrated into the production, storage, and supply chains. One of the research areas in Industry 4.0 is the possibility of using autonomous and/or intelligent industrial vehicles. The optimization of the management of the tasks allocated to these vehicles with adaptive behaviours, as well as the increase in *vehicle-to-everything communications* (V2X) make it possible to develop collective and adaptive intelligence for these vehicles, often grouped in fleets. Task allocation and scheduling are often managed centrally. The requirements for flexibility, robustness, and scalability lead to the consideration of decentralized mechanisms to react to unexpected situations. However, before being definitively adopted, decentralization must first be modelled and then simulated. Thus, we use a multi-agent simulation to test the proposed dynamic task (re)allocation process. A set of problematic situations for the circulation of autonomous industrial vehicles in areas such as smart warehouses (obstacles, breakdowns, etc.) has been identified. These problematic situations could disrupt or harm the successful completion of the process of dynamic (re)allocation of tasks. We have therefore defined scenarios involving them in order to demonstrate through simulation that the process remains reliable. The simulation of new problematic situations also allows us to extend the potential of this process, which we discuss at the end of the article.

**Keywords:** multi-agent planning, cooperative mobile robot, collective problem solving, multi-agent simulation, communication V2X.

## 1. Introduction

The smart factory leads to a strong digitalization of industrial processes [1, 2], but also to continuous communication between the different tools, systems and workstations, integrated into production, storage and supply chains. Among the challenges of Industry 4.0 are the development and optimization of data, product and material flows [3]. Certain technological bricks have been defined [4], in particular for the use of autonomous mobile systems [5]: automated guided vehicle (AGV), autonomous industrial vehicle (AIV), and other autonomous/intelligent mobile robot [6].

The deployment of fleets of AIVs is problematic on several levels: the location of the vehicles, the fluidity of traffic, the perception of disturbances in the environment (dynamics), the vehicle heterogeneity, the acceptability by employees and even their cooperation. Currently, to achieve their goals, vehicles have navigation autonomy linked to rails, physical or virtual beacons. They follow predetermined trajectories while detecting obstacles to avoid collisions. Their decision-making capacities are often limited to following these predetermined trajectories and stopping in the event of obstacles. During incidents, the presence of obstacles or broken down vehicles, for example, the modification of routes is

ensured by the central system or a supervisor, which sends a change of mission order to the vehicles.

The level of vehicle autonomy is well characterized in the context of the road vehicle: six levels of autonomous driving have been proposed [7, 8]. However, this is not the case for autonomous industrial vehicles. The autonomy of AIVs is often limited to local visibility of their environment, as well as visibility and knowledge of other AIVs operating in the same environment. The ability of AIVs in the same fleet to exchange information between themselves, with the active elements of the infrastructure that they encounter on their route, or with the human beings who operate in their environment, should improve the decision-making autonomy of these vehicles, make the solutions collectively defined more robust, and enable greater adaptability to changing traffic conditions [9].

For an AIV to be autonomous, it must be able to manage and control a set of tasks well discussed in the literature: perception [10, 11], mapping [12, 11], task allocation [13], localization [14, 15], path planning [16, 17] and path finding [18], motion planning [14], and vehicle management [19, 20]. The autonomy of an AIV fleet can be further increased if the AIVs can collectively manage and optimize the task allocation problem: that means they collectively assign the set of tasks to the set of AIVs with a high level of efficiency (for instance,

minimization of energy costs or mission completion time) [13, 21].

Task allocation and planning are often managed centrally, even semi-centrally when global and local planning are differentiated [22]. For the proper functioning of autonomous and dynamic systems, the requirements of flexibility, robustness and scalability, lead to consider decentralized mechanisms to react to unexpected situations. Autonomy and decentralization are two excessively linked notions to the extent that an autonomous system operates and make decisions autonomously, and a system is decentralized if the decision, which are made, are not centrally controlled [23, 24]. The problem of task allocation [25, 26], tasks which grouped together can constitute missions, must therefore be thought of in a decentralized way [13]. However, before being definitively adopted, decentralization must first be modelled and simulated [23]. The concept of multi-agent systems is well suited to perform this type of modelling and simulation [27, 28, 29]. Furthermore, many tools have been developed to facilitate agent-based modelling and simulation (ABMS) [30]. It should also be noted that multi-agent systems are also used to physically control AGVs [31, 32, 33].

Some studies have focused on leveraging data from other autonomous vehicles within a fleet to control the physical behaviour of a specific autonomous vehicle from the same fleet. This is the case with reference [34], which proposes a suspension control strategy based on a deep learning algorithm. Also, based on deep learning algorithms, techniques for controlling connected autonomous vehicles are well-documented in the literature. In [35], the authors addressed this topic for adaptive convoy control, while [36], within the same subject, specifically considers communication failure situations.

In this article, we start by presenting a state of the art on the allocation of tasks in centralized or decentralized modes as well as on the different types and standards of V2X communication. In section 3, we propose a dynamic task (re-)allocation process for an AIV fleet. In section 4, we successively present the agent model designed to simulate this task (re-)allocation process, the simulation interface used and then the different test scenarios proposed in an industrial context such as a smart warehouse. We discuss the different results obtained in simulation in section 5. Finally, we conclude on the proposed dynamic task allocation-reallocation process, and then we present different work perspectives.

## 2. State of the art

### 2.1. Task allocation for mobile multi-robots

Task allocation consists of optimally assigning a set of tasks to be performed by agents, actors, robots or processes, grouped and organized in a global system [37]. This is the case for mobile multi-robot systems [25, 26] or the AIV fleets addressed in this article [38, 39].

In global way, the multi-robot task planning includes two processes: 1) a multi-robot task decomposition that refers to how a team mission can be decomposed into

several subtasks which can be completed by the robots, and 2) the multi-robot task allocation that consist to determinate how each subtask can be assigned to one robot [40]. Thus, two major objective functions are defined for analysing a solution to the multi-robot task allocation problem: the makespan (number of time steps required for all robots to reach their tasks), and sum of costs (sum of time steps requires by each robot to reach its task) [41].

In the field of mobile robotics, the taxonomy presented in [13] makes it possible to better characterize the task allocation process. A task can be assigned to one or more robots, and several tasks can be assigned to heterogeneous robots or multitasking robots. These tasks can be allocated instantly or extended over time. As many combinations as exhaustively detailed by numerous surveys on the issue of Multi-robot task allocation [26].

Task allocation in multi-robot systems is complex and the tasks themselves may have many time, precedence or resource constraints [37]. It is then necessary to determine the objectives to be optimized, in particular among: the travel cost (time, distance, fuel or battery consumption), the fitness (quality of task performing), the reward (gain of task completing), the priority (urgency of task completing), and the utility (balance between cost and fitness-reward) [13].

For reasons of flexibility, robustness and scalability, we are interested in decentralized task allocation solutions. These solutions must be able to assign tasks to a fleet of homogeneous or heterogeneous robots. Different models of solutions are proposed in the literature, mainly the following three: 1) solutions based on optimization (exact algorithms, dynamic programming, heuristics and metaheuristics) [26]; 2) solutions based on the Contract Net Protocol, particularly in the field of multi-agent systems (an initiating agent sends a call for proposals to the entire community of agents, chooses the best proposal received, then informs all the agents choice) [42]; or 3) solutions based on the market concept (announcement by an auctioneer, submission by bidders, selection by the auctioneer and award by the auctioneer) [43]. Solutions based on the market concept can easily be applied in a distributed context, where each mobile robot is able to become an auctioneer [44]. For each situation, a single mobile robot is appointed auctioneer [45]. He retains this role until the situation is definitively managed.

### 2.2. V2X communication and cooperation

In order to be able to successfully perform all the tasks assigned to them, AIVs must coordinate and therefore cooperate and share information about their activity and their perception of the environment. Four types of coordination can then be implemented according to increasing degrees of autonomy: centralized (with the assistance of a coordinator), negotiated (with respect for a specific negotiation protocol), agreement (with dynamic protocol) and emergent (with self-organizing algorithms) [46]. The last three coordination approaches are characterized as decentralized, they therefore require that the AIVs be connected to each other [47, 48, 49] and/or communicate frequently [32]. The link between

V2X communication and cooperation, often presented from the perspective of cooperative control, is widely emphasized and extensively documented in the literature, as exemplified in [50] and [51].

The communication that takes place between vehicles is commonly referred to as *vehicle-to-vehicle communication* (V2V) [52]. While the literature widely covers V2V communications, it seems relevant to highlight specific variations, such as decentralized intersection traffic light synchronization [53], simulation of mixed traffic with cooperative lane changes [54], and traffic light optimization [55]. Vehicles can also request or receive information from the infrastructure, in particular to warn them of the presence of obstacles. This communication is then called *vehicle-to-infrastructure* (V2I) communication [52], and is highly developed in environments such as smart warehouses [56].

Many systems use these two last modes of communication to intelligently manage and coordinate connected vehicles in problematic situations such as crossing an intersection [57], accessing a highway [58], allocating a parking space in a smart car park [59], platooning and traffic flow optimization [46], or multiple vehicle cooperation and collision avoidance [60].

Communication between intelligent vehicles and pedestrians/humans is also used to increase vehicle autonomy and ensure better safety for humans in shared human-robot environments [61]. This communication is called *vehicle-to-pedestrian communication* (V2P) [62]. It also makes it possible to set up collaboration between humans and autonomous vehicles, for example when a vehicle has detected an obstacle that requires the intervention of a human operator to clear it. In the rest of the article, we will call *vehicle-to-everything communications* (V2X) all of the three communication modes presented above (V2V, V2I, and V2P) [63].

