(Article begins on next page)

Sen3Bot Net: a meta-sensors network to enable smart factories implementation

Marina Indri, Fiorella Sibona, Pangcheng David Cen Cheng
Dipartimento di Elettronica e Telecomunicazioni
Politecnico di Torino
Corso Duca degli Abruzzi 24, 10129 Torino, Italy
{marina.indri, fiorella.sibona, pangcheng.cencheng}@polito.it

***Abstract*—In the near future, an increasing number of mobile agents working closely with human operators is envisaged in smart factories. In industrial human-shared environments that employ traditional Automated Guided Vehicles, safety can be ensured thanks to the support provided by Autonomous Mobile Robots, acting as a net of meta-sensors. The localization and perception information of each meta-sensor is shared among all mobile platforms. In particular, the information about the dynamic detection of human presence is combined and uploaded in a shared map, increasing the awareness of the mobile robots about their surroundings in a specific working area.**

This paper proposes an architecture that integrates the meta-sensors with an existing net of Automated Guided Vehicles, with the aim of enhancing systems based on outdated mobile agents that seek for Industry 4.0 solutions without the necessity of a complete renewal. Simulations of test scenarios are provided in order to confirm the validity of the proposed architecture model.

***Index Terms*—Mobile robots, human detection, obstacle avoidance, multi-sensor data fusion.**

I. Introduction and state of the art

Over the last few years, mobile robots have been used widely in industry, since they increase the efficiency and flexibility in the production line. They are commonly classified as Automated Guided Vehicles (AGVs) and Autonomous Mobile Robots (AMRs). AGVs are usually employed for transportation tasks following pre-defined paths, whose guidance system is generally based on wired or magnetic navigation. This kind of system has strong dependencies on the infrastructure and is not reactive to the dynamic changes within the environment. On the other hand, AMRs are equipped with a heterogeneous set of on-board sensors and an intelligent decision-making system that allow them to move autonomously in the working space, while performing specific tasks.

In a working space where mobile robots and human operators coexist, safety is an important factor and hard constraints imposed by international standards must be satisfied. This may be difficult for setting up the AGVs. Indeed, industrial manufacturing systems that employ only AGVs may need to find a trade-off between an ad-hoc smarter solution and a more traditional one having a wider application even if less efficient. This becomes necessary since traditional AGVs have technical limitations and, on the other hand, their enhancement is expensive [1]. In contrast, AMRs have a better equipment, which allows them to perform reactive tasks, such as collision avoidance of dynamic obstacles and the relative autonomous path re-planning. The introduction of AMRs in industrial production systems improves manufacturing performance in terms of productivity, flexibility and costs, even without re-designing the production lines [2].

A distributed multi-agent system should efficiently split and execute tasks. However, the methodology adopted for this kind of system is usually focused on the algorithms for a particular category, such as motion planning for delivery operations. Nevertheless, the Highway Code implementation proposed in [3] focuses on how to handle safety between robots and human operators. Simulations are highly recommended, since the experimental testing in real world environments would be dangerous and time consuming. In this way, it is possible to perform a list of risks assessments according to the available ISO standards for Human-Robot Collaboration (HRC) operations [4]. Alternative ways that may ensure safety within the industrial environment include installing fixed sensors around the working space, for monitoring the activities of the mobile agents and the human operators. These sensors create virtual barriers that may deactivate or slower down the robot if the human presence is detected. In [5], particular areas are classified by colors, which indicate the operations carried out by the robots and the relative potential dangerousness for the human operator. Nonetheless, having a fixed network of sensors hinders flexibility.

Cyber-Physical Systems (CPS) are emergent technologies that integrate functionalities for connecting operations in the real world with computing and communication infrastructures through a well-defined network [6]. In this context, sensor systems that share the relative information from the real world to other agents permit to define collaborative planning in the manufacturing process and therefore, ensure safety within the working space. When considering CPS, many researchers have developed multi-agent system algorithms that measure and share the relative position of each robot and of all the obstacles within the environment.

The integration of information coming from different sources can be achieved by applying sensor data fusion methods. The combination of data from different sensors allows to overcome the limitations of each sensor, including a limited field-of-view. For example, in [7], a control method that preserves the visibility among robots when they are equipped with limited field-of-view sensors (such as LIDAR, cameras and optical sensors) is proposed. The aim is to maintain multiple lines-of-sight formed by the robots while they are moving. The visibility is modeled using graphs and the edges of the line-of-sight of each robot within the network.

Furthermore, a sensor fusion method for cooperative trail-following tasks is proposed in [8]. Each robot can periodically exchange the visual data with other robots, so the decision making depends on its local view and the shared information from others. The authors used their own framework, SF-Cooper, to control a group of mobile robots and three feature fusion methods: SVM (Support-Vector Machine), SOFTMAX and four-layer DNNs (Deep Neural Networks). They combined the visual information coming from ground and aerial robots, and tested it in the real-world environment, successfully dealing with the "limited view" problem, typically found in single-robot systems. Similarly in [9], the use of an air-ground robot combined with ground robots is presented, since the Unmanned Aerial Vehicle (UAV) allows an easy global view. By aligning the data from the UAV and the ground robots camera frames it is possible to estimate the global pose of each ground robot. Nevertheless, the combination of ground and aerial robots cannot be adopted easily since UAVs are not suitable for any indoor environment.

