Accepted manuscript published in Computer, vol. 56, no. 5, pp. 41-50, May 2023, doi: 10.1109/MC.2023.3236523.

On Neural Networks Redundancy and Diversity for Their Use in Safety-Critical Systems

Axel Brando
Barcelona Supercomputing Center

Isabel Serra
Barcelona Supercomputing Center

Enrico Mezzetti
Barcelona Supercomputing Center

Francisco J. Cazorla
Barcelona Supercomputing Center

Jon Perez-Cerrolaza
Ikerlan Technology Research Centre, Basque Research and Technology Alliance (BRTA)

Jaume Abella
Barcelona Supercomputing Center

***Abstract*—An increasing number of critical functionalities integrated in Embedded Critical Systems rely on Deep Learning technology, such as ground and onboard operations in avionics and decision-making functions in autonomous automotive systems. In this work, we summarize certain key safety aspects of the software and system development process, as required by domain-specific safety certification standards, at odds with the intrinsic stochastic and training-data-dependent nature of Deep Learning solutions. These are significant obstacles that must be addressed before Deep Learning solutions can be seamlessly adopted in Embedded Critical Systems. In this line, we propose a potential approach for developing Neural Network based safety functions using redundancy and diversity as main drivers. We also show and exemplify how redundancy and diversity can be developed in Neural Networks.**

■ **FROM ITS ORIGINS** in 1956, Artificial Intelligence (AI) has evolved considerably. Today, AI-based systems can work autonomously and take human-level decisions in a wide range of areas. A remarkably successful branch of AI is Machine Learning (ML), which allows computers to learn by themselves analyzing data. For example, data examples can implicitly specify the desired functionalities, operating rules and constraints. Inside the ML branch, we find Neural Networks (NN) that are particularly effective at solving complex problems such as image recognition and natural language processing. The emergence of Deep Learning (DL) technology, i.e. NN models that adapt to learn highly complex functionalities using vast amounts of data, constitutes an inflection point for the commoditization of AI.

In Embedded Critical Systems (ECS), the use of DL is increasingly wide-spreading in many domains like avionics, automotive, railway, and space [1]. For instance, in automotive perception and decision-making functions, DL solutions are a cornerstone in the development of future advanced (fully) autonomous systems. In fact, DL techniques are at the heart of the realization of advanced software functions such as computer vision (e.g., object detection and tracking), path planning, and driver-monitoring systems [2]. This is so because DL techniques have demonstrated their effectiveness in managing complex and heterogeneous problems, outperforming other algorithmic approaches, and providing a technical approach for the development of next-generation safety-critical systems such as autonomous systems. Hence, the future economic success of ECS industries depends on their ability to design, implement, qualify, and certify DL-based software products under bounded effort/cost.

Safety-related functions in ECS undergo a strict development process, as determined by generic (e.g., IEC 61508) and domain-specific safety standards like

ISO 26262 for automotive [3]. In particular, the development process defines a specific set of steps and techniques to guarantee that the risk of failure is residual. Those steps rely on software components that can be explicitly specified and use deterministic algorithms; further, test data is used during validation phases to collect evidence supporting that the developed software and system is correct by design.

Implementing safety-related functions with DL requires demonstrating compliance with the same stringent development process. However, the design of DL solutions combines stochastic algorithms and data, hence clashing with requirements on (i) explicit, complete and verifiable specification, (ii) the use of deterministic algorithms, as well as (iii) the separation of software and data for safety-related systems. Therefore, non-trivial DL-based solutions are, in general, incompatible with the state-of-the-art practice and safety standard requirements for developing safety-related systems. For example, current IEC 61508:2010 safety standard does not recommend AI techniques even for diagnostics.

In this context, and building on the ISO 5469 [4] standard (draft) Usage Level (UL) and class taxonomy, for the most stringent usage level where the NN implements at least one safety function (A1), we first analyze key features of AI-based components that clash with safety-related development processes. This precludes the adoption of those NN components straightforwardly as part of safety-related systems. We then define a potential approach for the development of NN-based safety-related functions, building on the concepts of redundancy and strong diversity. The proposed approach is defined in analogy to SIL4 railway odometry based safety functions that provide safe vehicle speed and distance measurements based on an ensemble of redundant, diverse and highly reliable sensors that are individually not compliant with safety standards. In explaining our approach, we develop how NNs can be made redundant and provably diverse based on argumentation. We further illustrate the proposed approach with a simplified practical example that also shows the limitations and relevant open challenges to overcome.

