Assurance Case Patterns for Cyber-Physical Systems with Deep Neural Networks

Ramneet Kaur(✉), Radoslav Ivanov, Matthew Cleaveland, Oleg Sokolsky, and Insup Lee

PRECISE Center, University of Pennsylvania, Philadelphia, USA
{ramneetk,rivanov,mcleav,sokolsky,lee}@seas.upenn.edu

Abstract. With the increasing use of deep neural networks (DNNs) in the safety-critical cyber-physical systems (CPS), such as autonomous vehicles, providing guarantees about the safety properties of these systems becomes ever more important. Tools for reasoning about the safety of DNN-based systems have started to emerge. In this paper, we show that assurance cases can be used to argue about the safety of CPS with DNNs by proposing assurance case patterns that are amenable to the existing evidence generation tools for these systems. We use case studies of two different autonomous driving scenarios to illustrate the use of the proposed patterns for the construction of these assurance cases.

Keywords: DNNs · Safety-critical CPS · Safety properties · Assurance case

1 Introduction

With recent advances in machine learning, there is much interest in using deep neural networks in safety-critical cyber-physical systems (CPS), such as self-driving vehicles [5], aircraft collision avoidance [22], and medical diagnoses [10]. The black-box nature of neural networks (NNs) makes it difficult to interpret their behavior on perturbed or even unseen inputs and therefore makes it challenging to provide safety guarantees about systems with NNs. To enable the use of NNs in safety-critical CPS, it is therefore important to convincingly demonstrate that CPS with NNs (CPSNN) are acceptably safe to use.

One way to argue about the safety of CPSNN is through assurance cases [1]. Systems developed for medical, transportation, infrastructure applications, etc. that significantly impact life, property, or environment need to get the approval of an independent entity such as a regulatory body. This approval process can be viewed as the manufacturer making the case that their system meets the

This work is supported in part by the Air Force Research Laboratory and the Defense Advanced Research Projects Agency as part of the Assured Autonomy program under Contract No. FA8750-18-C-0090.

A. Casimiro et al. (Eds.): SAFECOMP 2020 Workshops, LNCS 12235, pp. 82–97, 2020.
https://doi.org/10.1007/978-3-030-55583-2_6

criteria for certification and the regulatory body assessing this case to arrive at a certification decision. An assurance case provides a structure for making this case by using arguments supported by evidence to justify the claim in a hierarchical fashion. This hierarchical structure of the assurance case with explicit claims, arguments, and evidence has favored its use in the certification process [30]. For instance, the Food and Drug Administration (FDA) changed its approval process to enable the use of assurance cases for demonstrating the safety of insulin pumps [9] and the Federal Aviation Administration (FAA) accepts assurance cases to approve the safety of aviation systems [11].

Assurance cases have been used to assure the safety of the traditional CPS (CPS without NN components) in the past [3,12,26,35]. These cases analyze the system's specification by making use of the analytical techniques (such as proofs [35]) build on model-based development [3,35], hazard mitigation [12] or both [26]. The black-box nature of NNs makes it difficult to apply these analytical techniques for reasoning about CPSNN and thus structuring their assurance case in the way they are done for the traditional CPS.

Prior work has been done on proposing assurance case patterns for the safety-critical CPSNN [7,25,28]. Some of these patterns argue about the safe functionality [25] or performance [28] of machine learning components in the CPS. Others [7] argue about the acceptance of residual risk in these systems due to the functional insufficiency of machine learning. All of these patterns are specific to the assurance of the functional requirements (or features) of machine learning components in the CPS and do not provide assurance about specifications of the entire system. Also, the challenge of coming up with a provably exhaustive list of requirements for machine learning components in CPSNN makes it difficult to extend these patterns for the assurance of CPSNN.

We propose assurance case patterns for specifications of the closed-loop behavior of CPSNN. The main challenge in building an assurance case for CPSNN is the black-box nature of NNs, which makes it difficult to generate the evidence required for the assurance of CPSNN. An assurance case built for CPSNN should be structured in a way that is amenable to generating evidence about the NN for the assurance of the larger system.

A feasible approach to generating evidence for the assurance of CPSNN is to make use of the computational tools that have been developed recently to provide formal guarantees about these systems. These tools can be broadly classified into two categories. The first analyzes the NN separately from the rest of the system. Existing tools analyzing the behavior of NNs characterize the correctness of these networks based on robustness [6,24], safety guarantees [19,23] or properties [15,17] of these networks. These tools can generate evidence for NN-specific, component-level claims in the assurance case of CPSNN. The second analyzes the closed-loop behavior of CPSNN for both verification [21,32] and falsification [14,33]. These tools can generate evidence for system-level claims made in assurance cases for CPSNN. This classification of the evidence generation tools for the assurance of CPSNN into two categories motivates us to propose two assurance case patterns for CPSNN, one for each category.

The first pattern is based on an assume/guarantee argument. The system makes an assumption about some property of the NN. This assumption leaves us with a much simpler model of the CPSNN to analyze, allowing us to scale the existing verification [8] or falsification [13,16] techniques for the CPS to the CPSNN. Arguments to guarantee the assumed property of the NN need to be made and justified separately in this pattern. These guarantees are composed with the claim about the system with the assumed property of the NN to provide assurance about the CPSNN. Tools analyzing the behavior of NNs can be used to generate evidence for the NN-specific claims made in this pattern. The second pattern is based on a holistic approach to the assurance of CPSNN. This approach relies on the analysis of the closed-loop behavior of the system and does not require a separate specification for the NN. Tools analyzing the closed-loop behavior of the CPSNN can be used to generate system-level evidence in this pattern.