Managing dense information coming from a sensor may lead to false positives. For this reason, there are researchers that combined different methods with Kalman Filters in order to obtain robust measurements. The multi-sensor fusion strategy proposed in [10] is able to detect and eliminate spurious data before it undergoes the fusion procedure. This is achieved by using a Fault Detection and Exclusion (FDE) method based on the Kullback-Leibler Divergence between *a priori* and *a posteriori* distribution of the Information Filter. The framework was applied to a multi-robot system moving within an indoor environment with the aim to improve the localization integrity of the overall system, also known as the Collaborative Localization (CL), in which each robot detects the others and computes relative observations. This method was upgraded later, as presented in [11], by achieving trajectory tracking using Sliding Mode Control (SMC). Moreover, the information update of the robots is time-based. However, this kind of approach requires many computational resources, while an event-based update would reduce the communications, processor and memory requirements [12].

In [13], a hybrid distributed and centralized cooperative fusion is proposed; in particular, the authors refer this methodology as the Edge Cloud Cooperative Localization (ECCL), which combines several distributed Kalman Filters in nodes edge and a centralized system that works as a fusion unit in the cloud server. In order to ensure a robust data fusion, the authors applied a localization validation method called the Cooperative Redundancy Validation (CRV), that takes into account all the available observations.

A distributed control architecture for multi-robot task allocation is presented in [14]. Distributed messages are used for the communication system between the robots. Each robot has a task tree, that allows the communication with its teammates, the identification of its own tasks and the ones performed by the other robots. In addition, the architecture allocates tasks in such a way that the robots work together to complete the operations, respecting all the motion constraints.

Most of the algorithms for cooperative localization of mobile agents have been designed with the idea of introducing new robots with better functionalities to substitute the old ones. However, many worldwide industries are still working with non-collaborative robots and a total replacement would require a huge investment [15]. In contrast to larger firms, a complete renewal for Small, Medium and Micro Businesses (SMMEs) for becoming smart factories may be a problem, due to the high cost and limited resources [16].

By exploiting concepts related to CPS, it is possible to integrate intelligent agents, e.g., AMRs, with the existing elements within the industrial infrastructure without the need of a total technology replacement, while still obtaining the advantages of Industry 4.0 solutions. Moreover, since the future scenario envisages robots and human operators working very closely in the same environment, there is the need for a shared acceptance of the robots as part of the process and feedbacks must be taken into account. Indeed, the probabilistic behaviour of an AMR leads to a skeptical attitude from workers, since they cannot predict the unexpected motions of the robot [17].

This paper provides behaviour details on the architecture introduced in [18]. There, the system was described at a very high conceptual level, leaving out the expected behaviour specifications and the functional details, which, in fact, are provided by this work. Moreover, the meta-sensor AMR module, there only hinted at, is here taken into account as a fully functional module, whose desired features have implemented and tested in [19]. The meta-sensors act as mobile sensor units with the aim of supporting an existing net of traditional AGVs. To achieve this, the relative localization information of each AMR along with the information about its surroundings are fused and shared to all the mobile agents. Particular attention will be devoted to the dynamic detection of human operators; when one or more AMRs detect the presence of human workers in a specific area, the relative position of the latter ones will be updated on the shared map. The AGVs coordination interface will process all the gathered information from the meta-sensors and send the proper commands to the AGVs depending on (i) the dangerousness of the area and (ii) the activities of the operator. In this way, safety is ensured for the overall system, as well as the compliance with collaborative tasks between human operators and robots. In particular, the aim of the authors is to provide a framework that enhances the coordination and decision-making interface of the AGVs with the measurements obtained from the meta-sensors.

The paper is organized as follows: Section II presents the proposed procedure, at first providing some details about the architecture, then describing the blocks that compose it. The simulations are unfolded in Section III, where different test scenarios are reported. Finally, Section IV draws some conclusions and identifies some open issues.

II. META-SENSORS ARCHITECTURE FOR AGV SUPPORT

This work provides a complete overview on the behaviour of the architecture whose specifications are presented in [18]. With the aim of contextualizing this description, some relevant features of the system are given hereafter. The architecture is imagined to be integrated in a flexible production line setting, where traditional AGVs, workstations, cobots, and human operators co-exist. Moreover, it has the following main components: (i) a meta-sensor AMR fleet, (ii) the Sensors Synergy Center (SSC) and (iii) the AGV Coordination Center Interface (AGV CCI).

It is necessary here to clarify exactly what is meant by *meta-sensor*: each AMR, equipped with a heterogeneous selection of sensors, becomes a sensor itself, with the specific function of facilitating the monitoring of industrial scenarios, as a support to traditional or semi-autonomous fixed-path AGVs. The meta-sensor AMRs must not be considered as an evolution of traditional AGVs, but as AGVs enhancers, to enable smart factories benefits. Throughout this paper, the term Sen3Bot (Sentry roBot) will be used to refer to the meta-sensor AMR. Note that these two terms are used interchangeably. The Sen3Bot component, whose implementation is described in [19], has the capability of detecting and identifying humans. This ability is implemented through the real-time object detection system YOLO, by applying its convolutional neural network to the video stream of a low-cost IP camera placed on the mobile robot. This vision information is integrated with the corresponding distance value through camera-laser data sensor fusion, allowing to correctly place the identified humans within the plant map, and to impose a more conservative behaviour specifically around them (higher inflation radius). The SSC element is in charge of performing sensor-fusion and map traffic updates, receiving the current poses of the AGVs by interfacing the existing AGV Coordination Center (AGV CC), to take decisions for the Sen3Bots task allocation and execution based on the AGVs currently pursued tasks. The AGV CCI then allows to convert the significant data gathered by the SSC (through the Sen3Bots) into proper commands that the AGV CC will use to suitably adjust the AGVs motion.