BACKGROUND ON THE DEVELOPMENT PROCESS FOR SAFETY SYSTEMS

The development of safety-relevant systems must meet strict processes and technical requirements that aim to reduce to extremely low levels the probability of system failure due to systematic errors (e.g., specification, design and implementation errors made by humans, methods and tools) and random hardware errors (e.g., memory bit flip).

After defining the safety goals of the system, these are map into explicitly defined safety requirements, which determine the architectural design of the system (and its components/items) and the integrity level inherited by each component [3].

Components inheriting high-integrity levels may be decomposed into multiple lower integrity level components providing redundant functionality, yet diverse design so that they do not fail all of them systematically upon a single common error. For instance, a given software component can be implemented by disjoint teams building on the same specification so that human design errors are independent and the probability of common cause errors decreases.

The architectural design is based on complete, deterministic and explicit component specifications that can be verified, often with formal or semi-formal methods that allow proving that safety requirements are met. Besides, since random hardware errors cannot be avoided entirely, appropriate safety measures are set up to mitigate and control them. This process, which builds solely on complete, deterministic and explicit requirements, and deterministic software algorithm design, is completed by implementing the components and their integration. At every implementation and integration stage, appropriate data is used for testing (validation) to obtain empirical evidence of the safe operation of the system. However, data does not influence the design and it is used only for testing purposes. Besides, all possible software configuration parameter combinations shall be tested positively.

As classified by the ISO 5469 [4] UL taxonomy, AI technology can be used to implement a safety function (A1, A2), or a non-safety-related function that might interfere with safety function(s) (C), or be interference-free (D). And an AI-based solution can be developed in compliance with safety standard techniques (Class I techniques such as redundancy and diversity) or using compensation measures (Class II techniques such as diverse monitor).

DL-based software functionalities can play different roles in the safety of the system. The default approach to enabling the use of DL-based solutions as part of

safety-related systems consists in performing system decompositions, so that a non-DL-based deterministic monitor inherits its safety requirements along with those of the DL-based component it monitors (e.g., UL C with Class II safety-bag technique). This allows relieving DL-based components from any safety requirement in practice. In fact, upon a failure of those components, the monitor is typically in charge of either switching to a degraded mode of operation or another type of safe state. This approach has been suitable for assistance systems, such as Advanced Driving Assistance Systems in the automotive context, where upon a failure of the AI-based component, the monitor generally notifies the driver about the failure and transfers the control to the driver.

However, fully autonomous systems (e.g., drones, spacecraft, and level 4/5 autonomous cars) may simply lack a safe state since no driver or pilot may exist, and hence, the AI-based component must implement the safety function (A1). In this scenario, the AI-based components inherit the safety requirements. Therefore, an approach needs to be devised to tailor the architecture of those components to fit a safety-related development process despite the stochastic and data-based nature of those AI-based components, such as NN-based ones.

GAP ANALYSIS FOR THE USE OF NN IN ECS

Introduction to NN

In supervised learning, the NN globally works as a transformation from the input data (X) to the output response variable ($\boldsymbol{y}$) which usually approximates the desired value (such as labels in classification problems or real values in regression ones). Mathematically, the NN produces the function $\phi_{\boldsymbol{w}}: \mathbb{R}^K \rightarrow \mathbb{R}^O$ transforming $X \longmapsto \boldsymbol{y}$ and where $\boldsymbol{w}$ are the NN weights.

Despite their underlying model is often assumed to approximate a (deterministic) mathematical function, DL solutions can exhibit stochastic elements in the training process but also in the model evaluation. While some approaches have been proposed to enforce a deterministic training process [5], its stochastic nature is typically acknowledged [6], capturing the variability from the physical inputs to other uncertainty sources [7][8] that includes even the execution platform [9]. When it comes to the model itself, the implementation of variational inference methods using Variational Dropout or Bayesian networks with re-parametrization, for example, entails that the evaluation of the same model over the same input values will yield different outputs, as they depend on internal random variables. For these reasons, we consider that, by default and in the worst-case scenario, the NN training and model can exhibit a stochastic nature.