The ETSI (*European Telecommunication Standard Institute*) has defined standard communication messages for intelligent transport systems (ITS), which we have transposed and used in previous works [64]. The *Decentralized Environmental Notification Message* (DENM) (ETSI standard EN 302 637-3 [65]) are alert messages, issued at the time of an unexpected event in order to notify it, and therefore to cooperate by broadcasting information in the geographical area concerned. The *Cooperative Perception Messages* (CPM) (ETSI standard TR 103 562 [66]) provide warning and help vehicles make decisions on their route. Thus, if an AIV detects an obstacle obstructing an aisle of a warehouse, it can signal it to the other AIVs which can then recalculate and plan a new route, if necessary to be able to accomplish their tasks. An infrastructure camera can also detect obstacles and send this type of message to AIVs. Note that beyond AIVs, recent literature suggests various types of V2V communications. For instance, [67] proposes, in the field of urban transportation, a dynamic control strategy for a fleet of connected buses using a multi-agent system that leverages bus history along with traffic information. This approach enables each bus to adjust its movement based on weighted information from downstream buses.

Another type of V2V communication could be useful to improve cooperation between AIVs in carrying out

their tasks. Indeed, if an AIV finds itself blocked by obstacles in a warehouse aisle, broken down or generally unable to perform the task in progress, it sends a DENM message by default. It could then be useful for him to send a cooperative message to delegate the realization of his task with the necessary information.

We therefore propose a new *Cooperative Task Message* (CTM), which would allow in particular delegating a task. Häfner et al. [68] propose a protocol with four new types of messages, including the *Cooperative Response Message* (CRM) for transmitting the response to a request for cooperation. The AIV agents, modelled in the simulation part of this article (*cf.* §4), will use this type of message in feedback from the CTM messages to signify their agreement to take charge of a task for example.

### 3. Tasking process for AIV fleets

The common objective of the AIVs belonging to the same fleet is to perform all the tasks assigned to them while respecting a certain number of time and priority constraints. In this context, given  $T$  a set of tasks to be performed, a task  $\tau \in T$  is defined by the following tuple (1):

$$\tau = \langle \tau_{id}, \kappa, p_{start}, p_{end}, t_{start}, t_{end}, pr, \sigma \rangle \quad (1)$$

Where  $\tau_{id}$  is the task identifier;  $\kappa$  is the task category (for example in a warehouse, move goods, *i.e.*: “load the goods at a starting point to bring it to an ending point”);  $p_{start}$  is the task starting point;  $p_{end}$  is the task ending point;  $t_{start}$  is the task starting time;  $t_{end}$  is the task ending time;  $pr$  is the task priority; and  $\sigma$  is the task status.

A set of  $n$  tasks can be grouped to form a mission  $m$  defined as follows (2):

$$m = \langle \tau_1, \dots, \tau_n \rangle \quad (2)$$

Given two sets  $V$  and  $T$ , such that  $V = \{v_1, \dots, v_n\}$  is a set of  $n$  AIVs and  $T = \{\tau_1, \dots, \tau_m\}$  is a set of  $m$  tasks, we define the two following functions: the function  $\mathcal{C}$  which allows to calculate the cost  $c_{\tau_i, v_j} \in \mathcal{C}$  of performing a task  $\tau_i$  by an AIV  $v_j$  (3), and the function  $\mathcal{A}$  which allows to allocate each task  $\tau_i \in T$  to an AIV  $v_j \in V$ , depending on the cost  $c_{\tau_i, v_j}$  (4).

$$\mathcal{C} : T, V \rightarrow \mathcal{C} \quad (3)$$

$$\mathcal{A} : T, V, \mathcal{C} \rightarrow A \quad (4)$$

The task allocation process that we have defined is based on a market model type solution [45]. The flexibility of this solution allows a good adaptation for a decentralized system. In the rest of the article, we will apply and test this task allocation process to a fleet of homogeneous mobile robots, loading and unloading goods in a warehouse. Its process is depicted in Figure 1.

When a task has been defined by an organizational actor (call *supervisor* in the rest of the article), it is sent by the supervisor using a CTM message to an available AIV (*i.e.*, having no tasks to be accomplished at this time), who will play the role of auctioneer. Then, this AIV auctioneer will send an acknowledgment using a CRM message, upon receipt of the work package.

For greater efficiency, before starting the auctions, the AIV auctioneer can cluster certain tasks received. This involves, in particular, associating tasks having ending points and starting points in common. For example, consider the task  $\tau_1$  “Bring material from a parking source node  $n^{\circ}21$ , to a storage point node  $n^{\circ}13$ ” ( $\tau_1 = [21, 13]$ ). An associated task  $\tau_2$  could be to go to a parking lot after completing task  $\tau_1$ , for instance: “Take material from storage point node  $n^{\circ}13$  and bring it to parking node  $n^{\circ}25$ ” ( $\tau_2 = [13, 25]$ ). Then, the two clustered tasks  $\tau_1$  and  $\tau_2$  are represented by the mission  $m_i = \langle \tau_1, \tau_2 \rangle = \langle [21, 13], [13, 25] \rangle$  (in the tables of section §5, we simplify the writing of the mission  $m_i$  by using the triplet  $\langle 21, 13, 25 \rangle$ ).

Following the association of tasks to form missions, these are sent to all vehicles by the AIV auctioneer. Then, AIVs calculate costs to perform the various missions, taking into account a set of performance indicators (distance, energy, time, etc.), and produce their bids.

Each AIV returns all of its bids to the AIV auctioneer. This runs a simple optimization algorithm: the clustered tasks are listed in priority order. Subsequently, the AIV that bid with the lowest cost for a mission wins the auction (strategy of choice by the auctioneer for the least expensive proposal).

To summarize at the communication level, the AIV auctioneer allocates a task (eventually clustered tasks) to each chosen AIV, sending to it via a CTM message. The receipt of this CTM message by an AIV ends with a CRM message sent to the auctioneer in order to inform him of his acceptance. The different interactions between the supervisor, the AIV auctioneer and the other AIVs are identified in the sequence diagram of Figure 2. This allocation mechanism is also used for task reallocation. Indeed, a robot can re-auction a task becoming an auctioneer in turn to manage the reallocation of all or part of its tasks. The robot offering the best bid will then add the reallocated task(s) to the set of tasks it must perform.

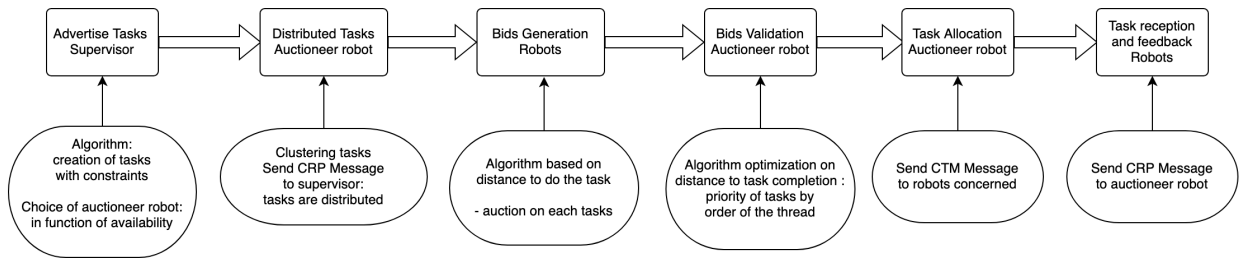


Figure 1. Task allocation process

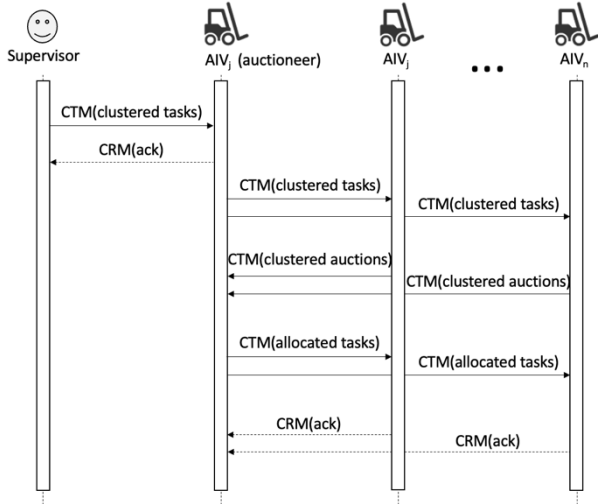


Figure 2. CTM and CRM messages exchanged during task allocation

## 4. Multi-agent simulation of AIV fleets

### 4.1. Agent model for AIVs

Many simulation-based approaches have been proposed in literature in the context of Industry 4.0,

mainly : Agent-based modelling and simulation, Discrete Event Simulation, System Dynamics, Virtual Reality, Augmented Reality, Artificial Intelligence, Petri Nets simulation, Hybrid Simulation (characterized by the combination of two or more simulation methods, i.e., multi-paradigm model), Digital Twins, or Virtual Commissioning [23]. Among these approaches, the use of the agent paradigm to simulate or model complex, interactive, adaptive, distributed or cooperative systems has become common [27]. Indeed, the properties for which the agent-based approach is most suitable are: modularity [69], decentralization [28, 24], autonomy [30], flexibility [70] and agility [23]. Agent-based systems (ABS) have thus been proposed in many engineering fields, such as for industrial applications [71], for intelligent manufacturing [72, 73, 74], for supply chain management [75], for autonomous vehicles [29, 76], or for AGV [31]. In this last area, the case study “AGV control in an industrial bakery” proposed by Mes and Gerrits is well representative of the developments of ABS to control AGVs in the context of industry 4.0 [33]. The distribution of agents (decentralization) allows the systems that implement them to be more flexible and reactive. On the other hand, the agent concept is well suited for modelling and simulating cyber-physical systems [77] including AGVs or AIVs.

The basic definition of an agent is something that acts, to which it is useful to add three key properties:

autonomous, interactive and adaptive. Thus, an agent is an autonomous entity that can adapt to and interact with its environment or other agents [42].

