Strojniški vestnik - Journal of Mechanical Engineering 60(2014)12, 815-831
© 2014 Journal of Mechanical Engineering. All rights reserved.
DOI:10.5545/sv-jme.2013.1556
Review Scientific Paper
Received for review: 2013-11-20
Received revised form: 2014-03-25
Accepted for publication: 2014-04-25

Towards the Maintenance Principles of Cyber-Physical Systems

Santiago Ruiz-Arenas[1,2,*] – Imre Horváth[1] – Ricardo Mejía-Gutiérrez[2] – Eliab Z. Opiyo[1]
[1]Delft University of Technology, Faculty of Industrial Design Engineering, the Netherlands
[2]Universidad EAFIT, Design Engineering Research Group, Colombia

Cyber-physical systems (CPSs) are rapidly proliferating in different applications. Their system features significantly differ from those of linear complex systems (LCSs). Consequently, they pose novel challenges with regard to ensuring the dependability of system operation. Maintenance of CPSs raises new theoretical and practical issues. To guarantee a high level of dependability, new and efficient system maintenance principles should be explored and operationalized in various contexts. This paper reports on the first results of the authors' work in this direction. A comprehensive literature review has been conducted with the objective of identifying the specific features of LCSs and CPSs. We analysed the major maintenance principles and approaches currently applied to complex systems to see how they can be applied to CPSs. We found that the existing maintenance principles have various relationships with CPSs: (i) some of them cannot be considered in the context of CPSs due to incongruent system features, (ii) some of them can be adapted due to certain partial congruencies, and (iii) some of them can be applied directly due to the congruency of some system features of LCSs with CPSs. It was also found and demonstrated through a number of practical examples that many specific maintenance principles need to be developed for CPSs. We assert that the system features of CPSs without parallel in LCSs primarily reveal what sort of new maintenance principles and approaches are needed. The ultimate goal of our on-going research is to define and test these new maintenance principles. In this paper, we identify and define these principles, starting from the unique system features of CPSs and aiming to develop a maintenance advisory system.

Keywords: complex systems, cyber-physical systems, maintenance principles, failures, maintenance advisory system

0 INTRODUCTION

Technical maintenance is an important multifaceted set of activities performed to preserve the operation of the system in a dependable and optimal state. It involves activities such as inspection, adjustment, replacement, repair, overhaul, and renewal. Maintenance increases the useful life and reliability of systems, reduces the size, scale and number of repairs, and the need for emergency repairs as well as the overall costs while increasing safety and security. Various policies have been conceptualized, which are operationalized through various maintenance principles that have been widely studied in the context of conventional engineering systems. However, maintenance becomes a challenging issue as the heterogeneity and complexity of systems increase. In general terms, the higher the amount of uncontrollable conditions, the more uncertain the physical world becomes [1].

The overall objective of our research is to address the challenge of maintenance of cyber-physical systems (CPSs), and to define a possible set of generic principles that can be applied in the development of system-specific maintenance plans and actions. We presumed that some maintenance principles of linear complex systems (LCSs) may be considered, but also that many new maintenance principles will likely be needed due to the distinct system features of CPSs. The literature suggests that the maintenance principles of LCSs have been derived by considering two complementary maintenance strategies, i.e. preventive maintenance and corrective maintenance. The principles associated with them lent themselves to the development and application of various design methods, such as redundancy, that are nowadays commonly used to enhance reliability, fault tolerant operation, and ease of repair [2]. In the framework of preventive maintenance, time-based plans are generated for periodic and systematic testing and the replacement of fault-prone elements to prevent sudden failures. Modern predictive maintenance intends to apply sensing technologies to monitor the status of the physical system components in real time, and to initiate the necessary maintenance actions. It also envisages equipping systems with reasoning capabilities to support automated decision making on the necessity of maintenance. With the goal of restoring its intended operation, corrective maintenance is carried out after the malfunctioning, failure, or breakdown of system component has been detected.

As mentioned above, the known principles of systematic maintenance have been developed and applied in the case of LCSs, whose behaviour is linear and remains so even under intensively varying operational circumstances. This is an important issue because CPSs typically operate as highly dynamic systems, while LCSs operate as steady-state systems. CPSs may even sometimes operate under unpredictably varying environmental conditions [3] and their performance may be mainly influenced by the effects of such external factors [1]. The resulting non-linear operation makes it difficult to

[*]Corr. Author's Address: Delft University of Technology, Faculty of Industrial Design Engineering, Landbergstraat 15, Delft, the Netherlands, s.ruizarenas@tudelft.nl

predict momentary system behaviour and to ensure permanent system availability. Our literature review explored a significant knowledge gap in the field of the maintenance of non-linear systems. The methods and tools currently used for controlling the physical part and the cyber part of CPSs are very different and often do not fit adequately [4]. To this end, systems science attempts to integrate knowledge from different engineering disciplines [5] and to facilitate a concurrent management of the computational part and the physical part of cyber-physical systems [6].

