On the Origins of Self-Explainability in Cyber-Physical Systems: Model-Based and Data-Driven Approaches

Laleh Akbari[1], Rolf Drechsler[1,2]
[1]Institute of Computer Science, University of Bremen, Germany
[2]Cyber-Physical Systems, DFKI GmbH, Bremen, Germany
{laleh,drechsler}@uni-bremen.de

Abstract

Cyber-Physical Systems (CPS) operate in complex environments where system behavior results from the interaction of software, physical processes, and changing external conditions. This inherent complexity makes it difficult to fully understand, verify, or explain how such systems behave. Explainability has therefore become an important system property, helping users make sense of system decisions, supporting developers in diagnosing unexpected behavior, and enabling systems to justify actions of other cooperating systems. This review classifies work in explainability according to a fundamental question: which system artifact is used to generate an explanation? Based on this criterion, we distinguish between model-based and data-driven approaches. The discussion considers how these approaches generate explanations, what assumptions they rely on, and the constraints that affect their practical use, while also pointing to issues that remain insufficiently addressed.

1 Introduction

The growing complexity of cyber-physical and autonomous systems increasingly challenges users in understanding system behavior. Self-explainable systems aim to mitigate this challenge by generating explanations from within the system itself. Instead of relying on external post-hoc analysis, such systems are expected to generate explanations grounded in their own knowledge. However, existing approaches differ substantially in where this explanatory knowledge originates. Some rely on explicit system models created at design time, while others derive explanations from data observed during system operation.
We approach this work with the following question in mind: Which system artifact is used to generate explanations? Applying this criterion, this paper organizes prior work in two major classes, and discusses hybrid approaches as combinations of both, addressing trade-offs between different approaches, common challenges and open research questions.
Our goal is to provide a structured comparison that supports readers in selecting an appropriate methodological direction based on their requirements. The presented methods are representative exemplars chosen to illustrate design principles within each category rather than a thorough survey with comprehensive coverage and recent state-of-the-art advances.

2 Classification Criterion

In a self-explainable system, an explanation should not be treated as an on-demand output; instead, it emerges from system artifacts such as models and logs that reflect assumptions about behavior and causality, as well as operational context. As a result, reliability of an explanation is fundamentally shaped by these underlying artifacts.

This work has (partially) been funded by the Deutsche Forschungsgemeinschaft (DFG) project no. 513623283 as part of the Research Training Group CAUSE.

We distinguish two major classes of explanatory artifacts:
Model-based: artifacts that are explicit representations of system behavior or causal relationships created by system designers.
Data-driven: artifacts that are learned or inferred from observations of system behavior, simulations, or data collected during operation.
This distinction focuses exclusively on the origin of explanatory knowledge and does not concern when explanations are generated or how they are presented. First, we describe representative model-based and data-driven approaches, then discuss hybrid approaches that combine elements of both.

3 Model-Based Approaches

Model-based work treats explainability as a property engineered into the system from the start. It relies on artifacts like formal specifications, behavior models, or domain ontologies and keeps them connected to runtime behavior so the system can justify decisions in terms of states, rules, and requirements.

3.1 Adaptive Feedback Loop

Schwammberger et al. [1] propose a model-driven process for making autonomous systems self-explainable by reusing existing formal specification models, timed automata in particular. The goal is to construct explanation models that can be used in the previously developed framework [2], an extension of the traditional MAPE loop used in self-adaptive systems.
Instead of developing separate models specifically for explanation, they reuse artifacts created during system design. They derive causal diagrams from the controller's timed-automata semantics which serve as explicit explanation models. The process then refines this diagram by removing parts irrelevant to the explanation purpose, tailors it to different explainee types, and finally enriches it

with environmental/semantic information such as traffic-rule meaning and natural-language annotations.
Feasibility is illustrated through a case study involving an automotive maneuver scenario.
The key advantage of this approach is reusing models, which reduces the effort required to generate explanation models. The generality of the approach allows it to be integrated into existing systems. Also, the model can be updated at runtime, and adapted to different users. However, expert involvement is required during model extraction, and frequent updates to explanation models at runtime remain challenging.

3.2 Ontology and Knowledge Graph

Aryan et al. [3] propose a web-based ExpCPS framework for explainable Cyber-physical energy systems. Its primary goal is to generate explicit explanations of system behavior by making internal events and their relationships understandable to users. The framework focuses on explanation generation rather than control or event detection, which are assumed to be handled externally.
The proposed approach is based on an ontology-driven knowledge graph that integrates heterogeneous data sources. These include static topology and dynamic measurement data from a simulated smart grid, external contextual data, and tacit expert knowledge. Causality is modeled using relations defined by domain experts that allow abstract causal knowledge to be contextualized into concrete relationships between system components.
The framework is evaluated using a simulated smart grid scenario, while system topology, events, and causality knowledge are represented in a knowledge graph. When an event requires explanation, a breadth-first graph traversal algorithm is applied to derive explanation paths based on causality-enriched relations.
The approach's main strengths lie in its structured integration of diverse data and its generic ontology design, which can be extended with domain-specific knowledge. However, it relies on expert knowledge to define causality relations and does not take causal loops into account. Evaluation is limited to a small simulated case, with no assessment of scalability or performance.

