ASSESSING ROBUSTNESS IN DATA-DRIVEN MODELING OF CYBER-PHYSICAL SYSTEMS

Maximilian Schmidt, Swantje Plambeck, Goerschwin Fey
Institute of Embedded Systems, Computer Engineering
Hamburg University of Technology
Am Schwarzenberg-Campus 1, 21073 Hamburg, Germany
firstname.lastname@tuhh.de

ABSTRACT

Robustness is a key factor in the design and analysis of Cyber-Physical Systems (CPS), ensuring that systems function correctly even under perturbations. This paper investigates robustness within the data-driven modeling process, focusing on three core aspects: system robustness, model robustness, and learner robustness. We survey existing notions of robustness and propose unified formal definitions for each aspect, analyzing their interdependencies and their contributions to overall CPS performance. Additionally, we introduce a method for assessing the robustness of models generated by data-driven learning that is independent of both the model's internal representation and the learning paradigm used. Our approach leverages input perturbations combined with probabilistic analysis to evaluate how well a learned model handles input variations, particularly when formal guarantees are challenging to obtain. To demonstrate the practical application of our method, we conduct a case study on a temperature control system, using decision trees to model system behavior. By perturbing test data and analyzing the resulting model outputs, we identify non-robust regions near decision boundaries, thereby revealing potential vulnerabilities. The proposed framework offers valuable insights for enhancing system design and lays the groundwork for future research into robust machine learning models for CPS.

Keywords Robustness · Cyber-Physical Systems · Data-Driven Modeling

1 Introduction

Cyber-Physical Systems (CPSs) integrate digital components with physical processes to intelligently monitor and control the physical world [26]. The digital components collect data from the environment and use it to make control decisions that are executed by actuators interacting with the physical world [24]. CPSs are ubiquitous in modern society, with applications ranging from automotive systems [15] and industrial automation [27] to smart grids [14] and healthcare [11].

The design and analysis of such systems present several challenges, primarily due to the need for accurate models that capture both physical dynamics and control logic. Such models are essential for testing and validating CPS designs in simulated environments, thereby reducing risks associated with real-world testing, such as operational disruptions or safety hazards when exploring extreme scenarios. Models also play a key role in identifying design flaws early, guiding system specifications, and supporting the overall development process.

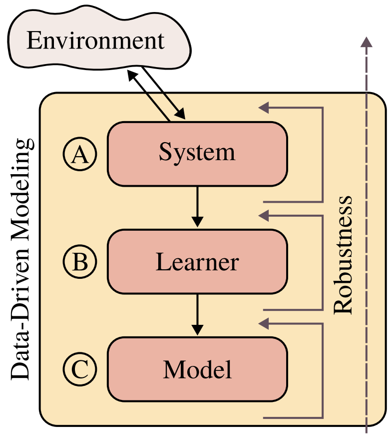

Figure 1: Workflow for data-driven modeling and the role of robustness analysis for the individual steps, as well as robustness as a property of the entire pipeline

Although systems can be precisely defined in theory, uncertainties in the physical environment—such as sensor inaccuracies, external disturbances, or even adversarial attacks—introduce deviations that can compromise system behavior. Given the inherent complexity of CPSs, exploring the entire behavioral space during the design phase is both computationally expensive and often impractical [17]. CPSs generate large amounts of data, which can be used to infer models through data-driven approaches. Observing the system during operation is thus more practical than analyzing it in full detail, provided that robust models are available to accurately predict system behavior.

