Safety-Assured Design and Adaptation of Learning-Enabled Autonomous Systems

Qi Zhu, Chao Huang, Ruochen Jiao, Shuyue Lan, Hengyi Liang, Xiangguo Liu,
Yixuan Wang, Zhilu Wang, Shichao Xu
Department of Electrical and Computer Engineering
Northwestern University
Evanston, Illinois, U.S.A.

ABSTRACT

Future autonomous systems will employ sophisticated machine learning techniques for the sensing and perception of the surroundings and the making corresponding decisions for planning, control, and other actions. They often operate in highly dynamic, uncertain and challenging environment, and need to meet stringent timing, resource, and mission requirements. In particular, it is critical and yet very challenging to ensure the safety of these autonomous systems, given the uncertainties of the system inputs, the constant disturbances on the system operations, and the lack of analyzability for many machine learning methods (particularly those based on neural networks). In this paper, we will discuss some of these challenges, and present our work in developing automated, quantitative, and formalized methods and tools for ensuring the safety of autonomous systems in their design and during their runtime adaptation. We argue that it is essential to take a *holistic* approach in addressing system safety and other safety-related properties, *vertically across* the functional, software, and hardware layers, and *horizontally across* the autonomy pipeline of sensing, perception, planning, and control modules. This approach could be further extended from a single autonomous system to a multi-agent system where multiple autonomous agents perform tasks in a collaborative manner. We will use connected and autonomous vehicles (CAVs) as the main application domain to illustrate the importance of such holistic approach and show our initial efforts in this direction.

CCS CONCEPTS

• **Computer systems organization → Embedded and cyber-physical systems**.

1 INTRODUCTION

Learning-enabled autonomous systems, such as self-driving vehicles and industrial robots, promise huge societal and economic benefits and have shown promising results in early adoptions. At the heart of these systems is their ability to sense and perceive the dynamic environment, reason about the situation considering various factors and uncertainties, and make decisions accordingly for system planning, control and other functions. They increasingly rely on the application of sophisticated machine learning techniques across the autonomy pipeline of sensing, perception, planning, control, and general decision making.

However, there are significant technical challenges for the design and runtime operation of these learning-enabled autonomous systems, hindering their wider adoptions. In particular, as many of these systems are safety-critical, it is essential and yet challenging to ensure the correctness of their safety-related functions, especially given the uncertainties of dynamic surrounding environment and changing inputs, the lack of analyzability for many machine learning techniques (particularly the ones based on deep neural networks), and the constant disturbances to system operations from environment interference, software and hardware faults, and malicious attacks [109].

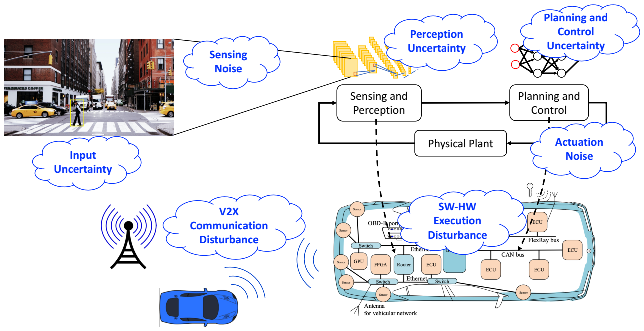

Figure 1: Illustration of uncertainties and disturbances in connected and autonomous vehicles.

As an example, Fig. 1 shows the various sources of uncertainties and disturbances in the operation of connected and autonomous vehicles (CAVs), including inherent uncertainties from the environment input, noises in sensing devices, uncertainties in the perception, planning, and control algorithms (where neural networks are used widely for the former and increasingly for the latter two), noises in the actuation devices, disturbances to the software-hardware execution of the algorithms, and disturbances to the

The authors are with the Department of Electrical and Computer Engineering, Northwestern University, 2145 Sheridan Road, Evanston, IL 60208-3118, U.S.A. Emails: {qzhu, chao.huang}@northwestern.edu, {RuochenJiao2024, ShuyueLan2018, HengyiLiang2018, XiangguoLiu2023, YixuanWang2024, ZhiluWang2018, ShichaoXu2023}@u.northwestern.edu. We gratefully acknowledge the support from National Science Foundation (NSF) grants 1834701, 1834324, 1839511, 1724341, and Office of Naval Research (ONR) grant N00014-19-1-2496.

ASPDAC '21, January 18–21, 2021, Tokyo, Japan
© 2021 Association for Computing Machinery.
ACM ISBN 978-1-4503-7999-1/21/01…$15.00
https://doi.org/10.1145/3394885.3431623

communication among vehicles and their surrounding infrastructures, i.e., the V2X (vehicle-to-vehicle, vehicle-to-infrastructure, etc.) communication. These uncertainties and disturbances could lead to unexpected scenarios and cause disastrous system failures, as evidenced by several high-profile accidents in practice. To address them and ensure system safety, a new set of design methods and tools is greatly needed.

In this paper, we argue that it is important to take a *holistic* approach in addressing system safety and other related properties for learning-enabled autonomous systems, across the different system layers vertically and the various modules in the autonomy pipeline horizontally. Vertically-speaking, we need to analyze how system safety may be affected by the algorithm design (e.g., design of neural networks) at the functional layer *and* by their implementations at the software and hardware layers (e.g., considering whether computation and communication may be carried out within deadlines, how functional correctness may be affected by transient faults, and in general whether the assumptions made at the functional layer can be preserved at the software and hardware layers). Horizontally-speaking, we need to analyze system safety in an end-to-end and closed-loop manner, i.e., from sensing to actuation and with the consideration of system plant. For instance, when considering the impact of sensing noises or adversarial attacks on sensing input, we should not stop at the sensing and perception module itself, but need to analyze the end-to-end effect on safety across sensing, perception, planning and control modules.

Enabling such holistic view for the design and operation of learning-enable autonomous systems requires developing new quantitative and formalized methods for individual layers and modules, *and* requires new formalization and methodologies for cross-layer and inter-module analysis, design and adaptation. Recently, we have made some progresses in developing safety-assured design and adaptation methods for individual modules and across system layers. In the rest of the paper, we will introduce these initial efforts from us, and also discuss the general challenges in designing safety-assured autonomous systems and other state-of-the-art methods. Section 2 focuses on methods for sensing and perception, Section 3 focuses on planning and control, Section 4 addresses multi-agent collaboration, and Section 5 concludes the paper.

2 SENSING AND PERCEPTION

In autonomous systems, many of the sensing and perception tasks are safety-critical and time-critical. For example, imagine an autonomous vehicle is driving and a pedestrian suddenly runs across the road, as shown in Fig. 2. To avoid an accident and ensure system safety, it is vital for the vehicle to correctly detect the pedestrian within a short time. In this section, we will discuss some of the techniques for ensuring the *functional safety* and *timing safety* of sensing and perception in learning-enable autonomous systems, as summerized in the figure.