Challenges

The development process of NNs clashes with the traditional development process of safety-relevant systems. Significant design phase gaps are:

- As described in the previous section, NNs have a stochastic nature, hence providing probabilistic results with confidence levels that depend on the uncertainty source.
- The NN functional requirements are specified using exemplary data that aims to implicitly specify the intended functionalities, rules and constraints. Instead of explicit requirements that can be verified for completeness and correctness.
- NNs build on an architecture that cannot be (usually) proven correct, as opposed to traditional software in safety-relevant systems.
- Finally, a relevant portion of the NN functional design is encoded as numerical weights determined by the training data instead of source-code software. This also clashes with established development processes where data is only used for testing and not for the design itself.

NNs to be used in safety-relevant systems also bring several concerns related to their validation, which further challenges the design to mitigate validation-phase challenges. For example:

- The non-deterministic nature of NNs makes it challenging to define proper test cases (e.g., worst scenario, equivalence classes, boundaries) or sufficiency measures (e.g., equivalence with minimum test coverage requirements), hence bringing uncertainty to the quality of tests.
- The open operational environment in which NN operate challenges defining when input data is within training range, and hence, when NNs can provide reliable predictions.

Overall, limitations in the design and validation of NNs against the requirements of a safety-relevant development process pose extra pressure on the design of the software components integrating NNs for safety-critical functionalities since their design must mitigate and control these gaps. As shown in the following

sections, instead of solving those gaps one by one by trying to make NNs behave as traditional safety software, which is against NN nature, we accept the stochastic nature of NNs and relate them to other scenarios in the context of safety-critical systems, where non-safety certified components can be ensembled to reach required safety integrity levels.

PROPOSED APPROACH

Our proposed approach targets the development of NN-based safety functions (A1) using redundancy and diversity (Class I and II techniques). The proposed design method and technique focus on systematic failure mitigation and target complex applications for which formal verification and safety-bag approaches are not generally applicable because safety requirements cannot be explicitly specified.

To that end, we extend and adapt the strategy used to develop odometry safety-critical systems (SIL4 odometry) to the development of NN-based safety-critical systems. In the proposed analogy, the ensemble of diverse and highly-reliable sensors used to build a SIL4 safety-critical system, where each sensor does not strictly require safety certification, is translated into an ensemble of diverse and highly-reliable NNs.

Building safety-critical functionalities with non-safety certified sensors: the case of the railway odometry safety critical-system

The railway onboard European Train Control System (ETCS) is a SIL4 ECS that supervises the train traveled distance and speed and activates the emergency brake if safe limits are exceeded. It relies on the speed and distance measurements provided by the onboard odometry safety function (SIL4) based on an ensemble of diverse and highly-reliable sensors such as doppler radars, accelerometers, and encoders. This diversity of physical principles of measurement, and sensor-specific characteristics (e.g., different manufacturers, hardware, communication protocol), reduce the probability of a Common Cause Failure (CCF). The odometry safety algorithm is commonly implemented as a SIL4 software/VHDL safety function executed on a SIL4 triplicated safety computer. However, each sensor does not require to be safety-certified, and the odometry safety algorithm safely combines the redundant estimates provided by the ensemble of sensors using error detection, mitigation and control techniques [10].

From non-safety sensors to NNs

In a safety-odometry system, such as the one previously described, each sensor should provide highly reliable estimates within the sensor's operational range under the uncertainty of circumstances that can arise in an open world. For example, a radar sensor should provide a highly reliable speed measurement within the usage conditions established in the product manual (e.g., installation angle and distance). Nonetheless, the odometry algorithm should be able to detect or mitigate known and unknown circumstances that could lead to wrong speed measurement inferences, such as incorrect installation angles and the presence of metal or snow on the track (which affects the doppler effect). To that end, the odometry algorithm shall be able to detect, control and mitigate errors using diverse and redundant sensors.

In this paper, we claim that DL-based systems can be built on top of NN components following the same approach adopted for non-safety sensors, namely using each NN as a potentially non-safety compliant component characterized by its external specifications, and safely ensembling multiple NNs redundantly in a way that diversity is also guaranteed. In particular, sensors are considered black boxes whose operation is determined only by the specification of their external behavior. Moreover, sensors may have fuzzy operation ranges (e.g., how much metal and where it must be located to impact the doppler effect) and provide stochastic responses subject to precision considerations. Analogously, NNs can be treated as black boxes, hence relieving the verification process from dissecting NN internals as done for traditional safety software. Moreover, the input data operational range for an NN is fuzzy (e.g., whether fog is excessive or not for object detection), and the result of inference is also stochastic (i.e. with confidence levels), as for sensors.