Other properties can be associated with the concept of agent: situated, social, flexible, proactive, and robust [33]; but also, mobile, intelligent, rational, temporally continuous, coordinative, cooperative, competitive, rugged (able to deal with errors and incomplete data robustly) [78]. These different properties will be discussed in section 5, in relation to the results obtained during the simulation of the scenarios proposed to illustrate this article.

The basic behaviour of an agent can be modelled by an automaton [79]: 1) an agent perceives inputs using detectors (sensors); 2) it analyses, processes and interprets its inputs (recognizes, normalizes, etc.); 3) it determines the actions to be performed by interpreting the data, examining its goals and its current state; then 4) it performs the actions through effectors (its outputs).

The design of an agent-based system (ABS), often studied from a processual or methodological point of view [80], supposes that the designer proceeds with a local vision to respect the fact that each agent manages its own knowledge and actions (autonomy). ABS design support languages are numerous [13, 81]. The support tools, both for agent-based modelling and for agent-based simulation, are also very numerous: JADE, GAMMA, Matlab, MASON, NetLogo, AnyLogic, SN2, Sinalgo, etc. [82, 83, 84].

In [85], we proposed a four-step ABS design method that largely refers to AUML (*Agent Unified Modelling Language*) [86]:

**Step 1.** Definition of use case diagrams (services provided by ABS).

**Step 2.** For each use case, draw sequence diagrams representing the interactions (exchanges of messages and scheduling) between the agents involved.

**Step 3.** From the sequence diagrams, which identify the agents, the objects and their interactions, create a class diagram: the objects are associated with classes, the

messages exchanged (requests for service between objects) are translated into operations on classes, parameters associated with operations are translated into class attributes – it may be possible to complete this diagram with a collaboration diagram.

**Step 4.** From the class diagram, define the behaviours of each agent using a state or activity diagram. These behaviours integrate a whole set of functionalities fulfilled by the agents. As an example, Mes and Guerrits specify a set of six generic functionalities to be fulfilled by AGV agents [33]: *DemandManagement*, *ParkManagement*, *VehicleScheduling*, *VehicleRouting*, *ConflictResolution* and *BatteryManagement*.

Figure 3 below presents the agent architecture developed to simulate the AIV traffic situations that we wish to study. Each agent of this architecture has its own knowledge (in particular on the other AIV agents, on the traffic environment, as well as on the paths and tasks allocated), and has functional capacities of observation, communication, decision and of action [39].

In this architecture, an industrial infrastructure (production lines, warehouses, etc.) is deployed in an environment, and is composed of a circuit and active elements such as beacons, tags, stations and cameras. These active elements are modelled as agents. Static or dynamic obstacles (e.g. human operators) may be present in the environment. Human workers or operators, with whom AIV agents can communicate to carry out cooperative activities, are also modelled as agents.

AIV agents, grouped in the same agent community, perform missions defined by paths on the traffic map. They are equipped with a radar that allows them to evolve in a partially known environment, and can thus avoid obstacles and collisions between them. The internal architecture of these agents is defined in such a way as to allow them to manage the four essential functions of an autonomous mobile robot: perception, localization, planning and control [87].

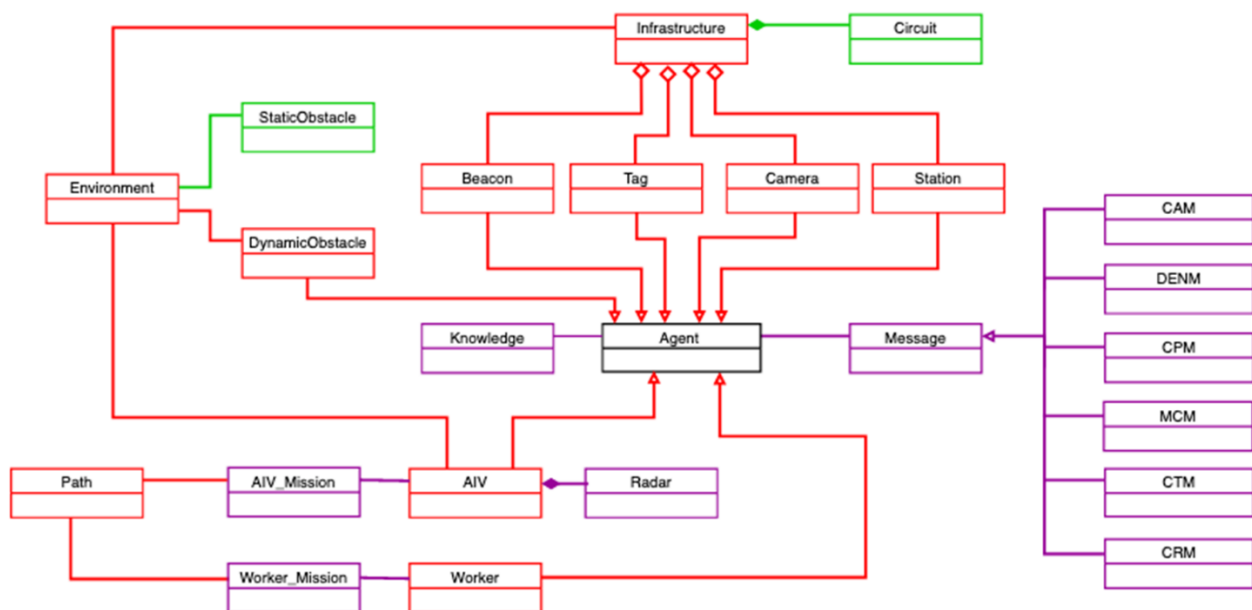
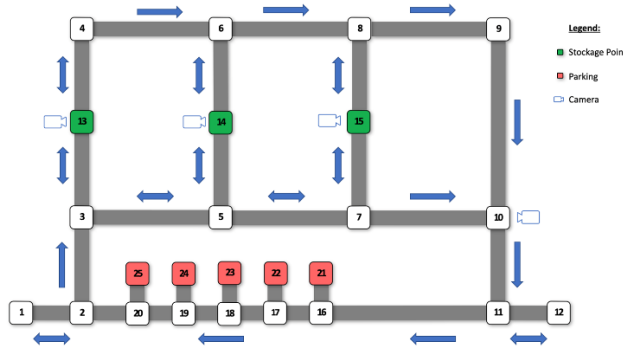


Figure 3. Simulator architecture: dynamic elements in red, static in green, and not related to the environment in purple

AIV agents cooperate with each other to optimize the performance of a set of missions, which are transmitted to them by a supervising agent who acts as an organizing service (these missions can be sent to the AIV in packets or in continuous flow). The active elements of the infrastructure (cameras, tags, beacons, stations, etc.) participate in the cooperation [88], in particular by contributing to the safety of AIV travel. To ensure this cooperation based on inter-agent communications in VTX mode, different types of standardized messages are used (CAM, MCM, CPM, CTM, CRM and DENM).

The environment chosen in this article, to illustrate our scenarios of problematic situations, is a typical warehouse presented in Bechtsis et al. [89]. Certainly, this environment is small, but it allows us to detail very finely and from an educational point of view all the scenarios that we have defined, in particular the 4 scenarios studied in the following sections. However, the framework presented in Figure 3 was developed to model larger warehouses and to simulate problematic traffic situations involving a large number of AIVs (for example, it is currently used to simulate the activity of a hundred baggage conveyor robots in an airport).

The warehouse circuit shown in figure 4.a comprises several intersections, in which the vehicles can arrive from different sides as in a warehouse. This type of traffic



plan offers the different characteristics of an industrial environment and allows us to carry out simulated experimental tests, in accordance with realistic scenarios of an industrial context. Five AIV agents are integrated into this environment corresponding to the five parking spaces available in this environment. One of the major interests of simulation is to be able to test the size of the vehicle fleet. Also, if the flow of tasks proposed to the AIVs becomes too great, leading to waiting times that are too long for the allocation of these tasks, the simulation conditions could be easily adapted in the environment with the addition of new AIVs and of AIVs parking lots.

The costs in distance between the different nodes of the circuit are represented in the directed graph of figure 4.b. They have been chosen and applied to favour certain directions of circulation. These costs are used to find the shortest paths in the graph (minimization of distance costs in this case), in order to optimize the times for performing tasks by AIV agents. The interested reader will find presentations and discussions on the different types of optimization algorithms that allow a mobile robot to determine a shortest path (Dijkstra, A\*, D\*, Genetic algorithm or Particle Swarm Optimization) in many surveys [14, 38, 90].

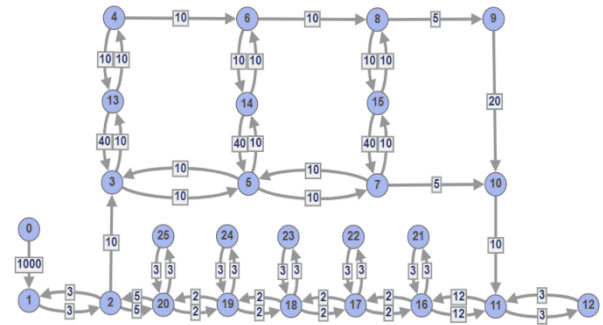


Figure 4. a) Representation of the circuit, and b) directed graph corresponding to the circuit

#### 4.2. Multi-agent simulation interface

The simulation interface presented in Figure 5 has been designed generically to integrate different types of traffic plans. We use and develop this simulator for various laboratory experiments, and for teaching engineering students. In order to facilitate community utilization, we are releasing the code of our project, on the Github page: [https://gitlab.inria.fr/jgrosset/AIV\\_Simulator](https://gitlab.inria.fr/jgrosset/AIV_Simulator).

Its implementation respects the agent model presented in Figure 3.

This interface is divided into five frames:

- Frame 1: visualization of the warehouse presented in figure 4.a. The white squares represent tag agents, used by AIV agents to locate themselves. AIV agents are visualized by coloured circles, and obstacles are surrounded by a red circle of variable size corresponding to their level of obstruction in the aisle. The four camera

agents of the infrastructure are identified by a black square evoking their viewing area of the aisle.