Should the research begin out of the knowledge of complex systems science and/or out of the currently evolving science of CPSs in order to derive these particular maintenance principles, or can some of the known maintenance principles of LCSs be reused? This is the central research question of this paper. The reasons this question has both theoretical and practical significance are that (i) many research activities have concentrated on transferring the fault tolerance-related system features of LCSs to CPSs, (ii) only a few of them have focused on the systematic development of maintenance principles that could provide efficient preventive or corrective maintenance solutions for CPSs, and (iii) the distinctive system features of CPSs have not been sufficiently studied for their influences on maintenance. Therefore, we first systematically surveyed and analysed the major system features and the currently applied maintenance principles of LCSs. Then, we investigated the applicability of the maintenance principles of LCSs to CPSs based on a comparison of the system features of LCSs and CPSs.

We note here that, for our purposes, system features have been perceived to consist of functional, structural, operational, interaction, application and behavioural attributes and characteristics that differentiate artefacts and service combinations. A first observation has been that the strong interdependency of the components of CPSs will probably make them more vulnerable to errors, attacks and failures [7]. We also observed that the available methods used in the context of maintenance of CPSs are premature and suffer from some fundamental limitations, such as their limited ability to deal with uncertain situations [8].

The reasoning model shown in Fig. 1 has been used in our explorative investigation. Our objective was to establish relationships between the system features of CPSs and the maintenance principles applied for LCSs. First, we investigated in which sense the system features of LCSs and CPSs differ from each other (Arrow A). Next, we considered which maintenance principles are commonly used in LCSs (Arrow B). Following that, we analysed what maintenance principles can be considered for CPSs, taking into account the differences between the system features of LCSs and CPSs (Arrow C). As the title of this paper suggests, the ultimate objective of our research is to explore specific maintenance principles for high-end CPSs (Arrow D).

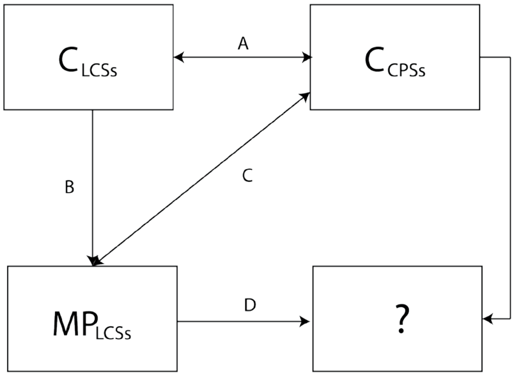


Fig. 1. *Reasoning model used in knowledge aggregation*

Therefore, the paper has been structured as follows: Section 1 presents the analysis of the system features of ordinary and complex systems, and Section 2 those of CPSs. This created a knowledge platform for the follow-up reasoning about the relevance of the maintenance principles of LCSs to CPSs. Section 3 analyses the principles currently applied in the maintenance of LCSs. It discusses various maintenance policies commonly applied to physical devices, as well as various failure management mechanisms that have been used in embedded systems. Section 4 projects the known maintenance principles of LCSs to generic CPSs. Section 5 discusses in which manner the maintenance principles with application potentials can be considered in developing general maintenance principles for CPSs. Section 6 discusses the need for dedicated additional maintenance principles for CPSs. Section 7 presents some examples that illustrate the consequences of applying the identified maintenance principles in a cyber-physical greenhouse. Section 8 evaluates the findings and proposes further research activities.

1 EXPOSITION OF SYSTEM FEATURES OF LINEAR COMPLEX SYSTEMS

Science differentiates systems based on complexity and behaviour. The complexity of systems is a measure influenced by factors, such as the (i) the number of components included in the system, (ii) the type of components that constitute the system, (iii) the number of sub-systems of different scales, (iv) the interconnections among co-located components, (v) the communications among geographically dislocated components, (vi) the interactions of the system with stakeholders, and (vii) the connections of the system with its environments [2]. In addition to these, (viii) the heterogeneity of the components, and (ix) the distinct material, energy and information flows within the system are also factors influencing system complexity. From an engineering point of view, system operation or behaviour can be linear or non-linear. Therefore, in our interpretation, systems can be categorized as simple and complex and as linear or non-linear. A system is linear if: (i) it is functionally and structurally reductionist, (ii) its output is directly proportional to its input, and (iii) it satisfies the superposition principle. Simple and compound reductionist engineering systems belong to this category. CPSs can also be linear systems, but the overwhelming majority of them fall into the category of non-linear complex systems. As such, they can be found in several alternative forms, e.g. as complicated, adaptive, evolving and replicating complex systems. Systems belonging to the category of non-linear complex systems have some sort of learning and self-organizing capabilities.