2.1 Functional Safety

Machine learning techniques, especially neural network based ones, have demonstrated significant improvements on sensing and perception over traditional methods. However, the impact of these methods on system safety has not been sufficiently studied, which impedes their further adoption in safety-critical systems. For instance, one major issue that has been raised recently is that neural networks may misclassify images with small but intentional perturbations, known as adversarial attacks [2, 27, 82]. To achieve better robustness against adversarial attacks and ensure system safety, we will need methods for 1) quantitatively evaluating the robustness of neural networks under input perturbations, and 2) designing neural networks with more robust structures.

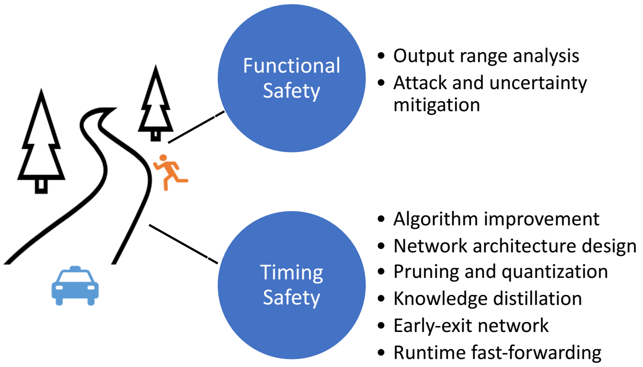

Figure 2: Functional and timing safety for the sensing and perception in learning-enabled autonomous systems.

2.1.1 Output Range Analysis for Neural Network Robustness. Output range analysis techniques could provide a quantitative bound for the output of a neural network when given an input space. More formally, it solves the following problem: given a neural network f and the input range $\mathcal{X}$, compute the output range of $f(\mathcal{X})$ or its overapproximation $\overline{\mathcal{Y}}$ such that $f(\mathcal{X}) \subseteq \overline{\mathcal{Y}}$. Such overapproximation can provide an explicit bound for determining whether the neural network output falls into an unwanted region. In the context of adversarial robustness, for a data point x with the concerned error bound ϵ, we can define the input space as the small L_∞-norm box $[x-\epsilon, x+\epsilon]$ and estimate the output range as $[l_1, \bar{l}_1] \times \cdots \times [l_g, \bar{l}_g] \cdots \times [l_n, \bar{l}_n]$ for n labels, where l_g ($\bar{l}_g$) denotes the lower (upper) bound of the output in terms of the ground truth. If $l_g \geq \bar{l}_i$, for $i = 1, \cdots, n$, we can safely say that the neural network is *locally robust* on x with respect to the perturbation ϵ.

State-of-the-art methods for output range analysis include symbolic interval propagation (SIP) and constraint programming (CP). SIP is developed from the earlier work of interval bound propagation (IBP). It denotes the range of a neuron with a symbolic representation, and propagates it layer-by-layer. For instance, ERAN uses symbolic zonotopes [79], while NNV adopts symbolic image-star representation [88, 89]. When handling nonlinear operations, symbolic intervals have to be concretized to range intervals and lose the dependencies between dimensions. CP methods encode the neural network as a constraint system and compute the output range with constraint programming techniques, such as mixed integer linear programming (MILP) [9, 16, 24, 66, 87] and semi-definite programming (SDP) relaxation [22, 71]. The main challenge for CP methods lies on the complexity. It needs to solve a large nonlinear programming problem encoding the entire network and thus suffers from the curse of dimensionality.

In [42], we present a new approach that leverages both SIP and CP. We first compute the interval relaxation for each operation with a propagation-based method (IBP or SIP) as the initialization step. Based on the initial range, we use a linear programming (LP) relaxation approach to better approximate the variable range. Then, the relaxation can be further tightened via MILP encoding, i.e., by dividing the input range for each operation into multiple segments. Our approach can iteratively improve the approximation precision by increasing the number of segments in the MILP encoding. In addition, we refine the variable range such that fewer integer variables are needed to achieve a similar approximation precision. Experiments on a number of common dataset demonstrate significant improvement of our approach over other state-of-the-art methods such as ERAN and NNV.

2.1.2 Attack and Uncertainty Mitigation. Knowing the local robustness of neural networks under input perturbations could help us design more robust networks or retrain existing ones to mitigate the impact of adversarial attacks. Note that beyond intentional input perturbations, there could also be inherent data and model uncertainties to the neural networks. The data uncertainties may be caused by random input noises or inherent randomness of the real data, while model uncertainties result from a lack of training data in certain areas of the input domain (i.e., test example is out of distribution). To estimate these two, we may use Assumed Density Filtering (ABF) for data uncertainties and quantile regression or Monte Carlo methods [25, 67] for model uncertainties. However, estimating and bounding these uncertainties is generally a very difficult task, given the large number of scenarios that may occur during operation.

Designing mitigation strategies against uncertainties and adversarial attacks is therefore a very challenging task. One promising direction is to conduct end-to-end analysis on how such uncertainties and attacks eventually affect safety at the system level and design the mitigation strategies accordingly.

2.2 Timing Safety

Besides functional safety, it is equally important to ensure that the system functionality can be carried out in time. In the case of sensing and perception modules, this means that their implementations at the software and hardware layers should be sufficiently efficient for meeting the timing constraints at real-time operation. Achieving both accuracy and efficiency could be challenging however, given the complexity of most neural network based approaches and the often limited resources on many autonomous systems. In the literature, there has been an extensive body of work addressing the efficiency of neural networks from different aspects. We only discuss a very small subset of them below to illustrate some widely-explored directions.

Algorithm improvement. Sometimes more efficient algorithms could be designed for the perception tasks. For instance, object detection [8, 95] and semantic instance segmentation [13, 103] are part of the main components in many autonomous systems. The algorithms for these perception tasks can be classified into two groups, the anchor-based methods and the anchor-free methods. Anchor-based method generate a large number of anchors where objects may exist on the image, and then classification, localization or segmentation may be performed based on the anchors. The algorithms include one-stage detectors such as SSD [61] and RetinaNet [59], two-stage detectors such as faster R-CNN [75] and R-FCN [12], and instance segmentation models [31, 47, 96]. However, a large amount of time in anchor-based methods is consumed in anchor generation and selection. Thus, anchor-free methods are proposed, which use predefined or learned object key-points/center-points rather than anchors for better generalization ability and faster detection speed. Such methods include YOLO v1 [74], CornerNet [51], FCOS [86], CenterMask [53], etc.