To evaluate the applicability of the proposed patterns, we consider two case studies. The first case study is about the safety specification of the closed-loop system from [20]. It consists of an NN-controlled F1/10 car [2] equipped with LiDAR, running in a known hallway environment. We make use of the holistic pattern to provide assurance about this system. The second case study is about the safety specification of the closed-loop system from [14]. It consists of an autonomous car with a perception-based NN and an emergency braking system (AEBS) [31], driving on a highway with a stationary car in front of it. We make use of the assume/guarantee pattern to provide assurance about this system.

Our contributions in this paper can be summarized as follows. First, we propose two assurance case patterns based on the existing tools for generating evidence for specifications of the closed-loop behavior of the CPSNN. Second, we illustrate the applicability of the proposed patterns with the help of two case studies. Third, we discuss directions for the development of new tools for the assurance of CPSNN with the help of undeveloped claims in the case studies.

2 Background

Here, we first define the assurance case and its goal structuring notation. We then describe the model-based assurance approach of the traditional CPS. Finally, we classify the existing tools that can be used to generate evidence in the assurance case of CPSNN into two categories with examples for each category.

2.1 Assurance Case and GSN

Assurance cases provide a structure for arguing about the safety of a system by making arguments that are supported by evidence to justify the safety claims about the system in a hierarchical fashion. It is defined as a "reasoned and compelling argument, supported by a body of evidence, that a system, service or organization will operate as intended for a defined application in a defined environment" [18].

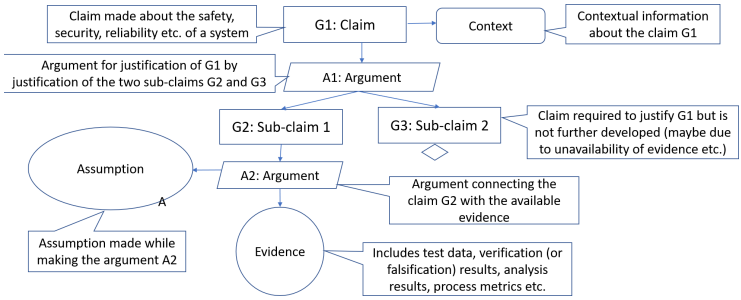

Fig. 1. An example of the hierarchical structure of assurance case made in GSN

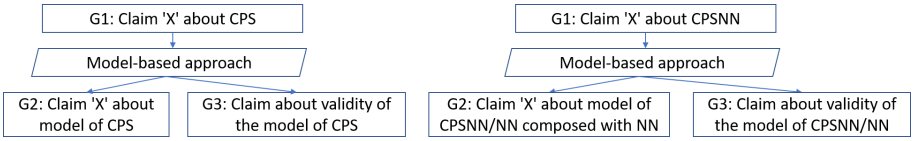

Fig. 2. Model-based approach for the assurance of CPS (left) and CPSNN (right)

Goal Structuring Notation (GSN) [18] is the most widely used graphical notation for representing assurance cases. The principle symbols of GSN are rectangles, parallelograms, circles, ovals with an 'A' at the bottom, rounded rectangles and rectangles with a diamond at the bottom representing claim (or goal), argument (or strategy), evidence (or solution), assumption, context and undeveloped claim in the assurance case, respectively. An example of the hierarchical structure of the assurance case made in GSN is shown in Fig. 1.

Claims, arguments and evidence (CAE) is another framework used to represent assurance cases. CAE leaves arguments as black boxes, while GSN makes them explicit through strategies. We were interested in the details of arguments and that is why we chose GSN over CAE.

2.2 Model-Based Approach for the Assurance of Traditional CPS

The model-based approach of building an assurance case targets the model-based development process of real-world systems [3]. This approach has been used for building assurance cases for the traditional CPS in the past [3,35]. The use of the model-based approach for providing assurance about CPS is motivated by the fact that most of the existing evidence for these systems is based on the models of these systems [27]. Verification tools such as Flow* [8] and falsification tools such as Breach [13] and S-TALIRO [16] are some examples of the existing tools that can be used to generate evidence for the model of the traditional CPS.

The structure of the assurance case for CPS based on the model-based argument was proposed in [35]. This structure is shown in the Fig. 2. It reflects the fact that the model-based evidence for a real-system is only as useful as the

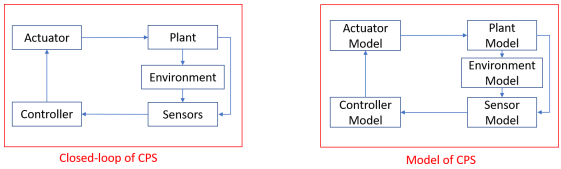

Fig. 3. Closed-loop of a traditional CPS (left) and its corresponding model (right)

model that is used to represent the system. Thus, in addition to the claim about the model of the CPS ($G2$), another claim about the validity of the model with respect to the real CPS ($G3$) needs to be made and justified.

2.3 Existing Tool-Based Evidence for the Assurance of CPSNN

The uninterpretable nature of NNs restricts the use of the traditional analytical methods to provide guarantees about CPSNN [25]. This motivates the use of computational tools based on techniques such as verification, falsification, and optimization to generate evidence about these systems. Existing tools based on these techniques that can be utilized to generate evidence for CPSNN can be classified into the following two categories.

Tools Analyzing the Component-Level Behavior of CPSNN. Work has been done in developing tools that analyze the behavior of NNs. These tools can be used to generate evidence for the NN-specific component-level claims made in the assurance case for CPSNN. Some examples of these tools are as follows. Verification tools such as Reluplex [23] and DLV(Deep Learning Verification) [19] can be used to provide evidence for the verification of the safety properties of NNs. Guarantees about the robustness of NNs can be generated with the help of the tools such as CNN-Cert [6] and POPQORN [24]. Other tools such as Sherlock [15] and LipSDP [17] can be used to assure tight bounds on the output set and global Lipschitz constant for NNs, respectively.