We had to be pragmatic in our research because adaptive, evolving and replicating complex systems are still in their infancy from an implementation point of view, and thus the knowledge about their behaviour and maintenance needs remains limited. In addition, there are many open theoretical and practical issues concerning the design, implementation and operation of these systems. Consequently, their utilization into the practical life remains rather limited. This explains why we have made the choice to focus only on complicated cyber-physical systems (C-CPSs) in the first phase of our research. Furthermore, in theory non-linear complex systems (NLC) CPSs are able to optimize their overall performance in cases of largely varying environmental conditions, changing internal relationships, and operational discontinuities. Likewise, we do not intend to address the issues related to so-called self-healing systems, which supposedly have the capability of automatically regaining functionality when components break down, or significant perturbations occur in the system.

Ordinary systems (OSs) are, usually, simple small-scale systems in which a single cause produces a single effect, which makes them reducible, composable and predictable in modelling and design. The basic assumption is that a small change in the input implies a small change in the output [9]. Examples of OSs are electro-mechanical systems, such as a coffee maker or a refrigerator, which have pre-programmed or adjustable control devices, and operate under steady-state conditions. These types of systems usually have only one energy source, one integrated functional unit, and one interface unit. As a consequence of their pre-programmed nature, no changes or updates are possible in their embedded software after their release to the market, and they cannot manage emergent real-time data [10].

LCSs are complex in the sense that they are composed of a diverse set of interconnected components, but do not have any capability to reorganize their structure, or change their designed functionality. The overall operation of LCSs is a union of the operation of their components. In other words, the aggregated functionality of the components determines the operation of the system as a whole, and no emergent behaviour occurs due to the interaction among the components or within the environment in which the system is embedded. They are closed systems with centralized architectures and control functions, which are aligned to the tendency of the so-called disappearing computer (that hides software components in a physical device) [11]. Therefore, these systems are controlled by a microprocessor-enabled embedded software system, and they typically perform (much) more complex operations than OSs [12].

The control function is realized through multiple feedback loops through which the software monitors and controls the whole system and its components in an optimized way [2]. The functional components of LCSs intensively interact with each other and the surrounding environment [11]. This type of system may be geographically distributed and decentralized, equipped with multiple energy sources, may have repetitions in the functional units, and the components may communicate by using wireless technologies. LCSs are widely used, for instance, in the automotive, electronics, avionics, railways, telecommunication, health, and security sectors [11]. In these systems, the maintenance of the physical components may be carried out by using preventive and corrective maintenance procedures.

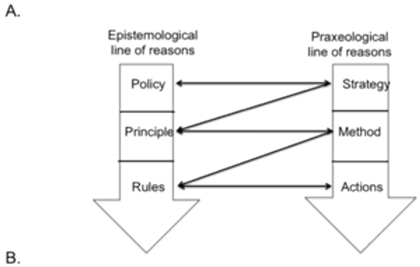
2 SYSTEM FEATURES OF CYBER-PHYSICAL SYSTEMS

CPSs came about as a result of the emergence of faster computer processors, the miniaturization of electronic components, broader communication bandwidths, and seamless integration of networked computing with everyday systems [13]. They blend physical technologies, software and middleware technologies, and cyber technologies. Future systems will make more extensive use of synergic technologies, which integrate hardware and cyber technologies [14]. Physical technologies enable the implementation of artefacts that can be recognized, located, operated, and/or controlled in the physical world [15]. Cyber technologies are used for capturing, analysing and processing sensed signals and data produced in the physical world for decision-making. Synergic technologies enable not only a borderless interoperation between physical and cyber elements, but also a holistic operation of the whole system. The design of the physical and computational aspects is becoming an integrated activity [16].

As mentioned above, CPSs link the physical world with the cyber world through the use of multiple sensor and actuator networks integrated under an intelligent decision system [17]. In other words, CPSs combine sensing and actuation with computation, networking, reasoning, decision making, and the supervision of physical processes [18]. With a view to their emergent nature, it seems expedient to differentiate low-end and high-end implementations of CPSs based on the extensiveness and sophistication of the resultant integrity [14]. Low-end implementations are linearly complex, closed architected, distributed and networked, sensing and reasoning enabled, smart and proactive, (often embedded and feedback controlled) collaborative systems. High-end implementations are non-linearly complex, open and decentralized, heterogeneous and multi-scale, intelligent and partly autonomous, self-learning and context-aware systems.