Network architecture design. One important way to improve efficiency is to explore various architecture designs for the neural networks. For instance, depth-wise separable convolution [10, 35, 76], group convolution [46, 91], and dilated convolution [100] have been proposed to reduce computation complexity. Many lightweight network architectures have been proposed in recent years, such as Shufflenet [102], MobileNet [35, 76], and EfficientNet [83]. Apart from hand-crafted network architectures, extensive research has also been conducted on automated neural architecture search (NAS) [11, 36], including for efficiency purpose.

Network pruning and quantization. The general idea of network pruning is to remove the "unimportant" part of neural networks to achieve more efficient designs. Such methods typically include 1) removing unimportant weight connections [14, 30, 52], 2) removing unimportant neurons [81], and 3) removing unimportant filters [32, 54]. The goal of network quantization is to reduce the number of bits for representing neural weights [1, 26, 101]. One special case in this topic is network binarization [60, 65, 73].

Knowledge distillation. The main idea of knowledge distillation is to train a smaller model to mimic a pre-trained, larger model or ensemble of models. This training process is sometimes referred to as "teacher-student", where the larger model is the teacher and the smaller one is the student [3, 34]. It has been applied to a variety of perception tasks for model compression, e.g., for object detection in [5, 55, 68, 90].

Early-exit networks. Deep neural networks (DNNs) often benefit from having a large number of layers, but sometimes many data points in classification tasks can be accurately classified with much less work. To improve efficiency, the idea of exiting early before the normal endpoint of the neural network has been proposed. In [69], convolutional layer features are used to assess the difficulty of input instances and conditionally activate the deeper layers of the network. In [84], additional side branch classifiers are added to a DNN to allow early exit with high confidence. Recently, the early-exit idea is applied to intermittent inference with non-uniformly compressed neural networks for energy harvesting devices [99].

Runtime fast-forwarding. Another idea for improving perception efficiency is to "fast-forward" through the unimportant frames, i.e., by skipping their processing. This is conceptually similar to the goal of data summarization but focuses more on runtime adaptation. In [49], we propose a video fast-forwarding framework called FFNet, to fast-forward a video stream on the fly based on reinforcement learning. The approach can significantly reduce the number of frames to be process (by more than 80% in the experiments) while effectively identifying the important ones (performing better than other summarization techniques in the literature). This approach has been further extended to multi-agent systems in [49] (more details in Section 4).

Note that for timing safety, it could be more important in safety-critical systems to ensure the *predictability* of timing behavior, i.e., timing in the worst cases, than the pure efficiency in the average sense. In [109], we provide more in-depth discussions on the concept of time-predictable DNNs.

3 PLANNING AND CONTROL

Planning, control, and general decision making for autonomous systems is another crucial element in ensuring the overall system safety. Given that these modules generate actuation signals that directly affect system state, it is essential to have quantitative and formalized methods for assessing their impact on system safety and designing them with safety assurance. In this section, we will start with discussing the formal definitions of control safety for autonomous systems, and then discuss safety under a single neural network based controller, under multiple model-based or neural network based controllers, and under timing uncertainties. Note that here we use "control" in a more general sense that may involve planning and other decision making functions.

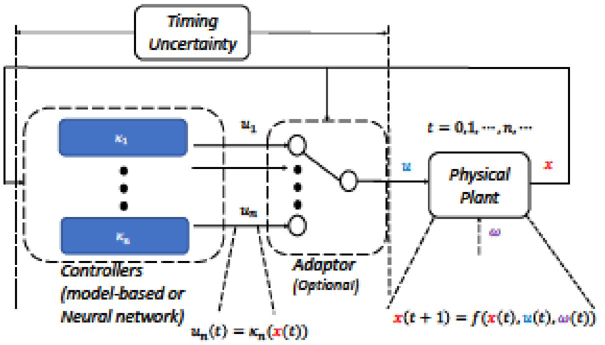

Figure 3: An illustration of control and adaptation in autonomous systems.

Fig. 3 shows an illustration of control and adaptation for autonomous systems, where multiple model-based or neural network based controllers may exist and an adaptor could be designed to switch among them. In terms of system safety, note that with the existence of external disturbances, the control module may not precisely follow the reference trajectory, and thus only ensuring safety of the reference trajectory does not guarantee the system safety. For model-based controllers that track the reference trajectory, system safety can be ensured by bounding the tracking error of the control before the trajectory planning. For instance in [18], the tracking error is bounded for trajectories that are modelled with the same ordinary differential equation (ODE) of the tracking control without considering external disturbance. This approach is extended in [19], where the trajectories are only required to be piece-wise continuous with respect to the ODE of the tracking control. As the planning models are usually coarser than the real system dynamics for efficiency purpose, the framework FasTrack [33, 80] proposes methods based on Hamilton-Jacobi (HJ) reachability and sum-of-squares (SOS) programming to bound the tracking error for such model-mismatched planning. Other works of safety-assured model-based control integrate the planning and control together, and leverage reachability analysis and robust invariant sets to guarantee the system safety, including methods such as barrier certificates [38, 70, 97], Taylor model [7, 28], and reachable-set based model predictive control (MPC) [39].

With the fast advancement of autonomous applications, new challenges have risen for ensuring control safety. First, due to the increasing application of machine learning techniques in control, analyzing system safety requires new methods for the cases of a neural network based controller or multiple switching controllers with some being neural network based. Then, complex decision making process may also bring timing uncertainties. In the rest of the section, we mainly focus on the safety verification of systems with neural network based controllers and with timing uncertainties. We start with a more formal definition of control safety (based on the concept of state reachability) that we consider.

Consider an autonomous system with state $x(t)$. The planning module is designed to build a reference trajectory $x_{ref}(t), t \in [0, T]$, which reaches $x_{ref}(T) \in X_f$ at time T. Here X_f is the target state space, with the constraint $x_{ref}(t) \notin O(t), \forall t \in [0, T]$, which requires avoiding any obstacles or unsafe range $O(t)$ at all times. The control module is designed to compute the control signal $u(t)$ to make the system follow the reference trajectory. The system dynamics can be modelled by an ODE $\dot{x} = f(x, u, w)$, where w is the external disturbance to the system. Formally, we can define the reachability and safety of the autonomous system as follows:

Definition 3.1 (Reachability). Given the initial state space X_0, time horizon T, target state space X_f and disturbance bound W, the system is reachable for the controller $u(t), t \in [0, T]$ if $\forall x(0) \in X_0, \forall w \in W, x(T) \in X_f$.

Definition 3.2 (Safety). Given the initial state space X_0, unsafe state space $O(t)$, and disturbance bound W, the system is safe for the controller $u(t)$ if $\forall x(0) \in X_0, \forall w \in W, x(t) \notin O(t)$.