Tools Analyzing the System-Level Behavior of CPSNN. Verification tools for properties of the CPSNN with NN controllers have been developed recently [21,32]. System-level verification has been so far applied to only these types of CPSNN. CPSNN with perception-based NNs do not lend themselves to these verification techniques due to the high dimensionality of their input space. This has led to the development of falsification tools [14,33] for analyzing the closed-loop of CPSNN with perception-based NNs. The absence of counterexamples from these falsification techniques provides evidence of the correct behavior of CPSNN with respect to its specification.

3 Assurance Case Patterns for CPSNN

We propose two assurance case patterns for the safety specifications of CPSNN. These patterns are built on the model-based approach for assurance.

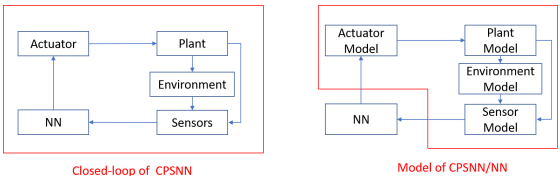

Fig. 4. An example of the closed-loop of a CPSNN (left) and the corresponding model of its CPSNN/NN composed with the NN (right)

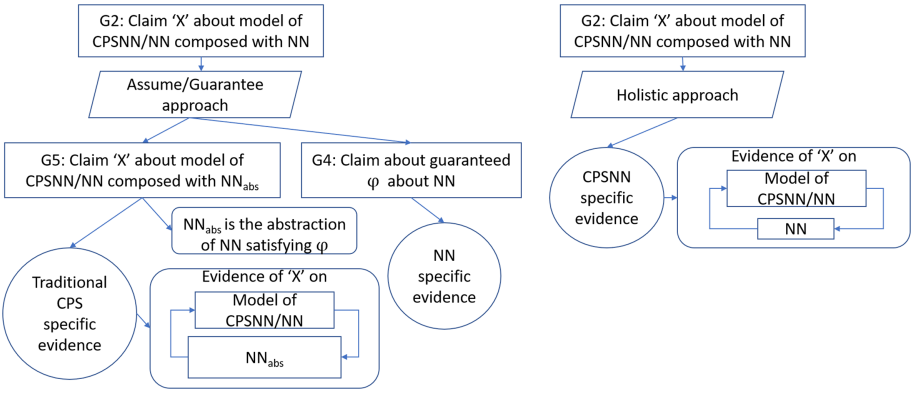

Fig. 5. Modular (left) and Holistic (right) patterns for the assurance of $G2$ in the model-based approach for CPSNN

Figure 3 shows the closed-loop structure of a traditional CPS and its corresponding model used in the model-based assurance approach for CPS [35]. The plant in this system is a physical object such as a vehicle or robot. Sensors collect information about the state of the plant and objects in the plant's environment. This sensory information is used by a controller to produce control actions, which are actuated by actuators on the plant. The model of a traditional CPS is the composition of the models of the individual components of the real-system.

The proposed patterns reflect the model-based assurance approach of the traditional CPS by modeling the sub-system of a traditional CPS present in a CPSNN. Figure 4 shows an example of the closed-loop structure of a CPSNN, where the controller in the traditional CPS is replaced by an NN in the CPSNN. We call a CPSNN without its NN components a CPSNN/NN. The model of a CPSNN/NN is equivalent to the model of the sub-system of CPS present in the CPSNN. This model is composed with the NN to close the loop in the system.

The Fig. 2 shows the model-based approach for the assurance of CPSNN. Assurance about the CPSNN is provided on the model of the CPSNN/NN composed together with the NN component via claim $G2$. Since assurance about CPSNN is provided on a system that approximates (via model) the CPSNN/NN

in the CPSNN, an additional claim, $G3$, is made about the validity of this approximation.

The proposed patterns are amenable to the existing evidence generation computational tools for the assurance of the claim $G2$ in the model-based assurance approach for CPSNN. These patterns differ from each other in their approach to generate this evidence. The first pattern analyzes the NN separately from the model of the CPSNN/NN for the assurance of $G2$ and makes use of the component-level tools for the assurance of CPSNN. The second pattern analyzes the closed-loop behavior of the CPSNN/NN model composed with the NN and does not require a separate claim about the NN in the system. This pattern makes use of the system-level tools for the assurance of CPSNN. Next, we describe the two assurance case patterns as shown in Fig. 5, in detail.

3.1 Pattern 1: Modular Pattern

The first pattern is based on the assume/guarantee approach for providing assurance about the claim $G2$ in the model-based assurance approach for CPSNN. Here, an abstraction of the NN, NN_{abs}, satisfying a property φ is considered. The assurance about the claim made in $G2$ is provided on the closed-loop of the model of CPSNN/NN composed with NN_{abs}, via claim $G5$ in this pattern. This abstraction of the NN leaves us with a much simpler closed-loop model of the CPSNN/NN composed with NN_{abs} to analyze. This analysis can be performed with the existing verification or falsification techniques for the traditional CPS.

Since $G2$ is assured on the model of CPSNN/NN composed with an abstraction of the NN satisfying φ, an additional claim, $G4$, about the satisfaction of φ by the NN needs to be made separately in this pattern. Tools analyzing NNs in terms of the safety guarantees [19,23], robustness [6,24] or properties [15,17] can be used to provide evidence for the guaranteed property of the NN. We call this pattern the modular pattern because the system-level assurance is generated by composing assurance claims about the modules, CPSNN/NN and NN, of the system in this pattern.

