Towards Better Trust in Human-Machine Teaming through Explainable Dependability

Marcello M. Bersani*, Matteo Camilli*, Livia Lestingi*, Raffaela Mirandola*, Matteo Rossi*, Patrizia Scandurra†
* Politecnico di Milano, Italy,
Email: {name}.{surname}@polimi.it
† University of Bergamo, Italy,
Email: patrizia.scandurra@unibg.it

***Abstract*—The human-machine teaming paradigm is increasingly widespread in critical domains, such as healthcare and domestic assistance. The paradigm goes beyond human-on-the-loop and human-in-the-loop systems by promoting tight teamwork between humans and autonomous machines that collaborate in the same physical space. These systems are expected to build a certain level of trust by enforcing dependability and exhibiting interpretable behavior. We present emerging results in this direction, with a novel framework aiming at achieving better trust in human-machine teaming leveraging formal analysis, as well as eXplainable AI. We illustrate our approach and the emerging results with an example from the healthcare domain.**

***Index Terms*—Human-machine teaming, formal analysis, statistical model checking, explainable AI**

I. Introduction

The increasing pervasiveness and capabilities of autonomous systems and their ability to work in complex environments has brought about the need for a different paradigm of interaction between humans and machines, called Human-Machine-Teaming (HMT). In this paradigm, machines and humans can be seen as teammates that collaborate leveraging their strengths and reducing their weaknesses. A successful collaboration yields a shared awareness, that is, the behavior of machines can be interpreted by humans, and the physiological characteristics of humans are reified into machines.

Awareness facilitates human-machine understanding and helps build *trust* in their collaboration. Here we adopt the definition of trust introduced in [18], as *"the attitude that an agent will help achieve an individual's goals in a situation characterized by uncertainty and vulnerability"*. This definition can be declined along different lines taking into account several dimensions including quality and social metrics [15]. As a first step towards trust creation in HMT, we adopt a vision where machines shall offer dependability guarantees and exhibit transparent behavior by explaining their actions in a human-interpretable format. However, the opaque nature of these systems has a negative effect on the level of trust since their decisions are typically fully autonomous and may be confusing for humans, or may even cause hazards in critical domains. At the same time, the widespread diffusion of AI technologies makes robotic agents flexible [4], [13], [16], able to respond to changing needs and expectations of humans, that may even act in an adversarial way. This latter point poses further challenges in understanding the meaning of trust in the other direction, that is, *machine trust* towards human behavior.

In this paper, we present our initial ideas and emerging results fostering *human trust* towards machine behavior in HMT. To this end, we propose the EASE[1] framework and its underlying high-level conceptual architecture. EASE leverages formal analysis, based on Statistical Model Checking (SMC) [11], as well as eXplainable AI, based on model-agnostic interpretable Machine Learning (ML) techniques [24]. Since formal analysis may be computationally expensive, we keep it in pre-production as an offline stage, where we can execute demanding activities without interfering with the running missions. Formal analysis feeds a binary classifier estimating the relations between uncertain and changing teaming factors and the mission outcome in terms of satisfaction of dependability properties. The classifier is then used online to predict the mission outcomes and explain them to human teammates.

The envisioned framework is illustrated with a running example in the healthcare domain featuring multiple sources of uncertainty in the human-machine interaction, such as diverse physical and physiological characteristics of the agents involved in the teamwork. The contributions of this work are as follows: (1) we tackle the human-to-machine HMT trust via dependability and explainability at runtime; (2) we introduce EASE, a pioneering framework combining SMC and interpretable ML; (3) we propose a three-layer software architecture supporting both offline and online stages of EASE.

Related Work. The self-adaptive systems community [1], [2] has been promoting a set of approaches to embed in systems a certain degree of autonomy and the ability to deal with foreseen/foreseeable changes through (ad-hoc) control loops. The role of humans in these systems has been a topic of discussion and debate, with researchers arguing for fully autonomous self-adapting systems (humans-out-of-the-loop), and others asserting the need of humans as external controllers and supervisors (human-on-the-loop) [12], [23], or as input providers for the system's work (human-in-the-loop) [22]. Recent work by Garlan et al. [6], [21], [22] formalizes the inclusion of the human-in/on-the-loop using stochastic models incorporating human personality traits and

[1]EASE stands for Exploration, AnalySis, and Explanation.

including explanations to facilitate the human understanding of the system operation through model checking.

