Collective awareness for abnormality detection in connected autonomous vehicles

Divya Thekke Kanapram (1,2), Fabio Patrone (1), Pablo Marin-Plaza (3), Mario Marchese (1), Eliane L Bodanese (2), Lucio Marcenaro (1) and David Martin Gomez (3), Carlo Regazzoni (1)

(1) Department of Electrical, Electronics and Telecommunication Engineering and Naval Architecture, University of Genova, Italy

(2) Centre for Intelligent Sensing, School of Electronic Engineering and Computer Science (EECS), Queen Mary University of London, UK

(3) Intelligent Systems Lab
Universidad Carlos III de Madrid, Spain
Email: {divya.kanapram@ginevra.unige.it, f.patrone@edu.unige.it, pamarinp@ing.uc3m.es, mario.marchese@unige.it, eliane.bodanese@qmul.ac.uk, lucio.marcenaro@unige.it, dmgomez@ing.uc3m.es, Carlo.Regazzoni@unige.it }

Abstract—The advancements in connected and autonomous vehicles in these times demand the availability of tools providing the agents with the capability to be aware and predict their own states and context dynamics. This paper presents a novel approach to develop an initial level of collective awareness (CA) in a network of intelligent agents. A specific collective self-awareness functionality is considered, namely agent-centered detection of abnormal situations present in the environment around any *agent* in the network. Moreover, the agent should be capable of analyzing how such abnormalities can influence the future actions of each *agent*. Data-driven Dynamic Bayesian Network (DBN) models learned from time series of sensory data recorded during the realization of tasks (agent network experiences) is here used for abnormality detection and prediction. A set of DBNs, each related to an *agent* is used to allow the agents in the network to reach synchronously aware about possible abnormalities occurring when available models are used on a new instance of the task for which DBNs have been learned.

A Growing Neural Gas (GNG) algorithm is used to learn the nodes variables and conditional probabilities linking nodes in the DBN models; a Markov Jump Particle Filter (MJPF) is employed for state estimation and abnormality detection in each agent using learned DBNs as filter parameters. Performance metrics are discussed to asses the algorithm's reliability and accuracy. The impact is also evaluated by the communication channel used by the network to share data sensed in a distributed way by each agent of the network. The IEEE 802.11p protocol standard has been considered for communication among agents. Performances of the DBN based abnormality detection models under different channel and source conditions are discussed. The effects of distances among agents and of the delays and packet losses are analyzed in different scenario categories (urban, suburban, rural).
Real data sets are also used acquired by autonomous vehicles performing different tasks in a controlled environment.

Index Terms—Self awareness, Collective awareness, Markov Jump Particle Filter, Dynamic Bayesian Network, Connected vehicles, Abnormality detection.

I. INTRODUCTION

Internet of things (IoT) is a concept that connects various physical objects and allows them to exchange data over the Internet. IoT is a forefront technology which helps to reduce human efforts by enabling autonomous control capabilities and intelligence in machines and makes human life easier. IoT can produce a network of intelligent systems when combined with machine learning and signal processing techniques. IoT with smart objects has many applications in the field of surveillance [1], transportation [2], crowd monitoring [3], etc. The term Internet of Vehicles (IoV) has been defined whenever smart physical objects are vehicles [4]. The number of vehicles is increasing exponentially as a consequence of the rapid increase in world population and the expansion of big cities. As a result, road accidents also increase dramatically by various reasons such as distracted driving, adverse weather conditions, speeding, unavailability of contextual aware data, etc. These factors highlight the need for making the objects 'self-aware' and sharing these awareness data among other objects, in order to develop and enrich contextual awareness. Each object needs to be aware not only of itself but also of the other objects and conditions of the surrounding area.

Inter-connectivity and efficient communication schemes are required to develop such collective awareness among smart objects which, in case of intelligent vehicles, can help assure the safety and efficiency in driving. By making objects self-aware, each object would be able to detect abnormal situations in the environment and make appropriate decisions to avoid accidents, threats, or other dangerous actions. If we consider a network of such agents which periodically communicate the acquired information among each other, the future knowledge and actions of one agent can affect the other ones' behaviour. For this reason, reliable communication among agents is mandatory to let them successfully cooperate. However, data exchange among objects can be adversely affected by different factors, such as the distance among them, the transmission delay, and the environmental conditions.

In this work, Dynamic Bayesian Networks (DBNs) [5] have been used as data-driven models learned from sensory data for the detection of abnormalities and the prediction of

future states of the agents. DBN's are hierarchical probabilistic models that make it possible to filter observations received from multiple sensors in order to understand and predict the possible object states that have generated the observations themselves. Data-driven DBN models require temporal dynamic relationships among object states that are learned from training data acquired along with the accomplishment of reference tasks by the agent. Such learned dynamic models make it possible to analyze time-series, i.e. time sequences of observations acquired in successive experiences of the same type to estimate if they occur in statistically normal (i.e. similar) ways. A probabilistic density function (PDF), namely the posterior of unknown states conditioned over observations is the typical recursive result obtained by a DBN filter at each time instant. A specific class of DBN models is here used, i.e. Switching Linear Dynamic Systems (SLDS) [6] or switching models. In such models, it is possible to represent a complex non-linear dynamic behavior with a probabilistic sequential combination of linear dynamic models. Switching random variables are used within the SLDS as higher-level hidden discrete variables one to one associated with different linear dynamic models defined in the DBN. The joint posterior of state and switching variable (superstate) of an object can be recursively estimated within a SLDS. Markov Jump Particle Filter (MJPF) [7] is an inference algorithm that allows estimation of posterior and prediction to be performed when a SLDS is available. Another example of Bayesian inference approach to be used for switching models, however dealing with non-linear dynamic models, is Rao-Blackwellized particle filter (RBPF) [8].

The first objective of this work is to make it possible awareness functionalities are exhibited by each agent of a network that is performing a collaborative task. Abnormality detection is considered as a first functionality necessary to reach collective awareness in the agent's network. As abnormality detection should be performed synchronously by all agents, communication of information required to obtain it must be considered. To achieve this goal, each agent has to learn in an offline training phase, a set of switching DBN models, one for each object in the network that is executing a collaborative task. In this way, during subsequent online situations, each agent will be provided by a model of expected normal behaviors of all agents in the network, including itself. A bank of MJPF filters will be applied to each DBN by each agent, that provides an estimation of posteriors and predictions for each agent. In addition, MJPF is provided of an abnormality detection capability, obtained by measuring the fitness of the dynamic models to current data at different levels of each DBN. Probabilistic distances between predicted priors and likelihood information inside each DBN node are used for this purpose. The second objective is to assess the impact of realistic information exchange among objects on the abnormality detection feature of collective awareness. With a reliable and efficient communication, agents should be capable of sharing ground truth observations acquired in a distributed way by each of them with all other agents in the network. Each agent should dispatch appropriately the ground truth observations received from each remote transmitting agent

to the appropriate MJPF where it can be compared with the respective state predicted by the relative DBN model to estimate the possible presence of abnormalities. In this way, each agent can estimate global abnormality conditions that can arise in any of the agents that compose the network. Self Awareness (SA) of single agents can so become Collective Awareness (CA) of the agent's network. Different distributed communication schemes at increasing complexity can be devised to reach such a CA. The agents could for example either communicate all their first-person observations to each other and then apply MJPFs for all agents to obtain abnormality estimation or communicate and share the outputs of only their own MJPFs using locally observed data. In this latter way, abnormality situations would be communicated, and local observations could remain private within the agent. In this paper, a first analysis is provided to indicate which communication strategy could be the best one depending on contextual parameters like the distance between the agents, the transmission delay, and the achievable data rates of the chosen communication protocol.

The main contributions of the paper can be summarized as follows:

- A method is provided to learn normality models for situations providing training data series, models represented as banks of DBNs. It is shown that an agent's network can use such learned models to online detect abnormal situations that occur in any of the intelligent objects of the network. Results of specific unsupervised learning algorithms used in the training phase to estimate the DBN models, like the GNG algorithm, are provided. Also, the results coming from a specific SLDS inference method working on learned DBNs, namely a MJPF model extended to become able to detect abnormalities, are discussed.
- The robustness of the distributed abnormality detection feature of CA with respect to a realistic communication channel model are discussed; performances are evaluated in order to assess, on the one hand, the reliability and accuracy of abnormality detection under perfect communication hypothesis, and, on the other hand, the robustness is also analyzed of the system model against packet losses and transmission delays of the communication channel among objects.

The remainder of this paper is structured as follows. Some of the main articles and works in the literature regarding self-aware vehicular networks are reported and summarized in Section II. Section III reports our proposed strategy for anomaly detection, describing in detail the principles exploited in the training phase, the steps included in the test phase, and how the communication among agents has been modelled. The experimental setup and the communication system are included in Section IV, proposed anomaly detection scheme and the communication among objects are described in Section V. Conclusions and possible future work are drawn in Section VI.

II. State of the Art

This section describes some of the related work regarding the development of self-awareness in agents and how such agents can perform better if they are part of a network. According to the statistics, about 75 billion "things" will be connected to the Internet by 2025, and a larger portion will be vehicles [9]. The number of vehicles equipped with IoT technology is increasing as a consequence of the rapid growth of vehicles numbers and the spread of the IoT technology, leading to a change from the conventional Vehicle Ad-hoc Networks (VANETs) [10] concept to the Internet of Vehicles (IoV) principle. Allowing inter-vehicles communication with the aim to make them smarter and self-aware is the main principle of IoV, where the concept of self-awareness mainly describes the cognitive capability of living entities such as humans. Self-awareness can be defined as the capability to observe itself as well as the surrounding environment, i.e. contextual awareness. If a vehicle is self-aware capable, it can be provided of models allowing it to detect abnormal situations and consequently decide emergency actions before the situation goes beyond its control. Moreover, such important information has to be shared by communications with other vehicles in the network to make all the surrounding entities aware of the overall situation. This shared awareness is here defined as collective awareness (CA).

In [11], the authors propose an approach to develop a multilevel self-awareness model by focusing on one agent. The developed self-awareness approach is learned by using multisensory data of a vehicle normally interacting in an environment. The model allows the agent to become able to detect abnormal situations present in its surrounding environment. The learning process of the self-awareness model for autonomous vehicles based on data collected from human driving is described in another work [12]. Other related works in this direction aim to enrich the experience of co-operative and secure driving [13], [14].