The systems belonging to the latter class of CPSs display organization without any predefined organizing principle and change their functionality, structure and behaviour by self-learning, self-adaption, or self-evolving. The previously mentioned C-CPSs are low-end implementations because they are not supposed to change their functionality or architecture, but to optimize their behaviour, for instance, energy efficiency (e.g., due to the necessity to operate during an extended period of time) [19], while operating under dynamically changing operating conditions or unforeseen circumstances. Some of these systems should operate in real-time applications and provide a precisely timed behaviour [20] as well as achieving a synergic interaction between the physical and the cyber worlds by integrating computational and physical processes [21].

The cyber and physical parts of the systems are interconnected and affect each other through information flows [22]. Due to this functional synergy, the overall system performance is of higher value than the total of the individual components [23]. This synergy is particularly important in the case of high-end CPSs, which exhibit properties such as self-organization [24]. In general, CPSs strive toward a natural human-machine interaction that also extends to the human cognitive domain [25]. These kinds of systems are also capable of exhibiting extensive remote collaboration [26]. Unlike LCSs, CPSs also work on non-dedicated networks [27]. CPSs are often connected in a hierarchical manner, as systems of systems, in which one system monitors, coordinates, controls and integrates the operation of other systems [28]. For this reason, they can be considered to be multi-dimensional complex systems [29]. Based on their functionality and characteristics, high-end CPSs can be used in areas such as transportation, health care, and manufacturing [28].


A.
Epistemological line of reasons
Praxeological line of reasons
Policy
Principle
Rules
Strategy
Method
Actions

B.

Epistemological line of reasoning		Praxeological line of reasoning	
Policy	Determines and provides maintenance directions through the formulation of specific rules. Every policy implies specific principles	**Strategy**	Defines the overall planning of maintenance in a company or application, including the identification of principles to be applied and to which extent.
Principle	Fundamental knowledge resource of maintenance policies	**Method**	Defines the way of execution of a maintenance action
Rule	Provides theoretical guidance for the execution of particular actions	**Action**	Execution and completion of an element of the strategy

Fig. 2. *Clarification of the main terms: A) relationships of the terms, and B) interpretation of the terms*

Some CPSs are mission critical systems (MCSs) because their correct functioning is critical to the success of a mission, provisioning an essential supply, or safeguarding security and well-being [30] and [31]. These are the systems that ensure proper and continuous operation of (for example) nuclear plants, automated robot control systems, and automatic landing systems for aircraft [32]. Any failure in MCSs can lead to loss of human life and to damage to the environment, and may cause losses in terms of supply and cost [33]. However, their operation is always characterized by the presence of uncertainty. This introduces challenges from the point of view of the dependability, maintenance and repair of mission critical non-linear cyber-physical systems [14]. In the long run, it is crucial to comprehensively analyse what the maintenance of these systems theoretically, methodologically, and practically means, and how it can be implemented in different systems.

3 OVERVIEW OF THE MAINTENANCE PRINCIPLES APPLIED TO LCSs

The terminologies related to system maintenance do not seem to be uniform in the literature. Terms such as "maintenance strategy", "principles" and "policy" are used with various interpretations and meanings, often even interchangeably or confusingly. In general, a policy is defined as a collection of rules that, depending on the most essential state variables, precisely specifies what to do in a particular situation [34]. From a managerial point of view, "strategy" is described as the definition of long-term goals, objectives and courses of action for a company, and the allocation of resources for the achievement of such objectives [35].

There are some other basic terms used in literature whose definition is often taken for granted. These are "principle", "method", "rule" and "action". We adopted the Oxford dictionary definitions [36]. Therefore, a principle is interpreted as "a fundamental source or basis of something"; a method as a "particular procedure for accomplishing or approaching something" in a systematic way, a rule as a set of explicit understood regulations, and an action as the logically separable procedural element of doing something. The application of these terms in our maintenance context is presented in Fig. 2A. This figure shows the interrelationships between the above-defined significant terms and separates them according to whether they are of epistemological (knowing) or praxiological (executional) flavour. Fig. 2B summarizes the above interpretation of the two groups.

Maintenance seeks to ensure the permanent availability of a system through the application of its basic principles. Consequently, these principles should be applicable to any system, including CPSs. However, it has been recognized that this claim is not apparent with regards to CPSs, because these systems should be considered differently from a maintenance perspective due to the inherent heterogeneity of their physical, software and cyberware components. The high level of synergy makes the maintenance of the three basic kinds of components inseparable from each other. This is in contrast with the classic view of the abstract machine architecture in which systems are composed of hardware and software components, which operate with a lower level of synergy [37]. The maintenance of LCSs is based on this classic view and, consequently, its maintenance principles are also based on it (Fig. 3).