3.3 Formal Explanation Patterns

Fey et al. [4] propose a generic explanation pattern for self-explanation in digitally controlled systems. The work demonstrates that explanation generation can be abstracted independently of any specific application domain. The proposed model considers the explainer, the addressee, and the explanation component as separate entities, and describes how beliefs, consistency, and counterfactual reasoning are represented. It also addresses the resolution of inconsistent beliefs across these components.
The pattern is instantiated using Mealy machines, which makes it possible to derive explanation paths directly from system behavior. This instantiation also illustrates how explanations can be adjusted to suit the addressee's level of knowledge. Through this process, the approach connects system inputs, outputs, and internal states in a structured way to the resulting explanations.
The work contributes toward the development of reusable explanation patterns applicable across different cyber-physical system domains. However, it does not discuss concrete implementation strategies for large-scale systems and does not include an experimental evaluation.

4 Data-Driven Approaches

Data-driven work derives explainability from observed behavior using logs, measurements, and patterns. Instead of relying on a complete design-time model, these approaches learn interpretable structures or rules, classify observed behaviors into reason-like categories, or construct local explanations.

4.1 Anomaly Classification

Ziesche et al. [5] address the problem of explaining anomalous behavior in autonomous systems. They do not assume availability of a model and rely on supervised learning.
In the first step, anomalous behavior is detected using a neural network-based model. Detected anomalies are then grouped into a small number of abstract classes with similar underlying causes. This grouping reduces the number of explanations that need to be considered. Each class is associated with a predefined natural-language statement that provides a coarse explanation.
The approach is evaluated using simulated autonomous driving scenarios. The reported results show high detection and classification accuracy. However, the reliance on simulated data and the limited detail of the generated explanations restricts the conclusions that can be drawn about real-world applicability.
This component is kept separate from safety-critical parts of the system, which gives it potential for integration with existing systems. However, it requires large labelled data sets, yet CPS generate lots of unsupervised data during operation.

4.2 Decision-Tree-based Classification

Plambeck et al. [6] focus on explaining CPS behavior that depends on environmental conditions and system configurations. In such systems, deterministic digital components interact with continuous and often noisy influences from the physical world, which makes it hard to determine which factors are responsible for observed outcomes.
To address this, the approach integrates decision-tree learning with clustering. The method begins by using application-specific categories. If no such categories are known, the grouping is obtained via clustering, enabling the approach to proceed without requiring predefined labels for the output space. These grouped outputs are then used to train a decision tree that relates environmental and configuration parameters to observable behavior. In this way, non-linear dependencies can be captured while keeping the resulting explanations understandable.
The approach is evaluated using a LiDAR-based real-time localization system mounted on a forklift operating in a logistics test environment. The results indicate that the

learned decision trees consistently identify influential factors and produce explanations that are easy to follow. However, the evaluation is limited by a small dataset and a restricted set of experimental conditions. Even though such a method is able to automatically identify relevant features, providing quality data and proper clustering is critical.

5 Hybrid Approaches

Hybrid approaches leverage design-time artifacts, such as transition functions, simulation models, and system topology, and combine them with run-time information learned from operational data, such as sensor measurements. We view hybrid methods as a bridge between model-based and data-driven approaches to self-explainability.

5.1 Causal Approximation

Reyd et al. [7] introduce CIRCE, a method that approximates a sufficient cause for a specific runtime observation without relying on a global causal model. Rather than constructing a complete causal graph in advance, the approach adapts a local explainability technique to generate a decision-tree surrogate model using data obtained from system simulations.

The method assumes a discrete Markovian state sequence and access to a system transition function. When a user poses a query, CIRCE performs targeted simulations, trains a local surrogate model, and extracts an explanation that is specific to the situation.

The approach is illustrated through a smart-home case study, which highlights that the objective is not to recover the preconditions encoded by existing rules, but rather to identify contextual causes that explain observed system behavior. In addition, a quantitative evaluation is conducted using the F1 score as the primary metric to assess the method across multiple evaluation questions.

The authors report advantages such as context-specific explanations, scalability with respect to the number of variables, and applicability to stochastic systems. At the same time, the method depends on the Markov assumption, requires a simulation model, and may become unstable when explanations rely on rare or poorly represented counterfactual cases.

5.2 Event to State Causal Tracing

Schreiberhuber et al. [8] introduce a domain-independent framework for explaining system states in cyber-physical systems, motivated by smart-grid-like operations where operators must understand anomalies. The framework stores system topology, detected events and states, and expert causal knowledge in a knowledge graph, and generates explanations as causal paths or root-cause trees for a queried state.