Safety under a single neural network based controller. Neural network based methods are increasingly being used in planning and control as they may provide better performance than model-based methods and also do not require building complex, costly, and error-prone physical models. The safety of neural network controlled systems (NNCSs) has been recently considered in [15, 43, 48]. These works share the similar methodology that a tractable model is first used to overapproximate the neural network controller and then the safety of the new system with the approximated controller is verified. Due to the overapproximation, the safety of the new system is a sufficient condition for the safety of the original NNCS, and hence the approach is sound. The differences among different works reside on their choice of the approximation model and the resulting approximation error. In [48], a differentiable neural network is equivalently transformed into a hybrid system, and then the entire system is verified as a hybrid system. In [15], authors focus on ReLU networks and use piecewise polynomials for approximation. In our work [43], we utilize Bernstein polynomials for approximation and propose a sampling-based error estimation approach based on Lipschitz constant of the network. Due to the fact that Bernstein polynomials could be a universal approximator for continuous functions, our approach can handle most common types of neural networks, including Sigmoid, ReLU, tanh, and combinations of them. However, this approach is not sufficiently efficient for networks that require a large number of samplings or networks with a large Lipschitz constant. To improve the efficiency of our approach, in [20], we develop a knowledge distillation approach to retrain a network for reducing its Lipschitz constant while maintaining similar performance, such that the resulting NNCS is easier to verify. Then in [21], we improve the sampling efficiency via parallel computing and GPU acceleration.

Safety under multiple model-based or neural network based controllers. In practice, it is common to have multiple controllers available for the same control function. They could be designed by different teams or based on different design methodologies, including well-established model-based controllers and emerging neural network based ones. They could each have their own advantages and disadvantages, e.g., some are more robust but less efficient while others are the opposite. Thus, it could be beneficial to conduct runtime adaptation (switching) among the controllers based on the current system state and mission requirements, as shown in Fig. 3. For instance, in [77], rule-based switching logic among application controllers and a safety controller is proposed to enhance system performance and fault tolerance. In [72], multiple controllers are dynamically-weighted based on reinforcement learning (RL) to compute the system control input for improving performance. In our work [45, 92], RL-based switching approaches are developed with safety guarantees based on invariant set computation, which is essential for safety-critical systems.

Safety under timing uncertainties. As shown in Fig. 1, at the software and hardware layers, there could be disturbances to system operations (e.g., computation, communication, or storage operations) due to environment interference, transient faults, or malicious attacks. The effects of many of such disturbances manifest as *timing uncertainties*, where the system operations may miss their deadlines or fail entirely. Under traditional hard timing constraints for safety-critical systems, any deadline misses will be deemed as system failures. However, many system functions (e.g., control, planning and perception functions) have some inherent robustness and can sustain occasional and bounded deadline misses from disturbances, while still meeting their performance and safety requirements. To more accurately capture system behavior under timing uncertainties (execution disturbances) and formally analyze their properties, we propose to leverage the *weakly-hard* paradigm. One common type of weakly-hard formalization is the so-called (m, k) constraint, where a system operation (i.e., a task) is allowed to miss at most m times during any k consecutive activations. By allowing deadline misses through weakly-hard constraints, we could exploit such flexibility to facilitate system retrofitting and runtime adaptation, for improving system performance, safety, security, etc.

The first key to leverage weakly-hard constraints is to formally verify that system functions can still meet their requirements (e.g., safety requirements) under the allowed deadline misses. In [44], we propose an approach based on exponential stability for formally verifying the safety of control functions under weakly-hard constraints, and further extend it to more general systems in [37] by leveraging descretization techniques and graph theory. Our experiments demonstrate that often the safety of control functions can still be proven in the cases where a single deadline miss is always followed by several successful completions. It becomes much harder to prove the control safety when consecutive deadline misses are allowed to occur. This shows that beyond the frequency of deadline misses (or disturbances in general), the *temporal pattern* of these misses has a significant impact on system safety, which is precisely what weakly-hard constraints try to capture.

Knowing whether the system can safely sustain missed deadlines could enable proactive skipping of operations. In [45], we propose a safety-assured RL-based online adaptation framework to proactively skip control steps for saving actuation energy. The skipping decision is made according to the real-time system state, which significantly reduces the pessimism from the worst-case analysis during offline verification. The online adaptation framework is shown to be effective in leveraging the robustness of controllers for energy saving, while maintaining system safety. In [93], we formally analyze control stability under various deadline miss patterns, and explore control periods that can improve performance while maintaining stability under deadline misses. Note that shorter sampling periods typically lead to better control performance and stability, however on a resource-limited platform, the increased execution load may lead to deadline misses that are detrimental to performance and stability. Thus, our approach provides a quantitative analysis of the tradeoff between the two factors, and helps identifying the control period that has the optimal performance with guaranteed stability.

We can also leverage weakly-hard paradigm at the design time to help system retrofitting. Our work in [58] analyzes how much deadline misses certain control functions can accommodate under safety requirements, and extracts those scheduling slack to facilitate adding monitoring tasks for improving system security. In [57], we develop an approach for analyzing control stability under deadline misses and leveraging the obtained slack to apply fault-tolerance techniques for transient errors, including explicit output comparison (EOC) and embedded error detection (EED) techniques [104].

4 MULTI-AGENT COLLABORATION

In some application domains, autonomous systems may collaborate with each other to improve system safety and performance. However, such collaboration often faces its own safety challenges. In this section, we will discuss these aspects.

Some well-known traffic accidents involving autonomous vehicles clearly demonstrate the limited capability of a single autonomous vehicle. For example, in March 2017 at Tempe, Arizona, an Uber autonomous vehicle going straight collided with a left-turn vehicle in the opposing direction [29], as shown in Fig. 4. This accident could be avoided if the autonomous vehicle had out-of-sight situational awareness or traffic signal phase plan. Many researchers thus argue the importance and necessity of collaboration between autonomous vehicles and Road Side Units (RSUs), which can extend perception range, improve detection accuracy, and lead to more efficient planning and control.

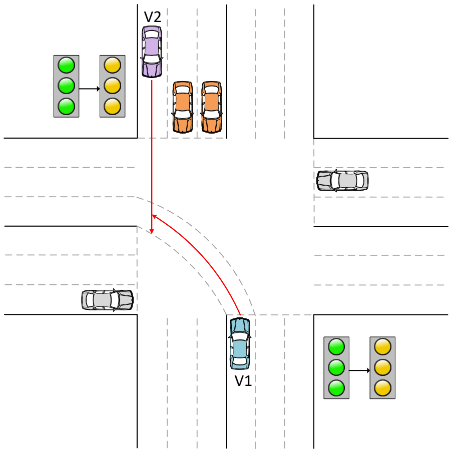

Figure 4: Illustration of the Uber accident. When the traffic signal turns yellow from green, two orange cars make a complete stop; while the autonomous vehicle V2 intends to get through the intersection before the traffic signal turns red as it cannot stop before the intersection. The view of the left-turn vehicle V1 is blocked by the two orange vehicles and it enters the intersection when the signal is yellow. Then V2 and V1 collide in the intersection.

