Towards Multi-View Test Specification in CPPS Engineering

Dietmar Winkler*†‡, Serafima Sherstneva*†, Stefan Biffl†‡
* Christian Doppler Laboratory for Security and Quality Improvement in the Production System Lifecycle,
† Inst. of Information Systems Eng., TU Wien and ‡ CDP, Austria. Email: {firstname.lastname}@tuwien.ac.at

***Abstract*—In context of Industry 4.0, the engineering of Cyber-Physical Production Systems (CPPSs) need to incorporate a heterogeneous set of engineering disciplines, data models, and artefacts. The quality of related data models and engineering artefacts is success-critical for the engineering process and the planned CPPS. Software and System tests aim at improving the quality of a CPPS. However, in CPPS, risk cases are often unknown and insufficiently covered by systematic testing methods, especially in heterogeneous environments. In this paper, we describe a multi-view test specification (MVTS) approach based on a risk analysis to systematically derive regular and negative/error test cases in CPPS engineering. We build on the PPR Asset Network (PAN) that provides the structure of a CPPS from product, process, resource perspective and their dependencies, and the Failure Mode and Effect Analysis (FMEA) to efficiently identify risks in CPPS engineering. We conceptually evaluate the MVTS approach with domain experts in a feasibility study to show benefits and limitations in context of traditional software testing. First results showed benefits of the MVTS approach with the help of the PAN and FMEA to systematically capture risks and derive test cases. While the execution of test cases is often limited to the regular systems behavior, negative test are often not executed because of possible physical damages. However, negative test cases can raise the awareness of possible critical risks during CPPS planning and design.**

***Index Terms*—Cyber-Physical Production Systems, PPR Asset Network, Multi-View Test Specification.**

I. Introduction

The engineering of Cyber-Physical Production Systems (CPPSs) in the Industrie 4.0 context requires efficient coordination of discipline-specific views (i.e., multi-views) and data exchange to support collaboration within an engineering team [2]. However, engineering teams need to include a heterogeneous set of engineering disciplines (such as mechanics, electrics, and software engineering), and data model and engineering artifacts, that belong to these disciplines and views. Dependencies between data models and artifacts are often implicitly given but not explicitly expressed. Therefore, data exchange and coordination is often limited [4] and make systematic testing inefficient, risky, and error-prone. Furthermore, the testing of CPPSs is often limited to regular cases with limited consideration of error cases that might cause physical damages. In CPPS environments, the *Failure Mode and Effect Analysis (FMEA)* is an established approach for risk assessment and mitigation [12]. However, FMEA models are often used as isolated source of information by individual experts and disciplines. Therefore, we see the need to support systematic test specification for CPPS engineering projects based on identified risks and related root causes.

In this paper, we present a **multi-view test specification (MVTS) approach based on a risk analysis to systematically derive regular and error cases in CPPS engineering.**

Based on this goal, we derive three main research questions:
RQ1. What are the basic requirements for a multi-view test specification process approach? Based on related work and discussions with domain experts, we identify a set of requirements to support systematic testing in CPPS environments.

RQ2. What are the main process steps for systematically defining test cases in CPPS environments? We build on the PAN generation process approach [13] and extend the process and the meta-model with focus on test case generation.

RQ3. What are the benefits and limitations of the MVTS approach? Based on discussions with domain experts, we explore benefits and limitations of the proposed approach.

We conceptually evaluate the multi-view test specification approach with domain experts to explore benefits and limitations in contrast to a traditional software testing approach. First results showed benefits of MVTS with the help of the PAN and FMEA to systematically capture risks and derive regular and error test cases. While regular test cases demonstrate the correct functional behavior of the CPPS, negative (or error) test cases can at least raise the awareness of possible critical risks during CPPS planning and design.

The remainder of this paper is structured as follows: Section II summarizes background and related work. We describe the MVTS approach in Section III and present the conceptual evaluation in Section IV. Section V discusses the results and limitations, and concludes the paper.

II. Background and Related Work

This section summarizes background and related work on Cyber-Physical Production Systems (CPPSs), the PPR Asset Network (PAN), and the Failure Mode and Effect Analysis (FMEA) as foundation for test case generation.

Cyber-Physical Production Systems (CPPSs) are required for flexible industrial production in Industrie 4.0 [6]. Their design involves several engineering disciplines, such as mechanical, electrical, and software engineering that represent different views on the CPPS project from discipline-specific

978-1-7281-2989-1/21/$31.00 ©2021 IEEE

perspectives [2]. Individual aspects of an Industrie 4.0 component include on *product* information (i.e., what product to build), *process* information (sequence of steps to construct the product), and *resource* information (resources needed to build the products within the production process) that form a PPR Asset [9]. Dependencies between different PPR Assets of a CPPS form a PPR Asset Network (PAN).