An example of instantiation of the modular pattern is the assurance about the safe reachable set of states of a NN controlled linear time-invariant dynamical plant under bounded perturbations. Bounded models for both the plant and measurements comprise the model of the CPSNN/NN in this system. This model composed with an NN_{abs} can be used to argue about the reachable (and hence safe) state by the plant with the help of the evidence of Theorem 1 from [34]. NN_{abs} is the abstraction of the NN satisfying the property of a bounded output for a given set of bounded input. This property of the NN can be verified by the existing tools [15,17] for providing NN specific evidence in this pattern.

3.2 Pattern 2: Holistic Pattern

The second assurance case pattern for the CPSNN is based on analyzing the whole system without decomposing it down into its modules. Therefore, we call this pattern as the holistic pattern. The evidence for the claim $G2$ about the closed-loop behavior of the model of CPSNN/NN composed with NN is generated by the existing system-level tools for CPSNN [14,21,32,33] in this pattern.

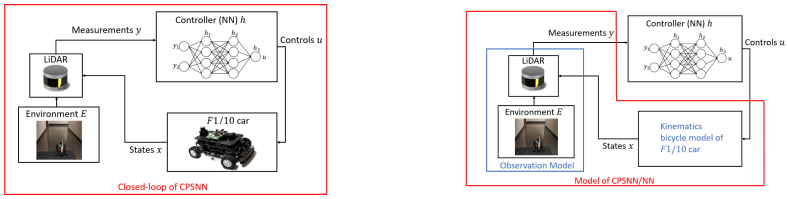

Fig. 6. Closed-loop system, S_1, of the CPSNN considered for the case study of the holistic pattern (left) and the corresponding model of its CPSNN/NN composed with the NN-controller in the system (right) [20]

An example of instantiation of this pattern is the assurance about the safe distance between two cars, where the follower car is equipped with an NN-based adaptive cruise control and a radar to measure the distance to the lead car. The dynamics model for the two cars traveling on a straight road together with the measurement model of the radar comprises the model of the CPSNN/NN, which when composed with the NN-controller can be used to argue about the safety of the system with the help of evidence generated by the verification tool from [32].

4 Case Studies

We provide case studies to illustrate the applicability of the proposed patterns.

4.1 Case Study for the Holistic Pattern

System Description and the Model of Its CPSNN/NN Composed with the NN in the System. We consider a CPS with an NN-controller from [20] for this case study. This system, S_1, is shown in Fig. 6. It consists of an autonomous F1/10 car [2] running at a constant throttle and low speed ($[0, 5]$ m/s) in an empty hallway. The car is equipped with a LiDAR to provide distance measurements from the hallway walls. These distance measurements are fed into an NN-controller which gives front steering commands to the car, thereby controlling the heading of the car. The NN is a small fully connected network with smooth activation functions. The safety property of interest for S_1 is that the car navigates the hallway without hitting its walls.

The model of the CPSNN/NN in S_1 composed with the NN-controller in the system is shown in Fig. 6. It contains three main components. First, the behavior of the car is captured by a continuous-time dynamical system that uses a control signal generated by the NN-controller as input and contains differential equations that represent the evolution of the system state. A Kinematic bicycle model, which is known to work well for front-steering cars at low speeds [29], is

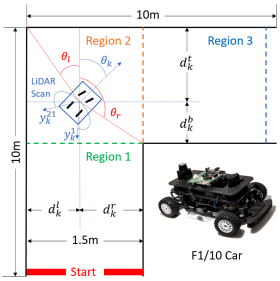

Fig. 7. Hallway divided into three regions depending on number of walls reachable by LiDAR [20]

used to represent the dynamics of the F1/10 car by the following equations:

$$\begin{aligned} &\dot{x} = vcos(\theta + \beta),\ \dot{y} = vsin(\theta + \beta),\ \dot{v} = -c_a v + c_a c_m (u - c_h) \\ &\dot{\theta} = \frac{Vcos(\beta)}{l_f + l_r} tan(\delta),\ \beta = tan^{-1}\left(\frac{l_r tan(\delta)}{l_f + l_r}\right), \end{aligned} \tag{1}$$

where v, θ, β and (x and y) is the car's linear velocity, orientation, slip angle and position respectively. u is the throttle input, δ is the heading input. c_a is an acceleration constant, c_m is a motor constant, c_h is a hysteresis constant. l_f and l_r are the distances from the car's centroid to the front and rear, respectively.

Second, the observation model captures how measurements supplied by the LiDAR are produced, based on the heading of the car relative to the walls and its position in the hallway. The behavior of the LiDAR at turns is different from the straight sections of the hallway. To accurately capture the dynamics of measurements, the hallway is therefore divided into three regions and a measurement model of LiDAR is provided for each region, as shown in Fig. 7. The measurement model of the LiDAR scan with 1081 rays for Region 2 (other regions are special case of region 2) is as described by the following equations:

$$y_k^i = \begin{cases} d_k^r / cos(90 + \theta_k + \alpha_i) & \text{if } \theta_k + \alpha_i \leq \theta_r \\ d_k^b / cos(180 + \theta_k + \alpha_i) & \text{if } \theta_r < \theta_k + \alpha_i \leq -90 \\ d_k^t / cos(\theta_k + \alpha_i) & \text{if } -90 < \theta_k + \alpha_i \leq \theta_l \\ d_k^l / cos(90 - \theta_k - \alpha_i) & \text{if } \theta_l < \theta_k + \alpha_i, \end{cases} \tag{2}$$

where k is the sampling step, $d_k^t, d_k^b, d_k^l, d_k^r$ are distances to the four walls. $\alpha_1, .., \alpha_{1081}$ are the relative angles for rays in the LiDAR scan with respect to the car's heading. θ_l and θ_r are the relative angles to the two corners of the turn.