Based on connectivity provided by vehicular ad-hoc networks, many connected vehicle applications have been proposed, such as Cooperative Adaptive Cruise Control (CACC) [94], Autonomous Intersection Management (AIM) [17, 78], and so on. In these applications, a group of connected vehicles collaboratively make decisions to improve traffic safety and efficiency, and can be viewed together as a multi-agent system. There is a rich literature on algorithm design at the functional layer for multi-agent systems. In our work [50], as mentioned in Section 2, we propose an adaptive and consensus-based framework, named DMVF, for distributed multi-agent video fast-forwarding in real-time. In DMVF, multiple stationary or mobile cameras communicate with each other and dynamically decide what fast-forwarding paces to use. DMVF offers high coverage, low processing rate, and small communication overhead, and can be applied to a variety of resource-limited multi-agent systems to balance performance, safety, and efficiency. In [40, 41], we propose a hierarchical MPC framework to efficiently control multi-agent systems with safety guarantees for linear and switched linear dynamical systems, respectively. In [63, 64], we focus on CAVs in particular and leverage the Markov decision process (MDP) model to capture network-level traffic information, which is then combined with a locally-optimal motion planner to improve long-term travel efficiency in mixed traffic stream. Our simulations demonstrate statistically significant efficiency for the subject vehicle and its surrounding vehicles in different traffic states.

While those designs based on multi-agent collaboration provide many benefits, significant challenges arise when considering the potential disturbances to V2X communication from environment interference, device faults, or malicious attacks. For instance, the work in [6] analyzes the vulnerability of intelligent traffic signal that collects Basic Safety Messages (BSMs) from vehicles for signal phase planning. It demonstrates that spoofing attacks from a single attacker can lead to a significantly worse system performance when compared with the case of no connectivity. The work in [98] shows that, in dense traffic conditions, the V2X communication delay can be as much as several hundred milliseconds, which may fail to meet the requirements of safety-critical applications. The work in [85] studies the string stability of CACC platoon under communication delays and shows that the string stability cannot be guaranteed when the communication delay exceeds an upper bound.

Our work [56, 105–108] conducted in-depth studies of CACC and AIM applications under communication delays and packet losses, and proposed delay-aware safety-assured designs. For instance, our recent work in [106] proposes a communication delay-tolerant protocol for general multi-lane intersection management. We model the proposed protocol as a finite state machine and define timeouts in the protocol to address communication delays and packet losses. We formally prove that our protocol is safe, deadlock free, and satisfies the liveness property. We also demonstrate that it outperforms the various designs of smart traffic signals based on simulations in an extended traffic simulation suite SUMO.

Interactions between autonomous vehicles and human-driven vehicles also have major impact on system safety. Our recent work [62] explicitly models driving attitudes of vehicles in a lane changing process, and investigates how the performance and safety of a single vehicle and that of the entire system will be influenced by the level of information sharing of personal driving attitudes among autonomous and human-driven vehicles. The work in [23] considers time-varying, probabilistic characteristics of human behavior, and proposes a Bayesian model to estimate the confidence in its human model. The robot can thus adapt its motion plan according to this confidence. The work in [4] proposes a hierarchical reinforcement and imitation learning (H-REIL) approach for improving the safety during the interaction between autonomous vehicles and human-driven vehicles in near accident scenarios.

5 CONCLUSION

In this paper, we discuss the challenges in addressing safety and safety-related properties in learning-enabled autonomous systems, argue the importance of taking a holistic approach, and introduce some of our initial efforts in this direction, along with other state-of-the-art methods.

REFERENCES

[1] R. Banner, I. Hubara, E. Hoffer, and D. Soudry. Scalable methods for 8-bit training of neural networks. In S. Bengio, H. Wallach, H. Larochelle, K. Grauman, N. Cesa-Bianchi, and R. Garnett, editors, *Advances in Neural Information Processing Systems*, volume 31, pages 5145–5153. Curran Associates, Inc., 2018.
[2] O. Bastani, Y. Ioannou, L. Lampropoulos, D. Vytiniotis, A. Nori, and A. Criminisi. Measuring neural net robustness with constraints. In *Advances in neural information processing systems*, pages 2613–2621, 2016.
[3] C. Buciluă, R. Caruana, and A. Niculescu-Mizil. Model compression. In *Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 535–541, 2006.
[4] Z. Cao, E. Bıyık, W. Z. Wang, A. Raventos, A. Gaidon, G. Rosman, and D. Sadigh. Reinforcement learning based control of imitative policies for near-accident driving. *arXiv preprint arXiv:2007.00178*, 2020.
[5] G. Chen, W. Choi, X. Yu, T. Han, and M. Chandraker. Learning efficient object detection models with knowledge distillation. In *Advances in Neural Information Processing Systems*, pages 742–751, 2017.
[6] Q. A. Chen, Y. Yin, Y. Feng, Z. M. Mao, and H. X. Liu. Exposing congestion attack on emerging connected vehicle based traffic signal control. In *NDSS*, 2018.

[7] X. Chen, E. Ábrahám, and S. Sankaranarayanan. Flow*: An analyzer for non-linear hybrid systems. In *International Conference on Computer Aided Verification*, pages 258–263. Springer, 2013.

[8] Y. Chen, D. Zhao, L. Lv, and Q. Zhang. Multi-task learning for dangerous object detection in autonomous driving. *Information Sciences*, 432:559–571, 2018.

[9] C.-H. Cheng, G. Nührenberg, and H. Ruess. Maximum resilience of artificial neural networks. In *International Symposium on Automated Technology for Verification and Analysis*, pages 251–268. Springer, 2017.

[10] F. Chollet. Xception: Deep learning with depthwise separable convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017.

[11] X. Chu, B. Zhang, and R. Xu. Moga: Searching beyond mobilenetv3. In *ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 4042–4046. IEEE, 2020.

[12] J. Dai, Y. Li, K. He, and J. Sun. R-fcn: Object detection via region-based fully convolutional networks. In *Advances in neural information processing systems*, pages 379–387, 2016.

[13] B. De Brabandere, D. Neven, and L. Van Gool. Semantic instance segmentation for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 7–9, 2017.

[14] M. Denil, B. Shakibi, L. Dinh, M. Ranzato, and N. De Freitas. Predicting parameters in deep learning. In *Advances in neural information processing systems*, pages 2148–2156, 2013.