Artificial intelligence and machine learning are two multi-disciplinary concepts which are growing in interest in a number of research studies in the past few years. However, we still lack a genuine theory that explains the underlying principles and methods that would tell how to design agents that can not only understand their environment but also be conscious of what they do. Moreover, agents have to understand the purpose of their own actions to take a timely initiative beyond the already programmed goals set by humans. Another important aspect is that agents should be able to incrementally learn from their own previous and current experiences and share learned models with other agents. The issues mentioned above are not new, and researchers in various fields of science (artificial intelligence, cognitive science, neuroscience, and robotics) have already addressed the problems in organization and operation of a system capable of performing perception, action, interaction and learning up to different levels of development [15]. The term "cognitive architectures" is commonly used in the Cognitive Sciences, Neuroscience, and Artificial Intelligence (AI) communities to refer to propositions of system organization models designed to mimic the human mind. Most of the previous works that aimed at developing cognitive architectures did not address the issue of self-awareness. Although, some neuroscientists have considered self-awareness as an expression of consciousness [16], others propose to ground it in the robust theoretical framework of integrated information theory [17]. Firstly, it is required to investigate if and how a machine can develop self-awareness and then how it can communicate with other self-aware agents to achieve common goals. To do so, it is important to understand the concept clearly and to propose a computational model that can account for it [18], [19]. To make the agent self aware, it is crucial to develop and integrate perceptual abilities for self-localization and environment interpretation, decision-making and deliberation, learning and self-assessment, and interaction with other agents. If we could bring all those up to the implementation level, it would be possible to make the agent self-aware, i.e., awareness of being in control of it's own actions and responsible for their outcomes [20]. In addition to that, such an integration of the results and characteristics of various subconscious deliberative processes (such as perception, action, and learning) in a shared global workspace [21] appears fundamental in humans to enable meta-cognitive processes such as the ability to report to oneself and to other agents about internal state, decisions and the way these decisions were made [22]. Additionally, it is vital to develop predictive models of agents [23].

To bring IoT into its next cognitive level, more sophisticated AI needs to be injected across the entire network to make it self-aware. Currently, autonomous vehicles may combine data from cameras, onboard sensors, and lidars, making them intelligent and able to learn and adapt to each possible situation. But if they are not connected, then we cannot call them smart. Uber and Tesla are self-driving vehicles, but they are not connected, and they do not cooperate with each other. The two strong technology trends, one in the mobile communications industry and the other in the automotive industry, are becoming intermixed and will provide new capabilities and functionality for future Intelligent Transport Systems (ITSs) and future driving. As the vehicles are continuously growing more aware of their environment due to the higher number of sensors they are equipped with, the amount of interactions rises, both in between vehicles and between vehicles and other road users. As a result, the significance and reliance on capable communication systems for ego-things/Machine to Machine (M2M) are becoming a key asset. On the other hand, the mobile communications industry has connected more than 5 billion people over the last 25 years, and the next step in wireless connectivity is to link all kinds of devices. According to Ericsson's technical mobility report published in 2017, around 29 billion connected devices are forecast by 2022, of which approximately 18 billion will be related to IoT.

In [11], the authors propose a method to develop a multilayered self-awareness in autonomous entities and exploit this feature to detect abnormal situations in a given context. Most of the related works [12], [24] use either position-related information to make inferences or the agents are not connected to each other. In an autonomous agent, the information related to the control plays a significant role in the prediction of future

states and actions of the entity. Moreover, it is imperative to establish a reliable and accurate network among the agents to allow them to communicate important awareness information in order to make the entire network fully aware of the context where the agents operate.

III. METHODOLOGY

This section first describes how the "awareness" can be modelled into the *things* that can generate "ego-things". Ego-things can be defined as intelligent autonomous entities that can perceive their internal as well as external parameters and adapt themselves when they face abnormal situations. Secondly, we investigate how the network of such ego-things can establish timely and efficient communication to develop collective awareness (CA).

The collective awareness in a network of ego-things is defined here as the capability of a set of ego-things in a network to understand whether perception-action information processing models they are provided of, are performing in a normal way. Normality is defined in a Bayesian inference sense, i.e. as the capability of dynamic models describing hidden object state characteristics as confirmed by observations of available agents ego-things' sensors. Such an ability is provided to each ego-thing in the network and concerns the whole set of agents. Communication is available in the network to exchange information necessary to detect abnormalities of all ego-things by each agent in the network. Each ego-thing can so achieve awareness not only about the fitness of its own models when predicting its own state but also about the possibility that abnormality conditions affect the actions of other cooperating agents with respect to predictions provided by their dynamic models. Such a collective awareness can trigger agents' decision systems to perform emergency routines or switching to other available modalities.

The collective awareness is based on detecting jointly and synchronously abnormal situations present in the context. It allows appropriate decisions that can be taken to maintain the stability of the entire network of systems.

The ego-things are equipped with various sensors. The collected data from each ego-thing have been initially synchronized and then categorized into different groups. In this work, we mainly consider the data related to the control part of the ego-thing along with the trajectory data in order to develop collective awareness. The proposed method can be divided into two parts: offline training and online testing. A block diagram representation of the proposed method is shown where offline training and online processing carried on by each ego-thing in the network is shown in Fig. 1. During the offline training phase, each ego-thing learns probabilistic filtering models from agent sensors' dynamic data series collected while collectively performing a reference situation task. This implies that during the training phase, all agents perform the task autonomously or in a teleoperated way. The collected data series provides information on data collected by sensors' observing esoperceptive and proprioceptive data. By assuming that observation models can remain invariant along the process (i.e. the model for estimating state likelihood from sensory

observations is given and fixed), a set of dynamic models is learned, composed by a discrete vocabulary of continuous conditional probabilities functions and by a transition matrix. Such models are organized within a Dynamic Bayesian Network. Such a process is repeated for each agent, and the set of DBNs related to each agent is made available to the collective ensemble of ego things. In the online phase, each DBN is used for filtering agent sensory data within each agent. The comparison of learned dynamic prediction models with incoming observations allows each agent to estimate the level of fitness and to measure abnormality of the collective situation in a distributed way. However, to this end, communications have to be maintained to allow each agent to filter and detect abnormalities also of other ego-things in the network. Filtering is performed using a Bayesian filter appropriate for the type of DBNs learned, i.e. switching models. Markov Jump Particle Filter (MJPF) has been here chosen as the dynamic probability models in learned DBNs are here linear and continuous Gaussian, so allowing Kalman filters to be used at a continuous level in switching models. In Figure 1, it is highlighted how such filters are here provided of the additional capability of measuring abnormalities, in addition to filtering, such capability is at the basis of CA.

A. Offline training phase

In the training phase, each ego-thing will learn a switching Dynamic Bayesian Network (DBN) model for itself, i.e. from the data collected by its own sensors, and one DBN model for each other ego-thing present in the network by exploiting the data generated by that ego-thing's sensors. In this work, ego-things are autonomous vehicles, and the number of vehicles is limited to two. In Fig. 1, the first part (grey shaded area) represents the training phase of an ego-thing, and the steps followed to learn the switching DBN models are explained below.

1) Data pre-processing and state estimation: First, the collected multisensory data are synchronized by using their time stamps. In this work, we considered as a case study the data sequences related to two low dimensionality sensorial data, namely odometry as representative of ego-thing's esoperceptive sense of position and steering as proprioceptive control information of the ego-thing. An initial basic generalized filter [25] has been applied to the data sequence for the estimation of generalized states. The generalized state estimation of acquired data is described below.

Let $Z_k^{e\overline{n}}$ be the measurements in the ego-thing $e\overline{n}$ at the time instant k and $X_k^{e\overline{n}}$ be the state associated to the measurement $Z_k^{e\overline{n}}$, such that $Z_k^{e\overline{n}} = g(X_k^{e\overline{n}}) + \omega_k$. $g(\cdot)$ is the function that maps states into observations and ω_k represents the noise of the sensors. Similarly, $e\overline{n}$ will also have measurements from all the other ego-things in the network, which can be represented as $Z_k^{e1}, \ldots, Z_k^{en}, \ldots, Z_k^{eN}, n \in \mathcal{N}, n \neq \overline{n}$, where N and $\mathcal{N}$ are the number and the set of ego-things in the network, respectively.

As explained in [26], [27], including time derivatives in hidden object states allows dynamic probabilistic flow models describing ego-thing states to be one to one related with
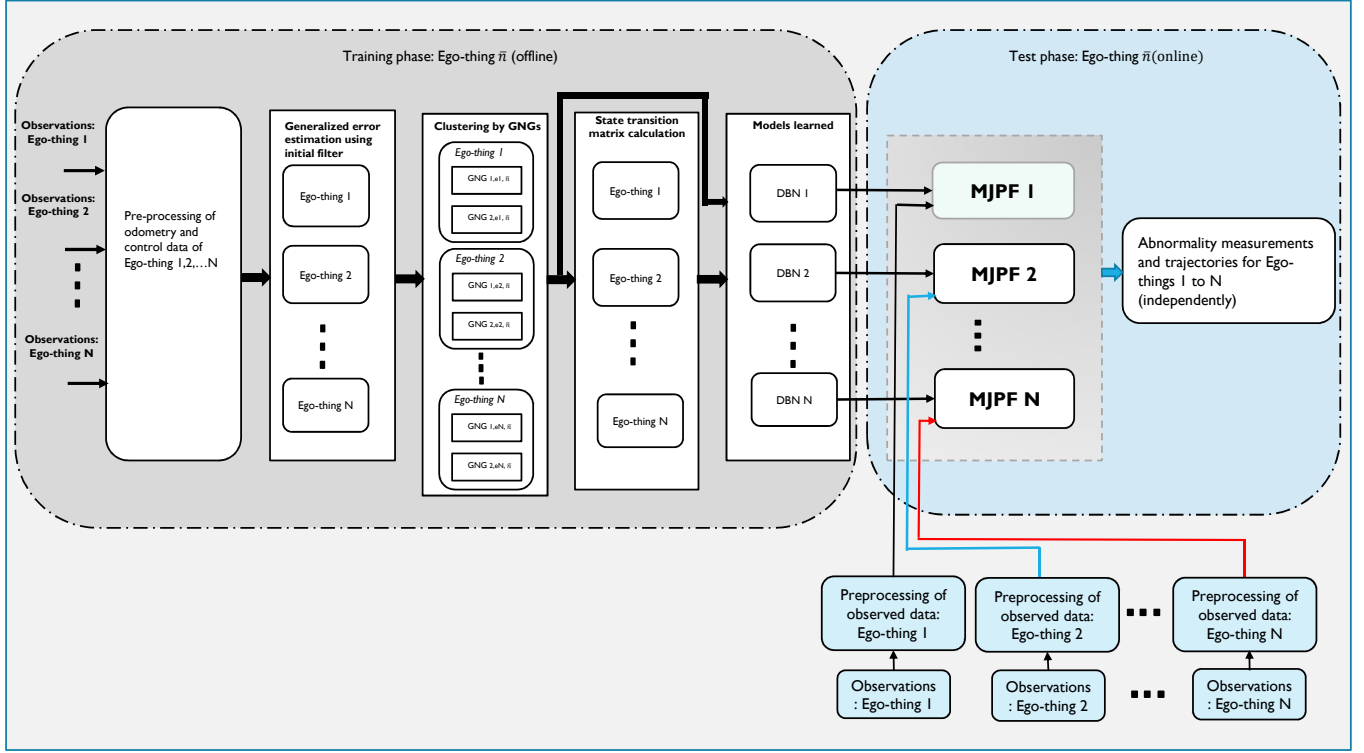
Fig. 1: Block diagram: training phase and test phase

descriptors of motion laws coming from the mechanical statistic, i.e.Lagrangian. Such flow models are represented in the moving reference system of each ego-thing so allowing data series to be described as relative not only to the estimated state of the object but also to how such a state is instantaneously changing. The generalized state of $e\overline{n}$ by considering only itself can be defined as:

$$\boldsymbol{X}_k^{e\overline{n}} = [X_k^{e\overline{n}} \;\; \dot{X}_k^{e\overline{n}} \;\; \ddot{X}_k^{e\overline{n}} \cdots X_k^{(L)e\overline{n}}]^\mathsf{T}, \tag{1}$$

where (L) indexes the L-th time derivative of the state.