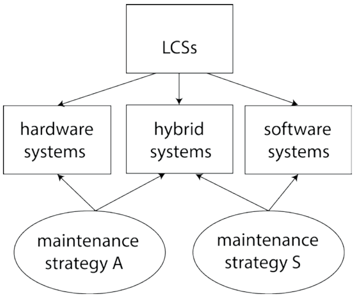


Fig. 3. *Articulation of maintenance strategies for LCSs*

The maintenance principles of LCSs are applicable from both physical (artefactual systems) and cyber (software systems) perspectives. The physical part maintenance is based on the assumption that every component of the system has a limited life cycle and, therefore, may be subjected to wear or breakdown. Therefore, for the physical part, the main (global) principles have been maintaining system availability and doing so in a cost-effective way. Based on this principle, two main approaches have been developed: preventive maintenance (PM) and corrective maintenance (CM). Principles of PM aim to avoid failures before they occur through preventive actions, such as revisions, exchanging components and repairs. The principles of CM allow a system to operate up to the occurrence of failures if the consequences of failures are not critical, or do not have an effect during a particular operation period [38]. PM may be conducted according to the principle of time-based maintenance (TBM) or of condition-based maintenance (CBM). The TBM principle (P_1) entails scheduling maintenance actions [39]. Therefore, knowledge management techniques should be applied in order to determine a schedule for conducting revisions, exchanging of components and repairs, while the CBM principle (P_2) is based on the completion of inspection activities by which maintenance actions will be initiated and completed. This principle can be applied to component that do not exhibit failure predictability or fail randomly, while scheduled maintenance principle may be applied to those components that show evident signs of wearing [38].

In contrast, CM can be implemented by following the principles of (i) failure-based maintenance (FBM), (ii) opportunistic-based maintenance (OBM), and (iii) design-out maintenance (DOM). OBM and DOM belong to this group of principles because both of them assume failures to occur in order to be identified. Therefore, the FBM principle (P_3) considers repair and making changing components once a failure had occurred [40], it deals with maintenance only if failures or breakdown occurs [41]. The principle of OBM (P_4) suggests completing general inspection of all of the components when any of them fails. It has been reported that combining PM activities can lead to savings in terms of system cost [42]. Therefore, the principle of OBM states that there is an opportunity to conduct general maintenance when a maintenance intervention is required for other components [43]. Finally, the principle of DOM (P_5) aims to use redesign to avoid the causes of failure. This principle is usually applied when breakdowns frequently occur [44]. The application of one or another principle depends on how likely the components exhibit wear characteristics and how random components fails.

In the context of maintenance of software (and knowledge) intensive information systems, the primary assumption is that there are obviously no physical (e.g. wearing) processes. Therefore, the primary maintenance principle for software systems is that failures should be self-avoided and self-managed by the system. Several principles may be applied in the case of software system for a proper fault management [45] and [8]. These may be based on the consideration of: (i) fault prevention, (ii) fault removal, (iii) fault detection and isolation, (iv) fault forecasting, (v) fault tolerance, and (vi) fault reporting.

The principle of fault prevention (P_6) seeks to avoid the occurrence of faults through preventive actions [46], while the principle of fault detection and isolation (P_{10}) aims to detect and determine whether a fault occurred in a particular system, by attempting to autonomously detect these faults and to isolate the affected component [47]. Having a different objective, the principle of fault removal (P_7) seeks to reduce the number of faults and their severity [48]. The principle of fault forecasting (P_9) allows predicting failures and their impact, based on the fault records [49]. It entails estimating the incidence and consequences of faults, based on the present number of faults. The principle of fault tolerance (P_8) aims to assure the continuity of system operation, despite the presence of faults, errors or attacks [50]. The principle of fault reporting (P_{11}) is based on alerting the user or operator in case there is a fault in order to allow actions to be taken [51]. All of the aforementioned principles for the software (and cyber) side are focused on autonomously taking actions such as identification, diagnosis, isolation, repair and/or reporting.

In addition to the abovementioned principles, the principle of e-maintenance and the principle of vaccination, which are preferred and commonly implemented in auto-immune systems (AIS), have also been identified and worked out. The principle of e-maintenance is based on the exploitation of particular ICT affordances for enhancing the effectiveness of maintenance decisions [52]. It entails making use of information technologies to exploit data required in decision-making. This principle is mostly applied in manufacturing plants where full system availability is required. The principle of vaccination has a natural analogy. In the context of human beings, the principle of vaccination seeks to create immunity to any particular disease by introducing a soft version of the disease in the body, and to generate a memory of the pathogens. This natural principle has been extended to software systems, and now it allows the adaptation of the system behaviour against new and evolving attacks [53]. It is usually applied to systems whose complexity levels are higher than of OSs. Since we have focused in our research on systems that need external management of maintenance, rather than taking care of it by themselves, we will not deal with the principles of e-maintenance and vaccination.