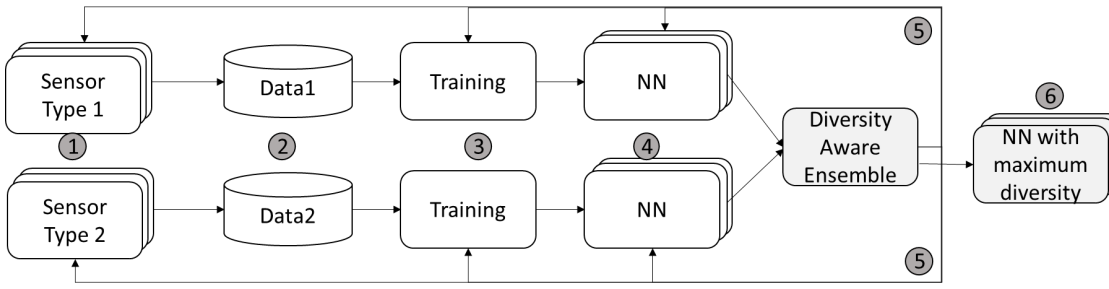

Figure 1: Example diversity approaches for proposed NN ensembles.

Required NN properties for the Realization of the Proposed Safety Approach

The proposed safety approach requires, among other aspects, appropriate management of the operational range and diverse redundancy (extending the analogy of the previously described railway odometry sensors).

Operational range: In our view, the concept of operation range for sensors can be managed through the notion of the distributional shift [11] and NN's epistemic uncertainty modeling [7].

- The former field corresponds to detecting changes in the distribution generated by the input data. Specifically, the goal is to identify and foresee outliers only using the input information before predicting the response variable.
- The latter field refers to the study of how the NN model definition (i.e. the selected hyper-parameters of the NN) limits the ability of the NN to predict in certain scenarios. Similarly, the goal is to infer the boundaries of the functions approximated by the NN's pre-selected architecture and generic hyper-parameters.

Diverse redundancy: At a conceptual level, the ultimate objective is to combine NN-produced predictions with enough dissimilarities so that their individual failures can be regarded as independent. Hence, the NN can be used redundantly to provide specific safety levels, as in the case of the sensors of the odometry system. It is noted that the failure rates imposed by high integrity levels in the corresponding safety standards (from 10^{-5} to 10^{-9} per hour of operation in the case of IEC 61508) are well beyond the success rate of a NN that are commonly in the range 75%-95% when they are considered accurate solutions. Hence, the required failure rates can only be realistically reached using diverse and redundant NNs, that in fact are realized at multiple scopes simultaneously, e.g. operating on the same input, and across multiple redundant and diverse data sources (e.g. multiple cameras, radars and LiDARs). Although the latter already brings (physical) diversity by construction, it also imposes significant system design and procurement costs, hence, we focus on the former in the remaining of this work.

When combining several NNs to obtain reliable results, either in the form of multiple predictions for the same input (Class I), or in the form of a prediction and a test of whether such prediction can be trusted (Class II), the key metric is independence between the networks. This is better understood with an example: If we can gain some confidence that two NNs, NN1 and NN2, are completely independent, then we can probabilistically combine their outputs. That is, the combined probability of misprediction is the multiplication of the individual misprediction probabilities.

A way to achieve independence is via diversity, which can be achieved and architected in different manners to prevent those inputs that lead to misprediction for one NN and systematically from leading to misprediction for another redundant NN. Figure 1 summarizes example diversity approaches for the development of NN ensembles:

1) Diversity may come from different inputs sources. For instance, we set two identical NNs trained identically, but linked to different cameras (sensors) in different car locations, thus capturing different perspectives of the same objects. Again, non-systematic mispredictions must also be justified.
2) We can set also up multiple NNs, e.g. NN1 and NN2, identical, but trained with different input data sets. We trust their prediction when it matches, and distrust it otherwise (Class I). Note that it remains to be argued that their different training avoids or mitigates to a sufficient extent the possibility to fail both systematically.
3) We can set up NN1 and NN2 with different designs (regarding the architecture, training hyper-parameters that defines a NN or even sharing only

part of the architecture [12]) and (randomly) change the initialization point of their weights, either with the same or different data for training. Non-systematic mispredictions must also be justified.