For this paper, we adopt the data-driven modeling approach shown in Figure 1, where a *learner* is tasked with inferring a model of the system from collected data. We use the term *system* to represent a real CPS interacting with its environment. A *model* is a simplified representation of the system that abstracts its complexity to facilitate analysis, simulation, and prediction. We specifically focus on the robustness of the model generation pipeline. As CPSs increasingly operate in environments vulnerable to security attacks, robustness is essential to ensure resilience against adversarial threats [7, 3]. In the context of data-driven modeling of CPSs, robustness encompasses several dimensions, including *(A)* the system's resilience to input perturbations [7], *(B)* the learner's ability to generate functioning models despite variations in training data [28], and *(C)* the predictive accuracy of the model under unforeseen conditions [6]. Generally, we refer to robustness as the capacity of an entity to maintain stable performance in its environment even under conditions that were not anticipated during design. We note that *resilience* typically refers to the ability to adapt and recover from disturbances, whereas robustness focuses on maintaining stability in the presence of perturbations. Throughout this paper, we use the term *robustness* in the sense of stability and resistance, while acknowledging that resilience is a complementary property. This contrasts with the notion of *reliability*, which typically refers to the system's ability to perform its intended function without internal failure. Additionally, robustness is often associated with the ability to recover from environmental errors [5].

It is important to note that while robustness aims to ensure stability and resistance to perturbations, this often comes at the expense of sensitivity. A highly robust system or controller may react more slowly to important changes in its environment compared to a more sensitive design. Thus, designers must carefully balance robustness and sensitivity, ensuring that stability does not unduly hinder the systems responsiveness.

This paper explores various definitions and implications of robustness within CPS modeling. Rather than advocating for a single superior notion of robustness, we build on existing definitions by combining, abstracting, and formalizing them to offer a more comprehensive perspective on robustness in CPSs. Different application contexts require distinct formalizations of robustness due to the unique challenges and requirements each context presents [7].

The main contributions of this paper are:

- An overview of various notions of robustness found in the literature.
- A unified notion of robustness for data-driven modeling pipelines that addresses systems, models, and learners, along with a discussion of the implications.
- A data-centric method to analyze model robustness through input perturbation.
- An application of the proposed framework to a hybrid system, demonstrating its practical utility.

The remainder of this paper is structured as follows: section 2 provides an overview of related work on robustness in CPSs. In section 3 we introduce a unified notion of robustness for data-driven modeling, focusing on systems, models, and learners. Section 3.1 explores the robustness of a system, section 3.2 discusses the robustness of a model, and section 3.3 examines the robustness of a learner. Section 4 presents a data-centric framework for analyzing model robustness using perturbation. Finally, section 6 concludes the paper with a summary of the key findings and implications for future research.

2 Related Work

The term *robustness* is used in various contexts, and its definition varies with the application domain. Robustness commonly refers to a system's ability to tolerate perturbations and continue functioning effectively despite deviations from expected behavior, such as disturbances, uncertainties, or adversarial attacks. Literature on robustness in CPSs spans multiple disciplines, including control theory, machine learning, and formal methods. Table 1 provides an overview of related literature and summarizes how robustness is approached in different contexts.

In control theory, robustness is primarily associated with a system's ability to maintain stability and minimize performance degradation in the presence of perturbations, with an emphasis on graceful degradation under disturbances. CPSs typically consist of physical and digital components, where the physical component describes continuous behavior and the digital component describes discrete behavior. Analyzing robustness separately for these components follows naturally from the different types of disturbances affecting them. There are also approaches that combine control and digital logic, such as hybrid system robustness, which addresses robustness across both layers of CPSs. Formal methods provide a theoretical foundation for robustness analysis, often employing temporal logic to specify and verify robustness properties of CPSs.

From a machine learning perspective, robustness is often framed in terms of a model's resilience to adversarial samples or out-of-distribution inputs, ensuring that predictions remain reliable under such conditions. Another key focus in machine learning is optimizing data selection or transformation to enhance the robustness of both learners and models.

3 Robustness in Data-Driven Modeling

In the context of data-driven modeling for CPSs, we consider robustness at three distinct levels: the system, the model, and the learner. Each of these components plays a critical role in the overall performance and reliability of the modeling pipeline. We begin by examining system robustness, which concerns the system's ability to maintain stable behavior under external perturbations. Next, we analyze model robustness, focusing on the models capacity to accurately predict system behavior despite input variations. Finally, we address learner robustness, which relates to the learners ability to generate reliable models when trained on imperfect or perturbed data. By investigating robustness at each of these levels, we aim to provide a comprehensive understanding of the challenges and requirements for robust data-driven CPS modeling.