As for the current state, it is apparent that an extensive set of maintenance approaches are available for LCSs, regardless of whether purely physical or purely software systems are considered. It can be argued that cyber-physical systems need some sort of combination or even a blending of these in order to be able to provide system dependability. As our survey and analysis has revealed, from the physical perspective, maintenance is conducted to avoid general system failures, or to reduce their probability, based on repairs, spare changes, and revision activities. From the software (and cyber) perspective, maintenance is orientated to the control functions of systems and they are usually kept operational through fault management. In the latter case, the intension is to assure system operation even in the presence of failures, or when facing any type of faults.

Our other observation has been that both hardware and software systems' possible failures are addressed by maintenance principles that have been developed for LCSs. The combination of the hardware- and the software-systems oriented maintenance principles works properly in LCSs systems. However, when the level of complexity of the target system increases and the operation of the system becomes non-linear, these changes cause a higher level of unpredictability. This has consequences on the applicability of the maintenance principles. In the case of complicated

CPSs, it is also important to analyse how the interactions among the system components happen under varying operational conditions, and how they may affect the operation of the system as a whole. In other words, it is necessary to investigate how maintenance principles should be adapted to meet the functional requirements of cyber-physical systems.

4 PROJECTING THE MAINTENANCE PRINCIPLES OF LCSs TO CPSs

A high-level, three-tier structure is proposed in [54] as a reasoning model for the maintenance of non-linear complex systems such as CPSs; it includes an environmental tier, a service tier, and a control tier. The environmental tier is related to the physical devices, the service tier is a typical computing environment with services in a service-oriented architecture (SOA), and the control tier is for decision making. This reasoning model clearly differentiates the methods and techniques that can be applied for artefactual systems and for software systems (Fig. 3). This differentiation is significant from the aspect of applying the traditional maintenance principles of LCSs in CPSs. Since we have decided to focus on complicated systems, this reasoning model has many limitations. Consequently, we have based our study on the model proposed by [14] which identifies three generic constituents: (i) physical technologies, (ii) cyber technologies, and (iii) synergic technologies. This model specifies that the maintenance policy of CPSs should consider these three constituents in their synergy (Fig. 4.) We use this reasoning model to facilitate the simultaneous consideration of the hardware, software and information content-related issues and principles of maintenance.

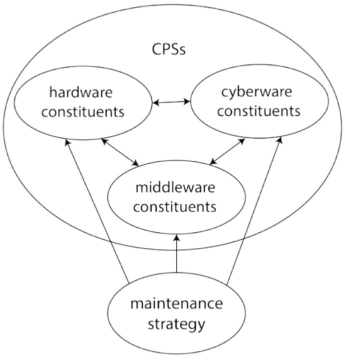


Fig. 4. *Doctrine of integral maintenance for CPSs*

The analysis concerning the congruencies of the system features of LCSs and CPSs, extended with the analysis of maintenance principles of LCSs in the previous chapter, provided a basis for us to determine which maintenance principles are transferable to CPSs. We investigate each of the maintenance principles from the aspect of transferability below. In the assessment, we take into account the similarities of the system features of the two kinds of systems, as well as the importance of the functions that they perform.

- *Schedule maintenance actions* (P_1)

This maintenance principle is appropriate for LCSs that operate continuously. In order to avoid system failures, the different common working cycles of the physical components are taken into consideration, together with signals concerning their state of wear. The states of the components and their criticality together determine when a maintenance action should be executed. As for the transferability of this principle to CPSs, we can argue that CPSs are dynamic systems whose actual operations cannot always be predicted with a high degree of probability. This dynamic operation affects the common working cycles of the physical components and thus the frequency of maintenance may be different for each of them. In the context of our exploration, it means that although this principle cannot be neglected it needs adaptation to be adequate for CPSs.

- *Support maintenance actions by monitoring activities* (P_2)

Even in the case of traditional LCSs, implementation of this maintenance principle requires augmentation with agents for operation monitoring. In principle, these agents can be embedded in CPSs that normally have a set of physical sensors, sensor networks, or software sensors. For this reason, this principle can be transferred to CPSs. The necessary maintenance actions of the system will be determined based on permanent monitoring, which can be applied even in the case of a non-linear behaviour of CPSs.

- *Conduct maintenance actions once a failure has occurred* (P_3)

Application of this principle to CPSs is far from straightforward, in particular when mission critical systems are considered. In the case of MCSs, continuous availability is not negotiable, and risk in the operation is usually not tolerable. It implies that general system failures, as well as cascade failures, should be avoided through engineering actions, or by dedicated system functions. This however implies that this maintenance principle should not be considered in the case of mission critical CPSs.