The l-th time derivative in $e\overline{n}$ at the time k by considering only itself can be approximated as:

$$X_k^{(l)e\overline{n}} = \frac{X_k^{(l-1)e\overline{n}} - X_{k-1}^{(l-1)e\overline{n}}}{\Delta k}, \tag{2}$$

where $X_k^{(0)e\overline{n}} = X_k^{e\overline{n}}$ and Δk is the uniform sampling time for all multisensory data.

The generalized state of $e\overline{n}$ by considering all the ego-things in the network can be written as:

$$\boldsymbol{C}_k^{e\overline{n}} = [\boldsymbol{X}_k^{e1} \;\; \boldsymbol{X}_k^{e2} \;\; \boldsymbol{X}_k^{e3} \cdots \boldsymbol{X}_k^{eN}]^\mathsf{T}, \tag{3}$$

2) Clustering by GNGs: When a data series is available, a generative filter capable of generating other instances provided of the same statistical properties as well as to predict future states has to be learned. Generative filters here used are hierarchical switching 2-Time Slice DBNs (2T-DBN) [28]. This generative filter, as shown in Fig. 2 is composed of hidden states at continuous and discrete levels. Generalized states are here used at the continuous level. Discrete hidden states are hierarchically higher and represent switching variables. For each value of such random variables, a different dynamic model has to be learned at the continuous level capable of predicting in a different way dynamics of states. This type of DBNs is capable of representing non-linear dynamic models by using a set of linear dynamic models. In order to learn such DBNs, the vocabulary of switching variables and the associated set of dynamic linear models must be learned from data. To this end, having used generalized states is particularly useful. In fact, a technique can be used as in [25] that allows defining an initial basic generalized filter [26] that operates on data series to produce an estimation of the dynamic model that should be associated to each state sparse point obtained by filtering the data sequence. Such a technique consists of an initial filter based on a single value switching variable; such a value corresponds to

a unique dynamic model that assumes no state change should be associated with values in the data series. When a data series violates this assumption, obtained derivatives of state correspond to errors with respect to such a hypothesis. Errors can be clustered to define a set of state-dependent linear dynamic models characterizing the state as varying according to average derivatives and their covariances. Jointly clustering in an unsupervised way states and errors allow one to obtain a vocabulary of regions. Each region is characterized by a compact part of the state space and by a compact subspace of the derivative state space. The average state derivative in the region subspace defines a different filter for each compact state subspace. Here we used an unsupervised clustering approach, the Growing Neural Gas (GNG) [29] method to obtain regions from generalized errors produced as outputs by the initial filter, i.e. sequences of coupled state estimations and errors. GNG clusters correspond to coupled compact regions of state points and errors. Derivative errors cluster encode the description of the expected dynamics that caused the data series to vary instead of following the hypothesis of the initial filter. Different ways of changing are coded as behaviors that have been found in a corresponding compact state region. Compact state region represents switching variables in the hierarchical DBN. GNG is not the unique possible clustering algorithm that could have been employed. K-means clustering [30], Self Organizing Map (SOM) [31], Neural Gas (NG) [32],etc are other possible choices. In comparison to K-means and SOM, NG converges faster and also it has other advantages. The Growing Neural Gas (GNG) algorithm is an improved version of the NG algorithm. In comparison to NG, it does not need any dynamically modifiable parameters. The Growing Neural Gas algorithm extends the Neural Gas algorithm by adding a local error measure for each node. A second addition here used, first proposed by [33], is the utility measure. By considering all the aforementioned advantages, we chose to use the GNG algorithm in this work.

The output of GNG consists of a set of clusters defined as nodes. In the proposed approach a separate GNG clustering is applied to states and derivatives obtained from the initial filter. Each node groups a subset of samples of states or derivatives that have a low distance with respect to the centre of mass of the region associated with the node. Iterative presentation of the same set of samples allows reorganization of nodes averages until convergence is reached. Nodes produced by GNG can be seen as a set of *letters* forming a vocabulary. A different vocabulary is formed for GNGs working on state and derivative samples produced by the initial filter. Nodes associated with the GNG working on derivatives form a vocabulary of dynamic linear models. The flow model of each dynamic model is defined by the centre of mass of the error in the respective node. On the other side nodes associated with the GNG working on states defines a vocabulary of regions, i.e. switching variables of the state space. The set of nodes produced as output by GNG l, i.e. related to the l-th derivative order, of the ego-thing $e\overline{n}$ can be defined as:

$$\boldsymbol{V}^{(l)e\overline{n}} = \{V_1^{(l)e\overline{n}}, V_2^{(l)e\overline{n}}, \ldots, V_{G^{(l)e\overline{n}}}^{(l)e\overline{n}}\}, \qquad (4)$$

where $G^{(l)e\overline{n}}$ is the set of nodes of the GNG l related to ego-thing $e\overline{n}$'s l-th derivative of the state.

$V_n^{(l)e\overline{n}}$ defines the node, and it is considered as a Gaussian random variable whose mean value is the average of samples and whose size corresponds to the variance of the samples themselves. $\boldsymbol{V}^{(l)e\overline{n}}$ can be seen as a *vocabulary* of order l composed by the relative nodes. The switching variables at the highest level of the DBN model learned by GNGs so computed in Fig.2 is the switching variable. Such a variable assumes values from the vocabulary learned by GNG working at the state level $l = 0$. Each region can so be seen as a switching variable: Each value of the variable indexes a region in the continuous state space corresponding to a Gaussian having as mean and covariance associated with the node. The dynamic model associated with that region is found by identifying a letter in the vocabulary of higher derivatives GNG nodes that specifies the velocity and higher-order generalized coordinates of a set of dynamic models that can be associated with the state region.

The compact regions of the derivative state space form a *vocabulary* composed of symbols associated with different dynamics of generalized states $\boldsymbol{X}_k^{e\overline{n}}$. In this work, generalized states include only states and their first order derivatives such as $\boldsymbol{X}_k^{e\overline{n}} = [X_k^{e\overline{n}} \ \dot{X}_k^{e\overline{n}}]^\intercal$. A generic element (letter) of the vocabulary describing clusters of state derivatives can be associated with an equation of a dynamic model to be used by a linear filter. Such a model can be written as

$$\boldsymbol{X}_{k+1}^{e\overline{n}} = A\boldsymbol{X}_k^{e\overline{n}} + BU_k + w_k \qquad (5)$$

where

$$A = \begin{bmatrix} I_j & 0_{j,j} \\ 0_{j,j} & 0_{j,j} \end{bmatrix} \ ; \ B = \begin{bmatrix} 0_{j,j} \\ I_j \Delta k \end{bmatrix}$$

variable j is related to the dimensionality of the state vector for data under consideration. I_j is an identity matrix of dimension j. $0_{j,j}$ is a zero $j \times j$ matrix. $w_k \sim \mathcal{N}(0, \sigma)$, encodes the noise produced by the system. U_k is a control vector that is defined from the average derivative of states obtained by GNG within a dynamic model region. The dynamic model to be chosen is the one of the state regions to which $\boldsymbol{X}_k^{e\overline{n}}$ belongs. A different dynamic model can be associated to different letters describing the same state space region.

By combining letters of nodes produced by GNG working on different derivatives, it is possible to obtain a set of *words* which define discrete states combined with dynamic models, so providing a semantic vocabulary whose elements combine centroids of different derivative orders. Such words computed at ego-thing $e\overline{n}$ are defined as:

$$\boldsymbol{W}^{e\overline{n}} = \{\varphi^{e\overline{n}}, \dot{\varphi}^{e\overline{n}}, \ldots, \varphi^{(L)e\overline{n}}\}, \qquad (6)$$

where $\varphi^{e\overline{n}} \in \boldsymbol{V}^{(l)e\overline{n}}$. $\boldsymbol{W}^{e\overline{n}}$ contains all possible combinations of switching variables and dynamic models. The switching variable acts as a variable at a higher hierarchical level that explains the states from a semantic viewpoint. The discrete switching variables (i.e., letters and words) of the learned DBN model shown in the pink shaded area in Fig. 2.

3) Estimation of state transition: The vocabularies are learned by applying initial filters and GNG clustering to each ego-thing $e\overline{n}$ sensory data acquired along a cooperative task performed with other ego-things. For example, in *scenario 1* (refer section IV) a cooperative driving task of two autonomous cars is considered. In order to allow each ego-thing to develop models that consider time evolution not only at continuous level but also as probabilistic transitions among *words* in the learned *vocabularies*, timestamps are assumed to be provided to data series and transition models to be used at the discrete level of DBNs are estimated. Such transition models allow switching variables to be predicted probabilistically at each moment by the DBN. Moreover, as the DBNs estimate at each time, a joint posterior probability over switching models and continuous states, the predictions provided by the transition model can be used as a source to obtain a further measurement of semantic abnormality. In particular, if predicted words do not match with evidence supported by observations of one agent, then such an agent can occur in a semantic abnormality.

The probabilistic transition matrix has been estimated from the data sequence by considering the transitions in time and such a matrix can tell the mapping of the variables in discrete space (i.e., word space). In other words, it can tell the probability of transition from word W_k^{en} at time instance k to the word W_{k+1}^{en} in next time instance $k+1$ shown in Fig. 2. We use this information for the prediction purpose at the word level.