The purpose of this work is to describe the meta-sensors AMR system behaviour. Some implementation details of components (i) and (ii) have already been described in [19], but a more functional overview of both elements, whose behaviour is inevitably strictly correlated, is provided in this work.

The S3B Net (Sen3Bot Network) is a network of autonomous and interacting hybrid agents, i.e, intelligent robots that can autonomously perform actions on the basis of a planning algorithm while being able to sense and act when an environmental change occurs. In addition, each meta-sensor AMR localizes itself implementing the Adaptive Monte Carlo Localization (AMCL) algorithm, provided by the ROS Navigation Stack.

Based on a recognized design workflow pattern [20], we describe the system according to the following characterizing blocks.

A. Task decomposition

The S3B Net main role is to ensure a safe motion for each AGV of the pre-existent system. This task can be achieved by taking advantage of the added value provided by the so-called *meta-sensor fusion*, i.e., the integrated information gathered from the involved monitoring Sen3Bots. It should be noted that, apart from its main task, each Sen3Bot can be employed for moving tools or bringing materials to human operators depending on the availability. Indeed, when not busy with its sentry role, the Sen3Bot can take on traditional AMR tasks.

B. Coalition formation

The coalition formation in the S3B Net is defined a-priori and depends on the critical level of the different areas of the factory visited by the AGVs during their tasks. To better understand what is meant here by critical level, some reference to standard definitions are introduced to explain the concept.

According to ANSI/ITSDF B56.5 “Safety Standard for driverless, automatic guided industrial vehicles and automated functions of manned industrial vehicles” [21], we can identify several zones in an industrial environment. Hazardous or restricted areas are not taken into account here, since by standard these are clearly marked areas where personnel has no access. What is relevant for the working scenario are non-restricted areas, defined in ANSI/ITSDF B56.5-2019 as areas where the guide-path is installed and that are shared by personnel. Furthermore, when considering the Sen3Bot fleet, based on a first draft of the standard for AMR Industrial Mobile Robots ([22], R15.08 Drafting Subcommittee presentation from the Autonomous Mobile Robot Conference, September 2019), we can identify the following regions: (i) free space, where the IMR (Industrial Mobile Robot) can plan a path, (ii) keep-out zone, excluded from the free space, and (iii) monitored space, corresponding to the volume around the IMR where perception systems can monitor. Taking into consideration the co-existence of both AGVs and meta-sensor AMRs in the working scenario, the Sen3Bot monitored space must cover as much as possible the non-restricted area that the supported AGV is going to cross, in order to provide a real-time awareness of the situation *before* the AGV even reaches the location.

Moreover, since today’s smart factories and the ones of the very near future have enhanced manual stations and cobots workstation fully integrated within the automated lines [23], we can identify zones where the presence of human operators is highly probable and therefore we consider them to be areas of interest.

We can thus distinguish the following areas:

1. Critical areas.
   1.1 Non-restricted areas with limited visibility for the

approaching AGV (blind intersections).

1.2 Areas where human operators are likely to pass, like transition areas. This area type may include also particular areas that may fall somewhere in between the definitions of areas of type 1 and 2.

2. Sub-critical areas. These include cobots workstations and manual stations locations where human operators are likely to be present, but expected to be mostly static.
3. Non-critical areas. These comprehend restricted areas known to be human free or full visibility areas where human passage is rare. Note that safety is anyway guaranteed by safety-rated sensors on the AGV.

As it can be easily understood, our areas of interest are critical and sub-critical ones: as soon as an AGV is assigned its path, if the latter crosses areas of interest, we can have different approaches for the a-priori coalition formation, depending on the level of criticality of the first crossed area. If a critical area (1.1 or 1.2) is crossed, two Sen3Bots are assigned to monitor it. On the other hand, when a sub-critical area is foreseen to be crossed, only one Sen3Bot is sent to the scene, to scout the area. Moreover, the final number of Sen3Bots needed to monitor the scene (until the AGV will overcome the area) also depends on a new real-time information gathered by the meta-sensor AMRs, once they reach their respective monitoring poses: the detected human operators' speeds and directions. These additional data are used to decide whether the number of Sen3Bots sent to the scene is suitable for the scenario. Note that areas of type 1.1 are the most critical ones, since making up for the lack of visibility is crucial to avoid undesired collisions. For this reason, if the crossed area is of type 1.1, two Sen3Bots are required at all times, until the AGV leaves the area. Less strict policies are adopted when considering types 1.2 and 2 areas. Table I summarizes the architecture coalition formation policy. As it can be seen, the detected human behaviour influences the final number of employed Sen3Bots. Indeed, if the detected humans are quite static, the information to be sent to the AGV does not require redundancy. Instead, a dynamic situation necessitates robust data to be shared: in this case, a redundant information about the human operators is preferred.