The third component that closes the loop is the NN-controller used in S_1 to control the heading of the car.

Existing Techniques that can be Used to Provide Assurance About the Model of CPSNN/NN in S_1 Composed with the NN in the System. The NN used as a controller in S_1 is well suited for the closed-loop verification of NN-controlled systems, supported by recent tools [21,32]. Verisig [21] is used

here to obtain the system-level evidence in the holistic pattern. Verisig operates directly on the NNs without approximating it some other function. This allows us to compose the model of CPSNN/NN in S_1 with its NN-controller and use Verisig to verify this composition for the safety specification of S_1.

Construction of Assurance Case for S_1 Based on Holistic Pattern. The assurance case for the safety specification of S_1 based on the holistic pattern is shown in Fig. 8. The top-level assurance claim, $G1$, states that "$\forall i \in I$, $\forall t \in T_i$, the distance of the car from the hallway walls in S_1 is always greater than zero". Here, 'I' is a set of initialization positions of the car in the hallway and for some $i \in I$, 'T_i' is a set of discretized time instants spent by the car on its trajectory starting from i. $G1$ is assured on the model of the CPSNN/NN in S_1 composed with the NN-controller in the system via claim $G2$ in the assurance case. This model-based assurance approach for S_1 requires an additional claim, $G3$, about the validity of the model used to represent the CPSNN/NN in S_1. Both $G2$ and $G3$ together imply the top-level safety claim $G1$ in this case.

$G2$ states that "$\forall i \in I$, $\forall t \in T_i$, the distance of the car from the hallway walls in the model of CPSNN/NN in S_1 composed with the NN-controller in the system is always greater than zero". It is a reachability property that can be checked by a verification tool. Thus, this branch of the argument follows the holistic approach to generate evidence for $G2$, which comes from the verification result obtained by Verisig as shown in Fig. 8.

The argument used for the assurance of the model validation claim, $G3$, is about the choice of the individual components that make the model of CPSNN/NN in S_1. Sub-claims $G4$ and $G5$ about the accuracy of the observation and the dynamics model, respectively, together provide assurance about the accuracy of the model of the CPSNN/NN in S_1. Characteristics of the F1/10 car in S_1 (it is a front steering car and its speed lies in [0, 5] m/s) makes it a suitable candidate for the kinematics bicycle model. The observation model is based on the ideal (noiseless) LiDAR operation in the known geometry of the hallway. Validating observation models is challenging in general due to the complex and uncertain behavior of the environment. This is one of the main directions for future work, as discussed in the discussion section.

4.2 Case Study for the Modular Pattern

System Description and the Model of Its CPSNN/NN Composed with the NN in the System. We consider the CPS with a perception-based NN from [14] for the case study of the modular pattern. This system, S_2, is shown in Fig. 9. It consists of an autonomous car (ego vehicle) driving on a highway through a desert with a stationary car in front of it. The vehicle is equipped with an automatic emergency braking system (AEBS) for avoiding collisions with preceding obstacles. The AEBS uses a perception-based NN and radar to get information about the preceding obstacles. It relies on the radar for obstacles at a distance less than or equal to 30 m from the vehicle. For obstacles farther than 30 m from the vehicle, the AEBS relies solely on the NN for detection. In

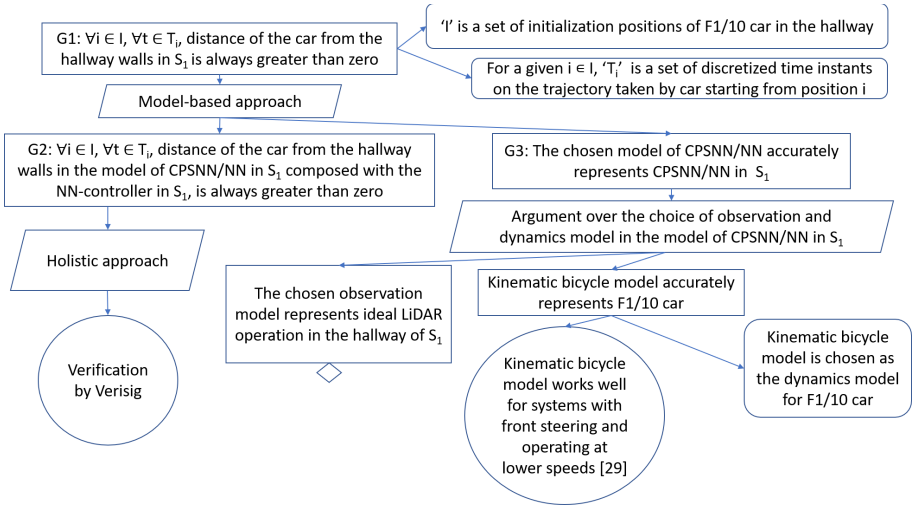

Fig. 8. Assurance case for the safety specification of S_1

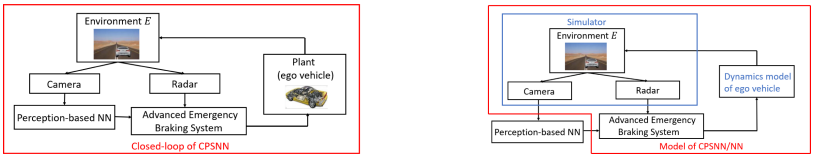

Fig. 9. Closed-loop system, S_2, of the CPSNN considered for the case study of the modular pattern (left) and the corresponding model of its CPSNN/NN composed with the perception-based NN in the system (right)

the event that an obstacle is detected, the AEBS issues a full braking command to the car. The input to the NN is provided by a camera that generates RGB images of size 1000×600. The safety property of interest for this system is that no collision happens between the vehicle and the stationary car.