System topology, events and states, and causal knowledge form an integrated data model that, together with an IoT data source, creates the Information layer. At the Function layer, an Event Detection Module analyzes IoT data to extract events of interest using either simple rules or pattern recognition models, and the State Derivation Module maps events to states. The Explanation Module maps expert-defined type causalities to instance-level actual causality by checking observed state sequences and filtering candidates with temporal and topological constraints. Explanations are obtained by recursively traversing the derived causal links from the trigger state to candidate root causes.

A feasibility study analyzes one month of sensor measurements from an EV charging garage. The approach offers traceable, inspectable explanations but depends on expert-crafted knowledge, treats event detection as a critical black box, and provides limited evaluation in terms of user study or scalability.

6 Comparison

Distinguishing between model-based and data-driven explanatory artifacts reveals an important trade-off that designers of explainable systems must navigate. From a system engineering perspective, this comparison suggests a design choice: should explainability be built primarily by modeling, by learning from data, or by combining both?

Model-based approaches can provide high faithfulness with respect to the model and offer traceability to system requirements. Their drawbacks include modeling effort and potential mismatch between model assumptions and real-world behavior.

Data-driven approaches can capture real-time behavior and adapt to changing conditions, but may suffer from noise and distribution shift. Their explanations may be difficult to validate causally.

Hybrid approaches can improve interpretability by grounding explanations in a system model while still leveraging data-driven learning capabilities.

Table 1 summarizes key differences between the two main approaches with respect to criteria relevant to explanation design and evaluation. Hybrid methods combine both paradigms; therefore, their pros and cons are encompassed by the two existing categories.

In practice, the choice of explanatory artifact constrains not only what kinds of system behavior can be explained, but also the scenarios supported for explanation.

This kind of decision-making is relevant both to deployed systems, where explainability has to be introduced afterwards, and to systems still under development, where designers must decide when to incorporate explainability and which artifacts to retain to support it.

Table 1 Comparison of approaches across key aspects

Dimension	Model-based	Data-driven
Computation effort	Upfront modeling; runtime reasoning	Training + inference; optional clustering
Robustness	Strong in-model; brittle out-of-model	Strong in-distribution; fragile in rare cases
Common failure reasons	Incomplete model; Incorrect model	Overfitting; spurious/noisy signal
Best-fit scenarios	Safety-critical; heavily regulated	Complex designs; evolving systems
Evaluation metrics	Coverage; fidelity to specification	Consistency; out-of-distribution performance

7 Discussion and Future Work

This paper proposed a perspective for organizing the explanation literature by focusing on the system artifact that provides the explanatory basis.
While preparing this review, we noticed that an important issue still needs to be addressed: comparing explanation methods with respect to explanation generation time, explanation size, and user-trust scores. Such comparisons would help characterize the practical differences between these approaches under consistent evaluation criteria.

8 Literature

[1] M. Schwammberger and V. Klös, "From Specification Models to Explanation Models: An Extraction and Refinement Process for Timed Automata," Sep. 28, 2022, arXiv:2209.14034.

[2] M. Blumreiter et al., "Towards Self-Explainable Cyber-Physical Systems," Aug. 13, 2019, arXiv:1908.04698.

[3] P. R. Aryan et al., "Explainable cyber-physical energy systems based on knowledge graph," in Proceedings of the 9th Workshop on Modeling and Simulation of Cyber-Physical Energy Systems, Virtual Event: ACM, May 2021, pp. 1–6.

[4] G. Fey, M. Fränzle, and R. Drechsler, "Self-Explanation in Systems of Systems," in 2022 IEEE 30th International Requirements Engineering Conference Workshops (REW), Aug. 2022, pp. 85–91.

[5] F. Ziesche, V. Klös, and S. Glesner, "Anomaly Detection and Classification to enable Self-Explainability of Autonomous Systems," in 2021 Design, Automation & Test in Europe Conference & Exhibition (DATE), Feb. 2021, pp. 1304–1309.

[6] S. Plambeck et al., "Explaining Cyber-Physical Systems Using Decision Trees," in 2022 2nd International Workshop on Computation-Aware Algorithmic Design for Cyber-Physical Systems (CAADCPS), May 2022, pp. 3–8.

[7] S. Reyd, A. Diaconescu, and J.-L. Dessalles, "CIRCE: a Scalable Methodology for Causal Explanations in Cyber-Physical Systems," in 2024 IEEE International Conference on Autonomic Computing and Self-Organizing Systems (ACSOS), Sep. 2024, pp. 81–90.

[8] K. Schreiberhuber et al., "Towards a State Explanation Framework in Cyber-Physical Systems," SIGENERGY Energy Inform. Rev., vol. 4, no. 4, pp. 142–154, Oct. 2024.