4) DBN model for all the agents: All the previous steps are the step by step learning process of the switching DBN models. Each ego-thing learns a total number of N switching DBN models in order to predict the future states of each entity in continuous as well as discrete levels. The set of DBNs learned by each ego-thing ei and ej is the same for each other ego-thing in the network, and can be written as:

$$\boldsymbol{DBN}^{ei} = \{DBN^{e1}, \cdots, DBN^{eN}\} = \boldsymbol{DBN}^{ej}, \forall i, j \in \mathbb{N} \quad (7)$$

The number of DBNs learned can be represented as shown in Fig.2. In each DBN, there are three levels such as measurements, continuous and discrete levels. The arrows and links (coloured in black) of such DBN are learned based on the *Scenario 1* task (see section IV). Moreover, the red and blue dotted arrows represent the influence of one ego-thing's action to the future states of the other ego-things in the network. These dotted arrows represent how one ego-things action can be influenced by the future actions of the other ego-things in the network and vice versa.

B. Online testing

In Fig. 1, the second part (shaded in blue) shows the block diagram representation of the online test phase. In this phase, we have proposed to apply a dynamic switching model called Markov Jump Particle Filter (MJPF) [34], [35] to make inferences on the DBN models learned in the training phase as shown in Fig. 2. MJPF is a Bayesian filter with a Kalman Filter (KF) is associated with each particle. In MJPF we use Kalman Filter (KF) [36] in state space (grey shaded area) and Particle Filter (PF) [37] in a higher hierarchical level called *word* level (pink shaded area) in Fig. 2. The blue and red arrows in Fig. 2 depict the information exchange between two ego-things, and, as a consequence, two DBN models. Those arrows tell how the future states of one ego-thing can influence the next states of the other one.

1) Estimation of future states: The MJPF is able to predict and estimate discrete and continuous states of the ego-things. In addition to that, it produces another information i.e abnormality measurements.

The data sequence (experience) never seen in the online training step is pre-processed and given as input to the MJPF applied on learned DBN models. The output of each MJPF is the estimation of future states of the associated ego-thing along with the probabilistic and spatial abnormality measurements.

MJPF uses PF for discrete variables, here corresponding to word variables; each particle used to approximate the joint posterior in MJPF is augmented with an associated continuous random variable characterized by a Gaussian probability. In our case the dynamic model describing changes of the continuous variable associated to a given word value is represented as in Eq. 5, while transition model is used as dynamic model at word level In the prediction step of the MJPF, a SIR [38] PF approach is used to predict new candidate particles at next step using transition model at discrete level. Each particle is enriched also of a Gaussian prediction of the continuous associated variable. This is done as in Kalman Filter, being dynamic models and observation models linear and variables Gaussian. Each predicted particle is so characterized as a word of given value with an associated prior probability at the continuous level. In the update step, the ground truth observations (belonging to each ego-thing) are used to first update the prior at continuous level, so obtaining the new posterior, and then providing a new weight to the particle word, based on the evidence that such a posterior provides to the specific word itself. In our approach these traditional MJPF are enriched by the computation of abnormality measurements as described in the next section, to allow agents to be aware of the fitness of their dynamic models to the observed sequences.

The posterior probability density function of MJPF belongs to ego-thing $e\overline{n}$ can be written as:

$$p(W_k^{e\overline{n}}, \boldsymbol{X_k}^{e\overline{n}}/Z_k^{e\overline{n}}) = p(\boldsymbol{X_k}^{e\overline{n}}/W_k^{e\overline{n}}, Z_k^{e\overline{n}})p(W_k^{e\overline{n}}/Z_k^{e\overline{n}})$$
$$(8)$$

where $W_k^{e\overline{n}}$ is the word in the higher hierarchical level and $\boldsymbol{X_k}^{e\overline{n}}$ is the continuous state in the state space belongs to ego-thing $e\overline{n}$ at time instant k.

As stated above, a different Kalman Filter is associated with each particle W_k^* and is different for each discrete zone (cluster). The Eq. 8 shows the link between the discrete state (i.e. words) and continuous state estimation. The KF associated to particle W_k^* is used to estimate the prediction on the continuous state $\boldsymbol{X_k}^{e\overline{n}}$ and to estimate $p(\boldsymbol{X_k}^{e\overline{n}}/W_k^{e\overline{n}}, Z_k^{e\overline{n}})$.

As explained before, each ego-thing has its own switching model as well as the model of other ego-things. At each

instant, the ego-thing predicts its own future states and future states of the other ego-things by the learned switching DBN models. By receiving the ground truth observations, the ego-thing can match with the predicted states and detect if any anomalies present. The observations from other ego-things can be received through the established wireless channel with a certain delay and loss. By making efficient communication between the ego-things, we can develop the collective awareness in the entire network of ego-things. Such collective awareness can tell if any abnormal situations happen anywhere in the network. Moreover, the collective DBN models can handle the uncertainty of the environment and the variability of observations.

2) Abnormality detection: Posterior probability estimation in MJPF is here enriched with computation of an additional information useful for self-awareness of individual ego-things. i.e., abnormality measurements. Such information is estimated to instantaneously allow each ego-thing to measure how well the learned models fit the currently observed sequence. To estimate the abnormality of a sequence, a statistical distance metric is here proposed that estimates the distance between predictions performed within MJPF at discrete and continuous levels and the sensory observations produced along an ego-thing experience. In this work, Hellinger Distance (HD) [39] is proposed as the metric to evaluate sequence abnormality.

Some important statistical distances include Bhattacharya distance [40], Hellinger distance [39], Total variation distance [41], etc. The Bhattacharyya distance measures the similarity of two probability distributions. It is closely related to the Bhattacharyya coefficient, which is a measure of the amount of overlap between two statistical samples or populations. Similarly, the Hellinger distance (closely related to, although different from, the Bhattacharyya distance) is used to quantify the similarity between two probability distributions. The Hellinger distance is defined between vectors having only positive or zero elements [42]. The datasets in this work are normalized, so the values vary between zero and one; there aren't any negative values. For this reason, Hellinger distance is more appropriate than using other distance metrics as an abnormality measure. The works in [35] and [43] used Hellinger distance as an abnormality measurement.

In this work, Hellinger distance is used as an abnormality measurement between predicted generalized states and observation evidence.

The Hellinger distance related to the Ego-thing $e\overline{n}$ can be written as:

$$\theta_k^{e\overline{n}} = \sqrt{1 - \lambda_k^{e\overline{n}}}, \tag{9}$$

where $\lambda_k^{e\overline{n}}$ is defined as the Bhattacharyya coefficient [44], such that:

$$\lambda_k^{e\overline{n}} = \int \sqrt{p(\boldsymbol{X}_k^{e\overline{n}}|\boldsymbol{X}_{k-1}^{e\overline{n}})p(Z_k^{e\overline{n}}|\boldsymbol{X}_k^{e\overline{n}})} \, \mathrm{d}\boldsymbol{X}_k^{e\overline{n}}. \tag{10}$$

The variable $\theta_k^m \in [0, 1]$, where values close to 0 indicate that ground truth observations match with predictions; whereas values close to 1 show the presence of an abnormality.

Once detected abnormal situations, the ego-thing has to take appropriate actions either by stopping itself or reducing the speed, etc. However, the decision making part is not included in this work.

C. Evaluating the model performance after the packet loss

Each ego-thing has its own model for prediction of the future states and the ground truth observations received from the sensors to check whether if any anomalies present in the environment around it. At the same time, the models for other ego-things can predict the future states and receive ground truth observations from the corresponding ego-things with a certain delay and loss in transmission. The DBN model of each agent is updated sequentially (with a certain delay) using the shared information. The delay and the loss depend on various factors such as the distance between the ego-things, the employed communication protocol, modulation scheme, and frequency, scenario conditions (urban, rural, ...), etc.

To check the model performance in predicting abnormal situations the true positive rate (TPR) and false positive rate (FPR) are calculated to build a set of Receiver Operating Characteristic (ROC) curves [45] corresponds to different K factor values [46]. The ROC curves plot TPR and FPR at different thresholds, where:

$$TPR = \frac{TP}{TP + FN} \; ; \quad FPR = \frac{FP}{FP + TN} \tag{11}$$

The true positive (TP) is defined as the number of times where abnormalities are correctly identified. False negative (FN) consists of the times that abnormalities are classified incorrectly. Accordingly, false positive (FP) are the times where anomalies are wrongly assigned to normal samples, and true negative (TN) represents the times where normal samples are correctly identified. In this work, mainly considered two parameters of ROC for measuring the performance of the model before and after the transmission loss are: *(i)* the area under the curve (AUC) of the ROC curves, which quantifies the performance of the DBNs' abnormal detection at several thresholds; *(ii)* the accuracy (ACC) measurement, which is defined as follows:

$$ACC = \frac{TP + TN}{TP + TN + FP + FN}, \tag{12}$$

1) Communications among ego-things: In the training phase, we assumed that each ego-thing has available all the required data describing all the other ego-things' status. However, in a real scenario, data exchange among ego-things through a wireless mean has to be considered. Different variables affect communication performance over time. They are mainly related to:

- objects' movement, such as object's velocity, acceleration, and moving direction;
- environment where the objects are located, such as urban or rural scenario, presence of obstacles, Line of Sight (LoS) or Non-LoS (NLoS) conditions;
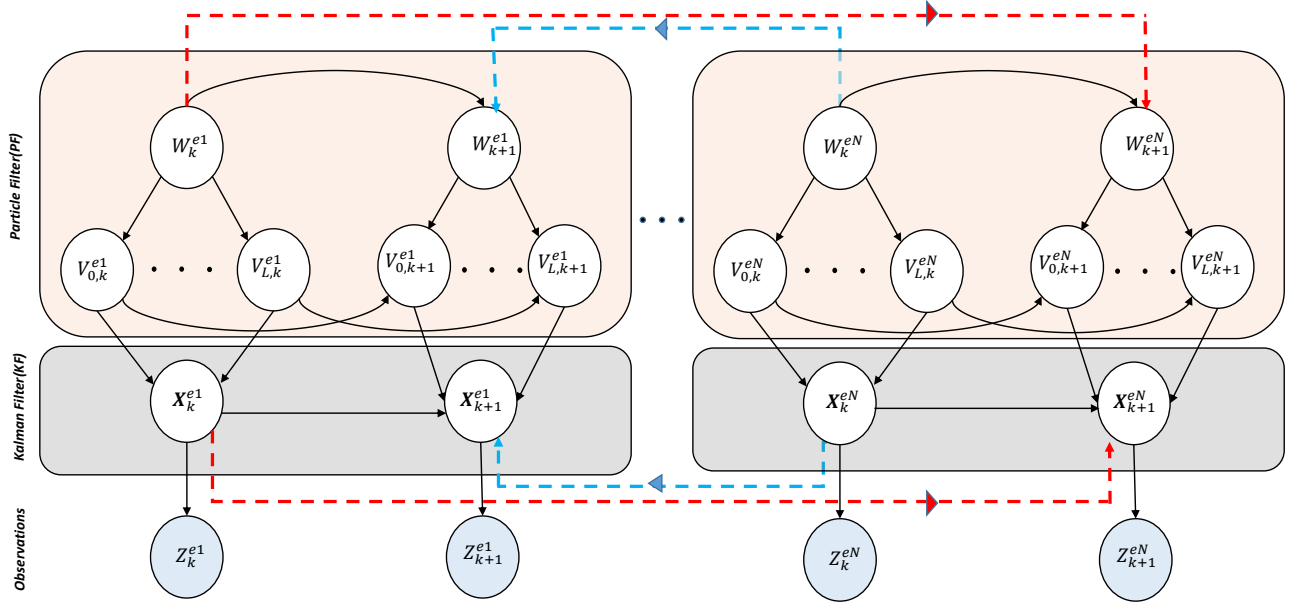
Fig. 2: Collective Dynamic Bayesian Network (CDBN) model

- chosen communication parameters, such as employed communication protocol and modulation, exploited frequency band, achievable data rate, transmission power, and received signal strength.