Reference	Area	Notion of Robustness
Tabuada et al. [25]	Control theory	Individual analysis of physical and digital components in CPSs
Rungger et al. [20]	Control theory	Deviation from nominal behavior proportional to the disturbance
Sontag [23]	Control theory	Input-state stability for robustness analysis
Bloem et al. [5]	Formal methods	Proportionality of environmental failures to system failures
Goebel et al. [10]	Formal methods	Homogeneous perturbations using pre-asymptotic stability
Castiglioni et al. [7]	Formal methods	Temporal logic for robustness analysis of CPSs
Hsieh [12]	Analysis	Resilience against resource failures in production systems
Cui et al. [9]	Analysis	Ratio of functioning nodes after an attack or error
Baker et al. [2]	Analysis	Risk and consequences of potential damage to the system
Zhang et al. [29]	Machine learning	Resilience against failures in digital components
Sehwag et al. [21]	Machine learning	Analyzing robustness with out-of-distribution samples
Ramoni et al. [19]	Machine learning	Robust learning under missing data
Bhagoji et al. [4]	Machine learning	Data transformations to enhance model robustness
Najafi et al. [16]	Machine learning	Role of unlabeled data in enhancing adversarial robustness
Odyurt et al. [17]	Monitoring	Variants of anomalies in traces of execution
Zhang et al. [30]	Monitoring	Cascading failures in complex systems and impact of failures
Shahrokni et al. [22]	Software	Software robustness
Hu et al. [13]	Design methods	Stability, security, and systematic design considerations

Table 1: Overview of related work on robustness in CPSs

3.1 Robustness of a System

Robustness of a system refers to its ability to maintain functionality in the presence of perturbations. A system

$$S : I \to O$$

defines a mapping from inputs to outputs, where I denotes the input space and O the output space. The systems behavior is characterized by this input-output relationship. Here, I and O may represent traces of inputs and outputs over time to capture time-dependent or stateful behavior. Deviations in system behavior arise when design assumptions are violated at runtime, which is inevitable due to partial knowledge about the system and its operating environment [20].

To formalize perturbations, we define a perturbation function $F : I \to I$ that modifies the input to the system, and we introduce a robustness predicate R that encapsulates the intended functionality of the system for assessing its robustness. Here, i denotes an original input, $F(i)$ is the perturbed input, $S(i)$ is the output corresponding to the original input, and $S(F(i))$ is the output under perturbation. The predicate $R \subset I \times I \times O \times O$ is defined to capture acceptable deviations in behavior; for instance, it may specify that the distance between $S(i)$ and $S(F(i))$ does not exceed a predefined threshold. A system is thus considered robust if

$$(i, F(i), S(i), S(F(i))) \in R \quad \forall i \in I.$$

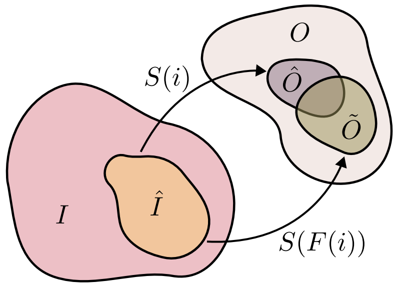

Figure 2: Input and output sets used to model robustness under input perturbations

The predicate may be restricted to a local subspace $\hat{I} \subset I$ to focus on specific regions of interest. Figure 2 illustrates the concept of robustness by showing how input perturbations affect the corresponding outputs.

A straightforward approach to assess robustness is to define a distance metric $d_o : O \times O \to \mathbb{R}$ between the intended output o and the perturbed output $\widetilde{o} = S(F(i))$. With a predefined threshold $\epsilon \in \mathbb{R}$, the robustness predicate R_ϵ is characterized by

$$d_o(\widetilde{o}, o) \leq \epsilon.$$

For this *bounded output stability* [1], the system is said to be ϵ-robust if R_ϵ holds for all perturbed outputs.