In order to define how a *static* and a *dynamic* behaviour are distinguished in the Sen3Bot detection, it is worth recalling how human obstacles are represented in the shared map. As described in [19], human obstacles are enclosed in *virtual cages*, i.e., they are provided a greater safety radius value with respect to other obstacles. Futhermore, a virtual obstacle is published in correspondence of each detected human. The extension of this virtual obstacle depends on the human bounding box extension (detected at the image processing stage). This sort of virtual cage around the human operator remains integral with the operator movements, dynamically adjusting to the detected person bounding box. Moreover, the human obstacle is conservatively enclosed since the safety distance is maintained based on the left and right edges of the vision derived bounding box, thus guaranteeing that all mobile platforms are aware of the obstacle extension. The described behaviour is shown in Figure 1.

Fig. 1. The Sen3Bot human obstacle avoidance behaviour.

Hence, for the sake of simplicity, the human behaviour is classified depending on the following criteria: when the center of the human obstacle moves more than $1\ m$ from the first detected position, in either direction, the human operator is considered to be dynamic. Otherwise, it is considered to be static. Furthermore, each Sen3Bot (if more than one are monitoring the scene) will provide such data from different perspectives, thereby allowing to capture a more complete information about the human motion. Note that the threshold value may be adjusted depending on the area features and user needs.

Additionally, depending on the area criticality and the behaviour of the detected human operators, a *priority index p* is assigned to every task pursued by a Sen3Bot. In such a way, it is possible to classify tasks based on their priority, to allow for flexibility in the case that a Sen3Bot is required for a higher priority task. Suppose a Sen3Bot is pursuing a classical AMR task, e.g., moving material, and the system requires a Sen3Bot to assist the passage of an AGV in a critical zone of type 1.1: if the mentioned Sen3Bot is eligible for being assigned this task (further details about task allocation will be given in Section II-C), it will interrupt the pursued low priority task and move to the critical area, postponing its initial task. The maximum priority is given by $p = 0$ and increased when the priority of the task decreases. The flowchart in Figure 2 provides a schema of the overall behaviour. As shown, if human operators are detected, the AGV CCI gathers the information from the shared map and through the AGV CC (which we assume corresponds to the generic pre-existent classical AGV manager) slows down the AGV when it approaches the area of interest.

C. Task allocation and execution

As the system becomes aware that an AGV has been assigned a task, the suitable number of Sen3Bots (as specified in Section II-B) are sent to the scene in order to monitor the AGV passage. In our case, task allocation is based on two criteria: (i) distance of the Sen3Bot from the area requiring the monitoring and (ii) the Sen3Bot availability/priority of the task it is busy with.

Before describing how task allocation is managed, it is worthwhile to provide a general assumption about the initial

TABLE I
COALITION FORMATION IN THE SEN3BOT NET

Area Type	Area Type ID	A-priori # of Sen3Bots	Overall human behaviour	Final # of Sen3Bots
Critical Area	1.1	2	static dynamic	2 2
Critical Area	1.2	2	static dynamic	1 2
Sub-critical Area	2	1	static dynamic	1 2

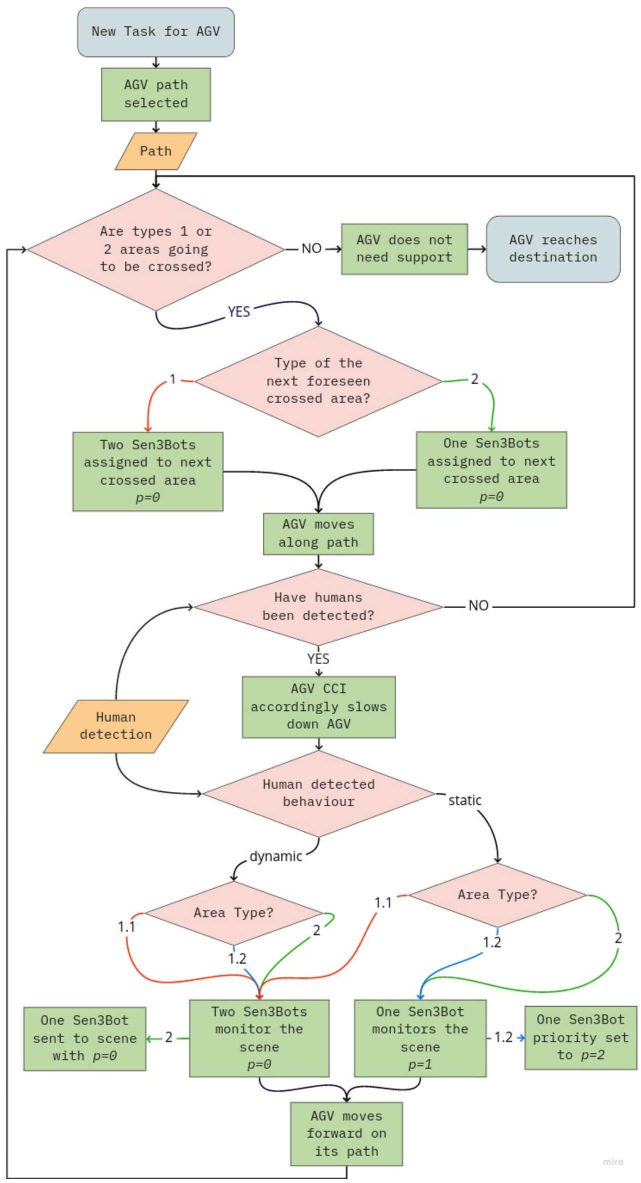

Fig. 2. Flowchart for the Sen3Bot Net coalition formation decision process.

pose of the Sen3Bots. We assume that the power charging stations for the Sen3Bots are positioned strategically, i.e., placed in such a way that the Sen3Bots monitor the most critical areas in the factory, while charging or while at *home* (in close proximity to the station). This enables the S3B Net to be responsive when areas requiring the most attention are involved. This allows to get rid of the limitations inevitably introduced by fixed sensors, enabling flexibility while ensuring safety. Note that if the Sen3Bot is not assigned any task, it goes back to its *home* pose and $p = 2$ is set (low priority); of course, $p = 0$ is set when the Sen3Bot is heading to its recharging station due to detected low battery.