The notion of HMT has been recently introduced in [10], [23], where interaction patterns yield a partnership exploiting the strengths of both actors. The framework introduced in [23] supports the elicitation of HMT options in a set of simulated operational contexts. The work presented in [10] extends the MAPE-K loop with human-related tasks and describes an infrastructure that supports this tight collaboration enforcing transparency, augmented cognition, and coordination.

Another orthogonal line of research focuses on conceptual frameworks for trust in autonomous systems per-se [9], and in human-machine interactions in general [4], [13], [16]. Trust definition and management encompass multiple facets elicited in recent surveys [13], [16], [25], including dependability and explainability aspects that represent our main focus. The lifecycle of trust management with a definition of properties, metrics and possible solutions in the area of social Internet of Things is illustrated in [15]. A roadmap on human-machine mutual understanding and collaboration is presented in [4].

Paper organization. Section II introduces the example we use to illustrate our envisioned approach EASE, which is then presented in Sec. III. A discussion about the potential societal impact of better (bidirectional) trust is presented in Sec. IV, while Sec. V concludes the paper and presents our future work.

II. A Running Example in Healthcare

Robots can be used in hospitals to accompany patients in the ward and assist healthcare workers in their daily activities. In this setting, human agents may be affected by physical or mental fatigue and act independently of a prescribed plan out of their free will. More generally, the environment in which HMT takes place is highly dynamic since the presence of people and the workload on care workers are subject to change.

In this work, HMT missions consist of a finite sequence of *services* provided by robots and initiated by humans, according to one or more *interaction pattern*s [19], [20]. Our running example adopts the "human follower" and "human leader" patterns as follows. Consider the portion of the ward consisting of an entrance, a waiting room, a doctor's office and a storage room. The sequence consists of four services. The robot reaches the entrance and *accompanies* the patient to the waiting room. It then heads towards the current position of the doctor, it meets the doctor and *follows* them to the storage room where they fetch the required tools. The robot *follows* the doctor back to the office. Finally, the robot returns to the waiting room and *escorts* the patient to the office where the doctor is waiting for them. The mission *succeeds* if all services are delivered within a given amount of time defined by the designer. Mission success is the targeted dependability requirement for the robotic application.

III. The EASE Approach

In this section, we present the main stages of our approach, depicted in Fig. 1: offline ① HMT modeling, ② exploration and analysis; and online ③ prediction and explanation. The

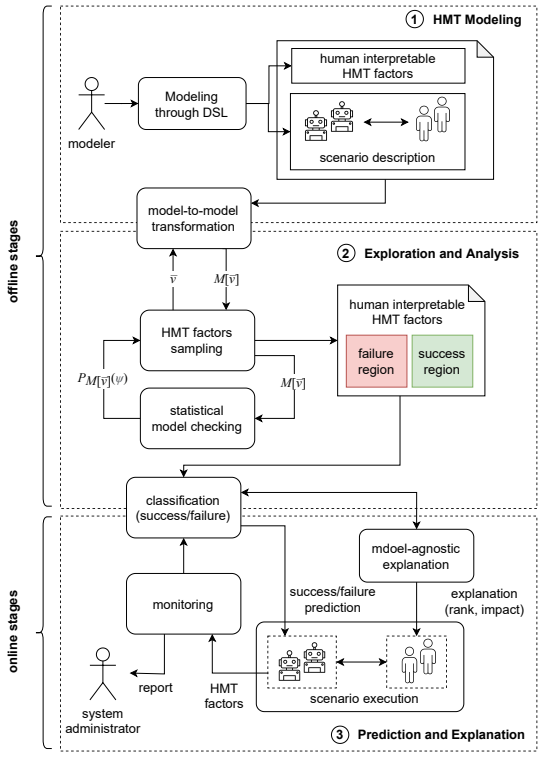


Fig. 1: EASE workflow.

underlying three-layer EASE architecture is depicted in Fig. 2. The top layer is responsible for assisting in the HMT modeling stage, the middle layer includes the main components allowing the exploration, analysis and prediction of the HMT behavior, while the bottom layer represents the HMT system itself. In the following, we describe the EASE stages as well as the involved components of the proposed architecture. We exemplify the key concepts through the running example of Sec. II.