Another perspective on robustness is that the system's deviation from nominal behavior is proportional to the magnitude of the environmental disturbance [5, 20]. We denote this *proportional stability* as R_κ using a distance metric for both input and output spaces. We define an input space distance metric $d_i : I \times I \to \mathbb{R}$ similarly to d_o, and the system is considered robust if

$$d_o(\widetilde{o}, o) \leq \kappa \cdot d_i(\widetilde{i}, i),$$

where $\kappa \in \mathbb{R}$ is a proportionality constant, $\widetilde{i} = F(i)$, and $\widetilde{o} = S(\widetilde{i})$. This proportional stability is a specialization of the Lipschitz continuity property and is commonly assumed in CPS robustness analyses.

We call any perturbed input sample that causes a violation of the robustness predicate an *adversarial sample*. By this we transfer the notion of adversarial examples from the context of machine learning to CPS and define *adversarial robustness* as the magnitude of the disturbance required to transform any original sample into an adversarial one [6]. Specifically, adversarial robustness in a local input space $\hat{I}$ is quantified as the minimum distance between any adversarial sample and its corresponding original sample:

$$\min_{i \in \widetilde{I}} \; d_i(i, F(i)) \text{ s.t. } (i, F(i), S(i), S(F(i))) \notin R.$$

This metric provides a means to quantify the system's resilience to adversarial perturbations, indicating the level of disturbance required before its robustness fails.

3.2 Robustness of a Model

A model M is designed to replicate the behavior of a system S for the purposes of analysis and prediction, typically by abstracting some of the system's complexity. The model represents a simplified version of the system, capturing its essential input-output characteristics: $M : I \to O$.

The robustness of a model refers to its ability to accurately predict the system behavior despite perturbations. Assuming the system satisfies a robustness predicate R, the model—intended to faithfully represent the system behavior—should also satisfy R under similar perturbations. Thus, the model is subject to the same robustness definitions as the system.

3.2.1 Models as Surrogates

To quantify the difference between the model output and the system output, we define a loss function $L : O \times O \to \mathbb{R}$ that enables the optimization of an accurate model representation of the system behavior. An accurate model is one that minimizes this loss function. Although system robustness cannot be directly inferred from model robustness, a sufficiently accurate model can serve as a surrogate, providing valuable insights into the robustness properties of the system.

This strategy offers two key advantages. First, by identifying regions where the model fails to meet robustness criteria, we can cross-reference these findings with the system to determine whether the issues stem from limitations in the model or reflect inherent weaknesses in the system. This feedback loop between model and system helps to clarify whether non-robust behavior is intrinsic to the system design or simply a byproduct of the modeling process.

Second, identifying non-robust regions via the model yields actionable insights for improving the system design, leading to enhanced overall performance. In this way, the model serves not only as a diagnostic tool but also as guidance for system improvements.

3.2.2 Limitation

If the system fails to satisfy a robustness predicate, then even an accurately constructed model cannot be expected to meet that predicate. This implies that the robustness of a model is inherently limited by the robustness of the system it represents. Therefore, while a model may be validated under nominal conditions, if the underlying system exhibits non-robust behavior in certain scenarios, the models predictions under similar conditions will likewise lack robustness.

3.3 Robustness of a Learner

A learner is an algorithm tasked with inferring a model from training data. The performance of the model is evaluated based on its ability to accurately predict system behavior. The optimal model is obtained by minimizing the loss function defined in section 3.2.1:

$$\min_{M} \sum_{i \in I} L(M(i), S(i)) \quad \forall i \in I.$$

The robustness of a learner $\mathcal{L}$ refers to its ability to produce an accurate model despite perturbations in the training data. Since the predictive performance of a model depends on the quality of the training data, learner robustness measures its capacity to adapt to variations in that data. Formally, we denote the learning process as $\mathcal{L}(D) = M$, where $D \subset I \times O$ is the training set of input-output samples and M is the model inferred by the learner. The optimal learner $\mathcal{L}^*$ is one that infers a model that perfectly captures the system behavior, i.e.,

$$\mathcal{L}^*(D)(i) = S(i) \quad \forall i \in I.$$

To assess learner robustness, we define a perturbation function F on the training set. We then compare the performance of the model trained on the original training set with that of the model trained on the perturbed training set. The formal notation for this comparison, analogous to the system robustness definition (cf. section 3.1), is omitted here for brevity.