4) The same case as those above but instead of using 2 NNs, we use a larger number and rely on voting to secure a prediction in many more cases [13]. Note that the voting process is a challenge on itself since we may need, for instance, non-homogeneous weights to consider different degrees of dependence across NNs, or even consider temporal redundancy if predictions occur periodically to properly bias the voting process.

5) We set a NN for the prediction and one or several NNs to validate/reject such prediction. Those other NNs may be trained to detect whether input data is out of the valid range, to predict specific casuistic for which the main NN is not so skilled or even improve the main NN learning process using an adversarial learning-based design, in the form of a Zero-sum game [14] (Class II).

Importantly, the approaches above can be mixed to avoid their individual systemic weaknesses. For instance, (1) mixing depends on the ability to deduce or interpret where the shortcomings of each model come from. Usually, the high dimensionality of the data (e.g. when we are tackling with images) prevents from easily deriving a clear mix. Therefore, we should rely and improve the work on high dimensional probabilistic modeling. Similarly, (2) mixing depends on expert-knowledge to ensure different input information provide richer and diverse sources of learning. On the other hand, (3) mixing is connected with the concept of expressivity of a certain neural network architecture or hyper-parameter selection. This constitutes an open research problem since it is well-known that, regardless of the number of parameters, NNs can approximate complex mathematical functions, but the boundaries of such estimation capabilities are difficult to define. Likewise, the voting process of a huge number of NNs in the ensemble, as proposed in (4), should consider how each of these NN were designed to weight its relevance in the vote and, similarly, any unsupervised learning process, as shown in (5), will depend on how the prediction and rejection NN were defined. Thus, the more misprediction sources are considered, the better.

A commonality in all approaches to achieve sufficient diversity is the fact that independence needs to be proven in accordance with common practice in the corresponding safety-critical domain. This may imply providing adequate argumentation as well as quantitative data validating that independence holds to a sufficient extent.

Motivating example

For illustration purposes, in this section we focus on a specific simplified realization of NN diversity, to show the potential feasibility of the proposed approach, the technical limitations and the open research challenges that still need to be addressed for the development and certification/qualification of diverse NN ensemble-based safety-critical systems. Other realizations and combinations thereof are possible.

We consider a set of 4 different NNs trained to classify between several traffic signals images. Each of these NNs was selected to have more than 96% of accuracy considering the overall classes in the validation set, with accuracy measured as the average non-weighted success rate across all the classes. Therefore, each NN is a reasonably good estimator from a functional perspective, but the systematic error rate is far from tolerable for a safety-related ECS.

The used input images are obtained from the German Traffic Sign Benchmarks, a multi-class benchmark proposed for a classification competition [15]. In order to train the NNs as supervised learning models[1], we divided the data set into 3 randomly selected subsets:

- the training set (containing 35209 images, 68% of the global data set), that is used to optimize the NN parameters;
- the validation set (including 4000 images, 8% of the global data set), that is used to decide when to stop the learning procedure avoiding phenomena such as overfitting; and
- the test set (considering 12630 images, 24% of the global data set) that is used to ultimately check the performance of the trained system in a separate set to compare to other models.

[1] The presented experiments were implemented using Tensorflow with Keras as a high-level API.

In order to study NN diversity, we focus on a simple case that already shows the potential benefits for safety applications. In particular, we build on the same NN in terms of architecture (i.e. layers and number of neurons) and input data. The four variants of the NN are generated by simply changing the initialization point of their weights. That is, as we saw previously, NN are optimized by changing their weight values and here we consider that each of the NNs will start from a different point so each NN will tend to end up with distinct weights values or, equivalently, being a different mathematical function. This occurs since the mathematical surface to approximate – defined by the weights values – is not convex, and therefore there exist several local minima when the optimization process of the weights is applied. Note that this simplified example exploits a single source of diversity, hence still keeping some lack of independence across NNs, which share, for instance, their architecture and input data for training.

In order to assess diversity, we compute the success rate of increasingly larger combinations (ensembles) of NNs, out of the four available. We start by deriving the success rate of NN0. Then, we compute the success rate of the ensemble NN0+NN1 so that if any of the two NNs successfully classifies the object of a given class in the image, we consider that their ensemble produces a good classification. Importantly, here the reported information is not a chosen class but a predictive set containing the true label, which is connected with the concept of Conformal prediction with relevant importance in certain contexts [16]. Following this idea, we create two further sub-ensembles, incrementally adding NN2 and NN3.