A. HMT Modeling

According to Fig. 1, the modeler starts the offline stage using the `ModelManager` component (see Fig. 2) to define the HMT interaction patterns of interest through a user-friendly Domain-Specific Language[2] (DSL). The modeler specifies the agents' characteristics, including the health status of humans, the age and the fatigue/recovery rates, the initial battery level of the robots, their discharge rates, and the initial position of all the agents in the teaming area (see Table I). These human-interpretable HMT factors are variables, each with its own type and domain. Hence, an assignment $\bar{v}$, mapping every variable to a domain value, characterizes human agents and robot agents, influences the HMT, and possibly affects, in turn, dependability aspects. Given an HMT mission and an assignment $\bar{v}$, the `ModelManager` automatically generates a formal representation of the pattern referred to as model instance $\mathcal{M}[\bar{v}]$, i.e., a Stochastic Hybrid Automata (SHA)

[2]DSL for Human-Robot Interactive Service Scenarios: sources publicly available at https://github.com/LesLivia/hri_dsl.

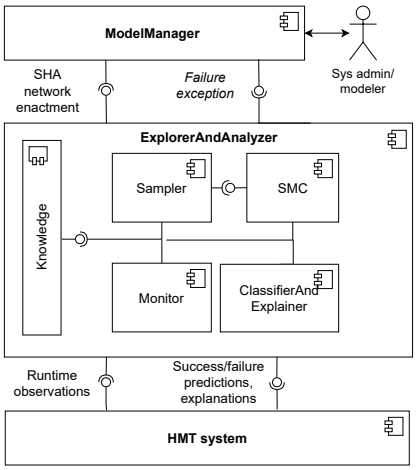


Fig. 2: High-level conceptual architecture of EASE.

TABLE I: Selected examples of HMT factors.

Variable	Agent	Type	Domain
Free Will	patient	categorical	{*dutiful*, *obedient*, *disobedient*}
Age/Fatigue	patient	categorical	{*young*, *elderly*}×{*healthy*, *sick*, *unsteady*}
Position x	doctor	continuous	$[0, 50]$ m
Position y	doctor	continuous	$[0, 8]$ m
Robot Speed	robot	continuous	$[30.0, 100.0]$ cm/s
Battery Level	robot	continuous	$[11.1, 12.4]$ V

network [3]. The formalism allows for the expression of non-linear dynamics of the physical variables and a stochastic characterization of human free will. Although the DSL is agnostic with respect to the verification tool, this work exploits an automated model-to-model transformation procedure to generate a UPPAAL model [11], [20].

Example 1 (Model instance): Figure 3 shows an extract of the SHA model instance that formalizes part of the running example of Sec. II.[3] The categorical *free will* parameter in Table I represents an uncertain aspect of the interaction captured by means of probabilistic edges. When a human agent is standing and receives the go command, she/he can either start or refuse walking with probability p_{do} and $1 - p_{\mathrm{do}}$, respectively. The value p_{do} depends on the free will profile specified by the modeler. The edges between locations *standing* and *walking*, labeled with condition $\gamma_{\mathrm{FreeWill}}$, model the human autonomous decision in terms of a dice roll. Another example is the age of the individual involved. Age determines the rate of fatigue and recovery of an individual in motion, i.e. an individual's muscular endurance/recovery capacity against a period of activity/rest. In Fig. 3 fatigue is modeled by means of a variable F, whose temporal behavior is defined through Ordinary Differential Equations f_{stand} and f_{walk} and value ranges in the interval $[0, 1]$. Parameters ρ and λ are the fatigue and recovery rates, which depend on the categorical parameter *fatigue profile* [20]. $\mathrm{F_{max}}$ is the maximum tolerated fatigue allowing the human to react to command go.

[3]Complete model available at github.com/LesLivia/hri_designtime.