Therefore, the **PPR Asset Network (PAN)** represents the structure of a CPPS including product, process, and resource information and dependencies between these PPR assets [13]. Biffl *et al.* [4] describe a coordination artifact that includes multi-disciplinary views on assets encapsulating change dependencies and engineering knowledge to enable/improve coordination capabilities to coordinate changes in CPPS engineering. Based on these coordination artifacts, risk assessment and mitigation can help engineers to systematically trace observed effects back to root causes across the PAN [3].

In CPPS engineering, the **Failure Mode and Effect Analysis (FMEA)** is an established approach for risk assessment and mitigation. Basically, the FMEA is an engineering and quality assurance method to identify and mitigate risks and potential production failures before a customer can be effected by poor product performance [12], [14]. A typical FMEA identifies known and potential failure modes along with their corresponding causes and effects, prioritizes them, and defines corrective actions. For example, the process FMEA focuses on failure modes occurring during the manufacturing and/or the assembly process; the design FMEA addresses product-level failure modes [10]. However, FMEA concepts and are often used isolated with limited connection to related engineering disciplines and models. In this paper, we aim at linking the FMEA to the PAN approach to support systematic testing.

Software and System Tests Software and System Testing focuses on the identification of defects in (software) engineering components. Therefore, testing focuses on (a) exploring *intended effects*, eg., whether or not a product (characteristic) matches expected requirements and/or a production process meets performance goals. Furthermore, the definition of software tests can be used to check for risks which could cause *undesired effects*. Antao *et al.* [1] reported on requirements for Testing IoT systems. In Feldmann *et al.*, the authors investigated how to manage inconsistencies in model-based engineering in automated production systems [5]. Based on Software Engineering best practices [11] we used equivalence classes [7] to support the test case specification for CPPS based on CPPS structure (i.e., the PAN) combined with risk analysis based on the FMEA.

III. Multi-View Test Specification Approach

In this section we focus on (a) requirements for testing CPPSs based on the structure of the CPPS and the FMEA; (b) present the MVTS process; and (c) the meta-model (based on [13]) for modeling test cases in context of the PAN.

Requirements. Based on discussions with domain experts from industry and academia (in the production automation systems domain), we have identified a set of requirements for an efficient multi-view test specification support in CPPS environments. Note that this set of requirements for the solution represent an excerpt of important requirements to be supported by the MVTS approach. The set of requirements include capabilities for (a) the representation of cause-effect relationships; (b) assessing and prioritizing risks; (c) supporting the definition of equivalence classes; and (d) supporting multi-views for test specification and test automation.

MVTS Process. Figure 1 presents a high-level process overview (in yellow) and more detailed sub-process steps related to individual main steps (in gray). Note because of space restrictions, we do not cover feedback cycles in this figure. Main process steps include: (1) *Building the PPR Asset Network (PAN).* Individual stakeholders identify PPR assets and dependencies, build the PAN and execute a set of verification and validation steps. See Winkler *et al.* [13] for more details. (2) *Conduct risk assessment*. Based on the PAN and the FMEA appraoch domain and quality experts conduct a risk assessment to identify root causes based on observed effects [3]. (3) *Identify risk drivers.* We used the ISO SQUARE standard [8] as foundation for eliciting critical requirements and risk drivers for testing a CPPS. This selection is usually done by domain experts for prioritization of critical requirements to be tested. (4) *Specify Test Cases.* This step includes the definition of equivalence classes related to critical parameters (identified during the risk assessment) and the creation of test cases. Note that we separate between regular and negative/error test cases (see example in Section IV). In CPPS, it is also important to address test cases that could be executed, measured, and controlled by test experts, e.g., for setting up the test cases. Measurement includes if the expected behavior could be measured and/or controlled. Finally step (5) includes the execution and/or observation of the test case execution.