The priority value p together with the distance of each Sen3Bot from the interest area are taken into account in order to identify the eligible Sen3Bots for standing sentry in the area. To facilitate a faster S3B Net response, the list of all Sen3Bots is sorted depending on how distant they are from the scene and then selected only if the task they have been assigned to is of priority 1 or 2 (i.e., medium or low priority). In the case the Sen3Bot has been assigned a $p = 1$ task, it is selected but can be replaced if a more distant Sen3Bot with $p = 2$ is found. The selection process iterates until the number N of required Sen3Bots is reached. Refer to Figure 3 for more details about the process. It should be noted that the total number of Sen3Bots available in the plant is mainly dependent on the number of areas of interest and on the client necessities; a good trade-off could be reached considering that a minimum coverage of all critical areas should be guaranteed.

Furthermore, the simulations reported in this work consider the case in which a central system manages the task allocation for a relatively small number of agents. The complexity of the overall system increases according to the number of mobile agents, so a distributed control strategy could be preferable in some cases. If the application scenario requires a very high number of mobile agents, the proposed architecture can be implemented as different distributed modules, each one acting in a particular area. However, in order to maintain synchronized the whole system, which includes other mechatronic systems within the smart factory, i.e., cobots, these modules still have to communicate with a centralized supervisor that assigns the tasks at a high level. This way, the performance of the entire manufacturing system can be improved through sequences of minimal corrective actions, as in [24], established by integrating the performance indicators of the production process with the capabilities and working functionalities of the robotic systems and agents.

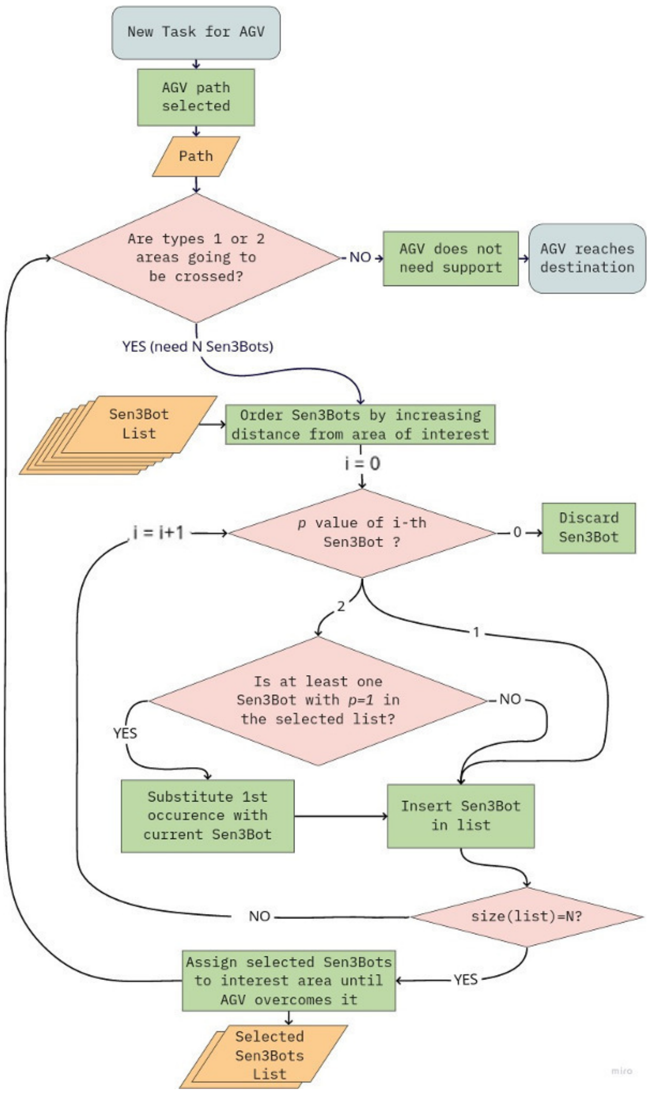

Fig. 3. Flowchart for the Sen3Bot Net task allocation decision process.

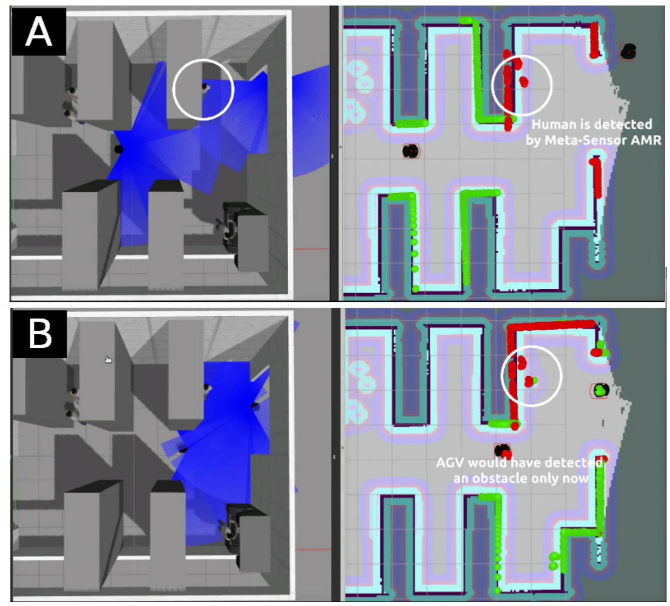

Fig. 4. Sen3Bot Net simulation: meta-sensor concept demonstration.