- Frame 2: application management and its various features. It is thus possible: 1) to simulate the four types of scenarios illustrated in the rest of the article, as well as a random scenario; 2) to generate obstacles randomly on the circuit; 3) to emulate a robot or camera failure; or 4) to view a model of the circuit with the node numbers, as in figure 4.b, by clicking on the circuit button.

- Frame 3: supervision of AIV agents. This frame makes it possible to visualize the missions assigned to the various AIV agents, their paths, their statuses and other information useful to the supervisor agent. When an AIV agent plays the role of auctioneer, he is graphically identifiable by a frame.

- Frame 4: supervision of camera agents. This frame makes it possible to identify their status, their position and their detection of obstacles. When a camera



agent has detected an obstacle, it is graphically identifiable by a black frame.

– Frame 5: task supervision. This frame makes it possible to monitor the progress of the performance of the tasks allocated to the AIVs. The states of the different task attributes are updated there: task identifiers, task starting point and ending point, and task states (attribute, in progress, blocked, completed).

The traffic plan chosen, and presented in the form of a directed graph in figure 4.b, makes it possible to start a set of problem scenarios that can be easily configured in the interface. In the following, we will focus on three scenarios called  $sc_1$ ,  $sc_2$  and  $sc_3$ . The missions will be named  $m_i$  and the nodes  $n_i$ . These different scenarios are detailed in the following subsection.



Figure 5. Multi-agent simulation interface

### 4.3. Dynamic task (re-)allocation scenarios

In [64], we proposed AIV agent blocking scenarios that highlighted the need to increase cooperation/communication between agents if we wanted to effectively manage these problems. In [39], we showed that the use of MCM, CAM and DENM messages made it possible to respond effectively to the problem of avoiding obstacles and collisions between AIV agents.

To verify the ability of AIV agents to carry out their missions while cooperatively managing the problems of obstacles or AIV agent breakdown, we have defined four new scenarios. These scenarios include five AIV agents, simulating five real AIVs of the same type, and illustrate the different types of V2X cooperation/communication that allow agents to anticipate problems and thus improve the performance of their missions collectively: V2I for the first scenario, V2V for the second, and V2P for the third. All information concerning the communication during these scenarios is resumed in the Table 10. The scenarios are run in the environment shown in Figure 4.a (a simplified warehouse). VIAs perform simple tasks: 1) load goods at a source storage point, then 2) drop them off at a destination storage point. Each AIV agent has knowledge of the environment, i.e. the position of aisles, intersections, parking lots, storage points, battery

replacement points and active elements of the infrastructure such as camera agents.

The first scenario  $sc_1$  is represented in figures 6.a and 6.b. It makes it possible to test the contribution of the cooperation between the AIV agents and the infrastructure agents for the performance of the tasks entrusted to the AIV agents. In this scenario, the camera agent placed to monitor the area around storage point n°14 detects an obstacle and send a CPM message to the five AIV agents. The AIV agents, whose mission involves passing through the aisle obstructed by the obstacle, are able to re-plan their rout in advance. Thus, this cooperation with the infrastructure makes it possible to avoid waiting for an AIV agent to detect the obstacle with its LIDAR and to warn the four other AIV agents when it passes near the obstacle (therefore saving time on detection). This scenario allows measuring the performance of a collective strategy including the infrastructure compared to a collective approach based only on V2V communications between AIV agents.

The  $sc_2$  scenario corresponds to an inability for an AIV agent to complete his mission. This one can be blocked by obstacles or have a breakdown but without this preventing it from communicating (figures 7.a and 7.b). In this case, it is the dynamic task allocation mechanism presented in section 3, which is launched to reallocate the unfinished mission. The blocked AIV agent



becomes an auctioneer. He transmits all the tasks he had to perform to the four other AIV agents using a CTM message. The AIV agents bid according to their situation and the tasks they are performing, which allows the AIV auctioneer agent to make his choice for the reallocation of the tasks he cannot complete.

Scenario *sc3*, depicted in Figure 8.a, illustrates the ability of an AIV agent to handle the blocking problem when a task cannot be completed due to an event occurring on the warehouse circuit or in the defined environment to perform the task. It is further assumed that this blocking could not be detected by an infrastructure agent and therefore that its resolution was carried out by an AIV agent. For example, if a stock point designated as a target in the mission of an AIV<sub>i</sub> agent is inaccessible (for instance, because of the presence of several obstacles), then the AIV<sub>i</sub> agent must be able to inform the four other AIV agents that its mission cannot be carried out, using a CPM message. Subsequently, a human or an AIV<sub>j</sub> agent having the ability to clear the obstacles in the aisle can intervene in response to the request made to it by the AIV<sub>i</sub> agent, by sending a DENM message (Figure 8.b). The task that could not be

performed before the human intervention is put back to auction as soon as the human has informed the AIV<sub>i</sub> agent that the aisle is clear again, by sending a CPM message. The AIV<sub>i</sub> agent then temporarily becomes an auctioneer to manage the reallocation of the task. This prevents the AIV<sub>i</sub> agent from waiting for human intervention to be able to continue its mission, and can possibly complete another tasks.

Scenario *sc4* presents a situation similar to Scenario *sc3*, where an AIV agent encounters what seems to be an obstacle, as detected by a camera. However, in this instance, the camera's assessment is faulty, and there is no actual obstruction present. Upon detecting the apparent obstacle, the camera notifies both the five AIV agents and the worker. However, upon closer inspection, it is revealed that the obstacle does not exist. Subsequently, the worker promptly sends two messages to the five AIV agents: a CPM message confirming the absence of an obstacle at the specified location, and a DENM message alerting about the failure of the camera agent.

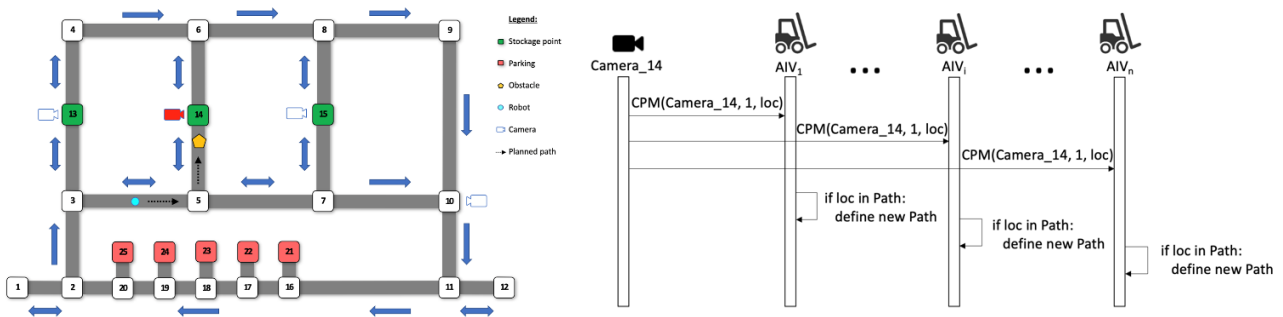


Figure 6. a) Simulation of the detection of an obstacle by a camera agent, b) sequence diagram of the scenario *sc1*

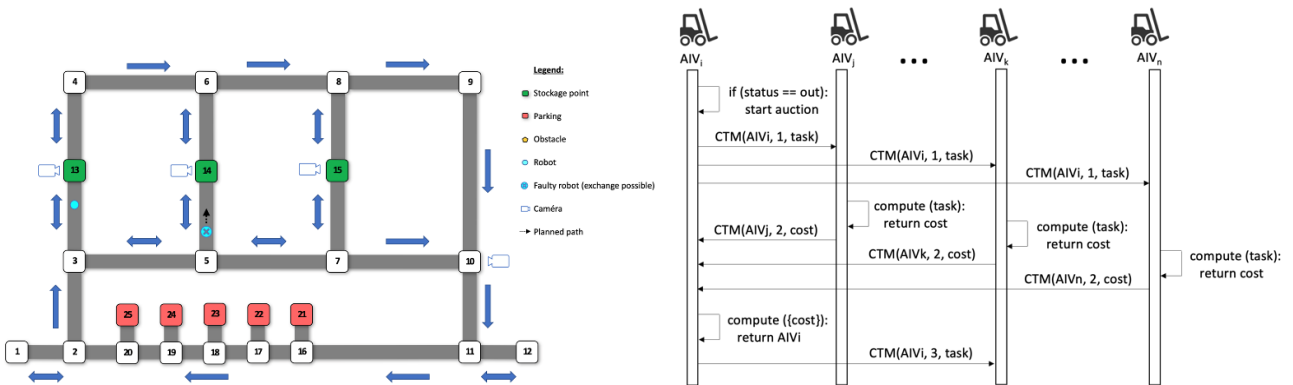


Figure 7. a) Simulation of an AIV agent breakdown at access point n°14, b) sequence diagram of the scenario *sc2*

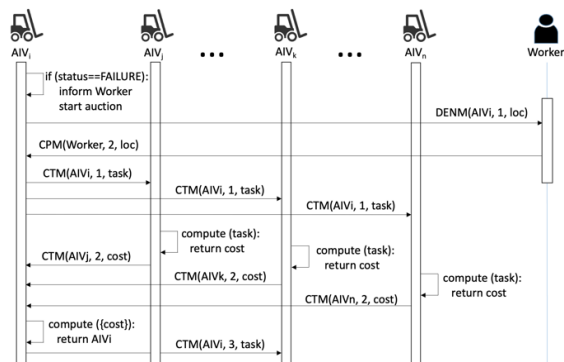
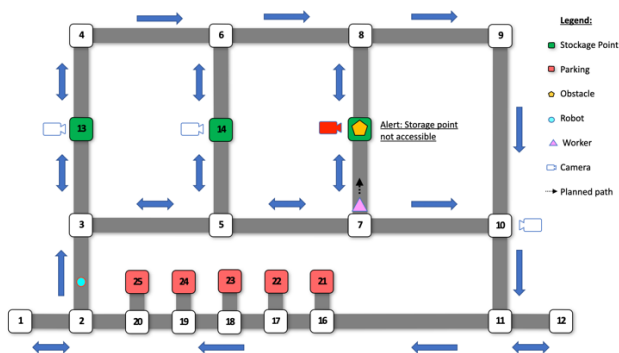