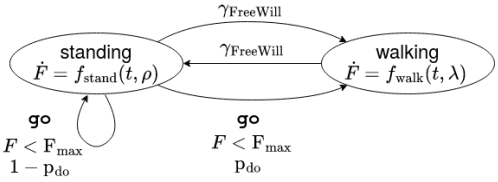


Fig. 3: Portion of SHA modeling a human walking.

B. Exploration and Analysis

Given an SHA network instance $\mathcal{M}[\bar{v}]$, dependability properties can be verified using Statistical Model Checking (SMC) [11] executed by the `SMC` component. SMC is cheaper than exhaustive state space exploration since it is based on a finite number of simulations of the target system. Based on these runs, SMC estimates the expression $\mathcal{P}_{\mathcal{M}[\bar{v}]}(\psi)$, which represents the probability of property ψ holding for a random run of $\mathcal{M}[\bar{v}]$. Formulae ψ are automatically-generated Metric Temporal Logic (MTL) properties [17] we use to check whether the HMT is successful within a given time limit. SMC determines the confidence interval $[p - \epsilon, p + \epsilon]$ for the actual probability of ψ holding. Thus, we say that $\mathcal{M}[\bar{v}]$ satisfies property ψ, denoted by $\mathcal{M}[\bar{v}] \models \psi$, if the lower bound value $p - \epsilon$ is greater than a user-defined probability threshold. As per Fig. 2, `SMC` interacts with the `Knowledge` component that stores the artifacts required by the formal analysis (i.e., SHA models, dependability properties, analysis outcomes).

Example 2 (Verification of dependability properties): Every service in our example (e.g., follow the doctor) is associated with a Boolean value $supplied_i$ that is true if the i-th service in the sequence succeeds. So, the generated dependability property ψ has the form $\Diamond_{\leq \tau} \bigwedge_i supplied_i$, where $\Diamond$ is the "eventually" operator and $\tau \in \mathbb{R}_{>0}$ a user-defined time limit.

The set of human-interpretable HMT factors induces a large, or even infinite, space of assignments that may break dependability requirements. According to Fig. 2, the component in charge of dealing with this issue is `Sampler` deliberately designed to increase knowledge through the exploration of the HMT factors rather than exhaustive enumeration. The sampling process can adopt different strategies based on the characteristics of the search space. Uniform random sampling is suitable in case the likelihood of successful and failing runs is comparable. If one of the two cases is a rare event, other search strategies can be adopted (e.g., simulated annealing, and genetic algorithms [5]). Thus, the search process can either maximize the lower bound $p - \epsilon$ or minimize the upper bound $p + \epsilon$, selectively pushing the evolutionary search towards successful or failing runs, respectively.

According to Fig. 1, the outcome of the exploration includes the *success region*, i.e., the region of the space that contains the assignments $\bar{v}$ such that $\mathcal{M}[\bar{v}] \models \psi$ holds; as well as the *failure region* that contains assignments such that $\mathcal{M}[\bar{v}] \not\models \psi$ holds. At the end of stage ②, having identified the success and failure regions, the robotic agents can be deployed in production.

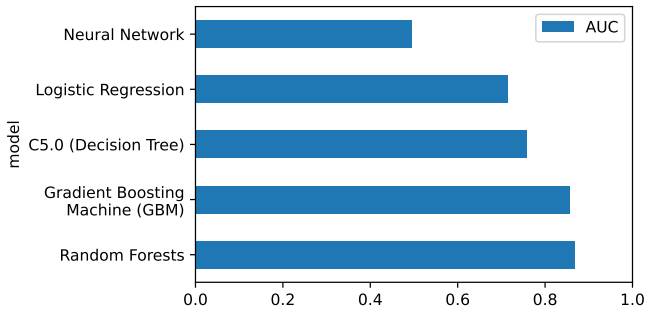
C. Prediction and Explanation

The `ClassifierAndExplainer` component retrieves from the `Knowledge` the data produced by the `Sampler`. Using this dataset, the `ClassifierAndExplainer` trains and validates a *binary classifier* [26] to bridge the gap between the offline and online stages. The classifier is built offline and it works online while the HMT mission is running to predict and explain the probability of success, as it learns the relationships between HMT factors (or features) and the corresponding mission outcome (i.e., the property ψ holds or not). Indeed, HMT factors may differ from those identified offline during exploration. In this case, the classifier can predict the mission outcome given new data points (assignments).