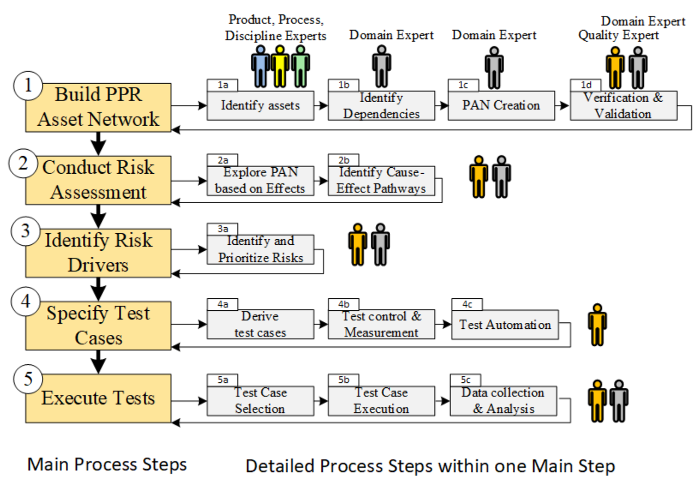

Fig. 1. MVTS approach.

MVTS Meta-Model. Based on the PAN meta-model [3], we extended the model to support test cases. Figure 2 presents the adaptation/extension of the meta-model for test cases (marked in yellow). Based on identified root causes (derived from the FMEA application [3]), a sequence of steps for test specification is defined. This sequence represents test scenarios that include a set of individual test cases [11]. Expected results directly address observed effects that need to tested.

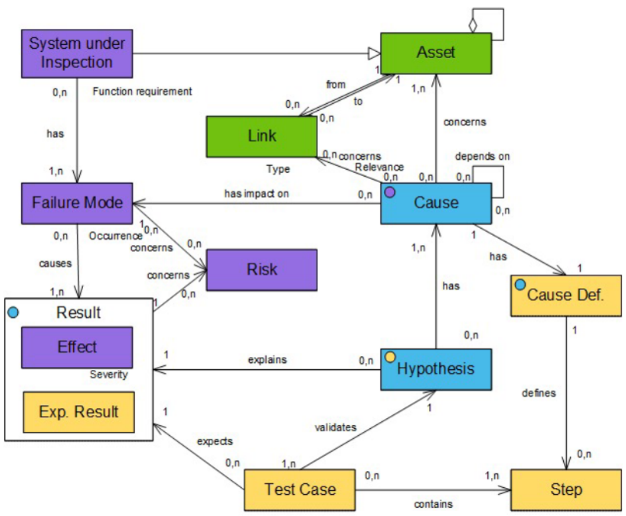

Fig. 2. Meta-Model of the PPR Asset Network (PAN) with Test Cases.

The process and the meta-model represent the building blocks for MVTS setup (and execution (out of scope of this paper)). Note that the model uses a graph database (such as Neo4J[1]) to efficiently store and query the PAN, FMEA, and test cases.

IV. Conceptual Evaluation

This section represents an example of the MVTS application in a general application domain. We have conceptually evaluated MVTS in three use cases: (a) in the software engineering domain with the example of a *bet-and-win platform*, (b) in the automation systems domain, where robots and conveyor belts are involved, as available in the *Center for Digital production (CDP)*[2], and in the automotive systems domain. Note that in this paper, we focus on the general approach. Figure 3 presents an example PAN (product, process and resources), the link to the FMEA model (FMEA resources), and test cases.

	EC_1	EC_2	Can measure	Can control	Levels of control
C_1	V_{11}	V_{12}	yes	yes	quality engineer
C_2	V_{21}	V_{22}	yes	yes	quality engineer
C_3	V_{31}	V_{32}	yes	yes	mechanical engineering

TABLE I
Equivalence Classes based on Effect Causes.

For an identified cause (Cx) Table I, equivalence classes for regular (EC1) and negative/error (EC2) value ranges have been defined (error cases in orange color). In addition attributes have been added, i.e., *can measure*, *can control*, and *level of control*. This refers to the possibility to run/control test cases or simply observe test case execution or observe defined parameters that could lead to risks and failures. Note that the level of control includes stakeholders who are capable of controlling the test case, e.g., by setting up the pre-conditions for test case execution.

Table II presents a snapshot of selected test cases (TCn) based on equivalence classes (combinations of values based on equivalence classes). Note that the test case spedification part includes expected results (i.e., effects to be observed (cf. Figure 2). Note that we also included Hypotheses (Hx) that formally represent assumption that result in test cases and expected results. Furthermore, we included columns that indicate the actual result, and capabilities for control and measurement.

TC N	C_1	C_2	C_3	H_n	Exp. Result	Act. Result	Can control	Can measure
TC 1	V_{11}	V_{21}	V_{31}	H0	OK		yes	yes
TC 2	V_{12}	V_{21}	V_{31}	H1	NOK		yes	yes
TC 3	V_{11}	V_{22}	V_{31}	H1	NOK		yes	yes
...							yes	yes
TC 8	V_{12}	V_{22}	V_{32}	H1	NOK		yes	yes

TABLE II
Test Case Specification based on Equivalence Classes.