The channel among ego-things has to be properly modelled in order to consider all the effects which can affect the obtained performance, such as scattering, diffraction, reflection, shadowing, and fading.

The effects on the wireless channel are addressed by large scale and small scale channel models. Large scale models cover effects such as path loss and the effects of the propagation environment over large distances. Small scale models, on the contrary, describe the behaviour in the time domain, taking into account the fast fading effects, i.e. multipath propagation. Large and small scale models are combined to realistically shape channel behaviours. Received power P_r is composed of the transmit power P_t, the large scale effects, i.e. path loss P_L, and the small scale effects ζ:

$$P_r = P_t P_L \zeta \qquad (13)$$

The path loss is the radio attenuation due to the communication mean. It is mainly affected by the communication frequency f and the distance d between source and destination. It can be computed as:

$$P_L = \left(\frac{\lambda^2}{(4\pi)^2 d^\alpha} \right) G_R G_T \qquad (14)$$

where $\lambda = 2\pi f$, α is the attenuation factor, G_R and G_T are the reception and transmission antenna gains, respectively.

The presence of objects and obstacles in the environment originates multiple copies of each transmitted signal, which can strengthen (if $\zeta > 1$) or weaken (if $\zeta < 1$) the original signal. This effect is called multipath fading and can be modelled as a Rayleigh, Rician, or Nakagami distribution.

Considering the current state-of-the-art, we focus on a Rician channel model based on a Rice distribution when LoS is present. Rice distribution can be expressed with parameters K and P_r, which are the Rician K factor and the received power, respectively, or as a function of ρ and σ, which are field strength of the LoS component and the field strength of scattered components, respectively.

The Rice distribution is:

$$p_Z(z) = \frac{z}{\sigma^2} exp \left(\frac{-z^2 - \rho^2}{2\sigma^2} \right) I_0 \left(\frac{z\rho}{\sigma^2} \right) \qquad (15)$$

where $z \geq 0$, ρ and σ are the signal strength of the dominant and of the scattered paths, respectively. Therefore, ρ^2 and $2\sigma^2$ are the average power of the LoS and NLoS multipath components, respectively. As the direct wave weakens, the Rice distribution becomes Rayleigh.

Rician K factor is defined as :

$$K = \frac{\rho^2}{2\sigma_0^2} \qquad (16)$$

It expresses the ratio between the dominant component to scattered waves. The stronger the line of sight component, the greater the K factor. In this way, the Rice distribution in eq. (15) can be expressed in terms of linear K factor as:

$$p_Z(z) = \frac{2z(K+1)}{P_r} exp\left(-K - \frac{(K+1)z^2}{P_r}\right) \cdot$$
$$\cdot I_0\left(2z\sqrt{\frac{K(K+1)}{P_r}}\right) \tag{17}$$

where I_0 is the modified Bessel function of first kind and zero order [47]. When $K \rightarrow \infty$, the Rice distribution tends to a Gaussian one, and when $K \rightarrow 0$, i.e. in case no dominant direct path exists ($\rho = 0$), the Rician fading reduces to a Rayleigh fading defined by:

$$p_Z(z) = \frac{z}{\sigma^2} exp(-\frac{z^2}{2\sigma^2}) \tag{18}$$

A more general fading distribution was developed whose parameters can be adjusted to fit empirical measurements. This distribution is called the Nakagami and is given by:

$$p_Z(z) = \frac{2m^m x^{2m-1}}{\Gamma(m)P_r^m} exp(\frac{-mz^2}{P_r}) \tag{19}$$

The Nakagami distribution is parametrized by P_r and the fading parameter m. For $m = 1$ it becomes Rayleigh fading, instead for $m = \frac{(K+1)^2}{2K+1}$ the distribution is approximately Rician with parameter K.

IV. EXPERIMENTAL SET UP

The scenario considered to validate the proposed methodology consists of two intelligent vehicles called iCab (Intelligent Campus Automobile), shown in Fig. 3b, with the capabilities of autonomous driving [48]. The vehicles are equipped with different sensors such as one lidar, a stereo camera, and encoders. Information about the control of the vehicles, i.e. steering angle (s) and power (p), along with the odometry data (x and y positions) are the data exchanged during the operative process. After collecting the data sets, a synchronization operation is performed in order to perfectly synchronize the collected data sets. The two vehicles follow the same movement trace, shown in Fig. 3a, keeping their position one after the other. For this reason, the iCab1 vehicle is called *leader* and the iCab2 vehicle is called *follower*. The dimension of the movement trace in the testing environment is $38mX33m$.
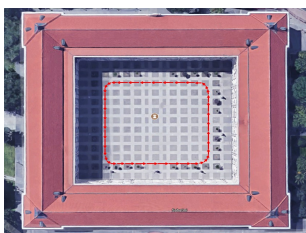
(a) Testing environment　　　(b) iCab platforms

Fig. 3: The vehicles and the environment used for the experiments.

To test the anomaly detection model, we used the data sets from the vehicles while they perform two different actions in the same test environment. The scenarios are:

- Scenario I *Perimeter monitoring*: iCab vehicles perform platooning operation in a closed environment, as shown in Fig. 6. In total, four laps (i.e., the platooning operation has been performed four times one after the other) has been performed and collected the data. The follower vehicle mimics the actions of the leader vehicle. This is the scenario used in the training phase to learn the switching DBN models.
- Scenario II *Emergency stop*: while both vehicles are moving along the rectangular trajectory of the perimeter monitoring task, a pedestrian suddenly crosses in front of the *leader* vehicle. As soon as the *leader* detects the presence of the pedestrian, the vehicle automatically executes an emergency brake and waits until the pedestrian crosses and then continues the task. At the same time, the follower detects the anomaly in the state of the leader and it also performs a stop operation until the leader starts its movement again. The datasets from this scenario have been used to test the switching DBN models learned in the training phase. We have used about 30% of the datasets to test the learned DBN models.
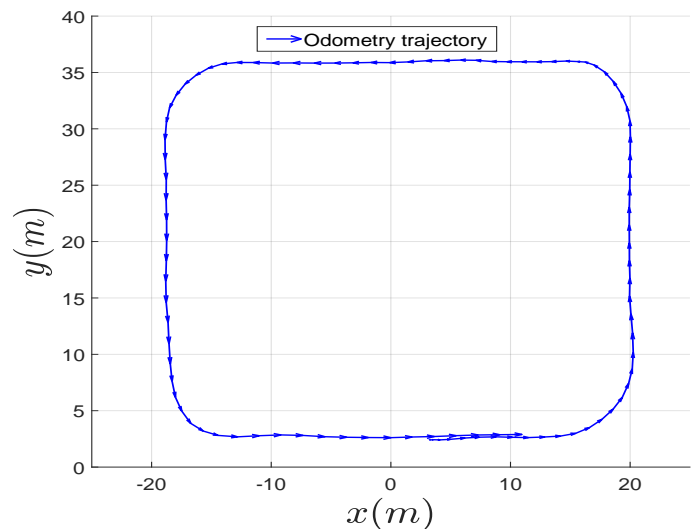

Fig. 4: Position data for perimeter monitoring task

Fig. 4 plots the one lap (from the four laps) odometry (x and y positions) data for the perimeter monitoring task. Fig.5 shows the steering angle w.r.t the vehicle's position (Fig. 5a) and the rotor power w.r.t the vehicle's position (Fig. 5b). For simplicity, in Fig 5, plotted only the data from one lap (i.e., about 800 data points).

To test the reliability and quantify the expected delay of the data exchange between the two vehicles, we have used the ONE simulator [49]. It is a network simulator designed for testing communications among moving objects. Moreover, we have analyzed how the packet loss and delay affect the proposed learned DBN models for abnormality detection.

Considering the current state-of-the-art, the IEEE 802.11p protocol is one of the most feasible and widely considered