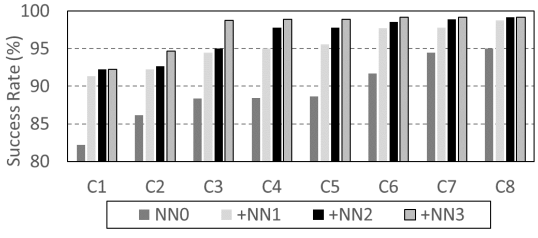

Figure 2: Success rate of different NN ensembles.

The results in Figure 2 show that for the 8 classes (C1-C8) for which NN0 produced bad prediction results, adding NN1, NN2, and NN3 progressively to the ensemble results in improved accuracy. In some cases, we see that just adding NN1 produces a significant increase in accuracy (e.g. for C1). For other cases the increment is steadier (C7). In other cases, the difference when adding a new NN to the ensemble is small (e.g. when adding NN2 to C2 and when adding NN2 and NN3 to C8). While diversity across the particular NNs used in the example is limited to a single source, it already provides some relevant – yet not full – independence. For instance, in the case of C1, if NNs lacked diversity, their combination would keep the success rate at 82% roughly constant. Instead, if they were fully independent, such success rate would grow to almost 97% when adding NN1 instead of only 91%, and to above 99% when adding NN2 instead of only 92%. Overall, we observe that there is exploitable diversity among NNs although their individual success rate is similar. Even small changes within the considered NN ensemble can help reducing the probability of a CCF and, crucially, such differences encourage further research on exploiting this existing information, and devising appropriate means to combine sources of diversity that provide a sufficient degree of independence.

Furthermore, from this simplified example, we can extract some insights into technical limitations and research open challenges. For example:

(a) Independence among diverse NNs. Exploiting redundant diversity techniques provides a potential approach to reducing the probability of CCF. However, even in this simplified example with initial random weights diversity, the obtained accuracy results were below ideal accuracy targets due to hidden dependencies. Hence, as for the odometry reference approach [10], the achievable independence and the probability of CCF should be analyzed, measured, and tested using safety standard compliant novel methods yet to be defined (see future work).

(b) Achieving extremely low probability of systematic errors. In generic DL-based applications, achieving a 99% accuracy might be considered suitable. However, this is not the generic case for safety-critical systems that must meet an extremely low probability of failure. For illustration purposes, if we assume that a new traffic sign must be correctly inferred every 6 seconds on average, a 99% accuracy leads to a probability of one erroneous classification every 600 seconds (10 minutes). And achieving the most demanding SIL4 minimum probability of failure (10^{-9} per hour of operation, approximately 114.000 years) would require an extremely high accuracy (99,9999999972%). As for the odometry reference approach, the exploitation of diverse redundancy can pave the way towards achieving such low probability of error. Yet, how to

combine different predictions from redundant NNs to increase accuracy is an open challenge in itself since approaches better than homogeneous voting may exist.

(c) Computational cost: The execution of redundant NNs entails a high computational cost, in a similar way to the increase in the economic cost of the odometry system due to the integration of redundant sensors. However, in the absence of a directly certifiable / qualifiable solution, the challenge in both cases is to develop a redundant solution with sufficient diversity to achieve the required safety integrity level in compliance with safety standards, at affordable effort/cost for the target domain.

CONCLUSIONS AND FUTURE WORK

The increasing adoption of NN and DL technology for ECS development in the avionics, automotive, rail and space domains (e.g. safety-critical autonomous systems) requires the definition of novel and bounded effort/cost technical approaches for their development and certification/qualification according to domain-specific standards. However, the stochastic and data-dependent nature of NN solutions inherently clashes with the compelling requirements imposed by such standards. In this work, we have elaborated and adapted the redundancy and diversity approach already used for developing SIL4 odometry safety-critical systems and proposed a safety argumentation with example diversity technical approaches for the development of NN ensemble-based safety-critical systems. We have used a simplified motivating example to show the potential feasibility of the approach, technical limitations, and open challenges for future work: e,g, exploring and assessing different diversity approaches, definition of analysis, test and measurement approaches for independence and CCF probability, definition of threshold values for acceptable diversity, and further refine the underpinned safety argumentation.

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Research Council (ERC) grant agreement No. 772773 (SuPerCom), the Horizon Europe Programme under the SAFEXPLAIN Project (www.safexplain.eu), grant agreement num. 101069595, and the Spanish Ministry of Science and Innovation under grant PID2019-107255GBC21/ AEI/10.13039/501100011033.