We have discussed expected capabilities for the MVTS approach with cooperation partners from acadmia and Industry in various domains, i.e., in software engineering, automation systems engineering, and the automotive domain. Based on a selected set of requirements (cf. Section III, we have analysed the capabilities in traditional software testing and the MVTS approach. Table III presents the summary of the capability assessment. While in traditional software testing equivalence classes are supported, non of the other requirements are supported. In the MVTS approach, all requirements (R1-R5) are well supported and represent a valuable input for all cooperation partners in their related domains.

ID	Requirement	Traditional Testing	MVTS Approach
R1	Representation of Cause-Effect Relationships	-	++
R2	Risk Assessment and Prioritization	-	+
R3	Support of Equivalence Classes	+	+
R4	Multi-View Test Specification	-	++
R5	Multi-View Test Automation	-	++

TABLE III
Analysis of traditional testing approaches and the MVTS approach.

V. Conclusion and Future Work

In this paper, we aim for a multi-view test specification (MVTS) approach based on a risk analysis to systematically derive regular and error cases in CPPS engineering. We have identified a set of five basic requirements (RQ1) that are needed to support the goal. Section III includes a process approach for MVTS application including main steps and related sub-steps (RQ2). The conceptual evaluation with cooperation partners include a set of benefits (RQ3) that refer to

[1]Neo4J: https://neo4j.com/
[2]Austrian Center for Digital Production (CDP): acdp.at/

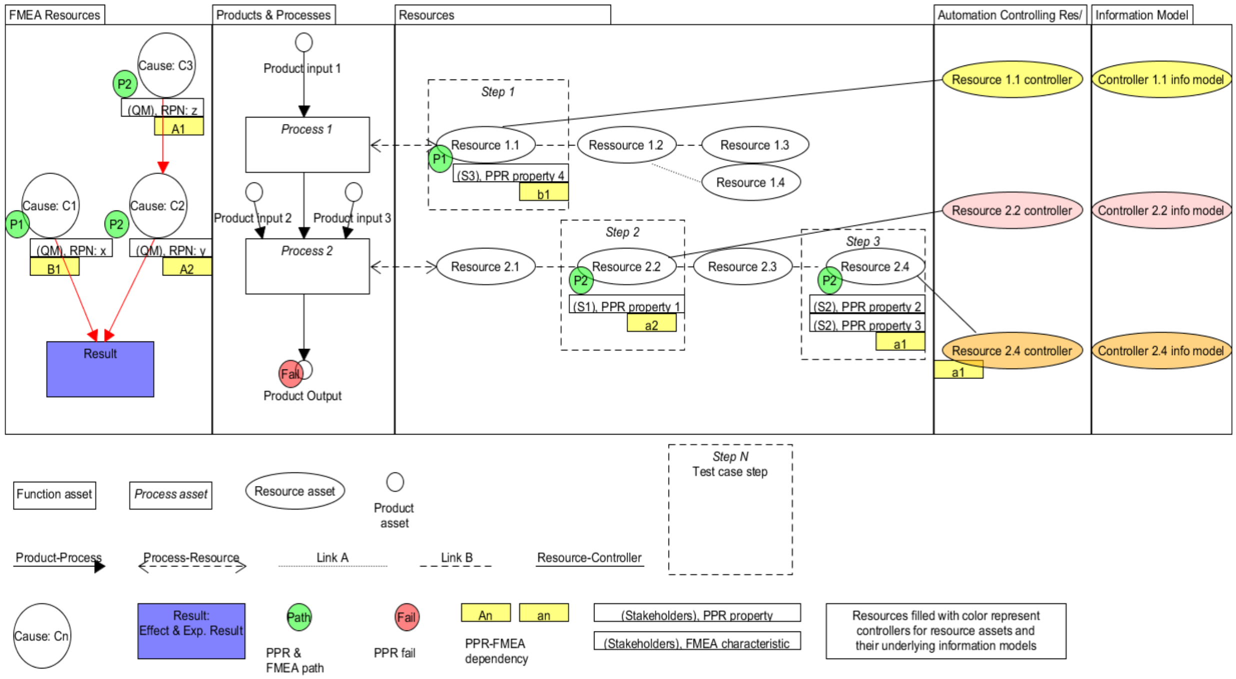

Fig. 3. Mutli-View Test Case Specification (MVTS) Application Example.

requirements R1-R3 in Table III. However, the main limitation focuses on the initial effort for setting up the PAN and FMEA model as foundation for test case specification and design. However, this additional effort will pay off during later engineering phases and/or operation where test cases can be automatically executed and/or defined critical parameters can be measurend and/or observed.