- *Conduct general maintenance once any of the system components fail* (P_4)

This principle is not associated with any particular system feature of LCSs and C-CPSs, but considers the entire system. For this reason, its applicability raises a concern regarding the fact that the overwhelming majority of C-CPSs are complex, decentralized systems, whose subsystems and modules may be characterized by some level of autonomy and operation profile. In other words, they may have and operate according to their own maintenance scenarios. Consequently, they may not need to go through general maintenance when failure in other subsystems or modules occurs. It has to be noted that this principle may be relevant to complex components.

- *Redesign to avoid the cause of failures* (P_5)

This principle is also not related to any particular system feature of LCSs or CPSs. The application of this principle entails re-designing the components and features of the system if they prone to be the source of recurrent failures [39]. Redesigning may be needed or be advantageous because recurrent failures can significantly affect the overall availability of CPSs and can increase the costs of operation. In the case of CPSs, further consideration is needed if the redesign is to focus on the hardware, software, or the information contents constituents, or any combination of them.

- *Avoid failures in the system by preventing the occurrence of faults* (P_6)

This principle works best in the case of systems such as LCSs whose operations are not highly dynamic. The reason is that the application of this principle assumes the conducting of reliability tests during the system development and installation stages. Consequently, possible failures and failure modes that may negatively affect the operation of the systems can be explored or predicted. However, CPSs are dynamic and highly complex systems, and their testing before full-scale operation cannot be exhaustive. The currently used testing approaches cannot cover all aspects of the operation of CPSs. Therefore, transferring this principle to CPSs necessitates adaptation.

- *Reduce the amounts of faults and their severity* (P_7)

This principle is applied during the design stage of the system with the objective of avoiding functional and structural failures. Although this principle in theory can be considered applicable to CPSs, it should be adapted to their system features. Multi-aspect fault propagation prevention methods and failure interaction evaluation methods will certainly be needed to make this principle applicable and efficient.

- *Assure continuity of system operation despite the presence of faults* (P_8)

This principle can be transferred to CPSs because the intelligence (i.e. sensing, reasoning and actuator capabilities) embedded in these systems can support its implementation by detecting the faults of physical components and the malfunctioning of software components, and activating protection mechanisms. Decentralization of the system operation and control also allows conducting an adaptive resource management. In addition, the application of preventive and corrective measures such as redundancy, reconfiguration and replacement, may be used to avoid complete system failure. However, the large possible number of functional connections complicates the identification of affected components and the prevention of fault propagation. Other advanced characteristics of CPSs, such as self-organization and self-adaptation, can take care of assuring the continuity of system operation, and in general, facilitate the application of this principle. Therefore, high-end CPSs will be able to transfer tasks from failed components to components that are working properly while the fault is eliminated. As a result, this principle can be the main principle for the maintenance strategy of CPSs.

- *Predicting failures on the system* (P_9)

The main objective of applying this principle is to forecast faults and failures and systematically avoid them. It is the most effective principle for systems with limited complexity and operational linearity, such as LCSs. Incongruities of the system features of CPSs and LCSs affect its applicability to CPSs, as the currently applied predictive and/or probabilistic models developed for LCSs are not appropriate for CPSs. However, the self-diagnosis and self-adaptation capabilities of CPSs may contribute to the effective application of this principle.

- *Detect and isolate faults* (P_{10})

Traditionally, this principle is operationalized rather “manually”. However, the intelligence and autonomy of CPSs may significantly influence the application of this principle. CPSs may be equipped with capabilities to detect fault events autonomously, and may analyse the consequences of emergent faults. Furthermore, the interactions among components allow the extraction of information for different devices to conduct performance tests, which contribute to the detection of whether the system operates properly.

Decentralization makes it possible to properly manage resources during the execution of these tests. This resource management will avoid system overloads and, therefore, the occurrence of faults or errors in processing. As a result, we argue that P_{10} can directly be applied to CPSs.

- *Alerting the operator in case of fault* (P_{11})

This principle can be transferred to CPSs because its implementation only entails application of information technologies in physical devices as long as LCSs and CPSs have physical features. The differences between system features do not affect the applicability of this principle.

5 OPERATIONALIZATION OF RELEVANT MAINTENANCE PRINCIPLES FOR CPSs

The above analysis shows that there are different relationships between the generic system features of CPSs and the maintenance principles that have been used in LCSs. Four categories of relationships can be identified: (i) non-applicable principles (P_x), (ii) adaptable principles (P_a), (iii) exportable principles (P_e), and (iv) additional (P_n) principles (Fig. 5). Two maintenance principles, i.e. P_3 and P_4, belong to the category of non-applicable principles (i.e. $P_x = (P_3, P_4)$). These seem to be problematic in the context of maintenance of CPSs due to their probable criticality and decentralization. Conducting maintenance actions once a failure has occurred in mission critical CPSs is not logical. Likewise, conducting general maintenance on a system that is capable of managing the consequences of failures in their separate or autonomous parts is also illogical.