Figure 8. a) Simulation of a blocked task at access point n°15, b) sequence diagram of the scenario *sc3*

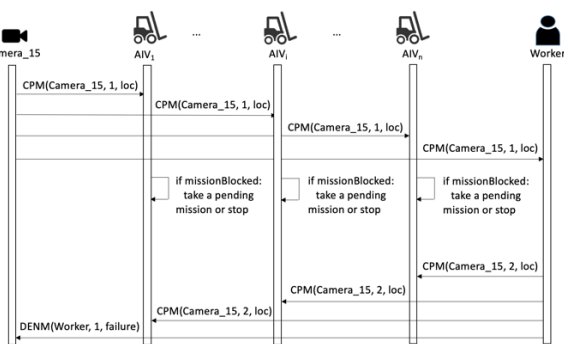
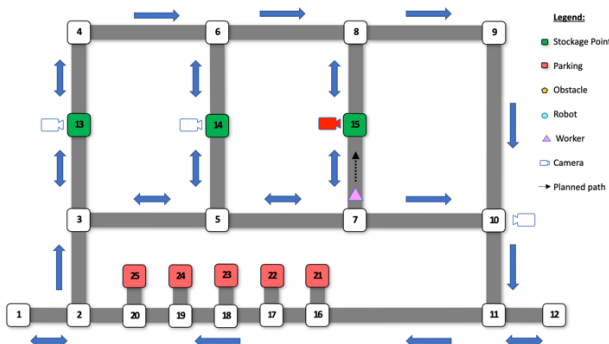


Figure 9. Simulation of a worker checking for an obstacle at storage point 15 detected by the camera, b) sequence diagram of the scenario *sc4*

## 5. Results and discussion

The four scenarios presented in the previous section were tested with the same dataset. The different types of V2X communications illustrated in the scenarios are summarized in the Table 9. The choice therefore fell on an allocation of tasks by packet, rather than continuously. A supervisor agent sends 10 tasks to an available AIV<sub>i</sub> agent (when the AIV agents have no more missions to perform, they inform the supervisor agent). The AIV<sub>i</sub> agent starts by clustering the tasks in missions, and then offers them up for auction.

The four scenarios were analysed with the following performance indicators:

- tasks to be performed,
- tasks fully completed,
- difference between the number of tasks to be performed and the number of tasks fully performed by each AIV agent,
- total distance covered by each AIV agent,
- load distribution (in particular to distribute the energy costs between the different vehicles and facilitate the management of battery charging).

Table 1 corresponds to the performance of all the tasks by the AIV agents, without them encountering any problem. This table includes the performance indicators listed above: the tasks requested then allocated to each AIV agent, the tasks actually performed by the AIV agents, the ratio between the tasks allocated and

performed, as well as the total distance covered by each AIV agent (distance in meters in this case).

Agent	Requested and allocated tasks	Completed tasks	Ratio	Distance covered
AIV1	<21,14,25>	<21,14,25>	2/2	124
AIV2	<22,13,22>	<22,13,22>	2/2	116
AIV3	<23,15,21>	<23,15,21>	2/2	112
AIV4	<24,14,23>	<24,14,23>	2/2	114
AIV5	<25,13,24>	<25,13,24>	2/2	114
Global / Supervisor	[[23,15],[25,13]] [[22,13],[24,14]] [[21,14],[15,21]] [[13,24],[13,22]] [[14,23],[14,25]]	<21,14,25> <22,13,22> <23,15,21> <24,14,23> <25,13,24>	10/10	580

Table 1. Test sets for the four scenarios

The results obtained during the execution of the scenario *sc1* corresponding to figure 6.a are given in table 3. The obstacle was detected by the camera, which saves time on obstacle detection. Indeed, we tested this same scenario by disabling the camera agent. It was necessary to wait for the AIV4 agent to arrive near the obstacle for it to be detected by its LIDAR. These results appear in Table 2. Thus, the total distance covered is 640 in the case where the camera agent detects the obstacle (Table3), and 710 if the camera agent is deactivated (Table 2). This makes it possible to verify that cooperation with the

infrastructure via camera agents can save time for the detection of obstacles, in particular by anticipating problems, and thus minimize distances for the performance of the missions of AIV agents.

The scenario *sc2* was simulated in two cases:

- the delegation of a complete mission (two clustered tasks),
- the delegation of the second part of a mission (only one task).

These delegations of mission by an AIV agent can occur when the latter is unable to perform the mission in progress, following a breakdown or a blockage in an aisle for example. Thus, in tables 4 and 5, it is possible to observe that the AIV2 agent could not finalize its mission because its number of tasks performed is not equal to its number of tasks to be carried out. In table 5, the AIV2 agent was able to perform one task out of two of its mission. He then started the task reallocation process, which resulted in the second uncompleted task being auctioned off. This task was won and performed by the AIV1 agent. The latter therefore perform three tasks, whereas two tasks had initially been assigned to him. The second test for task delegation corresponds to table 5 where it is possible to see that the entire mission of the AIV2 agent has been reallocated. In this case, it was the AIV1 agent who took over the complete mission, while minimizing the overall distance covered.

Agent	Requested and allocated tasks	Completed tasks	Ratio	Distance covered
AIV1	<21,14,25>	<21,14,25>	2/2	<b>164</b>
AIV2	<22,13,22>	<22,13,22>	2/2	116
AIV3	<23,15,21>	<23,15,21>	2/2	112
<b>AIV4</b>	<24,14,23>	<24,14,23>	2/2	<b>204</b>
AIV5	<25,13,24>	<25,13,24>	2/2	114
Global / Supervisor	[[23,15],[25,13]] [[22,13],[24,14]] [[21,14],[15,21]] [[13,24],[13,22]] [[14,23],[14,25]]	<21,14,25> <22,13,22> <23,15,21> <24,14,23> <25,13,24>	10/10	<b>710</b>

Table 2. Obstacle obstructing an aisle, detected by the AIV4 agent

Agent	Requested and allocated tasks	Completed tasks	Ratio	Distance covered
AIV1	<21,14,25>	[[21,14],[14,25]]	2/2	<b>144</b>
AIV2	<22,13,22>	[[22,13],[13,22]]	2/2	116
AIV3	<23,15,21>	[[23,15],[15,21]]	2/2	112
<b>AIV4</b>	<24,14,23>	[[24,14],[14,23]]	2/2	<b>154</b>
AIV5	<25,13,24>	[[25,13],[13,24]]	2/2	114
Global / Supervisor	[[23,15],[25,13]] [[22,13],[24,14]] [[21,14],[15,21]] [[13,24],[13,22]] [[14,23],[14,25]]	[[21,14],[14,25]] [[22,13],[13,22]] [[23,15],[15,21]] [[24,14],[14,23]] [[25,13],[13,24]]	10/10	<b>640</b>

Table 3. Obstacle obstructing an aisle, detected by a camera agent

Agent	Requested and allocated tasks	Completed tasks	Ratio	Distance covered
AIV1	<21,14,25>	<21,14,25>	2/2	124
<b>AIV2</b>	<22,13,22>		<b>0/2</b>	<b>24</b>
AIV3	<23,15,21>	<23,15,21> <22,13,22>	4/2	<b>236</b>
AIV4	<24,14,23>	<24,14,23>	2/2	114
AIV5	<25,13,24>	<25,13,24>	2/2	114
Global / Supervisor	[[23,15],[25,13]] [[22,13],[24,14]] [[21,14],[15,21]] [[13,24],[13,22]] [[14,23],[14,25]]	<21,14,25> <23,15,21> <22,13,22> <24,14,23> <25,13,24>	10/10	<b>612</b>

Table 4. Breakdown during part 1 of the AIV2 mission

Agent	Requested and allocated tasks	Completed tasks	Ratio	Distance covered
AIV1	<21,14,25>	<21,14,25> [13,22]	3/2	<b>234</b>
<b>AIV2</b>	<22,13,22>	[22,13]	1/2	<b>44</b>
AIV3	<23,15,21>	<23,15,21>	2/2	112
AIV4	<24,14,23>	<24,14,23>	2/2	114
AIV5	<25,13,24>	<25,13,24>	2/2	114
Global / Supervisor	[[23,15],[25,13]] [[22,13],[24,14]] [[21,14],[15,21]] [[13,24],[13,22]] [[14,23],[14,25]]	<21,14,25> [[13,22],[22,13]] <23,15,21> <24,14,23> <25,13,24>	10/10	<b>618</b>

Table 5. Breakdown during part 2 of the AIV2 mission

The results of the simulated dataset with a problem accessing a stock point appear in Table 6. They correspond to the simulation of the *sc3* scenario with the blocking of stock point n°15 identified in Figure 8. We can notice that the AIV3 agent who had two tasks related to the deposit or the retrieval of stock at stock point n°15 could not perform his tasks. Only eight of the ten tasks provided by the supervisor agent could be performed in this scenario. It is therefore necessary in this case, that an AIV agent or a human can come and unblock the situation (Figure 8.b).

The modified scenario, with human intervention and all tasks completed, is named *sc3'*. It is defined in Figure 8.b and the simulation results are presented in Table 7. Furthermore, the supervisor agent can also be informed so that it does not request the performance of other tasks related to this storage point as long as it is not accessible.