III. Simulation: Case Scenarios

In this section we report some simulation scenarios with the aim of demonstrating the S3B Net behaviour. The video featuring the considered demo cases is available online at [25].

The simulations have been performed using the Kinetic Kame distribution of ROS [26], with the corresponding compatible version of the Gazebo Simulator [27]. It must be noted that version 7 of Gazebo does not allow to spawn animated person models (actors) in the simulated environment. However, this does not hinder the simulation aim of demonstrating the architecture behaviour. The created Gazebo *world* represents a portion of a plant accessible to human operators (non-restricted area). Rows of racks and a workstation are present, to provide the minimum conditions for a demonstrative simulation. Figures 4, 5 and 6 show the top view for each simulated scenario: on the left the simulated industrial workspace, on the right the *rviz* visualization, where laser data of all mobile robots, along with the cost maps are visible.

The first reported scenario is depicted in Figure 4: the left mobile platform represents a traditional AGV (green laser points), while the right one simulates a meta-sensor agent (red laser points). This test focuses on the core role of the meta-sensor AMR concept, that is integrating the AGV knowledge about the path it is going to travel along before reaching the area of interest. In particular, as seen in Figure 4A, the AGV is aware of the human presence, even if it is out of its scope, thanks to the SSC processed information coming from the S3B Net. Should the AGV on-board obstacle avoidance sensors the only to be considered, the human operator would be detected too late (Figure 4B): the Sen3Bots meta-sensor fusion allows for a smart and safer behaviour. Note that the simulation does not consider the coalition formation rules, to simplify the scene for the sake of clarity.

The second simulation, shown in Figure 5, reports the case in which a critical area of type 1.1 (see Section II-B for area types reference) has to be crossed by the AGV: two Sen3Bots are sent to the scene and inform the AGV about the presence of one human operator. Figure 5A identifies the different mobile robots involved in the simulation. The AGV CC gathers this information from the SSC and accordingly modifies the AGV motion as it approaches the scene (Figure 5B). Then, since the human activity is perceived as dynamic, the two Sen3Bots are instructed to both stand sentry in the area (Figure 5C).

Last but not least, the third scenario (Figure 6) shows the situation in which an AGV has to cross a workstation area, usually classified as area of type 2, which is positioned in a way that visibility is hindered by racks and thus considered a critical area 1.2. (refer to Figure 6A for a recap of each mobile robot role in the simulation). Two Sen3Bots are preliminarly sent to the scene. However, during the AGV passage the human obstacles are detected to be quite static, allowing for the

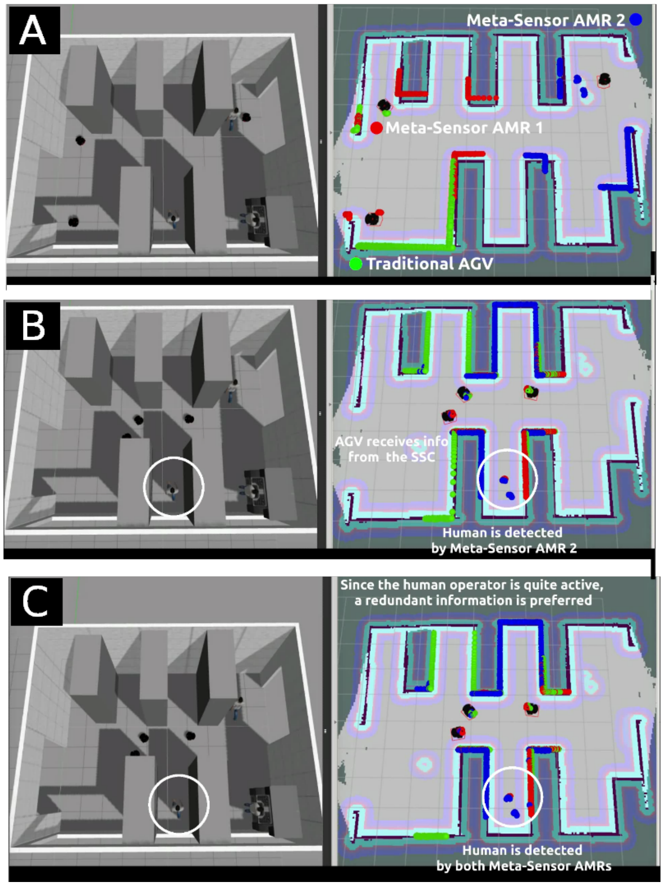

Fig. 5. Sen3Bot Net simulation: critical area 1.1 and dynamic human operator.