(a) Steering angle w.r.t position
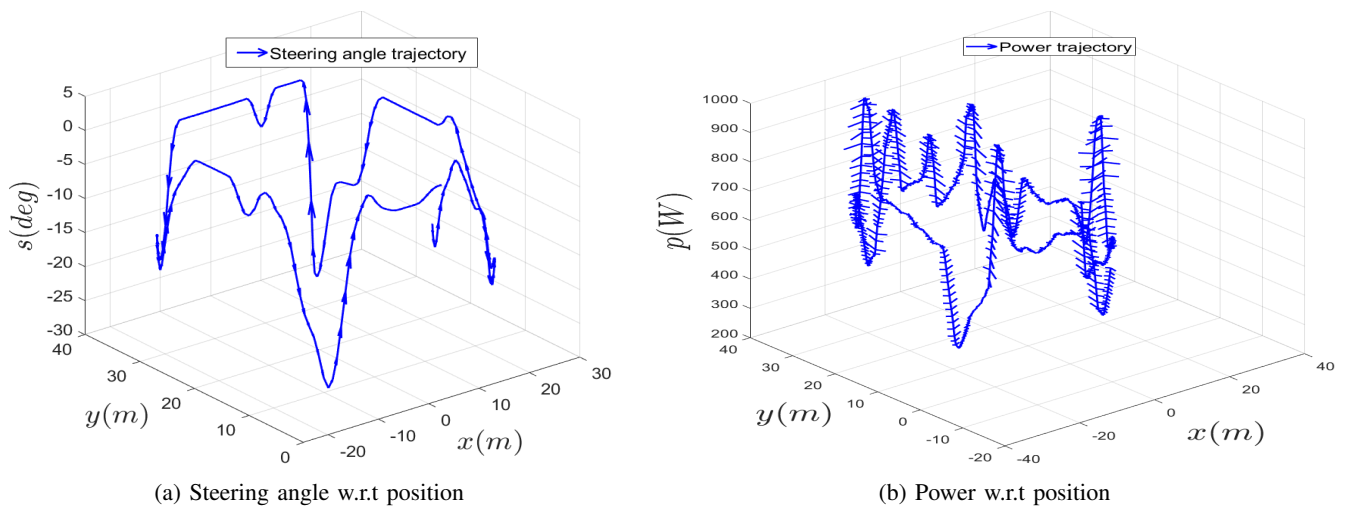

(b) Power w.r.t position

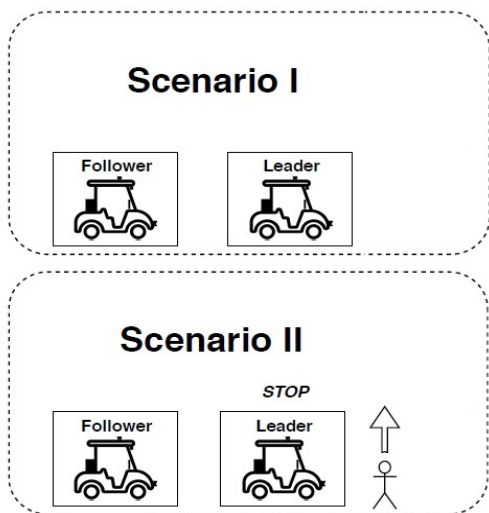Fig. 5: Control variables plotted w.r.t position

Fig. 6: Scenarios considered for the anomaly detection test

in the inter-vehicles communication scenario, especially in autonomous vehicle networks [50].

A new interface has been created in the ONE simulator in order to be able to model the inter-vehicle channel as a Rician channel and to set different values for its parameters, including transmitted power, central frequency, and Rician K factor.

We assumed that the data to be communicated between the ego-things are: XY position, steering angle (s), and power (p) of the rotor of the iCab vehicles together with a time stamp. In this way, we assume that the amount of data to be sent is 4 Bytes for the position + 2 Bytes for the steering angle + 2 Bytes for the rotor power + 4 Bytes for the time stamp. The total data payload size is 12 Bytes. Assuming UDP, IP, and IEEE 802.11p as transport, network, and data link layer protocols, respectively, the overall size of each data packet is 12 + 8 + 20 + 28 + 6 = 74 Bytes.

V. RESULTS

Results of offline learning of DBN models are here not described in all their steps but just providing some general overview. Then, the application of models learned in the online test phase is described in more detail to highlight their application to the agents of the ego-thing network.

A. Offline training phase

In order to collect training and test data sets, the vehicles performed platooning operation four times one after another. The size of each data sequence is about 3200 (i.e. 800 samples per each). The null force filter [25] with a single switching variable corresponds to a unique dynamic model that assumes no state change produces error sequence when the data sequence violates from this rule. The DBN model in the discrete level (i.e. word level) has been learned separately for each ego-thing while they were doing the same co-operative task. The initial filter [25] has been applied to all the agents in the network. The error sequence produced by this initial filter has been clustered to define state-dependent linear dynamic models characterizing the state as varying according to average derivatives and their covariance.

The total number of clusters (nodes) obtained by clustering the states and errors (obtained from the initial filter) were 35 for state space and also 35 clusters for state derivatives with corresponding dynamic models. The GNG reached convergence in this number. Each node cluster corresponds to a letter with respect to respective vocabularies, and the word list has been composed by the different possible combinations of letters from state and state derivative vocabularies (i.e. switching variables and related dynamic models). Then a unique label is given to each letter combinations, and at last, 442 such unique combinations of letters (i.e. words) were kept. A transition matrix at the discrete level was then estimated for each agent whose size was $442 X 442$. This information constitutes the DBN model of the MJPF filter to be applied to each agent.

B. Online test phase

In the online test phase, the data set of scenario II, i.e. emergency stop (refer Sec. IV), has been employed to check the prediction capability of switching DBN models learned in the training phase and to detect the presence of abnormal situations in the environment. The model was able to detect the abnormality situation due to the emergency brake obtaining high values of the Hellinger distance metric, as shown by the red part in Fig. 7 (iCab1 - leader) and 8 (iCab2 - follower).

As can be seen in both figures, there is a significant rise in the Hellinger distance abnormality measures during the abnormality intervals. However, the abnormality peak of the follower vehicle is not as high as the leader's one. The main reason is that after the emergency stop of the leader, the follower gradually decreased its speed rather than doing an emergency brake. We set the abnormality threshold to 0.4 (indicated by the blue dotted line in Fig. 7 and 8) considering the average Hellinger distance value of 0.2 when vehicles operate in normal conditions.

As described before, both vehicles have their own DBN model as well as the model for other vehicles. We have shown here the performance of the DBN model for the leader vehicle in the leader itself and the follower by plotting ROC curves and comparing AUC and ACC parameters. The model for the leader in the follower vehicle can predict the future states of the leader vehicle and detect if any abnormal situations present in the environment around the leader vehicle. Once the follower vehicle detects the abnormal situation of the leader, it should adapt its own behaviour by changing its future action by appropriate decisions. It should be pointed out that in this work the focus is on abnormality detection as a basic step of collective awareness, while the impact on such additional information on decision making and online learning of new actions is beyond the scope of this paper. The important factor that needs to be considered here is the effect of the communication channel over the transmitted data between the vehicles. Such transmission loss causes performance degradation of the DBN model and consequently the abnormality detection capability as well.

The DBN model for the leader vehicle inside the follower vehicle estimates the abnormality situation of the leader after receiving real-time observed data (i.e. steering angle and power) from the leader over the wireless channel. Due to the impact of the communication channel over the transmitted data, the DBN model performance decreases, and we have investigated how it affects the capability of detecting abnormal situations. The performance measure we used in this work is ROC curve parameters such as AUC and ACC. The main factors that affect the transmission loss are the data rate of different modulation schemes, the distance between the vehicles, and the Rician K-factor.

The IEEE 802.11p standard operates at 5.9 GHz central frequency, offers 10 MHz bandwidth, and allows sending data with different modulations and data rates range from 3 to 27 Mbps [51]. We have fixed a maximum communication range to 100 m, considering that the data loss is considerable and beyond a possible realistic reliability requirement if the

distance is higher. We performed different tests changing the values of data rate, modulation, and K-factor as shown in Table I. High K-factor values refer to rural scenarios where the presence of obstacles, buildings, etc. has a lower impact on the achieved performance. The sensitivity column shows the minimum values of the Signal-to-Noise Ratio (SNR) at the receiver to guarantee successful data reception [52].

TABLE I: Simulation parameters [52]

Data rate (Mbits/sec)	Modulation	Sensitivity (dBm)	K factor
3	BPSK	-85	0,1.8,2.6,3
9	QPSK	-80	0,1.8,2.6,3
18	16QAM	-73	0,1.8,2.6,3
27	64QAM	-68	0,1.8,2.6,3

The DBN model performance in terms of ROC curve has been plotted for the leader vehicle for data rates 18 Mbps in Fig. 9 and 27 Mbps in Fig. 10, respectively. These figures show the reliability of the communications in different scenarios, from completely rural ($K = 3$) to urban ($K = 0$). The blue curve refers to the case without transmission among ego-things, i.e. the ideal case of complete knowledge, and has been inserted as a comparison.

When the data rate is 18Mbps, our learned DBN model performance is not degrading much as compared to the no transmission loss case, the performance is in the acceptable range and the model well predicts the abnormal situations. The performance degradation (in terms of AUC) and the accuracy in prediction (in terms of ACC) for 18 Mbps and 27 Mbps data rates are summarized in Tables II and III, respectively. When the environment changes from rural to suburban to urban, the performance of the model again degrades. Finally, when in case of no line of sight component (LOS) ($K = 3$) the value of AUC and ACC in the ROC curve further reduced.

TABLE II: ROC features: Data rate = 18Mbps

K-factor	Area under the curve(AUC)	Accuracy(ACC)
No loss	**0.9039**	**0.9826**
3	0.86665	0.9814
2.6	0.8444	0.9814
1.8	0.7788	0.9764
0	0.7059	0.9764

TABLE III: ROC features: Data rate = 27Mbps

K-factor	Area under the curve(AUC)	Accuracy(ACC)
No loss	**0.9039**	**0.9826**
3	0.8653	0.9801
2.6	0.8409	0.9777
1.8	0.7743	0.9764
0	0.6994	0.9764

Moreover, the distance between the vehicles plays a role in packet losses. To analyze the relationship between distance, delay, and data packet loss, we focused on Scenario I but changing the velocity trace of the follower, in order to let the distance among them change during the simulation.

Fig. 12 and 11 show how the distance between the two vehicles and the communication delay between them change
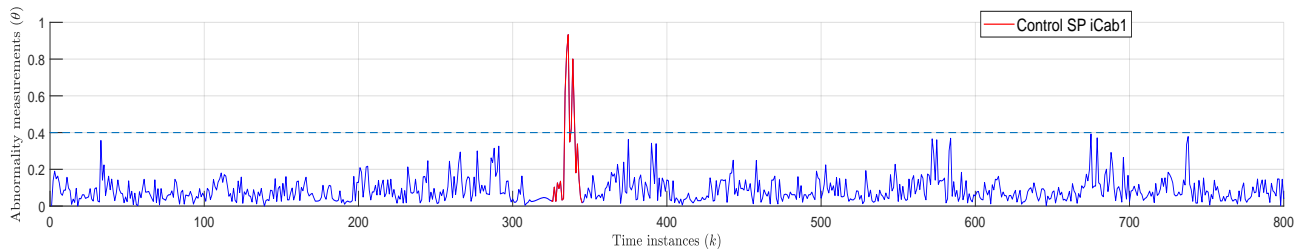
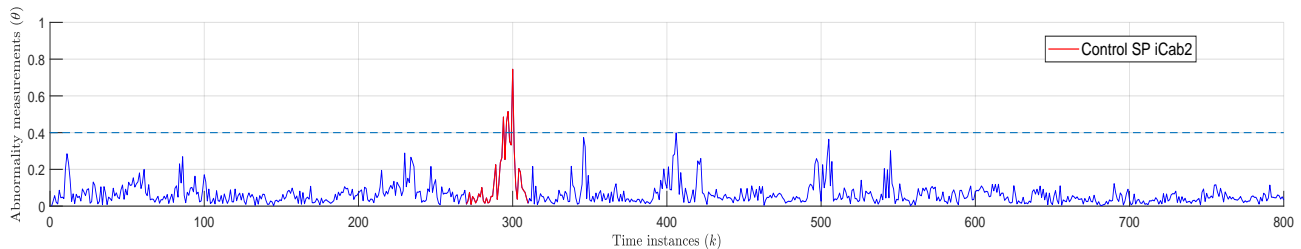Fig. 7: Abnormality measurements plot for iCab 1