[15] S. Dutta, X. Chen, and S. Sankaranarayanan. Reachability analysis for neural feedback systems using regressive polynomial rule inference. In *Proceedings of the 22nd ACM International Conference on Hybrid Systems: Computation and Control*, pages 157–168, 2019.

[16] S. Dutta, S. Jha, S. Sankaranarayanan, and A. Tiwari. Output range analysis for deep feedforward neural networks. In *NASA Formal Methods Symposium*, pages 121–138. Springer, 2018.

[17] A. B. Eriksen, C. Huang, J. Kildebogaard, H. Lahrmann, K. G. Larsen, M. Muniz, and J. H. Taankvist. Uppaal stratego for intelligent traffic lights.

[18] C. Fan, U. Mathur, S. Mitra, and M. Viswanathan. Controller synthesis made real: Reach-avoid specifications and linear dynamics. In H. Chockler and G. Weissenbacher, editors, *Computer Aided Verification*, pages 347–366, Cham, 2018. Springer International Publishing.

[19] C. Fan, K. Miller, and S. Mitra. Fast and guaranteed safe controller synthesis for nonlinear vehicle models. In S. K. Lahiri and C. Wang, editors, *Computer Aided Verification*, pages 629–652, Cham, 2020. Springer International Publishing.

[20] J. Fan, C. Huang, W. Li, X. Chen, and Q. Zhu. Towards verification-aware knowledge distillation for neural-network controlled systems. In *IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pages 1–8. IEEE, 2019.

[21] J. Fan, C. Huang, W. Li, X. Chen, and Q. Zhu. Reachnn*: A tool for reachability analysis ofneural-network controlled systems. In *International Symposium on Automated Technology for Verification and Analysis (ATVA)*, 2020.

[22] M. Fazlyab, M. Morari, and G. J. Pappas. Safety verification and robustness analysis of neural networks via quadratic constraints and semidefinite programming. *arXiv preprint arXiv:1903.01287*, 2019.

[23] J. F. Fisac, A. Bajcsy, S. L. Herbert, D. Fridovich-Keil, S. Wang, C. J. Tomlin, and A. D. Dragan. Probabilistically safe robot planning with confidence-based human predictions. *arXiv preprint arXiv:1806.00109*, 2018.

[24] M. Fischetti and J. Jo. Deep neural networks as 0-1 mixed integer linear programs: A feasibility study. *arXiv preprint arXiv:1712.06174*, 2017.

[25] Y. Gal. Uncertainty in deep learning. *University of Cambridge*, 1(3), 2016.

[26] R. Gong, X. Liu, S. Jiang, T. Li, P. Hu, J. Lin, F. Yu, and J. Yan. Differentiable soft quantization: Bridging full-precision and low-bit neural networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, October 2019.

[27] I. J. Goodfellow, J. Shlens, and C. Szegedy. Explaining and harnessing adversarial examples. *arXiv preprint arXiv:1412.6572*, 2014.

[28] E. Goubault and S. Putot. Forward inner-approximated reachability of non-linear continuous systems. In *Proceedings of the 20th International Conference on Hybrid Systems: Computation and Control*, pages 1–10, 2017.

[29] O. Grembek, A. A. Kurzhanskiy, A. Medury, P. Varaiya, and M. Yu. Introducing an intelligent intersection. *ITS Reports*, 2018(13), 2018.

[30] S. Han, J. Pool, J. Tran, and W. Dally. Learning both weights and connections for efficient neural network. In *Advances in neural information processing systems*, pages 1135–1143, 2015.

[31] K. He, G. Gkioxari, P. Dollár, and R. Girshick. Mask r-cnn. In *Proceedings of the IEEE international conference on computer vision*, pages 2961–2969, 2017.

[32] Y. He, X. Zhang, and J. Sun. Channel pruning for accelerating very deep neural networks. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 1389–1397, 2017.

[33] S. L. Herbert, M. Chen, S. Han, S. Bansal, J. F. Fisac, and C. J. Tomlin. Fastrack: A modular framework for fast and guaranteed safe motion planning. In *2017 IEEE 56th Annual Conference on Decision and Control (CDC)*, pages 1517–1522, 2017.

[34] G. Hinton, O. Vinyals, and J. Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[35] A. G. Howard, M. Zhu, B. Chen, D. Kalenichenko, W. Wang, T. Weyand, M. Andreetto, and H. Adam. Mobilenets: Efficient convolutional neural networks for mobile vision applications. *arXiv preprint arXiv:1704.04861*, 2017.

[36] C.-H. Hsu, S.-H. Chang, J.-H. Liang, H.-P. Chou, C.-H. Liu, S.-C. Chang, J.-Y. Pan, Y.-T. Chen, W. Wei, and D.-C. Juan. Monas: Multi-objective neural architecture search using reinforcement learning. *arXiv preprint arXiv:1806.10332*, 2018.

[37] C. Huang, K.-C. Chang, C.-W. Lin, and Q. Zhu. Saw: A tool for safety analysis of weakly-hard systems. *32nd International Conference on Computer-Aided Verification (CAV'20)*, 2020.

[38] C. Huang, X. Chen, W. Lin, Z. Yang, and X. Li. Probabilistic safety verification of stochastic hybrid systems using barrier certificates. *TECS*, 16(5s):186, 2017.

[39] C. Huang, X. Chen, E. Tang, M. He, L. Bu, S. Qin, and Y. Zeng. Navigating discrete difference equation governed wmr by virtual linear leader guided hmpc. In *2020 International Conference on Robotics and Automation*, 2020.

[40] C. Huang, X. Chen, Y. Zhang, S. Qin, Y. Zeng, and X. Li. Hierarchical model predictive control for multi-robot navigation. In *IJCAI*, 2016.

[41] C. Huang, X. Chen, Y. Zhang, S. Qin, Y. Zeng, and X. Li. Switched linear multi-robot navigation using hierarchical model predictive control. In *International Joint Conference on Artificial Intelligence 2017*, 2017.

[42] C. Huang, J. Fan, X. Chen, W. Li, and Q. Zhu. Divide and slide: Layer-wise refinement for output range analysis of deep neural networks. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, 39(11):3323–3335, 2020.

[43] C. Huang, J. Fan, W. Li, X. Chen, and Q. Zhu. Reachnn: Reachability analysis of neural-network controlled systems. *ACM Transactions on Embedded Computing Systems (TECS)*, 18(5s):1–22, 2019.

[44] C. Huang, W. Li, and Q. Zhu. Formal verification of weakly-hard systems. In *the 22nd ACM International Conference on Hybrid Systems: Computation and Control (HSCC)*, 2019.