Training samples that mislead the learner into generating suboptimal models are termed adversarial training samples. The learner's resilience against these adversarial samples is a key aspect of its robustness. Mathematical frameworks can formally guarantee robustness properties, as demonstrated by generalization results for neural networks and splitting criteria for decision trees [8].

Robust learning is closely related to learning under missing or limited observation data. Perturbations in training data involve more than merely adding or omitting samples; for CPSs, common issues include noise and non-equidistant sampling rates that distort the true distribution and lead to inaccurate representations. Furthermore, if the system generating the training data is not robust, the learner cannot necessarily compensate for perturbed data, as such perturbations may lead to inconsistent learning outcomes.

4 Analyzing Robustness

Robustness has long been a focus in various fields: system robustness is extensively studied in control theory, while learner robustness is commonly investigated in machine learning, particularly in the context of adversarial examples. Having discussed the three core components of the data-driven modeling pipeline—systems, models, and learners—we now shift our focus to model robustness.

4.1 Methodology

The method for evaluating robustness largely depends on the type of model being analyzed. For black-box models, robustness assessment typically involves perturbing the inputs and observing the resulting outputs to detect any deviations. This input-output approach does not require insight into the internal structure of the model. In contrast, white-box models permit structural analysis, which enables direct inference of robustness properties from the model internals.

In addition to explicit formalization, a data-driven and learning-based approach can be applied to robustness analysis. Instead of relying solely on formal methods or direct structural analysis, this approach uses data to predict potential robustness violations. By analyzing input-output pairs and their perturbations, models can be trained to detect patterns indicative of robustness issues. This predictive method is especially useful when formal robustness guarantees are challenging to obtain in highly complex models or systems.

4.2 Input Perturbation

We propose a black-box method for assessing robustness across the entire data-driven modeling pipeline, with a particular focus on the learned model. Our approach evaluates how the model responds to variations in the input space by systematically applying perturbations to the test data. This enables a comprehensive analysis of the models ability to generalize beyond its training data and remain robust under input variations.

Analyzing model robustness through sample testing is analogous to evaluating a model on a test set where the test set comprises perturbed samples. This method is particularly useful for complex models, such as (deep) neural networks, where deriving robustness properties analytically is challenging.

4.3 Probabilistic Robustness Analysis

To perform a probabilistic robustness analysis, we begin by training a model M on a dataset D that captures the behavior of the system S. By perturbing each input sample from the dataset with a randomly drawn perturbation F, we augment the test dataset and conduct a probabilistic assessment of the models robustness.

We choose a distance metric $d_o : O \times O \to \mathbb{R}$ to quantify the difference between the model output on the original input and the output on the perturbed input. The choice of perturbation function F and distance metric d_o must be based on the model type and the application domain. For example, the perturbation function may involve adding Gaussian noise to the input, while the distance metric d_o may be the L_0, L_2, or L_∞ distance.

It should be noted that the probabilistic robustness analysis does not provide formal guarantees. The statistical measures obtained are sensitive to the chosen perturbation function and distance metric. The coverage of the perturbation space may be limited, and thus the conclusions drawn from this analysis should be interpreted with caution. Future work will explore methods to enhance the formal coverage of the analysis.

This approach provides a flexible and scalable method for evaluating robustness, particularly for black-box models where formal analysis is impractical. By systematically perturbing inputs and assessing the impact on model outputs, we can determine how well the model maintains its functionality under diverse conditions. Due to its probabilistic nature, this approach quantifies