Finally, the results of the scenario *sc4* presented in Figure 9 appear in Table 8. The results obtained are the same as those in Table 7. The time differences compared to the scenario *sc3* occur if there are tasks continuous tasks. Because the AIVs will continue to pass through this area, and they will not have to wait for clearance by a worker or another robot related to an obstacle, as in scenario *sc3*. This scenario *sc4* demonstrates qualitative gains and robustness in processing. It highlights the importance of verifying information provided by the infrastructure because it can be faulty. This scenario underscores the importance of reliable infrastructure and effective communication channels among human

operators, AIVs, and surveillance systems to address inaccuracies and ensure operational efficiency.

Agent	Requested and allocated tasks	Completed tasks	Ratio	Distance covered
AIV1	<21,14,25>	<21,14,25>	2/2	124
AIV2	<22,13,22>	<22,13,22>	2/2	116
AIV3	<23,15,21>		0/2	22
AIV4	<24,14,23>	<24,14,23>	2/2	114
AIV5	<25,13,24>	<25,13,24>	2/2	114
Global / Supervisor	[[23,15],[25,13]] [[22,13],[24,14]] [[21,14],[15,21]] [[13,24],[13,22]] [[14,23],[14,25]]	<21,14,25> <22,13,22> <24,14,23> <25,13,24>	8/10	490

Table 6. Stock point n°15 has become inaccessible

Agent	Requested and allocated tasks	Completed tasks	Ratio	Distance covered
AIV1	<21,14,25>	<21,14,25>	2/2	124
AIV2	<22,13,22>	<22,13,22>	2/2	116
AIV3	<23,15,21>	<23,15,21>	2/2	112
AIV4	<24,14,23>	<24,14,23>	2/2	114

AIV5	<25,13,24>	<25,13,24>	2/2	114
Worker	<1,15,1>	<1,15,1>	2/2	116
Global / Supervisor	[[23,15],[25,13]] [[22,13],[24,14]] [[21,14],[15,21]] [[13,24],[13,22]] [[14,23],[14,25]]	<21,14,25> <22,13,22> <23,15,21> <24,14,23> <25,13,24>	10/10	580 AIVs + 116 Worker

Table 7. The stock point n°15 has become inaccessible, then cleared by a worker (human operator)

Agent	Requested and allocated tasks	Completed tasks	Ratio	Distance covered
AIV1	<21,14,25>	<21,14,25>	2/2	124
AIV2	<22,13,22>	<22,13,22>	2/2	116
AIV3	<23,15,21>	<23,15,21>	2/2	112
AIV4	<24,14,23>	<24,14,23>	2/2	114
AIV5	<25,13,24>	<25,13,24>	2/2	114
Worker	<1,15,1>	<1,15,1>	2/2	116
Global / Supervisor	[[23,15],[25,13]] [[22,13],[24,14]] [[21,14],[15,21]] [[13,24],[13,22]] [[14,23],[14,25]]	<21,14,25> <22,13,22> <23,15,21> <24,14,23> <25,13,24>	10/10	580 AIVs + 116 Worker

Table 8. A worker checks the presence of camera-detected obstacle

Scenario	Problem / Solution	Descriptive Figure / Result Table	Communication
<b>Sc0</b> Nominal case	AIVs perform all tasks (mission, allocation, path planning) without encountering any problems.	<b>Figure 4:</b> Representation of the circuit <b>Table 1:</b> Test sets for the 3 scenarios	<b>I2V:</b> Supervisor gives tasks to auctioneer AIV <b>V2V:</b> CTM and CRM messages for the task allocation (auctions, reception and feedback)
<b>Sc1</b> Obstructed aisle	If an obstacle obstructs an aisle, then an AIV that has to cross this aisle must quickly replan its path. <i>Cooperation AIV ↔ infrastructure</i>	<b>Figure 6:</b> Simulation of the detection of an obstacle by a camera agent <b>Table 2:</b> Obstacle detected by an AIV <b>Table 3:</b> Obstacle detected by a camera	<b>I2V:</b> - Supervisor gives tasks to auctioneer AIV - Camera send a CPM "Obstacle detected" <b>V2V:</b> CTM and CRM messages for the task allocation (auctions, reception and feedback)
<b>Sc2</b> AIV breakdown	An inability for an AIV to complete its mission (obstacles or breakdown). <i>Cooperation between the AIVs (for sharing missions).</i>	<b>Figure 7:</b> Simulation of an AIV breakdown at access point n°14 <i>Breakdown of an AIV mission during part 1 (Table 4) or part 2 (Table 5)</i>	<b>I2V:</b> Supervisor gives tasks to auctioneer AIV <b>V2V:</b> - CTM and CRM messages for the task allocation (auctions, reception and feedback) - CTM messages to share missions
<b>Sc3</b> Inaccessible stock point	An inability for an AIV to complete its mission due to an event occurring in the defined environment to perform the task (stock point is inaccessible). <i>Cooperation AIV ↔ Worker (to remove an obstacle).</i>	<b>Figure 8:</b> Simulation of the access point n°15 blocked <b>Table 6:</b> Stock point n°15 has become inaccessible <b>Table 7:</b> Stock point n°15 has become inaccessible, and then cleared by a worker.	<b>I2V:</b> - Supervisor gives tasks to auctioneer AIV - Camera send CPM: obstacle detected <b>I2P:</b> CPM message for an obstacle at stock point <b>V2V:</b> - CTM and CRM messages for the task allocation (auctions, reception, and feedback) - CTM messages to share missions <b>V2P:</b> DENM messages for a blocking problem <b>P2V:</b> CPM message indicating no more obstacle
<b>Sc4</b> Camera failure	If a camera sends false information, this information has to be checked. <i>Cooperation AIV ↔ Worker or between the AIVs (to verify information sent by the camera)</i>	<b>Figure 9:</b> Simulation of a worker checking for an obstacle at stock point n°15 <b>Table 8:</b> A worker checks the presence of camera-detected obstacle	<b>I2V:</b> - Supervisor gives tasks to auctioneer AIV - Camera send CPM: obstacle detected <b>I2P:</b> CPM message for an obstacle at stock point <b>V2V:</b> - CTM and CRM messages for the task allocation (auctions, reception, and feedback) - CTM messages to share missions <b>V2P:</b> DENM messages for a blocking problem <b>P2V:</b> - CPM message indicating no more obstacle - DENM message for the camera failure

Table 9. The various V2X communications in the different scenarios proposed

We indicated in section §4.1 that a certain number of properties can be associated with the concept of agent: situated, social, flexible, proactive, robust, mobile, intelligent, rational, temporally continuous, coordinative,

cooperative, competitive, rugged (able to deal with errors and incomplete data robustly). The four scenarios presented in this article, as well as the one proposed in a previous article [39], which we will call *Sc0*, make it

possible to verify the relevance of agent-based simulation. Indeed, all of the above properties are addressed during the realization of these scenarios (Table 10), where the AIVs agents will:

- carry out their missions in an environment where they will be located;
- communicate with each other, with the infrastructure and with workers, to establish collective intelligence;
- pursue a common objective of carrying out all tasks by cooperating with each other, with the infrastructure or with a worker;

- re-plan their paths and missions, or reallocate tasks, if necessary;
- listen to other AIVs and active elements of infrastructure, and continue to act even if they are blocked;
- coordinate themselves by using an auction mechanism for the allocation of tasks;
- collectively check possibly incorrect information and communicate with a worker to resolve any problems;
- act even when having incomplete data when receiving information without the AIVs being able to verify it themselves.

Scenarios	Agent properties												
	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10	P11	P12	P13
Sc0	●	●	◐		◐	●	◐	●	●		●		
Sc1	●		●		●	●	◐	●	●		●		◐
Sc2	●	●	●	◐	◐		●	●	◐	●	●	●	◐
Sc3	●	●	●	●	◐		●	●	◐	●	●	●	◐
Sc4	●	●	◐	◐	●	●	●	●	●		●		●

Table 10. The symbols means agent properties are captured ● or partially captured ◐, and the properties are : **P1** – Situated; **P2** – Social; **P3** – Flexible; **P4** – Proactive; **P5** – Robust; **P6** – Mobile; **P7** – Intelligent; **P8** – Rational; **P9** – Temporally continuous; **P10** – Coordinative; **P11** – Cooperative; **P12** – Competitive; **P13** – Rugged.

## 6. Conclusion and perspectives

In the context of the current smart factory, mobile robots must become increasingly autonomous in order to perform their missions effectively, i.e. optimize their activity according to performance indicators such as distances covered, energy consumed, time for perform missions, availability, etc. Autonomy and decentralization are two excessively linked notions to the extent that an autonomous system operates and make decisions autonomously, and a system is decentralized if the decision, which are made, are not centrally controlled. Therefore, we proposed a dynamic task (re-)allocation process model for autonomous industrial vehicles, managing their activity in a decentralized context. We then developed a multi-agent application to be able to simulate this process and test it on different scenarios of problematic traffic situations. The proposed scenarios allow us to move towards strong cooperation between AIV agents, but also between AIV agents and infrastructure agents (cameras, tags, beacons, etc.). The V2X communication implemented to enable this cooperation is an essential element of our decentralized agent-based simulation approach. We have shown that it brings more flexibility and robustness in the management of problematic dynamic situations. We wish to accentuate these types of cooperation to increase the autonomy of the AIVs that we use in real experiments.

The different perspectives that emerge from our work are data fusion and shared memory of AIV agents. For example, how to merge data related to the detection of an obstacle by an AIV agent and by a camera agent at different times. We want also to work on ways to verify

the presence of an obstacle, for example by asking an AIV agent to go and verify the presence of it. A shared memory would allow AIV agents to have, for example, global information on task delegation requests, but also to map the environment. For this, we plan to suppress the CRM messages, and to choose to return to all AIV agents the mission assigned to them. These prospects for enhanced cooperation would make it possible to increase the autonomy and efficiency of autonomous industrial vehicles. Furthermore, we continue to develop the simulation platform to integrate fleets of heterogeneous robots, therefore with robots that will not be able to perform all the defined tasks.