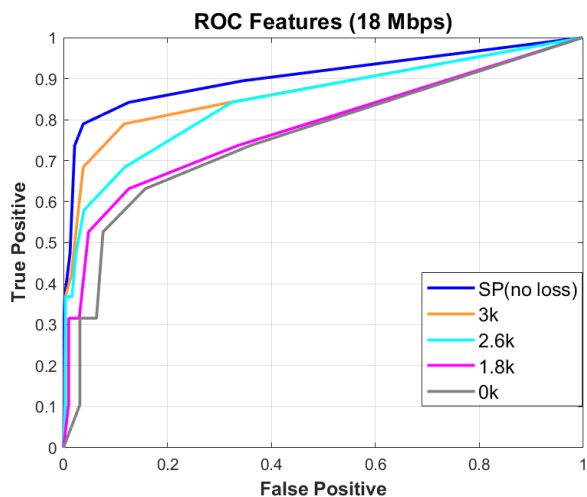Fig. 8: Abnormality measurements plot for iCab 2

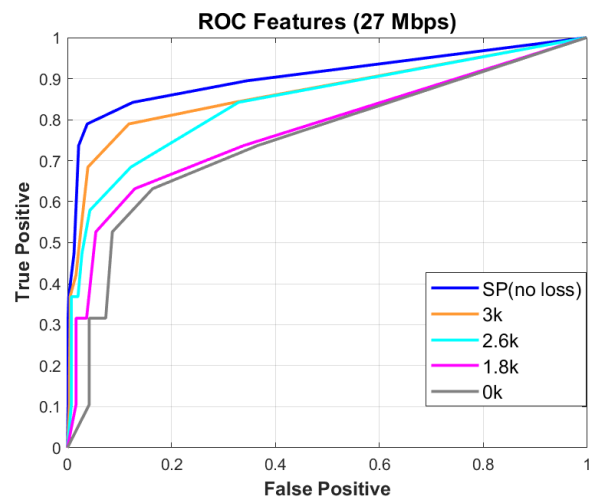Fig. 9: Receiver Operating Curve (18 Mbps)

Fig. 10: Receiver Operating Curve (27 Mbps)

over time, respectively, while the received SNR (blue plot) and the data packet losses (green dots) are shown in Fig. 13. It is evident from the figures that when the distance increases the packet loss also increases. Fig. 13 also shows the power in free space (red line) and sensitivity threshold (light blue line) of the power corresponds to the data rates we considered and finally the lost packets as it did not satisfy the threshold limit of the minimum received power. The overall amount of packets lost is shown in Table IV.

Considering the shown results, different considerations can be pointed out about which data ego-things should exchange to increase the reliability of the described system. For example, if the current distance between vehicles allows obtaining a packet loss ratio below a certain threshold, the vehicles can decide to communicate the ground truth observations to the other vehicles in the network to better detect if there are any

TABLE IV: Packet loss ratio for different K values and different data rates

K-factor	Packet loss ratio (18 Mbps)	Packet loss ratio (27 Mbps)
3	0.0025	0.0037
2.6	0.0037	0.0074
1.8	0.0099	0.0160
0	0.0310	0.0410

abnormalities in the environment. Otherwise, if the vehicle approaches the border of the transmission range or the distance between them exceed a certain threshold, it would be more appropriate to communicate only the abnormality measurements as soon as it detected rather than communicating all the ground truth observations. The transmission loss is directly proportional to the distance, such that if we transmit more data the loss also increases. To reduce the impact of false alarm or
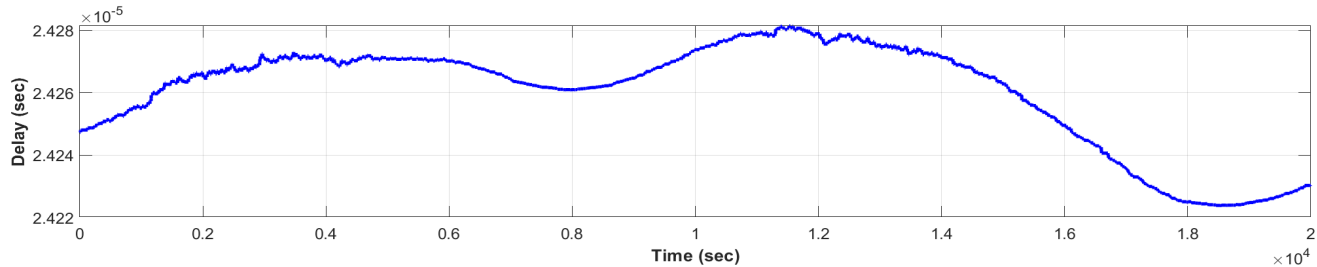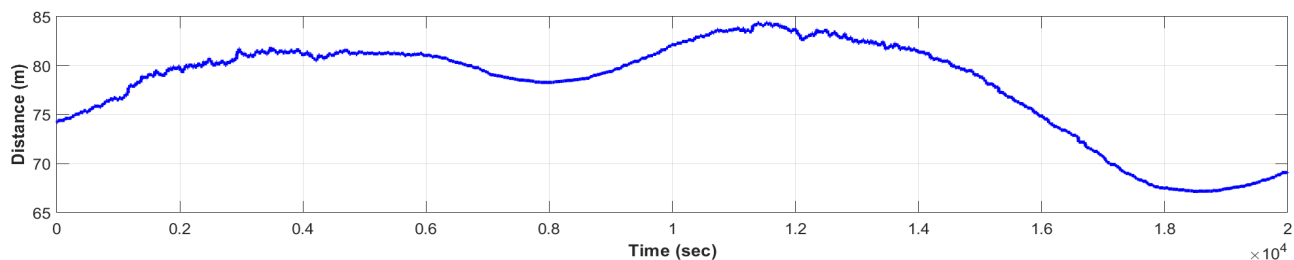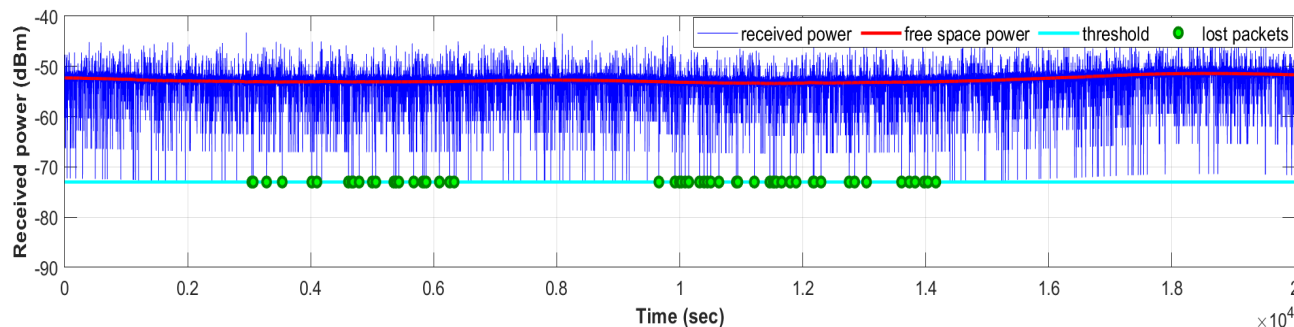
Fig. 11: Delay vs time

Fig. 12: Distance vs time

Fig. 13: Received power and packet losses

missed detection in sensing the abnormal situation, in such situations (higher distances), communicate only abnormality measurements could be more appropriate to give an indication to other ego-things in the network. Although in small distances it is recommended to exchange the observed data itself to detect abnormalities with an acceptable delay.

VI. CONCLUSION AND FUTURE WORK

This paper proposed a method to develop a collective awareness (CA) model and to recognize abnormal situations in smart object networks. Each entity learns a collective awareness model (i.e., a set of Dynamic Bayesian Network (DBN) models) describing the normal behaviour of itself and all the other entities in the network. The considered abnormality metric is based on the Hellinger distance between the predicted states by the learned DBN models and the real-time ground truth observations. A Markov Jump Particle Filter (MJPF) is employed to infer the future states of the entities. The abnormality metric values calculated in each of the DBN models suggest that our method provides good performance in detecting the environmental abnormalities. Moreover, informa-

tion exchange among entities has been considered in order to enhance the proposed strategy.

The considered test scenario is composed of two smart vehicles, one (the follower) following the other (the leader), which move along a predefined track. Communication performance has been collected in order to verify the reliability of the data exchange, quantify the expected performance in terms of delay and loss and consider how these performances could affect the abnormality detection process. We investigated the DBN model performance in the case where each object communicates the ground truth observations to the other entities in the network. To compare the performance with different parameters of the considered channel model (Rician model), such as K-factor, distance, and data rates, we have plotted ROC curves and calculated reliability (Area Under the Curve - AUC) and accuracy (ACC) metrics.

In the future, this work can be extended by learning new DBN models whenever the entities pass through the new experiences. Another direction could be optimizing the model in a way that the same model could be used for all the objects in the network only by changing specific parameters. Moreover, the design of a decision making module capable

to use abnormality situations with respect to available DBN models in order to adapt its own actions in unknown scenarios (by considering different test scenarios) is a topic of future directions of the research.

REFERENCES

[1] G. Ding, Q. Wu, L. Zhang, Y. Lin, T. A. Tsiftsis, and Y.-D. Yao, "An amateur drone surveillance system based on the cognitive internet of things," *IEEE Communications Magazine*, vol. 56, no. 1, pp. 29–35, 2018.

[2] J. A. Guerrero-Ibanez, S. Zeadally, and J. Contreras-Castillo, "Integration challenges of intelligent transportation systems with connected vehicle, cloud computing, and internet of things technologies," *IEEE Wireless Communications*, vol. 22, no. 6, pp. 122–128, 2015.

[3] J. Gubbi, R. Buyya, S. Marusic, and M. Palaniswami, "Internet of things (iot): A vision, architectural elements, and future directions," *Future generation computer systems*, vol. 29, no. 7, pp. 1645–1660, 2013.

[4] M. Gerla, E.-K. Lee, G. Pau, and U. Lee, "Internet of vehicles: From intelligent grid to autonomous cars and vehicular clouds," in *2014 IEEE world forum on internet of things (WF-IoT)*. IEEE, 2014, pp. 241–246.