Future work will include a set of case studies in industry environments in the selected application domains to formally evaluate the MVTS approach.

Acknowledgment

The financial support by the Christian Doppler Research Association, the Austrian Federal Ministry for Digital & Economic Affairs and the National Foundation for Research, Technology and Development is gratefully acknowledged.

References

[1] Liliana Antão, Rui Pinto, João Reis, and Gil Gonçalves. Requirements for testing and validating the industrial internet of things. In *2018 IEEE International Conference on Software Testing, Verification and Validation Workshops (ICSTW)*, pages 110–115, 2018.

[2] Stefan Biffl, Arndt Lüder, and Detlef Gerhard. *Multi-Disciplinary Engineering for Cyber-Physical Production Systems: Data Models and Software Solutions for Handling Complex Engineering Projects*. Springer, 2017.

[3] Stefan Biffl, Arndt Lüder, Kristof Meixner, Felix Rinker, Matthias Eckhart, and Dietmar Winkler. Multi-view-model risk assessment in cyber-physical production systems engineering. In S. Hammoudi, L. Ferreira Pires, E. Seidewitz, and R. Soley, editors, *Proceedings of the 9th Int. Conf. on Model-Driven Engineering and Software Development, MODELSWARD*, pages 163–170. SCITEPRESS, 2021.

[4] Stefan Biffl, Juergen Musil, Angelika Musil, Kristof Meixner, Arndt Lüder, Felix Rinker, Danny Weyns, and Dietmar Winkler. Industry 4.0 asset-based coordination in production systems engineering. In *Proc. of the 23rd IEEE Int. Conf. on Business Informatics (CBI)*. IEEE, 2021.

[5] Stefan Feldmann, Sebastian JI Herzig, Konstantin Kernschmidt, Thomas Wolfenstetter, Daniel Kammerl, Ahsan Qamar, Udo Lindemann, Helmut Krcmar, Christiaan JJ Paredis, and Birgit Vogel-Heuser. Towards effective management of inconsistencies in model-based engineering of automated production systems. *IFAC*, 48(3):916–923, 2015.

[6] Roland Heidel, Martin Hankel, Udo Döbrich, and Michael Hoffmeister. *Basiswissen RAMI 4.0: Referenzarchitekturmodell und Industrie 4.0-Komponente Industrie 4.0*. Beuth Verlag, 2017.

[7] Wen-ling Huang and Jan Peleska. Complete model-based equivalence class testing. *International Journal on Software Tools for Technology Transfer*, 18(3):265–283, 2016.

[8] ISO/IEC. Iso/iec 25041:2012: Systems and software engineering - systems and software quality requirements and evaluation (square) - evaluation guide for developers, acquirers and independent evaluators, international standard, 2012.

[9] Miriam Schleipen and Rainer Drath. Three-view-concept for modeling process or manufacturing plants with AutomationML. In *14th IEEE Int. Conf. on Emerging Technologies and Factory Automation*, pages 1–4. IEEE, 2009.

[10] Kapil Dev Sharma and Shobhit Srivastava. Failure mode and effect analysis (FMEA) implementation: a literature review. *Journal of Advanced Research in Aeronautics and Space Science*, 5:1–17, 2018.

[11] Andreas Spillner and Tilo Linz. *Software Testing Foundations: A Study Guide for the Certified Tester Exam-Foundation Level-ISTQB® Compliant*. dpunkt. verlag, 2021.

[12] Diomidis H Stamatis. *Risk Management Using Failure Mode and Effect Analysis (FMEA)*. Quality Press, 2019.

[13] Dietmar Winkler, Petr Novák, Kristof Meixner, Jirí Vyskocil, Felix Rinker, and Stefan Biffl. Product-process-resource asset networks as foundation for improving CPPS engineering. In *26th IEEE International Conference on Emerging Technologies and Factory Automation, ETFA 2021, Vasteras, Sweden, September 7-10, 2021*, pages 1–4. IEEE, 2021.

[14] Zhongyi Wu, Weidong Liu, and Wenbin Nie. Literature review and prospect of the development and application of fmea in manufacturing industry. *The International Journal of Advanced Manufacturing Technology*, pages 1–28, 2021.