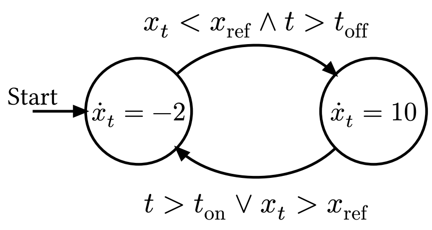

Figure 3: Automaton of the temperature control system

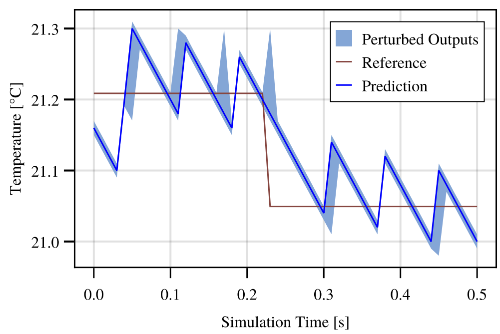

Figure 4: Predictions of the thermostat model with a probabilistic robustness analysis showing outputs under perturbations

robustness in terms of statistical measures but does not provide formal guarantees. We leave an exploration of the quality and coverage of this probabilistic analysis for future work.

5 Case Study

In this case study, we apply our probabilistic robustness analysis method to a simple CPS: a temperature control system. We use this system to illustrate how input space perturbations can reveal insights into the robustness of a model trained to predict system behavior.

5.1 Temperature Control System

The temperature control system consists of two main components: the boiler and the thermostat. The thermostat monitors the current temperature and compares it with a reference temperature. Based on this comparison, it switches the boiler on or off. The hybrid system, illustrated in Figure 3, operates in two modes: on and off.

To prevent overheating, the system enforces a predefined maximum duration for which the boiler can remain on, after which it automatically switches off. Additionally, a cooldown mechanism ensures that the system cannot immediately switch back to the on mode after being turned off, thereby introducing a delay before reheating is possible.

The system takes one input: the reference temperature $x_{\text{ref}} \in \mathbb{R}$. The internal state of the system is represented by the current temperature $x_t \in \mathbb{R}$. The rate of temperature change is governed by its first derivative: $\dot{x}_t = -2$ in the off mode and $\dot{x}_t = 10$ in the on mode.

5.2 Robustness Analysis

To analyze the robustness of the system, we first simulate the temperature control system for 100,000 samples, thereby generating a dataset for training. The dataset encodes the system input as a trace consisting of the current temperature x_t^n and the reference temperature x_{ref}^n over multiple time steps, while the system output is the next temperature x_t^{n+1}:

$$i = [x_t^0, x_{\text{ref}}^0, \ldots, x_t^n, x_{\text{ref}}^n, x_{\text{ref}}^{n+1}] \quad o = x_t^{n+1}.$$

We split the dataset into training and test sets and train a decision tree model to predict the next temperature based on the input trace. Decision trees are well-suited for learning the behavior of hybrid systems, as noted in prior work [18]. The trained model achieves a mean absolute error of 0.001 on the test set, indicating high accuracy in predicting system behavior.

Next, we assess the robustness of the model by applying perturbations to the last input reference. Specifically, we perturb the input by adding a disturbance $f \in \mathcal{U}_{[-\delta,\delta]}$, where $\mathcal{U}$ denotes a uniform distribution, i.e., $F(i) = i + f$. The model robustness is measured using the absolute deviation between the models output on the perturbed input $\tilde{o} = M(F(i))$ and its output on the original input $o = M(i)$, given by $d_o(\tilde{o}, o) = |\tilde{o} - o|$. Each input sample from the test set is perturbed 100 times using randomly drawn disturbances, and the model behavior is observed.

Figure 4 illustrates the simulation results, showing the predicted model output (in blue) along with the disturbed outputs represented as an area around the prediction. The robustness analysis reveals non-robust regions in the output, particularly near the decision boundaries between the systems on and off states. These regions exhibit greater sensitivity to input perturbations, highlighting areas where the model predictions are less robust.

This case study demonstrates the utility of probabilistic robustness analysis in identifying potential vulnerabilities in CPS models. By systematically perturbing the input space, we gain a deeper understanding of the model's response to variations, which can inform improvements in system design or guide further model refinement. If access to the actual system is available, these insights can also be used to conduct targeted testing and analysis.

In addition to detecting robustness violations, the model can serve diagnostic purposes, such as identifying faulty samples and classifying the nature of the faults. This extends the robustness analysis beyond mere detection, offering a deeper understanding of the perturbations or failures that lead to robustness violations.