## Acknowledgments

The authors would like to thank the Brittany region for funding the *VIASIC* project as part of the ARED-2021-2024 call for projects concerning the strategic innovation area: “The economy at the service of industry for intelligent production”.

## References

- [1] Lu Y. (2017). Industry 4.0: A survey on technologies, applications and open research issues. *Journal of Industrial Information Integration*, 6: 1–10.
- [2] Xu X., Lu Y., Vogel-Heuser B., Wang L. (2021). Industry 4.0 and Industry 5.0 – Inception, conception and perception. *Journal of manufacturing systems*, 61: 530-535.

- [3] Liao Y., Deschamps F., de Freitas Rocha Loures E., and Ramos L.F.P. (2017). Past, present and future of Industry 4.0 - a systematic literature review and research agenda proposal. *International Journal of Production Research*, 55: 3609–3629.
- [4] Pereira A.C. and Romero F. (2017). A review of the meanings and the implications of the Industry 4.0 concept. *Procedia Manufacturing*, 13: 1206-1214.
- [5] Benotsmane R., Dudás L., Kovács G. (2018). The Concept of Autonomous Systems in Industry 4.0. *Advanced Logistic Systems-Theory and Practice*, 12(1); 77-87.
- [6] Andreasson H, Bouguerra A, Cirillo M, Dimitrov DN, Driankov D, et al. (2015). Autonomous Transport Vehicles: Where We Are and What Is Missing. *IEEE Robotics & Automation Magazine*, 22(1): 64-75.
- [7] Wang J., Zhang L., Huang Y., Zhao J., and Bella F. (2020). Safety of autonomous vehicles. *Journal of advanced transportation*, 1-13.
- [8] Wiseman Y. (2021). Autonomous vehicles. In *Encyclopedia of Information Science and Technology*, Fifth Edition, IGI Global, pp. 1-11.
- [9] Peres RS, Jia X, Lee J, Sun K, Colombo AW, et al. (2020). Industrial artificial intelligence in industry 4.0-systematic review, challenges and outlook. *IEEE Access*, 8, 220121-220139.
- [10] Rosique F., Navarro P.J., Fernández C., and Padilla A. (2019). A systematic review of perception system and simulators for autonomous vehicles research. *Sensors*, 19 (3): 648.
- [11] Ma Y., Wang Z., Yang H., and Yang L. (2020). Artificial intelligence applications in the development of autonomous vehicles: A survey. *IEEE/CAA Journal of Automatica Sinica*, 7(2): 315-329.
- [12] Rubio F., Valero F., and Lopis-Albert C. (2019). A review of Mobile Robots: Concepts, Methods, Theoretical Framework, and Applications. *International Journal of Advanced Robotic Systems*, 16 (2): 1-22.
- [13] De Ryck M., Versteijhe M., Debrouwere F. (2020). Automated guided vehicle systems, state-of-the art control algorithms and techniques. *Journal of Manufacturing Systems*, 54: 152-173.
- [14] Zghair N.A.K., and Al-Araji A.S. (2021). A one decade survey of autonomous mobile robot systems. *International Journal of Electrical and Computer Engineering*, 11(6): 4891.
- [15] Djoko-Kouam M., Fougères A.-J. (2023). Towards a bijective Co-simulation model between physical and virtual environments, adapted to a platform for autonomous industrial vehicles. *Automation, Control and Intelligent Systems*, 11(2): 27-44.
- [16] García E., Villar J.R., Tan Q., Sedano J., and Chira C. (2023). An efficient Multi-Robot Path Planning solution using A\* and coevolutionary algorithms. *Integrated Computer-Aided Engineering*, 30(1): 41-52.
- [17] Rafai A.N.A., Adzhar N. and Jaini N.I. (2022). A review on path planning and obstacle avoidance algorithms for Autonomous mobile robots. *Journal of Robotics*, vol. 2022, Article ID 2538220, 14 pages.
- [18] Aizat M., Azmin A., and Rahiman W. (2023). A Survey on Navigation Approaches for Automated Guided Vehicle Robots in Dynamic Surrounding. *IEEE Access*, 11: 33934-33955.
- [19] Raj R. and Kos A., (2022). A Comprehensive Study of Mobile Robot: History, Developments, Applications, and Future Research Perspectives. *Applied Sciences*, 12, 6951.
- [20] Hewawasam H. S., Ibrahim Y., and Appuhamillage G. K. (2022). Past, Present and Future of Path-Planning Algorithms for Mobile Robot Navigation in Dynamic Environments. *Industrial Electronics Society*, 3: 353 – 365.
- [21] Ye X., Deng Z., Shi Y., and Shen W. (2023). Toward Energy-Efficient Routing of Multiple AGVs with Multi-Agent Reinforcement Learning. *Sensors*, 23(12), 5615.
- [22] Cardarelli E., Digani V., Sabattini L., Secchi C., and Fantuzzi C. (2017). Cooperative cloud robotics architecture for the coordination of multi-AGV systems in industrial warehouses. *Mechatronics*, 45: 1-13.
- [23] de Paula Ferreira W., Armellini F., and De Santa-Eulalia L. A. (2020). Simulation in industry 4.0: A state-of-the-art review. *Computers & Industrial Engineering*, 149, 106868.
- [24] Tavcar J., and Horvath I. (2019). A review of the principles of designing smart cyber-physical systems for run-time adaptation: Learned lessons and open issues. *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 49: 145–158.
- [25] Jia X, Meng M.Q. (2013). A survey and analysis of task allocation algorithms in multi-robot systems. *IEEE int. conf. on robotics and biomimetics, ROBIO 2013*, pp. 2280–2305.
- [26] Khamis A, Hussein A, Elmogy A. (2015). Multi-robot task allocation: a review of the state-of-the-art. *Coop Robots Sensor Netw*, 2: 31–51.
- [27] Macal C.M. (2016). Everything you need to know about agent-based modelling and simulation. *J. Simul*, 10: 144–156.
- [28] Kamdar R., Paliwal P., and Kumar Y. (2018). A state of art review on various aspects of multi-agent system. *Journal of Circuits, Systems and Computers*, 27(11), 1830006.
- [29] Jing P., Hu H., Zhan F., Chen Y., Shi Y. (2020). Agent-based simulation of autonomous vehicles: A systematic literature review. *IEEE Access*, 8: 79089-79103.
- [30] Abar S., Theodoropoulos G.K., Lemarini P., and O'Hare G.M. (2017). Agent Based Modelling and Simulation tools: A review of the state-of-art software. *Computer Science Review*, 24: 13-33.
- [31] Vis I.F.A. (2006). Survey of research in the design and control of automated guided vehicle systems. *Eur J Oper Res*, 170(3): 677–709.
- [32] Draganjac I, Miklic D, Kovacic Z, Vasiljevic G, Bogdan S. (2016). Decentralized control of multi-AGV systems in autonomous warehousing applications. *IEEE Trans Autom Sci Eng*, 13(4): 1433–47.
- [33] Mes M. and Gerrits B. (2019). Multi-agent systems. In H. Zijm et al. (eds), *Operations, Logistics and*

*Supply Chain Management*, Lecture Notes in Logistics, Springer Nature.