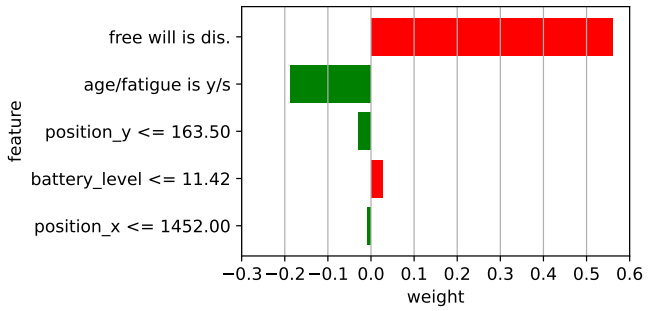
Example 3 (Training and validation): Figure 4a shows the outcome of an evaluation process considering different classification models (e.g., Random Forests, Neural Network) built using a dataset of 1000 assignments split into 80% training and 20% validation using stratified sampling. The evaluation is based on the Area Under the receiver operator characteristic Curve (AUC) technique [14]. As the Random Forests exhibits the highest performance (0.87), it is deployed in `ClassifierAndExplainer` to predict the mission outcome based on the HMT factors collected by `Monitor`.

According to Fig. 1, model-agnostic explanations are produced using either *local* or *global* interpretable ML techniques [24]. Since our goal is to provide humans with fast feedback on the ongoing run, in our current implementation, we rely on the Local Interpretable Model-agnostic Explanations (LIME) method [24] to probe repeatedly the classifier and understand the predicted outcome (either success or failure). LIME tests the effect of local variations synthetically injected into the current HMT factors sampled from the field. Starting from the current values $\bar{v}$, the `ClassifierAndExplainer` components uses LIME to generate a new dataset made of perturbations of $\bar{v}$ and the corresponding outcome given by the classifier. The new dataset is used to train an interpretable model (e.g., a regression model), which is weighted by the proximity of the perturbations to $\bar{v}$. If the model shows that the mission is likely to fail, the explanations illustrate the reason for the failure to the human teammates. The system administrator can also use the explanations to change the mission and trigger a new offline analysis stage.

Example 4 (LIME explanations): Figure 4b shows a local explanation generated by LIME. It shows the relative importance of the HMT factors and quantifies the extent to which their value contributes to the likelihood of observing a failure (or a success). For instance, the patient's free will *disobedient* (weight 0.56) represents the main reason for the failure. Other features, like the age/fatigue profile (weight -0.19), reduce the probability of failure, estimated by LIME to be 0.99 under the current assignment of HMT factors.



(a) AUC model performance



(b) LIME explanation

Fig. 4: Preliminary results from our running example.

IV. Discussion

In this section, we discuss our work, focusing on the limitations of the EASE approach. We also formulate future research questions that drive our research activity in HMT.

a) Mutual human-machine understanding: While human trust has been widely explored, a machine-centric perspective, where machines learn from humans and trust them, is a more recent multidisciplinary endeavor. A continuous learning cycle, aiming at obtaining a refined understanding of human behaviors and physical/emotional states, together with the explanations of machine behaviors entail mutual understanding between the two collaborating roles of the system. Quality guarantees, obtained through formal analysis of dependability properties, together with mutual understanding are, in our vision, the essential drivers for building mutual trust. The learning cycle requires multiple efforts in several research fields (e.g., machine intelligence, cognitive and affective model implementations, biophysical signal processing) but is key to the development of the next generation of HMT applications. The following questions deserve discussion by the research community: $\mathbf{Q_{a1}}$ is the common definition of trust [18] general enough to cover all these facets? $\mathbf{Q_{a2}}$ how can we realize machine-interpretable explanations for human behavior? $\mathbf{Q_{a3}}$ how can the concept of awareness be defined and how does it relates to the concept of mutual trust? $\mathbf{Q_{a4}}$ what are the societal and ethical concerns in this vision?