6 Conclusion

Robustness is a critical attribute in the design and analysis of CPSs, ensuring that systems, models, and learners can withstand and adapt to disturbances, uncertainties, and variations in their operational environment. This paper has explored the definitions and implications of robustness for data-driven modeling from three perspectives: systems, models, and learners. Overall, robustness is a multifaceted concept that is pivotal in ensuring reliability, security, and performance. Our exploration underscores the importance of robust design and analysis in building resilient and trustworthy CPSs that can operate effectively despite inevitable disturbances and uncertainties. Future work will continue to refine robustness metrics and develop methods to enhance the robustness of systems, models, and learners. Furthermore, the probabilistic robustness analysis presented here, while flexible, lacks formal guarantees and its coverage is inherently limited. The discussion on learner robustness, in particular, would benefit from further investigation into strategies such as adversarial training and robust loss formulations. Finally, additional work is needed to more precisely formalize the robustness predicates and ensure that all components of the modeling pipeline are thoroughly evaluated.

Acknowledgments

This work is funded by BMBF project AGenC no. 01IS22047A.

References

[1] B. D. O. Anderson and J. B. Moore. New Results in Linear System Stability. *SIAM Journal on Control*, 7(3):398–414, Aug. 1969.

[2] J. W. Baker, M. Schubert, and M. H. Faber. On the assessment of robustness. *Structural Safety*, 30(3):253–267, May 2008.

[3] A. Banerjee, K. K. Venkatasubramanian, T. Mukherjee, and S. K. S. Gupta. Ensuring Safety, Security, and Sustainability of Mission-Critical Cyber–Physical Systems. *Proceedings of the IEEE*, 100(1):283–299, Jan. 2012.

[4] A. N. Bhagoji, D. Cullina, C. Sitawarin, and P. Mittal. Enhancing robustness of machine learning systems via data transformations. In *2018 52nd Annual Conference on Information Sciences and Systems (CISS)*, pages 1–5, Princeton, NJ, Mar. 2018. IEEE.

[5] R. Bloem, K. Greimel, T. A. Henzinger, and B. Jobstmann. Synthesizing robust systems. pages 85–92, Dec. 2009.

[6] N. Carlini and D. Wagner. Towards Evaluating the Robustness of Neural Networks. In *2017 IEEE Symposium on Security and Privacy (SP)*, pages 39–57, San Jose, CA, USA, May 2017. IEEE.

[7] V. Castiglioni, M. Loreti, and S. Tini. RobTL: A Temporal Logic for the Robustness of Cyber-Physical Systems, Dec. 2022.

[8] H. Chen, H. Zhang, S. Si, Y. Li, D. Boning, and C.-J. Hsieh. Robustness Verification of Tree-based Models. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[9] P. Cui, P. Zhu, P. Xun, and Z. Xia. Enhance the robustness of cyber-physical systems by adding interdependency. In *2017 IEEE International Conference on Intelligence and Security Informatics (ISI)*, pages 203–203, July 2017.

[10] R. Goebel and A. R. Teel. Preasymptotic Stability and Homogeneous Approximations of Hybrid Dynamical Systems. *SIAM Review*, 52(1):87–109, Jan. 2010.

[11] S. A. Haque, S. M. Aziz, and M. Rahman. Review of Cyber-Physical System in Healthcare. *International Journal of Distributed Sensor Networks*, 10(4):217415, Apr. 2014.

[12] F.-S. Hsieh. Robustness Analysis of Cyber-Physical systems based on Discrete Timed Cyber-Physical Models. In *2021 IEEE World AI IoT Congress (AIIoT)*, pages 0250–0254, May 2021.

[13] F. Hu, Y. Lu, A. V. Vasilakos, Q. Hao, R. Ma, Y. Patil, T. Zhang, J. Lu, X. Li, and N. N. Xiong. Robust Cyber–Physical Systems: Concept, models, and implementation. *Future Generation Computer Systems*, 56:449–475, Mar. 2016.

[14] S. Karnouskos. Cyber-Physical Systems in the SmartGrid. In *2011 9th IEEE International Conference on Industrial Informatics*, pages 20–23, Lisbon, Portugal, July 2011. IEEE.