- [34] Du Y., Chen J., Zhao C., Liao F. and Zhu M. (2023). A hierarchical framework for improving ride comfort of autonomous vehicles by deep reinforcement learning with external knowledge. *Computer-Aided Civil and Infrastructure Engineering*, 38(8): 1059-1078.
- [35] Li D., Wu J., Zhu F., Chen T. and Wong Y.D. (2023). Modeling adaptive platoon and reservation-based intersection control for connected and autonomous vehicles employing deep reinforcement learning. *Computer-Aided Civil and Infrastructure Engineering*, 38(10): 1346-1364.
- [36] Shi H., Zhou Y., Wang X., Fu S., Gong S. and Ran B. (2022). A Deep Reinforcement Learning based Distributed Connected Automated Vehicle Control under Communication Failure. *Computer-Aided Civil and Infrastructure Engineering*, 37(15): 2033-2051.
- [37] Gerkey B. P. and Mataric M. J. (2004). A formal analysis and taxonomy of task allocation in multi-robot systems. *The International journal of robotics research*, 23(9): 939-954.
- [38] Karur K., Sharma N., Dharmatti C., and Siegel J.E. (2021). A survey of path planning algorithms for mobile robots. *Vehicles*, 3: 448-468.
- [39] Grosset J., Ndao A., Fougères A.-J., Djoko-Kouam M., Couturier C., and Bonnin J.-M. (2023). A cooperative approach to avoiding obstacles and collisions between autonomous industrial vehicles in a simulation platform. *Integrated Computer-Aided Engineering*, 30 (1), pp.19-40.
- [40] Yan Z., Jouandeau N., and Cherif A.A. (2013). A survey and analysis of multi-robot coordination. *International Journal of Advanced Robotic Systems*, 10(12): 399.
- [41] Stern R., Sturtevant N., Felner A., Koenig S., Ma H., Walker T., ... and Boyarski E. (2019). Multi-agent pathfinding: Definitions, variants, and benchmarks. *In Proceedings of the International Symposium on Combinatorial Search*, 10(1): 151-158.
- [42] Wooldridge M. (2009). *An introduction to multiagent systems*. John Wiley & Sons.
- [43] Dias M.B., Zlot R., Kalra N., and Stentz A. (2006). Market-based multirobot coordination: a survey and analysis. *Proceedings of the IEEE*, 94(7): 1257-1270.
- [44] Daoud A., Balbo F., Gianessi P., and Picard G. (2021). ORNInA: A decentralized, auction-based multi-agent coordination in odt systems. *AI Communications*, 34(1): 37-53.
- [45] Hussein A. and Khamis A. (2013). Market-based Approach to Multi-robot Task Allocation. *IEEE Int. Conf. on Individual and Collective Behaviors in Robotics*, pp. 69-74.
- [46] Mariani S, Cabri G, and Zambonelli F. (2021). Coordination of autonomous vehicles: taxonomy and survey. *ACM Computing Surveys*, 54(1): 1-33.
- [47] Zeng Y., Wang M., and Rajan R.T. (2022). Decentralized Coordination for Truck Platooning. *Computer-Aided Civil and Infrastructure Engineering*, 37(15): 1997-2015.
- [48] Ding C., Dai R., Fan Y., Zhang Z., Wu X. (2022). Collaborative Control of Traffic Signal and Variable Guiding Lane for Isolated Intersection under Connected and Automated Vehicle Environment. *Computer-Aided Civil and Infrastructure Engineering*, 37(15): 2052-2069.
- [49] Zhou A., Peeta S., and Wang J. (2023). Cooperative Control of a Platoon of Connected Autonomous Vehicles and Unconnected Human-driven Vehicles. *Computer-Aided Civil and Infrastructure Engineering*, 38(18): 2513-2536.
- [50] Yi R., Zhou Y., Ou J., Wang X., Ding F., and Nie Q. (2023). A 2D connected automated vehicle car-following control algorithm. *Computer-Aided Civil and Infrastructure Engineering*, 38(18): 2560-2575.
- [51] Mohebifard R. and Hajbabaie A. (2022). Trajectory Control in Roundabouts with a Mixed-fleet of Automated and Human-driven Vehicles. *Computer-Aided Civil and Infrastructure Engineering*, 37(15): 1959-1977.
- [52] Li L., Wen D., and Yao D.Y. (2014). A survey of traffic control with vehicular communication. *IEEE Trans. Intell. Transp. Syst.*, 15(1): 425-432.
- [53] Ma C., Yu C., Zhang C. and Yang X. (2023). Signal Timing at an Isolated Intersection under Mixed Traffic Environment with Self-organizing Connected and Automated Vehicles. *Computer-Aided Civil and Infrastructure Engineering*, 38(14): 1955-1972.
- [54] Li Q., Li X., Huang Z., Halkias J., McHale G. and James R. (2022). Simulation of Mixed Traffic with Cooperative Lane Changes. *Computer-Aided Civil and Infrastructure Engineering*. 37(15): 1978-1996.
- [55] Zhang Z., Guo M., Fu D., Mo L. and Zhang S. (2022). Traffic Signal Optimization for Partially Observable Traffic System and Low Penetration Rate of Connected Vehicles. *Computer-Aided Civil and Infrastructure Engineering*, 37(15): 2070-2092.
- [56] Youssef A., El Khoreby M., Issa H., Abdellatif A. (2022). Brief Survey on Industry 4.0 Warehouse Management Systems. *International Review on Modelling and Simulations*, 15(5): 340-350.
- [57] Namazi E, Li J, Lu C. (2019). Intelligent intersection management systems considering autonomous vehicles: A systematic literature review. *IEEE Access*, 7: 91946-91965.
- [58] Rios-Torres J, Malikopoulos A. (2016). A survey on the coordination of connected and automated vehicles at intersections and merging at highway on-ramps. *IEEE Transactions on Intelligent Transportation Systems*, 18(5): 1066-1077.
- [59] Lin T., Rivano H., and Le Mouél F. (2017). A survey of smart parking solutions. *IEEE Transactions on Intelligent Transportation Systems*, 18(12): 3229-3253.
- [60] Muzahid A.J.M., Kamarulzaman S.F., Rahman M.A. et al. (2023). Multiple vehicle cooperation and collision avoidance in automated vehicles: survey and an AI-enabled conceptual framework. *Scientific Reports*, 13(1): 603.
- [61] Dąbrowska A., Giel R., and Werbińska-Wojciechowska S. (2021). Human safety in autonomous transport systems—review and case study. *Journal of KONBiN*, 51(1): 57-71.
- [62] He S., Li J., and Qiu T. Z. (2017). Vehicle-to-pedestrian communication modeling and collision



- avoiding method in connected vehicle environment. *Transportation Research Record*, 2621(1): 21-30.
- [63] Kokuti A., Hussein A., Marin-Plaza P., de La Escalera A., and Garcia F. (2017). V2X communications architecture for off-road autonomous vehicles. In 2017 IEEE Int. Conf. on Vehicular Electronics and Safety (ICVES), pp. 69-74.
- [64] Grosset J., Fougères A.-J., Djoko-Kouam M., and Bonnin J.-M. (2022). Collective obstacle avoidance strategy - an agent-based simulation approach. Int. Conf. on Adv. in Signal Processing and Artificial Intelligence (ASPAI'2022), Corfu, Greece, October.
- [65] ETSI, T. (2014). [DENM] ETSI EN 302 637-3 V1.2.1. Intelligent Transport Systems (ITS); Vehicular Communications; Basic set of Applications; Part 3: Specification of Decentralized Environmental Notification Basic Service. *ETSI, Standard*.
- [66] ETSI, T. (2019). 103 562 V2.1.1; Intelligent Transport Systems (ITS). Vehicular Communications; Basic set of Applications; Analysis of the Collective Perception Service (CPS); Release 2. *Standard. European Telecommunications Standards Institute: Sophia Antipolis, France*.
- [67] Shi H., Nie Q., Fu S., Wang X., Zhou Y. and Ran B. (2022). A Distributed Deep Reinforcement Learning Based Integrated Dynamic Bus Control System in a Connected Environment. *Computer-Aided Civil and Infrastructure Engineering*, 37(15): 2033-2032.
- [68] Häfner B., Jiru J., Roscher K., Ott J., Schmitt G.A., and Sevilimis Y. (2020). CVIP: A Protocol for Complex Interactions Among Connected Vehicles. In 2020 IEEE Intelligent Vehicles Symposium (IV), oct. 2020, p. 510-515.
- [69] Leitao P., Karnouskos S., Ribeiro L., Lee J., S0trasser T., and Colombo A.W. (2016). Smart agents in industrial cyber-physical systems. *Proceedings of the IEEE*, 104(5): 1086-1101.
- [70] Rodrigues N., Oliveira E., and Leitao P. (2018). Decentralized and on-the-fly agent-based service reconfiguration in manufacturing systems. *Computers in Industry*, 101: 81-90.
- [71] Leitão P., Mařík V., and Vrba P. (2012). Past, present, and future of industrial agent applications. *IEEE Transactions on Industrial Informatics*, 9(4): 2360-2372.
- [72] Monostori L., Vánca J., and Kumara S.R. (2006). Agent-based systems for manufacturing. *CIRP annals*, 55(2): 697-720.
- [73] Leusin M.E., Kück M., Frazzon E.M., Maldonado M.U., and Freitag M. (2018). Potential of a multi-agent system approach for production control in smart factories. *IFAC-PapersOnLine*, 51(11): 1459-1464.
- [74] Fougères A.-J., Ostrosi E., (2019). Holonic Fuzzy Agents for Integrated CAD Product and Adaptive Manufacturing Cell Formation. *Journal of Integrated Design and Process Science*, 23(1): 77-102.
- [75] Vieira A.A., Dias L.M., Santos M.Y., Pereira G.A., and Oliveira J.A. (2020). Supply chain data integration: A literature review. *Journal of Industrial Information Integration*, 19: 100161.
- [76] Dong R., Woodman R., Jennings P.A., Brewerton S., Birrell S.A., Higgins M.D. (2021). A simulation study of platooning AV fleet service in shared urban environments with uncertainties. *Transportation Engineering*, 4: 100062.
- [77] Vogel-Heuser B., Lee J., Leitão P. (2015). Agents enabling cyber-physical production systems. *at-Automatisierungstechnik*, 63(10): 777-789.
- [78] Odell J. (2007). Agent Technology: What is it and why do we care? *Enterprise Architecture*, 10(3): 1-25.
- [79] Fougères A.-J. (2011). Modelling and simulation of complex systems: an approach based on multi-level agents. *Int. J. of Computer Science Issues*, 8(6): 8-17.
- [80] Biswas P.K. (2008). Towards an agent-oriented approach to conceptualization. *Applied Soft Computing*, 8(1): 127-139.
- [81] Bergenti F., Gleizes M.-P., Zambonelli F. (2004). Methodologies and Software Engineering for Agent Systems. Kluwer.
- [82] Macal C.M., North M.J. (2010). Tutorial on agent-based modelling and simulation. *Journal of Simulation*, 4: 151-162.
- [83] Dorri A., Kanhere S.S., Jurdak R., (2018). Multi-Agent Systems: A survey. *IEEE Access*, 6, 28573-28593.
- [84] Antelmi A., Cordasco G., d'Ambrosio G., de Vinco D. (2022). Experimenting with agent-based model simulation tools. *Applied Sciences*, 13, 13.
- [85] Fougères A.-J. (2004). Agents to cooperate in distributed design process. In Proc. of IEEE Int. Conf. on Systems, Man and Cybernetics (SMC'04), The Hague, Netherlands, pp. 2629-2634.
- [86] Odell J., Parunak H.V.D., and Bauer B. (2000). Extending UML for agents. In Proc. of the AOIS Workshop at the 17th Nat. conf. on Artificial Intelligence, Austin, Texas.
- [87] Siegwart R., Nourbakhsh I. R., and Scaramuzza D. (2011). Introduction to Autonomous Mobile Robots. Cambridge, MA, USA: MIT Press.
- [88] Belbachir A., El Fallah-Seghrouchni A., Casals A., and Pasin M. (2019). Smart mobility using multi-agent system. *Procedia Computer Science*, 151: 447-454.
- [89] Bechtsis D., Tsolakis N., Vlachos D., Srai J.S. (2018). Intelligent autonomous vehicles in digital supply chains: a framework for integrating innovations towards sustainable value networks. *Journal of cleaner production*, 181: 60-71.
- [90] Qin H., Shao S., Wang T., Yu X., Jiang Y., Cao Z. (2023). Review of Autonomous Path Planning Algorithms for Mobile Robots. *Drones*, 7: 211.