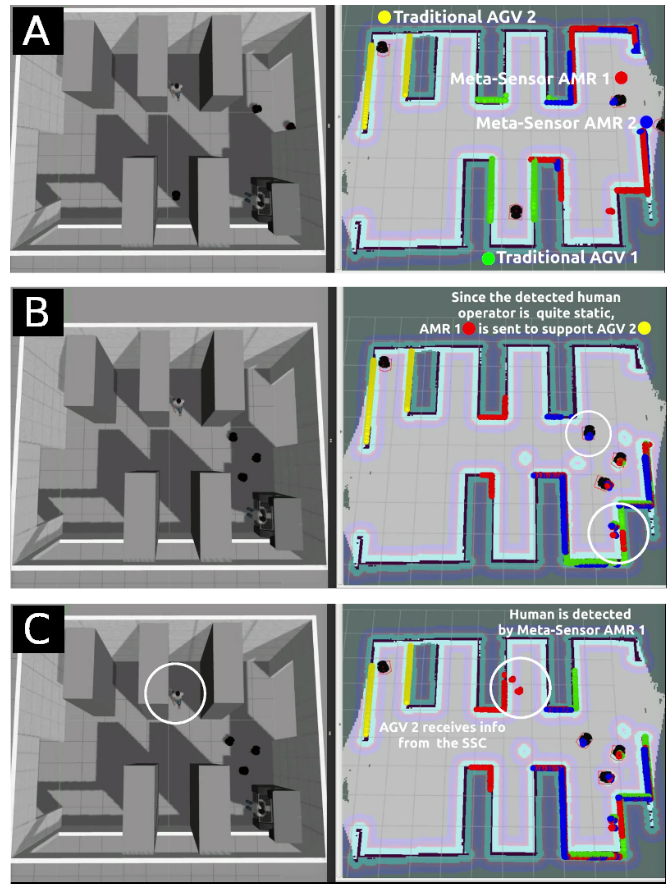

Fig. 6. Sen3Bot Net simulation: critical area 1.2 and static human operator.

system to set one of the meta-sensor AMRs to a priority $p = 2$ (Figure 6B). Then, as shown in Figure 6C, as the Sen3Bot is set available for pursuing other tasks, a monitoring is requested in a very near area and meta-sensor AMR 1 is selected and sent to the requested sentry pose (for reference on task allocation see Section II-C).

IV. Conclusions and Future Works

In this paper we presented a complete description of the Sen3Bot Net behaviour. Our architecture allows to upgrade obsolete pre-existent systems integrating it with intelligent AMRs, allowing for deployment in a higher number of facilities avoiding huge renewal costs. This is in contrast with the on-going trend of entirely substituting the existing mobile robot set-up with a new network of intelligent AMRs. Safety is virtually assured by using the AMRs as a supplement to the AGVs sensor equipment, with the aim of improving the environment perception capability. Using ROS as a development tool, promotes code-reuse through the *ROS Namespace* feature: the code running on each Sen3Bot is the same and threads belonging to a single robot are identified through a unique prefix. The latter approach permits to foster scalability.

As a next step, simulations should be enriched with further test scenarios so as to cover as much as possible all the hazardous situations that may occur in a real industrial environment. Moreover, a more realistic simulation would include animated person models (actors), but the current solution has been adopted to ensure full software compatibility between ROS and Gazebo.

Furthermore, the testing of the framework on real mobile robots will for sure have to take into account that (i) the laboratory working conditions are quite different from the real use case in an industrial environment, and (ii) the robots that will be used for testing the algorithm are laboratory demonstrators running ROS. For instance, ROS network setup heavily depends on TCP/IP standards, leading to some issues, e.g., reliability problems and delays, to be dealt with in an industrial context. It must be noted that in view of a real application of the architecture, each facility may have different AGV network management systems, leading to an ad-hoc implementation of the AGV CCI.

Further specific improvements can be aimed at analyzing and managing those situations in which several policies could be implemented. For example, when the AMR is far from the crossed area of interest, the AGV motion could be simply slowed down by a certain amount or completely stopped, depending on the specific characteristics of the environment and of the production line, leaving the different policies as options to be customized.

References

[1] G. Demesure, D. Trentesaux, M. Defoort, A. Bekrar, H. Bril, M. Djemaï, and A. Thomas, "Smartness versus embeddability: a tradeoff for the deployment of smart AGVs in industry," in *Service Orientation in Holonic and Multi-Agent Manufacturing*. Springer, 2018, pp. 395–406.

[2] G. Fragapane, D. Ivanov, M. Peron, F. Sgarbossa, and J. O. Strandhagen, "Increasing flexibility and productivity in Industry 4.0 production networks with autonomous mobile robots and smart intralogistics," *Annals of Operations Research*, pp. 1–19, 2020.

[3] A. Liaqat, W. Hutabarat, D. Tiwari, L. Tinkler, D. Harra, B. Morgan, A. Taylor, T. Lu, and A. Tiwari, "Autonomous mobile robots in manufacturing: Highway Code development, simulation, and testing," *The International Journal of Advanced Manufacturing Technology*, vol. 104, no. 9-12, pp. 4617–4628, 2019.

[4] R. Inam, K. Raizer, A. Hata, R. Souza, E. Forsman, E. Cao, and S. Wang, "Risk assessment for human-robot collaboration in an automated warehouse scenario," in *2018 IEEE 23rd International Conference on Emerging Technologies and Factory Automation (ETFA)*, vol. 1. IEEE, 2018, pp. 743–751.

[5] Q. Liu, P. Hua, A. Sultan, L. Shen, E. Mueller, and F. Boerner, "Study of the integration of robot in cyber-physical production systems," in *2019 International Conference on Cyber-Enabled Distributed Computing and Knowledge Discovery (CyberC)*. IEEE, 2019, pp. 367–370.