[45] C. Huang, S. Xu, Z. Wang, S. Lan, W. Li, and Q. Zhu. Opportunistic intermittent control with safety guarantees for autonomous systems. *DAC*, 2020.

[46] G. Huang, S. Liu, L. Van der Maaten, and K. Q. Weinberger. Condensenet: An efficient densenet using learned group convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2752–2761, 2018.

[47] Z. Huang, L. Huang, Y. Gong, C. Huang, and X. Wang. Mask scoring r-cnn. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6409–6418, 2019.

[48] R. Ivanov, J. Weimer, R. Alur, G. J. Pappas, and I. Lee. 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*, pages 169–178, 2019.

[49] S. Lan, R. Panda, Q. Zhu, and A. K. Roy-Chowdhury. Ffnet: Video fast-forwarding via reinforcement learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6771–6780, 2018.

[50] S. Lan, Z. Wang, A. K. Roy-Chowdhury, E. Wei, and Q. Zhu. Distributed multi-agent video fast-forwarding. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 1075–1084, 2020.

[51] H. Law and J. Deng. Cornernet: Detecting objects as paired keypoints. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 734–750, 2018.

[52] Y. LeCun, J. Denker, and S. Solla. Optimal brain damage. *Advances in neural information processing systems*, 2:598–605, 1989.

[53] Y. Lee and J. Park. Centermask: Real-time anchor-free instance segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 13906–13915, 2020.

[54] H. Li, A. Kadav, I. Durdanovic, H. Samet, and H. P. Graf. Pruning filters for efficient convnets. *arXiv preprint arXiv:1608.08710*, 2016.

[55] Q. Li, S. Jin, and J. Yan. Mimicking very efficient network for object detection. In *Proceedings of the ieee conference on computer vision and pattern recognition*, pages 6356–6364, 2017.

[56] H. Liang, M. Jagielski, B. Zheng, C.-W. Lin, E. Kang, S. Shiraishi, C. Nita-Rotaru, and Q. Zhu. Network and system level security in connected vehicle applications. In *Proceedings of the International Conference on Computer-Aided Design*, ICCAD '18, pages 94:1–94:7, New York, NY, USA, 2018. ACM.

[57] H. Liang, Z. Wang, R. Jiao, and Q. Zhu. Leveraging weakly-hard constraints for improving system fault tolerance with functional and timing guarantees. In *2020 IEEE/ACM International Conference On Computer Aided Design (ICCAD)*, pages 1–9, 2020.

[58] H. Liang, Z. Wang, D. Roy, S. Dey, S. Chakraborty, and Q. Zhu. Security-driven codesign with weakly-hard constraints for real-time embedded systems. In *2019 IEEE 37th International Conference on Computer Design (ICCD)*, pages 217–226, Nov 2019.

[59] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 2980–2988, 2017.

[60] X. Lin, C. Zhao, and W. Pan. Towards accurate binary convolutional neural network. In *Advances in Neural Information Processing Systems*, pages 345–353, 2017.

[61] W. Liu, D. Anguelov, D. Erhan, C. Szegedy, S. Reed, C.-Y. Fu, and A. C. Berg. Ssd: Single shot multibox detector. In *European conference on computer vision*, pages 21–37. Springer, 2016.

[62] X. Liu, N. Masoud, and Q. Zhu. Impact of sharing driving attitude information: A quantitative study on lane changing. In *2020 IEEE Intelligent Vehicles Symposium (IV)*, 11 2020.

[63] X. Liu, N. Masoud, Q. Zhu, and A. Khojandi. A markov decision process framework to incorporate network-level data in motion planning for connected and automated vehicles. In *Transportation Research Board 100th Annual Meeting*, 1 2021.

[64] X. Liu, G. Zhao, N. Masoud, and Q. Zhu. Trajectory planning for connected and automated vehicles: Cruising and platooning in mixed traffic. In *Transportation Research Board 99th Annual Meeting*, 1 2020.

[65] Z. Liu, B. Wu, W. Luo, X. Yang, W. Liu, and K.-T. Cheng. Bi-real net: Enhancing the performance of 1-bit cnns with improved representational capability and advanced training algorithm. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[66] A. Lomuscio and L. Maganti. An approach to reachability analysis for feed-forward relu neural networks. *arXiv preprint arXiv:1706.07351*, 2017.

[67] A. Loquercio, M. Segu, and D. Scaramuzza. A general framework for uncertainty estimation in deep learning. *IEEE Robotics and Automation Letters*, 5(2):3153–3160, 2020.

[68] R. Mehta and C. Ozturk. Object detection at 200 frames per second. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 0–0, 2018.

[69] P. Panda, A. Sengupta, and K. Roy. Conditional deep learning for energy-efficient and enhanced pattern recognition. In *2016 Design, Automation & Test in Europe Conference & Exhibition (DATE)*, pages 475–480. IEEE, 2016.

[70] S. Prajna and A. Jadbabaie. Safety verification of hybrid systems using barrier certificates. In *HSCC*, pages 477–492. Springer, 2004.

[71] A. Raghunathan, J. Steinhardt, and P. S. Liang. Semidefinite relaxations for certifying robustness to adversarial examples. In *Advances in Neural Information Processing Systems*, pages 10877–10887, 2018.

[72] S. Ramakrishna, C. Harstell, M. P. Burruss, G. Karsai, and A. Dubey. Dynamic-weighted simplex strategy for learning enabled cyber physical systems. *Journal of Systems Architecture*, page 101760, 2020.

[73] M. Rastegari, V. Ordonez, J. Redmon, and A. Farhadi. Xnor-net: Imagenet classification using binary convolutional neural networks. In *European conference on computer vision*, pages 525–542. Springer, 2016.

[74] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[75] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[76] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen. Mobilenetv2: Inverted residuals and linear bottlenecks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4510–4520, 2018.

[77] D. Seto, B. Krogh, L. Sha, and A. Chutinan. The simplex architecture for safe online control system upgrades. In *Proceedings of the 1998 American Control Conference. ACC (IEEE Cat. No. 98CH36207)*, volume 6, pages 3504–3508. IEEE, 1998.

[78] G. Sharon and P. Stone. A protocol for mixed autonomous and human-operated vehicles at intersections. In *International Conference on Autonomous Agents and Multiagent Systems*, pages 151–167. Springer, 2017.

[79] G. Singh, T. Gehr, M. Mirman, M. Püschel, and M. Vechev. Fast and effective robustness certification. In *Advances in Neural Information Processing Systems*, pages 10802–10813, 2018.

[80] S. Singh, M. Chen, S. L. Herbert, C. J. Tomlin, and M. Pavone. Robust tracking with model mismatch for fast and safe planning: An sos optimization approach. In M. Morales, L. Tapia, G. Sánchez-Ante, and S. Hutchinson, editors, *Algorithmic Foundations of Robotics XIII*, pages 545–564, Cham, 2020. Springer International Publishing.