[5] K. P. Murphy and S. Russell, "Dynamic bayesian networks: representation, inference and learning," 2002.

[6] E. Fox, E. B. Sudderth, M. I. Jordan, and A. S. Willsky, "Nonparametric bayesian learning of switching linear dynamical systems," in *Advances in Neural Information Processing Systems*, 2009, pp. 457–464.

[7] O. L. V. Costa, M. D. Fragoso, and R. P. Marques, *Discrete-time Markov jump linear systems*. Springer Science & Business Media, 2006.

[8] S. Särkkä, A. Vehtari, and J. Lampinen, "Rao-blackwellized particle filter for multiple target tracking," *Information Fusion*, vol. 8, no. 1, pp. 2–15, 2007.

[9] *Internet of Things (IoT) connected devices installed base worldwide from 2015 to 2025 (in billions)*, https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/.

[10] A. Dua, N. Kumar, and S. Bawa, "A systematic review on routing protocols for vehicular ad hoc networks," *Vehicular Communications*, vol. 1, no. 1, pp. 33–52, 2014.

[11] M. Baydoun, M. Ravanbakhsh, D. Campo, P. Marin, D. Martin, L. Marcenaro, A. Cavallaro, and C. S. Regazzoni, "A multi-perspective approach to anomaly detection for self-aware embodied agents," *arXiv preprint arXiv:1803.06579*, 2018.

[12] M. Ravanbakhsh, M. Baydoun, D. Campo, P. Marin, D. Martin, L. Marcenaro, and C. S. Regazzoni, "Learning multi-modal self-awareness models for autonomous vehicles from human driving," *arXiv preprint arXiv:1806.02609*, 2018.

[13] N. Lu, N. Cheng, N. Zhang, X. Shen, and J. W. Mark, "Connected vehicles: Solutions and challenges," *IEEE Internet of Things Journal*, vol. 1, no. 4, pp. 289–299, Aug 2014.

[14] B. Parno and A. Perrig, "Challenges in securing vehicular networks," in *Workshop on hot topics in networks (HotNets-IV)*. Maryland, USA, 2005, pp. 1–6.

[15] A. Morin, "Levels of consciousness and self-awareness: A comparison and integration of various neurocognitive views," *Consciousness and cognition*, vol. 15, no. 2, pp. 358–371, 2006.

[16] B. Hood, *The self illusion: How the social brain creates identity*. Oxford University Press, 2012.

[17] C. Koch, M. Massimini, M. Boly, and G. Tononi, "Neural correlates of consciousness: progress and problems," *Nature Reviews Neuroscience*, vol. 17, no. 5, p. 307, 2016.

[18] A. Chella and R. Manzotti, "Machine consciousness: a manifesto for robotics," *International Journal of Machine Consciousness*, vol. 1, no. 01, pp. 33–51, 2009.

[19] P. R. Lewis, A. Chandra, S. Parsons, E. Robinson, K. Glette, R. Bahsoon, J. Torresen, and X. Yao, "A survey of self-awareness and its application in computing systems," in *Self-Adaptive and Self-Organizing Systems Workshops (SASOW), 2011 Fifth IEEE Conference on*. IEEE, 2011, pp. 102–107.

[20] P. Haggard and M. Tsakiris, "The experience of agency: Feelings, judgments, and responsibility," *Current Directions in Psychological Science*, vol. 18, no. 4, pp. 242–246, 2009.

[21] S. Dehaene and L. Naccache, "Towards a cognitive neuroscience of consciousness: basic evidence and a workspace framework," *Cognition*, vol. 79, no. 1-2, pp. 1–37, 2001.

[22] M. N. Shadlen and R. Kiani, "Consciousness as a decision to engage," in *Characterizing consciousness: from cognition to the clinic?* Springer, 2011, pp. 27–46.

[23] A. K. Seth, K. Suzuki, and H. D. Critchley, "An interoceptive predictive coding model of conscious presence," *Frontiers in psychology*, vol. 2, p. 395, 2012.

[24] S.-K. Han, W.-S. Ra, and J. B. Park, "Tdoa/fdoa based target tracking with imperfect position and velocity data of distributed moving sensors," *International Journal of Control, Automation and Systems*, vol. 15, no. 3, pp. 1155–1166, 2017.

[25] D. Campo, A. Betancourt, L. Marcenaro, and C. Regazzoni, "Static force field representation of environments based on agents' nonlinear motions," *EURASIP Journal on Advances in Signal Processing*, vol. 2017, no. 1, p. 13, 2017.

[26] K. Friston and S. Kiebel, "Predictive coding under the free-energy principle," *Philosophical Transactions of the Royal Society of London B: Biological Sciences*, vol. 364, no. 1521, pp. 1211–1221, 2009.

[27] B. Balaji and K. Friston, "Bayesian state estimation using generalized coordinates," in *Signal Processing, Sensor Fusion, and Target Recognition XX*, vol. 8050. International Society for Optics and Photonics, 2011, p. 80501Y.

[28] D. Koller and U. Lerner, "Sampling in factored dynamic systems," in *Sequential Monte Carlo Methods in Practice*, 2001.

[29] B. Fritzke, "A growing neural gas network learns topologies," in *Proceedings of the 7th International Conference on Neural Information Processing Systems*, ser. NIPS'94. Cambridge, MA, USA: MIT Press, 1994, pp. 625–632. [Online]. Available: http://dl.acm.org/citation.cfm?id=2998687.2998765

[30] J. MacQueen *et al.*, "Some methods for classification and analysis of multivariate observations," in *Proceedings of the fifth Berkeley symposium on mathematical statistics and probability*, vol. 1, no. 14. Oakland, CA, USA, 1967, pp. 281–297.

[31] T. Kohonen, "Self-organized formation of topologically correct feature maps," *Biological cybernetics*, vol. 43, no. 1, pp. 59–69, 1982.

[32] B. Fritzke, "A growing neural gas network learns topologies," in *Advances in neural information processing systems*, 1995, pp. 625–632.

[33] ——, "A self-organizing network that can follow non-stationary distributions," in *International conference on artificial neural networks*. Springer, 1997, pp. 613–618.

[34] M. Baydoun, D. Campo, V. Sanguineti, L. Marcenaro, A. Cavallaro, and C. Regazzoni, "Learning switching models for abnormality detection for autonomous driving," in *2018 21st International Conference on Information Fusion (FUSION)*. IEEE, 2018, pp. 2606–2613.

[35] D. Kanapram, D. Campo, M. Baydoun, L. Marcenaro, E. L. Bodanese, C. Regazzoni, and M. Marchese, "Dynamic bayesian approach for decision-making in ego-things," in *2019 IEEE 5th World Forum on Internet of Things (WF-IoT)*, April 2019, pp. 909–914.

[36] G. Welch, G. Bishop *et al.*, "An introduction to the kalman filter," 1995.

[37] N. J. Gordon, D. J. Salmond, and A. F. M. Smith, "Novel approach to nonlinear/non-gaussian bayesian state estimation," *IEE Proceedings F - Radar and Signal Processing*, vol. 140, no. 2, pp. 107–113, April 1993.

[38] N. Yang, W. F. Tian, Z. H. Jin, and C. B. Zhang, "Particle filter for sensor fusion in a land vehicle navigation system," *Measurement science and technology*, vol. 16, no. 3, p. 677, 2005.

[39] R. Beran *et al.*, "Minimum hellinger distance estimates for parametric models," *The annals of Statistics*, vol. 5, no. 3, pp. 445–463, 1977.

[40] A. Bhattacharyya, "On a measure of divergence between two statistical populations defined by their probability distributions," *Bull. Calcutta Math. Soc.*, vol. 35, pp. 99–109, 1943.

[41] S. Verdú, "Total variation distance and the distribution of relative information," in *2014 Information Theory and Applications Workshop (ITA)*. IEEE, 2014, pp. 1–3.

[42] H. Abdi and N. J. Salkind, "Encyclopedia of measurement and statistics," *Thousand Oaks, CA: Sage. Agresti, A.(1990) Categorical data analysis., New York: Wiley. Agresti, A.(1992) A survey of exact inference for contingency tables. Statist Sci*, vol. 7, pp. 131–153, 2007.

[43] R. Lourenzutti and R. A. Krohling, "The hellinger distance in multicriteria decision making: An illustration to the topsis and todim methods," *Expert Systems with Applications*, vol. 41, no. 9, pp. 4414–4421, 2014.

[44] A. Bhattacharyya, "On a measure of divergence between two statistical populations defined by their probability distributions," *Bulletin of the Calcutta Mathematical Society*, vol. 35, pp. 99–109, 1943.

[45] J. A. Hanley and B. J. McNeil, "The meaning and use of the area under a receiver operating characteristic (roc) curve." *Radiology*, vol. 143, no. 1, pp. 29–36, 1982.

[46] S. Zhu, T. S. Ghazaany, S. M. R. Jones, R. A. Abd-Alhameed, J. M. Noras, T. Van Buren, J. Wilson, T. Suggett, and S. Marker, "Probability distribution of rician k-factor in urban, suburban and rural areas using real-world captured data," *IEEE Transactions on Antennas and Propagation*, vol. 62, no. 7, pp. 3835–3839, July 2014.

[47] A. Goldsmith, *Wireless communications*. Cambridge university press, 2005.

[48] P. Marin-Plaza, A. Hussein, D. Martin, and A. d. l. Escalera, "Global and local path planning study in a ros-based research platform for autonomous vehicles," *Journal of Advanced Transportation*, vol. 2018, 2018.

[49] A. Keränen, J. Ott, and T. Kärkkäinen, "The one simulator for dtn protocol evaluation," in *Proceedings of the 2nd international conference on simulation tools and techniques*. ICST (Institute for Computer Sciences, Social-Informatics and . . . , 2009, p. 55.

[50] A. M. Vegni and V. Loscri, "A survey on vehicular social networks," *IEEE Communications Surveys & Tutorials*, vol. 17, no. 4, pp. 2397–2419, 2015.

[51] "Ieee standard for information technology– local and metropolitan area networks– specific requirements– part 11: Wireless lan medium access control (mac) and physical layer (phy) specifications amendment 6: Wireless access in vehicular environments," *IEEE Std 802.11p-2010 (Amendment to IEEE Std 802.11-2007 as amended by IEEE Std 802.11k-2008, IEEE Std 802.11r-2008, IEEE Std 802.11y-2008, IEEE Std 802.11n-2009, and IEEE Std 802.11w-2009)*, pp. 1–51, July 2010.

[52] A. Bazzi, B. M. Masini, and A. Zanella, "Cooperative awareness in the internet of vehicles for safety enhancement," *EAI Endorsed Transactions on Internet of Things*, vol. 3, no. 9, 8 2017.