A general model of the CPSNN/NN in S_2 composed with the perception-based NN as shown in Fig. 9 is described as follows. It contains four components. First, the dynamics model represents the behavior of the ego vehicle on the highway. Second, a simulator captures how observations of the environment supplied by the camera and radar are produced, based on the position of the ego vehicle. Third, the AEBS algorithm used as it is. The last component is the perception-based NN in S_2 that closes the loop in the system.

Existing Techniques that can be Used to Provide Assurance About S_2. Since the input dimension for the NN in S_2 is very large, verifying S_2 for its specification is challenging. Most of the existing tools for the assurance of CPSNN with perception-based NN are based on falsification techniques [14,33]. We consider the falsification tool developed for the perception-based NN in [14]

to generate the NN-specific evidence in the modular pattern for the assurance of S_2. This tool approximates the NN to a lower-dimensional input function $\tilde{f}$ and finds falsifying examples for the NN from this lower-dimensional input space. The idea is to explore only realistic modifications in the input space of the NN, instead of exploring the high-dimensional input space for finding falsifying examples for the NN. The input space of $\tilde{f}$ is analyzed to find misclassifications by $\tilde{f}$. These misclassifications are then concretized back into the input images for NN to check for misclassifications by the NN.

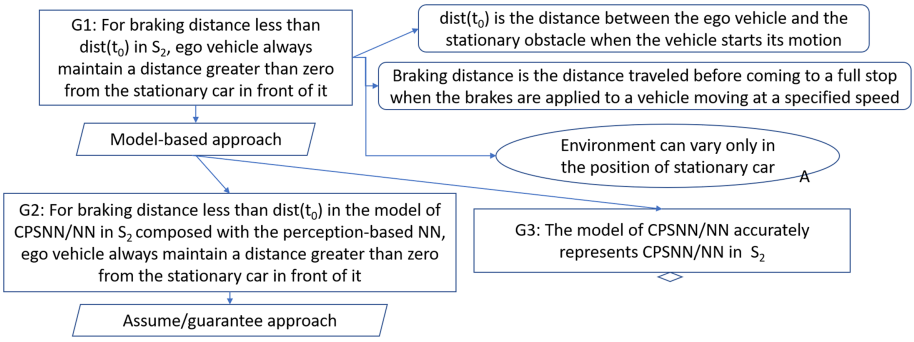

Fig. 10. Model-based approach for the assurance of S_2

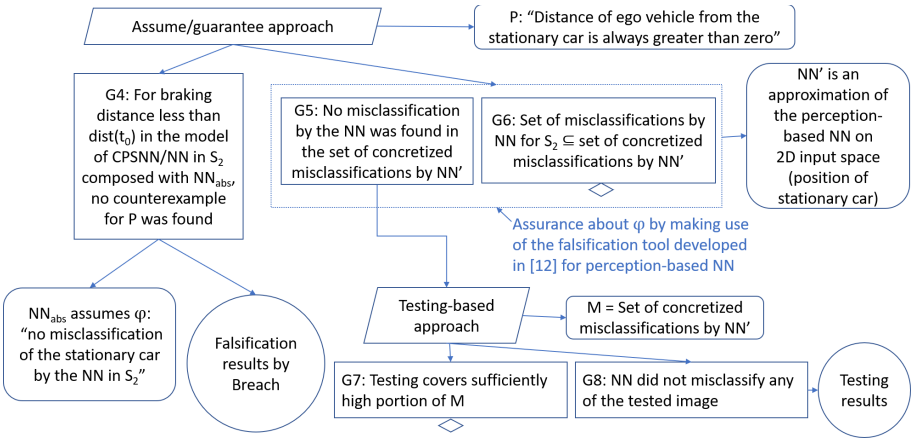

Fig. 11. Assurance of the claim $G2$ for S_2 by assume/guarantee approach

Construction of Assurance Case for S_2 Based on the Modular Pattern. The assurance case for the safety specification of S_2 based on the modular pattern is described as follows. The safety claim, $G1$, about S_2 is that "if the initialization distance between the two cars is greater than the ego vehicle's braking

distance, then the vehicle always maintains a distance greater than zero from the stationary car". We assume that the environment of S_2 can vary only in the 2-dimensional position of the stationary car. As shown in Fig. 10, $G1$ is assured on the model of the CPSNN/NN in S_2 composed with the perception-based NN in the system, via claim $G2$ in the assurance case. The claim $G3$ about the accuracy of the model of CPSNN/NN in S_2, required in addition to the claim $G2$ for completing the model-based argument for the assurance of $G1$ is marked as an undeveloped claim here. It can be developed by making arguments about the choice of the individual components composing the model of CPSNN/NN in S_2 as done for the justification of the model validation claim for assurance of S_1.

Figure 11 shows the assume/guarantee assurance approach for the claim $G2$ about the model of the CPSNN/NN in S_2 composed with the perception-based NN. An abstraction of the NN, NN_{abs}, satisfying the property φ of no misclassifications of the stationary car by the network, is considered here. NN_{abs} is composed with the model of the CPSNN/NN in S_2 to provide assurance about $G2$ via falsification claim $G4$ in the assurance case. The evidence for $G4$ is generated by Breach, a falsification tool for traditional CPS.

Since $G2$ is assured on the model of CPSNN/NN composed with an abstraction of the NN satisfying φ, an additional claim about the guaranteed satisfaction of φ by the NN is required to be justified to complete the assume/guarantee argument for the assurance of $G2$. Claims $G5$ and $G6$ together provide this guarantee by making use of the falsification tool for the perception-based NNs from [14]. The NN is approximated by a 2-dimensional input function $\tilde{f}$. The input to $\tilde{f}$ is the position of the stationary car in the 2D plane.