■ REFERENCES

1. J. Athavale et al (2020), "AI and Reliability Trends in Safety-Critical Autonomous Systems on Ground and Air," In DSN-W.
2. El Sallab, Ahmad, et al. (2017). "Deep reinforcement learning framework for autonomous driving" Electronic Imaging.
3. ISO 26262(-1/11) road vehicles - functional safety, 2018
4. Schneider, V. Artificial intelligence and functional safety - a summary of the current challenges. Report, TÜV SUD Rail GmbH, 27th May 2021
5. Nagarajan, P. et al. (2018). Deterministic implementations for reproducibility in deep reinforcement learning. arXiv:1809.05676
6. Izmailov, P., et al. (2021). What are Bayesian neural network posteriors really like?. In International conference on machine learning. PMLR
7. Kendall, A., and Gal, Y. (2017). What uncertainties do we need in bayesian deep learning for computer vision?. Advances in neural information processing systems.
8. Der Kiureghian, A., and Ditlevsen, O. (2009). Aleatory or epistemic? Does it matter?. Structural safety, 31(2), 105-112.
9. Alcon, M et al. (2020). Timing of autonomous driving software: Problem analysis and prospects for future solutions. In IEEE RTAS.
10. J. Perez, et al. (2010). "Codesign and Simulated Fault Injection of Safety-Critical Embedded Systems Using SystemC," In EDCC
11. Fang, T et al . Rethinking importance weighting for deep learning under distribution shift. Advances in Neural Information Processing Systems, 2007.
12. Antorán, J. et al. (2020). Depth uncertainty in neural networks. Advances in neural information processing systems, 33, 10620-10634.
13. Dong, X. et al. (2020). A survey on ensemble learning. Frontiers of Computer Science, 14(2), 241-258.
14. Liu, Y., Li et al. (2019). Generative adversarial active learning for unsupervised outlier detection. IEEE Transactions on Knowledge and Data Engineering, 32(8), 1517-1528.
15. Stallkamp, J. et al. (2011). The German traffic sign recognition benchmark: a multi-class classification competition. In The 2011 international joint conference on neural networks (pp. 1453-1460). IEEE.

16. Angelopoulos, A. N. et al. (2020, September). Uncertainty Sets for Image Classifiers using Conformal Prediction. In International Conference on Learning Representations.

Axel Brando is a mathematician and computer scientist who is working as a researcher at the BSC. He holds a PhD and Master's degree in Artificial Intelligence. His research has been conducted in the field of uncertainty modelling by using Deep Learning and he has published papers in top-level AI conferences and journals like NeurIPS and AISTATS. Contact him at axel.brando@bsc.es.

Isabel Serra is a mathematician and a senior researcher in the CAOS group at BSC. She has worked on the development of methods for extreme value prediction applied to Computer Science and Earth Science to precisely and reliably predict extreme events. Contact her at isabel.serra@bsc.es.

Enrico Mezzetti, holds a PhD in Computer Science from University of Bologna and Padova. His research interests cover the verification of critical embedded systems, specializing in worst-case timing analysis of multicore systems. Dr. Mezzetti has been co-authoring more than 50 publications in peer-reviewed international conferences and journals in the field of embedded real-time systems verification. Contact him at enrico.mezzetti@bsc.es.

Francisco J. Cazorla, is the co-head of the group on Interaction between the Operating System and the Computer Architecture (CAOS) at BSC. Dr. Cazorla has led three projects on the use of advanced multicore hardware in embedded critical systems. He has several works on the analysis of complexities of using AI-based algorithms in critical domains like avionics, space, or automotive. Contact him at francisco.cazorla@bsc.es.

Jon Perez-Cerrolaza, (Senior Member, IEEE) received his Ph.D. degree in computer science from TU Wien. He is currently a Principal Researcher at Ikerlan in the field of dependable autonomous systems, and his research interests include dependability, machine learning, and cybersecurity technologies. He has worked for more than 15 years in the development and certification of industrial and transportation domain safety-critical systems such as SIL4 onboard railway signalling systems (ERTMS/ETCS). Contact him at jmperez@ikerlan.es.

Jaume Abella is the co-head of the group on Interaction between the Operating System and the Computer Architecture (CAOS) at BSC. His research interests include timing, functional safety and AI in the context of high-performance critical real-time systems. He is a member of IEEE and HiPEAC. Contact him at jaume.abella@bsc.es.