[15] M. Lukasiewycz, S. Steinhorst, F. Sagstetter, W. Chang, P. Waszecki, M. Kauer, and S. Chakraborty. Cyber-Physical Systems Design for Electric Vehicles. In *2012 15th Euromicro Conference on Digital System Design*, pages 477–484, Cesme, Izmir, Turkey, Sept. 2012. IEEE.

[16] A. Najafi, S.-i. Maeda, M. Koyama, and T. Miyato. Robustness to Adversarial Perturbations in Learning from Incomplete Data. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc., 2019.

[17] U. Odyurt, A. D. Pimentel, and I. Gonzalez Alonso. Improving the robustness of industrial Cyber–Physical Systems through machine learning-based performance anomaly identification. *Journal of Systems Architecture*, 131:102716, Oct. 2022.

[18] S. Plambeck and G. Fey. Decision Tree Models of Continuous Systems. In *2022 IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA)*. IEEE, Sept. 2022.

[19] M. Ramoni and P. Sebastiani. Robust Learning with Missing Data. *Machine Learning*, 45(2):147–170, 2001.

[20] M. Rungger and P. Tabuada. A Notion of Robustness for Cyber-Physical Systems. *IEEE Transactions on Automatic Control*, 61(8):2108–2123, Aug. 2016.

[21] V. Sehwag, A. N. Bhagoji, L. Song, C. Sitawarin, D. Cullina, M. Chiang, and P. Mittal. Analyzing the Robustness of Open-World Machine Learning. In *Proceedings of the 12th ACM Workshop on Artificial Intelligence and Security*, pages 105–116, London United Kingdom, Nov. 2019. ACM.

[22] A. Shahrokni and R. Feldt. A systematic review of software robustness. *Information and Software Technology*, 55(1):1–17, Jan. 2013.

[23] E. D. Sontag. Input to State Stability: Basic Concepts and Results. In A. A. Agrachev, A. S. Morse, E. D. Sontag, H. J. Sussmann, V. I. Utkin, P. Nistri, and G. Stefani, editors, *Nonlinear and Optimal Control Theory: Lectures given at the C.I.M.E. Summer School Held in Cetraro, Italy June 19–29, 2004*, Lecture Notes in Mathematics, pages 163–220. Springer, Berlin, Heidelberg, 2008.

[24] M. Szczodrak, Y. Yang, D. Cavalcanti, and L. P. Carloni. An open framework to deploy heterogeneous wireless testbeds for Cyber-Physical Systems. In *2013 8th IEEE International Symposium on Industrial Embedded Systems (SIES)*, pages 215–224, Porto, June 2013. IEEE.

[25] P. Tabuada, S. Y. Caliskan, M. Rungger, and R. Majumdar. Towards Robustness for Cyber-Physical Systems. *IEEE Transactions on Automatic Control*, 59(12):3151–3163, Dec. 2014.

[26] W. M. Taha, A.-E. M. Taha, and J. Thunberg. *Cyber-Physical Systems: A Model-Based Approach*. Springer International Publishing, Cham, 2021.

[27] K. Thramboulidis. A cyber–physical system-based approach for industrial automation systems. *Computers in Industry*, 72:92–102, Sept. 2015.

[28] J. Uesato, J.-B. Alayrac, P.-S. Huang, R. Stanforth, A. Fawzi, and P. Kohli. Are Labels Required for Improving Adversarial Robustness?, 2019.

[29] J. J. Zhang, K. Liu, F. Khalid, M. A. Hanif, S. Rehman, T. Theocharides, A. Artussi, M. Shafique, and S. Garg. Building Robust Machine Learning Systems: Current Progress, Research Challenges, and Opportunities. In *Proceedings of the 56th Annual Design Automation Conference 2019*, DAC '19, pages 1–4, New York, NY, USA, June 2019. Association for Computing Machinery.

[30] X. Zhang, H. Ma, and C. K. Tse. Assessing the Robustness of Cyber-Physical Power Systems by Considering Wide-Area Protection Functions. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 12(1):107–114, Mar. 2022.