[81] S. Srinivas and R. V. Babu. Data-free parameter pruning for deep neural networks. *arXiv preprint arXiv:1507.06149*, 2015.

[82] C. Szegedy, W. Zaremba, I. Sutskever, J. Bruna, D. Erhan, I. Goodfellow, and R. Fergus. Intriguing properties of neural networks. *arXiv preprint arXiv:1312.6199*, 2013.

[83] M. Tan and Q. V. Le. Efficientnet: Rethinking model scaling for convolutional neural networks. *arXiv preprint arXiv:1905.11946*, 2019.

[84] S. Teerapittayanon, B. McDanel, and H.-T. Kung. Branchynet: Fast inference via early exiting from deep neural networks. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pages 2464–2469. IEEE, 2016.

[85] B. Tian, X. Deng, Z. Xu, Y. Zhang, and X. Zhao. Modeling and numerical analysis on communication delay boundary for cacc string stability. *IEEE Access*, 7:168870–168884, 2019.

[86] Z. Tian, C. Shen, H. Chen, and T. He. Fcos: Fully convolutional one-stage object detection. In *Proceedings of the IEEE international conference on computer vision*, pages 9627–9636, 2019.

[87] V. Tjeng, K. Xiao, and R. Tedrake. Evaluating robustness of neural networks with mixed integer programming. *International Conference on Learning Representations*, 2019.

[88] H.-D. Tran, S. Bak, W. Xiang, and T. T. Johnson. Verification of deep convolutional neural networks using imagestars. *International conference on Computer-Aided Verification*, 2020.

[89] H.-D. Tran, X. Yang, D. M. Lopez, P. Musau, L. V. Nguyen, W. Xiang, S. Bak, and T. T. Johnson. NNV: The neural network verification tool for deep neural networks and learning-enabled cyber-physical systems. In *32nd International Conference on Computer-Aided Verification (CAV)*, July 2020.

[90] W. Wang, W. Hong, F. Wang, and J. Yu. Gan-knowledge distillation for one-stage object detection. *IEEE Access*, 8:60719–60727, 2020.

[91] X. Wang, M. Kan, S. Shan, and X. Chen. Fully learnable group convolution for acceleration of deep neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2019.

[92] Y. Wang, C. Huang, and Q. Zhu. Energy-efficient control adaptation with safety guarantees for learning-enabled cyber-physical systems. In *2020 IEEE/ACM International Conference On Computer Aided Design (ICCAD)*, pages 1–9. IEEE, 2020.

[93] Z. Wang, H. Liang, C. Huang, and Q. Zhu. Cross-layer design of automotive systems. *IEEE Design Test*, pages 1–1, 2020.

[94] Z. Wang, G. Wu, and M. J. Barth. A review on cooperative adaptive cruise control (cacc) systems: Architectures, controls, and applications. In *2018 21st International Conference on Intelligent Transportation Systems (ITSC)*, pages 2884–2891. IEEE, 2018.

[95] B. Wu, F. Iandola, P. H. Jin, and K. Keutzer. Squeezedet: Unified, small, low power fully convolutional neural networks for real-time object detection for autonomous driving. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, pages 129–137, 2017.

[96] S. Xu, S. Lan, and Z. Qi. Maskplus: Improving mask generation for instance segmentation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, March 2020.

[97] Z. Yang, C. Huang, X. Chen, W. Lin, and Z. Liu. A linear programming relaxation based approach for generating barrier certificates of hybrid systems. In *FM*, pages 721–738. Springer, 2016.

[98] Y. Yao, L. Rao, X. Liu, and X. Zhou. Delay analysis and study of ieee 802.11p based dsrc safety communication in a highway environment. In *2013 Proceedings IEEE INFOCOM*, pages 1591–1599, 2013.

[99] W. Yawen, W. Zhepeng, J. Zhenge, S. Yiyu, and H. Jingtong. Intermittent inference with nonuniformly compressed multi-exit neural network for energy harvesting powered devices. *arXiv preprint arXiv:2004.11293*, 2020.

[100] F. Yu and V. Koltun. Multi-scale context aggregation by dilated convolutions. *arXiv preprint arXiv:1511.07122*, 2015.

[101] D. Zhang, J. Yang, D. Ye, and G. Hua. Lq-nets: Learned quantization for highly accurate and compact deep neural networks. In *Proceedings of the European Conference on Computer Vision (ECCV)*, September 2018.

[102] X. Zhang, X. Zhou, M. Lin, and J. Sun. Shufflenet: An extremely efficient convolutional neural network for mobile devices. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 6848–6856, 2018.

[103] Z. Zhang, S. Fidler, and R. Urtasun. Instance-level segmentation for autonomous driving with deep densely connected mrfs. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 669–677, 2016.

[104] B. Zheng, Y. Gao, Q. Zhu, and S. Gupta. Analysis and Optimization of Soft Error Tolerance Strategies for Real-Time Systems. In *2015 International Conference on Hardware/Software Codesign and System Synthesis (CODES+ISSS)*, pages 55–64, October 2015.

[105] B. Zheng, C. W. Lin, H. Liang, S. Shiraishi, W. Li, and Q. Zhu. Delay-aware design, analysis and verification of intelligent intersection management. In *2017 IEEE International Conference on Smart Computing (SMARTCOMP)*, pages 1–8, May 2017.

[106] B. Zheng, C.-W. Lin, S. Shiraishi, and Q. Zhu. Design and analysis of delay-tolerant intelligent intersection management. *ACM Transactions on Cyber-Physical Systems*, 4(1):1–27, 2019.

[107] B. Zheng, C.-W. Lin, H. Yu, H. Liang, and Q. Zhu. CONVINCE: A Cross-Layer Modeling, Exploration and Validation Framework for Next-generation Connected Vehicles. In *Computer-Aided Design (ICCAD), 2016 IEEE/ACM International Conference on*, November 2016.

[108] B. Zheng, M. O. Sayin, C. W. Lin, S. Shiraishi, and Q. Zhu. Timing and security analysis of vanet-based intelligent transportation systems: (invited paper). In *2017 IEEE/ACM International Conference on Computer-Aided Design (ICCAD)*, pages 984–991, Nov 2017.

[109] Q. Zhu, W. Li, H. Kim, Y. Xiang, K. Wardega, Z. Wang, Y. Wang, H. Liang, C. Huang, J. Fan, and H. Choi. Know the unknowns: Addressing disturbances and uncertainties in autonomous systems : Invited paper. In *2020 IEEE/ACM International Conference On Computer Aided Design (ICCAD)*, pages 1–9, 2020.