$G5$ states that the NN does not misclassify any of the concretized images from the set S of misclassification by $\tilde{f}$. The testing-based approach is used to sample the input space of $\tilde{f}$ for finding S. A claim supported by testing argument is only as good as the coverage by testing. So, we need to argue about the sufficiency of the sampling method used to cover the input set of $\tilde{f}$, via claim $G7$ in this case. The sufficiency of the sampling methods used to generate inputs for testing either in terms of high coverage or coverage of the corner cases could be used to generate evidence for $G7$. In addition to $G7$, a claim about no misclassifications by NN on concretized images of S is required to complete the argument for the assurance of $G5$. This is done via claim $G8$ supported by testing results.

Since $G5$ is the assurance of φ on the set S of misclassifications by $\tilde{f}$, an additional claim about the set of misclassifications by the NN as a subset of concretized S needs to be made. This is done via undeveloped claim $G6$ in the assurance case. $G6$ reflects the safety of the approximation function $\tilde{f}$ for the NN. By safe approximation in the context of falsification, we mean that any element in the input space of $\tilde{f}$ lying outside its set of misclassifications is not a concretized misclassification by NN. We propose the development of techniques that can be used to generate evidence about this claim as it will enable the reduction of misclassification space of the NN and enhance the use of falsification for providing assurance about systems with high-dimensional input space of NN.

5 Discussion and Conclusion

In this paper, we proposed two model-based assurance case patterns for the safety specifications of CPSNN. These patterns are amenable to the existing tools for generating evidence for the assurance of CPSNN. We used two case studies, one for each pattern, to illustrate the applicability of the proposed patterns. We note that the two case studies are no more than illustrations and our goal is not to try and convince the reader that these systems are, in fact, safe. Therefore, we did not try to fully elaborate the arguments, nor tried to uncover assurance deficits in each of the systems. Instead, we aimed to consider development and analysis technologies available in the literature to see how suitable they would be to supply evidence for arguments following each of the patterns. These patterns are designed to help expose challenges involved with the evidence generated from the existing tools. The undeveloped claims in the case studies, for instance, attempt to formulate these challenges as requirements of new (or enhancement of existing) tools for the system-level assurance of the CPSNN. This is different from the approach proposed in a recent work [4] to identify component-level gaps in the assurance of the CPSNN.

One of the main challenges in the model-based assurance argument that needs to be addressed is the assurance about the accuracy of the observation (or simulation) model used to capture measurements (or observations) in the environment. The complex and uncertain nature of the environment makes it difficult to precisely capture it in a model. Improving the scalability of the existing system-level verification tools for CPSNN is another challenge that needs to be addressed to strengthen the evidence for the holistic pattern. Formalization of requirements for the perception-based NNs is required to help develop tools for the assurance of these NNs in the modular pattern. Addressing these challenges form one of the directions of future research for us. Another promising direction to extend this work is to extend the proposed design-time patterns with arguments to include safety monitoring and runtime adaptation tools for assurance.

References

1. Adelard: ASCAD - the Adelard Safety Case Development (ASCAD) Manual (1998)
2. F1tenth. http://f1tenth.org/
3. Ayoub, A., Kim, B.G., Lee, I., Sokolsky, O.: A safety case pattern for model-based development approach. In: Goodloe, A.E., Person, S. (eds.) NFM 2012. LNCS, vol. 7226, pp. 141–146. Springer, Heidelberg (2012). https://doi.org/10.1007/978-3-642-28891-3_14
4. Bloomfield, R., Khlaaf, H., Conmy, P.R., Fletcher, G.: Disruptive innovations and disruptive assurance: assuring machine learning and autonomy. Computer **52**(9), 82–89 (2019)
5. Bojarski, M., et al.: End to end learning for self-driving cars. arXiv preprint arXiv:1604.07316 (2016)