[6] Y. Lu, "Industry 4.0: A survey on technologies, applications and open research issues," *Journal of Industrial Information Integration*, vol. 6, pp. 1–10, 2017.

[7] H. A. Poonawala and M. W. Spong, "Cooperative visibility maintenance in se (3) for multi-robot-networks with limited field-of-view sensors," *Control Theory and Technology*, vol. 15, no. 4, pp. 246–257, 2017.

[8] M. Geng, S. Liu, and Z. Wu, "Sensor fusion-based cooperative trail following for autonomous multi-robot system," *Sensors*, vol. 19, no. 4, p. 823, 2019.

[9] J. Zhang, R. Liu, K. Yin, Z. Wang, M. Gui, and S. Chen, "Intelligent collaborative localization among air-ground robots for industrial environment perception," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 12, pp. 9673–9681, 2018.

[10] J. Al Hage, M. E. El Najjar, and D. Pomorski, "Multi-sensor fusion approach with fault detection and exclusion based on the kullback–leibler divergence: Application on collaborative multi-robot system," *Information Fusion*, vol. 37, pp. 61–76, 2017.

[11] B. Abci, J. Al Hage, M. E. B. El Najjar, and V. Cocquempot, "Multi-robot autonomous navigation system using informational fault tolerant multi-sensor fusion with robust closed loop sliding mode control," in *2018 21st International Conference on Information Fusion (FUSION)*. IEEE, 2018, pp. 1–5.

[12] L. Marín, Á. Soriano, M. Vallés, Á. Valera, and P. Albertos, "Event based distributed kalman filter for limited resource multirobot cooperative localization," *Asian Journal of Control*, vol. 21, no. 4, pp. 1531–1546, 2019.

[13] L. Yin, Q. Ni, and Z. Deng, "Intelligent multisensor cooperative localization under cooperative redundancy validation," *IEEE transactions on cybernetics*, 2019.

[14] J. Blankenburg, S. B. Banisetty, S. P. H. Alinodehi, L. Fraser, D. Feil-Seifer, M. Nicolescu, and M. Nicolescu, "A distributed control architecture for collaborative multi-robot task allocation," in *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*. IEEE, 2017, pp. 585–592.

[15] A. Khalid, P. Kirisci, Z. Ghrairi, K. Thoben, and J. Pannek, "Towards implementing safety and security concepts for human-robot collaboration in the context of Industry 4.0," in *39th International MATADOR Conference on Advanced Manufacturing (Manchester, UK)*, 2017, pp. 0–7.

[16] L. Gumbi and H. Twinomurinzi, "SMME Readiness for Smart Manufacturing (4IR) Adoption: A Systematic Review," in *Conference on e-Business, e-Services and e-Society*. Springer, 2020, pp. 41–54.

[17] A. Markis, M. Papa, D. Kaselautzke, M. Rathmair, V. Sattinger, and M. Brandsttter, "Safety of mobile robot systems in industrial applications," 05 2019.

[18] M. Indri, L. Lachello, I. Lazzero, F. Sibona, and S. Trapani, "Smart Sensors Applications for a New Paradigm of a Production Line," *Sensors*, vol. 19, no. 3, 650, 2019.

[19] M. Indri, F. Sibona, and P. David Cen Cheng, "Sensor data fusion for smart amrs in human-shared industrial workspaces," in *IECON 2019 - 45th Annual Conference of the IEEE Industrial Electronics Society*, vol. 1, 2019, pp. 738–743.

[20] Y. Rizk, M. Awad, and E. W. Tunstel, "Cooperative heterogeneous multi-robot systems: A survey," *ACM Computing Surveys (CSUR)*, vol. 52, no. 2, pp. 1–31, 2019.

[21] "ANSI/ITSDF B56.5-2019, Safety Standard for Driverless, Automatic Guided Industrial Vehicles and Automated Functions of Manned Industrial Vehicles (Revision of ANSI/ITSDF B56.5-2012)," American National Standards Institute/Industrial Truck Standards Development Foundation, Geneva, CH, Standard, 2019.

[22] Preview of R15.08 Industrial Mobile Robot Safety, by Michael Gerstenberger (Chair of the R15.08 Drafting Subcommittee). Accessed: May 2020. [Online]. Available: https://www.robotics.org/userAssets/riaUploads/file/AMR19_Michael_Gerstenberger.pdf

[23] SPARC. Robotics 2020 Multi-Annual Roadmap. Accessed: May 2020. [Online]. Available: https://www.eu-robotics.net/cms/upload/topic_groups/H2020_Robotics_Multi-Annual_Roadmap_ICT-2017B.pdf

[24] A. Bonci, M. Pirani, and S. Longhi, "Robotics 4.0: Performance improvement made easy," in *22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA 2017)*, 2017.

[25] Meta-Sensor AMRs - DEMO. Accessed: May 2020. [Online]. Available: https://youtu.be/6kzC73qn7bI

[26] M. Quigley, K. Conley, B. P. Gerkey, J. Faust, T. Foote, J. Leibs, R. Wheeler, and A. Y. Ng, "ROS: an open-source Robot Operating System," in *ICRA Workshop on Open Source Software*, 2009.

[27] Gazebo Official site. Accessed: May 2020. [Online]. Available: http://gazebosim.org/