b) Reference Architecture for HMT: The identification of a software reference architecture as a shared vocabulary to discuss HMT applications represents an open challenge. Even though the conceptual architecture in Fig. 2 is an initial sketch specific to the EASE approach, it represents our initial effort in describing and reasoning about the architectural characteristics

of HMT systems that account for the key elements underlying trust, such as explainability. Research questions to be considered are: $\mathbf{Q_{b1}}$ what are the architectural characteristics of HMT systems? $\mathbf{Q_{b2}}$ what are the key concerns in the design of trustable and explainable HMT systems? $\mathbf{Q_{b3}}$ how to enact a monitoring and learning process to achieve bidirectional awareness? $\mathbf{Q_{b4}}$ how can the machine explain itself to humans (i.e., self-explainability)? $\mathbf{Q_{b5}}$ how to make the system able to adapt itself (i.e., self-adaptation) as needed for the achievement of its goals despite changes, when humans and machines interact more closely according to the HMT paradigm?

c) Formal guarantees in HMT: HMT systems are typically employed in critical domains (e.g., healthcare). These systems must be designed with strong, ideally provable, guarantees of dependability properties, or more in general, with quality aspects specified in a formal fashion. A natural starting point is to consider formal methods (e.g., SMC in our approach). However, such methods must deal with the inherent variability and uncertainty in humans as well as other relevant phenomena of the environment. This leads to a very high-dimensional space of uncertain and changing factors that cannot be reasonably explored following exact and exhaustive methods. We believe that a major challenge in the design of HMT is finding the right balance between formality and feasibility. So, future research questions are: $\mathbf{Q_{c1}}$ how can we formally model and guarantee trust in human behavior? $\mathbf{Q_{c2}}$ what are the relevant quality metrics for explanations in this context? $\mathbf{Q_{c3}}$ which formalisms can be used to model explanations? $\mathbf{Q_{c4}}$ what are the scalable approaches that can be reasonably adopted to generate formally-verified explanations?

V. Future Plans

We plan to extend our vision by providing an enhanced concept for a co-trust cycle of continuous human-machine learning, and the realization and experimentation of such a vision through the EASE framework and its concrete HMT-based software architecture. We also plan to investigate how to include into the SHA model other multi-disciplinary cognitive-related aspects and conduct variability model analysis. We want to mitigate uncertain (quality) attributes of the system under scrutiny and of the SHA model itself, especially with the availability of statistical inference techniques, such as Bayesian reasoning [7]. We would like to provide formal guarantees also for ethics-related concerns while designing, developing, deploying, and executing HMT systems. We also intend to define different levels of explainability [8] and the meta-requirements that an HMT system shall satisfy to meet the corresponding explainability level. Finally, we plan to extend the framework with the notion of prescriptive analytics. According to the analysis of the machine's progress and plans, the Exploration and Analysis layer could be endowed with a recommender component responsible for determining an optimal course of action and providing prescriptions for the machine and for the humans. This would make them more cognitive of critical situations and allow both to explore the solution space in a coordinated manner.

References

[1] International symposium on software engineering for adaptive and self-managing systems- seams series, 2006-2022. Accessible at https://dblp.org/db/conf/seams/index.html.

[2] International conference on autonomic computing and self-organizing systems- acsos series, 2020-2022. Accessible at https://dblp.org/db/conf/acsos/index.html.

[3] R. Alur, C. Courcoubetis, N. Halbwachs, T. Henzinger, P.-H. Ho, X. Nicollin, A. Olivero, J. Sifakis, and S. Yovine. The algorithmic analysis of hybrid systems. *Theor. Comp. Sci.*, 138(1):3–34, 1995.

[4] C. R. B. Azevedo, K. Raizer, and R. Souza. A vision for human-machine mutual understanding, trust establishment, and collaboration. In *2017 IEEE CogSIMA*, pages 1–3, 2017.

[5] C. Blum and A. Roli. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Comput. Surv.*, 35(3):268–308, sep 2003.