6. Boopathy, A., Weng, T.W., Chen, P.Y., Liu, S., Daniel, L.: CNN-Cert: an efficient framework for certifying robustness of convolutional neural networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 33, pp. 3240–3247 (2019)
7. Burton, S., Gauerhof, L., Heinzemann, C.: Making the case for safety of machine learning in highly automated driving. In: Tonetta, S., Schoitsch, E., Bitsch, F. (eds.) SAFECOMP 2017. LNCS, vol. 10489, pp. 5–16. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-66284-8_1
8. Chen, X., Ábrahám, E., Sankaranarayanan, S.: Flow*: an analyzer for non-linear hybrid systems. In: Sharygina, N., Veith, H. (eds.) CAV 2013. LNCS, vol. 8044, pp. 258–263. Springer, Heidelberg (2013). https://doi.org/10.1007/978-3-642-39799-8_18
9. Chen, Y., Lawford, M., Wang, H., Wassyng, A.: Insulin pump software certification. In: Gibbons, J., MacCaull, W. (eds.) FHIES 2013. LNCS, vol. 8315, pp. 87–106. Springer, Heidelberg (2014). https://doi.org/10.1007/978-3-642-53956-5_7
10. De Fauw, J., et al.: Clinically applicable deep learning for diagnosis and referral in retinal disease. Nat. Med. **24**(9), 1342–1350 (2018)
11. Denney, E., Pai, G.: Safety considerations for UAS ground-based detect and avoid. In: 2016 IEEE/AIAA 35th Digital Avionics Systems Conference, pp. 1–10 (2016)
12. Denney, E., Pai, G., Habli, I.: Towards measurement of confidence in safety cases. In: 2011 International Symposium on Empirical Software Engineering and Measurement, pp. 380–383. IEEE (2011)
13. Donzé, A.: Breach, a toolbox for verification and parameter synthesis of hybrid systems. In: Touili, T., Cook, B., Jackson, P. (eds.) CAV 2010. LNCS, vol. 6174, pp. 167–170. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-14295-6_17
14. Dreossi, T., Donzé, A., Seshia, S.A.: Compositional falsification of cyber-physical systems with machine learning components. In: Barrett, C., Davies, M., Kahsai, T. (eds.) NFM 2017. LNCS, vol. 10227, pp. 357–372. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-57288-8_26
15. Dutta, S., Chen, X., Jha, S., Sankaranarayanan, S., Tiwari, A.: Sherlock-a tool for verification of neural network feedback systems: demo abstract. In: Proceedings of the 22nd ACM International Conference on Hybrid Systems: Computation and Control, pp. 262–263 (2019)
16. Fainekos, G.E., Sankaranarayanan, S., Ueda, K., Yazarel, H.: Verification of automotive control applications using S-TaLiRo. In: 2012 American Control Conference (ACC), pp. 3567–3572. IEEE (2012)
17. Fazlyab, M., Robey, A., Hassani, H., Morari, M., Pappas, G.: Efficient and accurate estimation of Lipschitz constants for deep neural networks. In: Advances in Neural Information Processing Systems, pp. 11423–11434 (2019)
18. Group, A.C.W., et al.: Goal structuring notation community standard (2018)
19. Huang, X., Kwiatkowska, M., Wang, S., Wu, M.: Safety verification of deep neural networks. In: Majumdar, R., Kunčak, V. (eds.) CAV 2017. LNCS, vol. 10426, pp. 3–29. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-63387-9_1
20. Ivanov, R., Carpenter, T.J., Weimer, J., Alur, R., Pappas, G.J., Lee, I.: Case study: verifying the safety of an autonomous racing car with a neural network controller. arXiv preprint arXiv:1910.11309 (2019)
21. Ivanov, R., Weimer, J., Alur, R., Pappas, G.J., Lee, I.: Verisig: verifying safety properties of hybrid systems with neural network controllers. In: Proceedings of the 22nd ACM International Conference on Hybrid Systems: Computation and Control, pp. 169–178. ACM (2019)

22. Julian, K.D., Kochenderfer, M.J.: Neural network guidance for UAVs. In: AIAA Guidance, Navigation, and Control Conference, p. 1743 (2017)
23. Katz, G., Barrett, C., Dill, D.L., Julian, K., Kochenderfer, M.J.: Reluplex: an efficient SMT solver for verifying deep neural networks. In: Majumdar, R., Kunčak, V. (eds.) CAV 2017. LNCS, vol. 10426, pp. 97–117. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-63387-9_5
24. Ko, C.Y., Lyu, Z., Weng, T.W., Daniel, L., Wong, N., Lin, D.: POPQORN: quantifying robustness of recurrent neural networks. arXiv preprint:1905.07387 (2019)
25. Kurd, Z., Kelly, T., Austin, J.: Developing artificial neural networks for safety critical systems. Neural Comput. Appl. **16**(1), 11–19 (2007)
26. Lin, C.L., Shen, W.: Applying safety case pattern to generate assurance cases for safety-critical systems. In: 2015 IEEE 16th International Symposium on High Assurance Systems Engineering, pp. 255–262. IEEE (2015)
27. Nicolescu, G., Mosterman, P.J.: Model-Based Design for Embedded Systems. CRC Press, Boca Raton (2009)
28. Picardi, C., Hawkins, R., Paterson, C., Habli, I.: A pattern for arguing the assurance of machine learning in medical diagnosis systems. In: Romanovsky, A., Troubitsyna, E., Bitsch, F. (eds.) SAFECOMP 2019. LNCS, vol. 11698, pp. 165–179. Springer, Cham (2019). https://doi.org/10.1007/978-3-030-26601-1_12
29. Polack, P., Altché, F., d'Andréa Novel, B., de La Fortelle, A.: The kinematic bicycle model: a consistent model for planning feasible trajectories for autonomous vehicles? In: Intelligent Vehicles Symposium (IV), pp. 812–818. IEEE (2017)
30. Rushby, J.: The interpretation and evaluation of assurance cases. Comp. Science Laboratory, SRI International, Technical report, SRI-CSL-15-01 (2015)
31. Taeyoung, L., Kyongsu, Y., Jangseop, K., Jaewan, L.: Development and evaluations of advanced emergency braking system algorithm for the commercial vehicle. In: Enhanced Safety of Vehicles Conference, ESV, pp. 11–0290 (2011)
32. Tran, H.D., Cai, F., Diego, M.L., Musau, P., Johnson, T.T., Koutsoukos, X.: Safety verification of cyber-physical systems with reinforcement learning control. ACM Trans. Embed. Comput. Syst. (TECS) **18**(5s), 1–22 (2019)
33. Tuncali, C.E., Fainekos, G., Ito, H., Kapinski, J.: Simulation-based adversarial test generation for autonomous vehicles with machine learning components. In: 2018 IEEE Intelligent Vehicles Symposium (IV), pp. 1555–1562. IEEE (2018)
34. Wang, Y.S., Weng, T.W., Daniel, L.: Verification of neural network control policy under persistent adversarial perturbation. arXiv preprint arXiv:1908.06353 (2019)
35. Weimer, J., Sokolsky, O., Bezzo, N., Lee, I.: Towards assurance cases for resilient control systems. In: 2014 IEEE International Conference on Cyber-Physical Systems, Networks, and Applications, pp. 1–6. IEEE (2014)