[6] J. Cámara, M. Silva, D. Garlan, and B. R. Schmerl. Explaining architectural design tradeoff spaces: A machine learning approach. In *ECSA*, volume 12857 of *LNCS*, pages 49–65. Springer, 2021.

[7] M. Camilli, R. Mirandola, and P. Scandurra. Taming model uncertainty in self-adaptive systems using bayesian model averaging. SEAMS '22, New York, NY, USA, 2022. ACM.

[8] M. Camilli, R. Mirandola, and P. van Scandurra. XSA: explainable self-adaptation. In *2022 37th IEEE/ACM International Conference on Automated Software Engineering (ASE)*, 2022. To appear.

[9] E. Cioroaica, B. Buhnova, and T. Kuhn. Predictive simulation within the process of building trust. In *ICSA Comp.*, pages 47–48. IEEE, 2022.

[10] J. Cleland-Huang, A. Agrawal, M. Vierhauser, M. Murphy, and M. Prieto. Extending mape-k to support human-machine teaming. SEAMS '22, page 120–131. ACM, 2022.

[11] A. David, K. G. Larsen, A. Legay, M. Mikučionis, and D. B. Poulsen. Uppaal SMC tutorial. *Intl. Journal on Software Tools for Technology Transfer*, 17(4):397–415, Aug 2015.

[12] R. de Lemos. Human in the loop: What is the point of no return? SEAMS '20, page 165–166. ACM, 2020.

[13] B. Gebru, L. Zeleke, D. Blankson, M. Nabil, S. Nateghi, A. Homaifar, and E. Tunstel. A review on human–machine trust evaluation: Human-centric and machine-centric perspectives. *IEEE Trans. on Human-Machine Systems*, 52(5):952–962, 2022.

[14] J. A. Hanley and B. J. McNeil. The meaning and use of the area under a receiver operating characteristic (ROC) curve. *Radiology*, 143(1):29–36, 1982. PMID: 7063747.

[15] W. Z. Khan, Q.-u.-A. Arshad, S. Hakak, M. K. Khan, and Saeed-Ur-Rehman. Trust management in social internet of things: Architectures, recent advancements, and future challenges. *IEEE Internet of Things Journal*, 8(10):7768–7788, 2021.

[16] B. C. Kok and H. Soh. Trust in robots: Challenges and opportunities. *Current Robotics Reports*, 1(4):297–309, Dec 2020.

[17] R. Koymans. Specifying real-time properties with metric temporal logic. *Real-Time Systems*, 2(4):255–299, Nov 1990.

[18] J. D. Lee and K. A. See. Trust in automation: Designing for appropriate reliance. *Human Factors*, 46(1):50–80, 2004. PMID: 15151155.

[19] L. Lestingi, M. Askarpour, M. M. Bersani, and M. Rossi. Formal verification of human-robot interaction in healthcare scenarios. In *SEFM*, volume 12310 of *LNCS*, pages 303–324. Springer, 2020.

[20] L. Lestingi, C. Sbrolli, P. Scarmozzino, G. Romeo, M. M. Bersani, and M. Rossi. Formal modeling and verification of multi-robot interactive scenarios in service settings. FormaliSE '22, page 80–90. ACM, 2022.

[21] N. Li, J. Cámara, D. Garlan, and B. R. Schmerl. Reasoning about when to provide explanation for human-involved self-adaptive systems. In *IEEE ACSOS 2020,*, pages 195–204. IEEE.

[22] N. Li, J. Cámara, D. Garlan, B. R. Schmerl, and Z. Jin. Hey! preparing humans to do tasks in self-adaptive systems. In *SEAMS 2021*, pages 48–58. IEEE.

[23] A. M. Madni and C. C. Madni. Architectural framework for exploring adaptive human-machine teaming options in simulated dynamic environments. *Systems*, 6(4), 2018.

[24] C. Molnar. *Interpretable Machine Learning*. 2 edition, 2022.

[25] S. Sagar, A. Mahmood, Q. Z. Sheng, J. K. Pabani, and W. E. Zhang. Understanding the trustworthiness management in the social internet of things: A survey, 2022.

[26] V. Vapnik. An overview of statistical learning theory. *IEEE Trans. on Neural Networks*, 10(5):988–999, 1999.