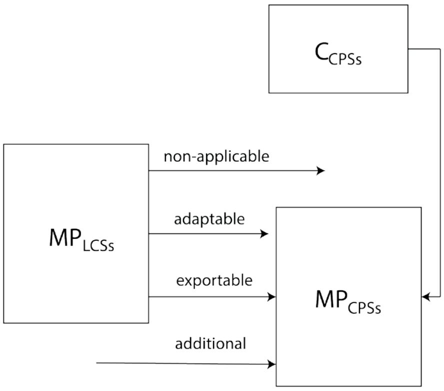


Fig. 5. *Roadmap towards maintenance principles for CPSs*

The rest of the principles seem to be applicable but in different ways. There are principles that can be applied without any modifications. These have been named exportable principles. They are: $P_e = (P_2, P_5, P_8, P_{10}, P_{11})$. They can be used without modifications, but the way of applying these depends on the CPS in question. The remaining group of principles can be applied only after a purposeful adaptation. They are: $P_a = (P_1, P_6, P_7, P_9)$. Our observation has been that certain system features of CPSs will require additional (not yet specified) maintenance principles, because they cannot be addressed by the principles known to be applicable to LCSs. Since we primarily focused on the reusability of maintenance principles of LCSs in the context of CPSs, these additional novel principles have not yet been explored in our study. In the next section, we will discuss the essence of these additional principles.

As mentioned above, the group (P_a) comprises those principles that should be and can be adapted. Their adaptation needs further considerations of the system features. In the following paragraphs, we consider the adaptations that should be made. For instance, the transfer of principle P_1 to CPSs needs knowledge about the lifecycle of components, their failure modes and effects, as well as about the specific forms and opportunities for automation of maintenance activities, such as revision, repairs, and spare-part changes. The very reason this principle needs adaptation is that the abovementioned activities greatly differ from those associated with LCSs. Some enabling methods, such as failure mode and effect analysis (FMEA) [55], fault tree analysis (FTA) [56], hazard and operability study (HAZOP) [57], and component fault tree [58] can be applied to conduct specific failure analyses.

However, the use of these methods requires a large amount of data and information about operation of complex systems, which may be difficult to obtain [59]. Furthermore, since the use of this principle in LCSs requires a high-level of human involvement, some sort of automation of the scheduled revision activities seems to be necessary. The adaptation should consider the self-monitoring and self-repair potentials of CPSs. It is important to note that P_1 was originally developed for physical components that provide observable indication (signals) of wear. Further studies are needed to investigate how this principle can be applied to electronic components, which are normally subjected to random failures only [60].

Principle P_6 also requires adaptation in order to provide optimal results for CPSs. The adaptation should consider new different types of tests, which take into account the effects of unexpected external and internal events. Currently, there are limitations in terms of what can be tested through functional or performance simulations and runtime tests. They should be able to deal with unique faults and failures of CPSs, which do not occur in LCSs. It seems to be necessary to design new protocols for behavioural and performance tests in order to determine how they should be conducted, which values are expected as the key performance indicators, and how to aggregate these in distributed and decentralized systems.

Principle P_7 implies the consideration and inclusion of fault avoidance and system maintenance at the design stage of the product development process. While various methodologies have been elaborated for LCSs, they do not seem to be directly applicable in the design processes of CPSs. It is necessary to include new design criteria based on the system features of CPSs, as well as quality standards and test procedures in the design processes of CPSs, which involves hardware, software and information platforms design. Further research is needed to develop comprehensive verification and validation methodologies for CPSs and subsystems that can be applied in the early phase of the design process. It is also imperative to investigate how the designed systems will respond to faults and what the impacts and consequences of the potential faults may be.

Principle P_9 places emphasis on the run-time prediction of possible failures and black outs of CPSs. This principle assumes predictive and/or probabilistic system models that are actualized in run-time and can prognosticate system operation based on the evaluation of subsequent system states. The predictive models currently applied for LCSs are not transferable to CPSs due to the dynamic nature and operational conditions of these systems. Relevant predictive models should be able to capture the internal dynamics of the systems, the dynamic interaction of the systems together with their environment, and the dynamics of the embedding environments. To effectively apply this principle, forecasting mechanisms are needed that are capable of forecasting future faults and failures of CPSs based on operation or application history information. This may be enabled by information provided by networked sensors, tracking the frequency, and amount of failures reported in the system, and even learnt from the conditions under which these faults occurred.

Group (P_e) comprises those principles that can be used in the maintenance of CPSs without adaption. However, it has to be mentioned that while these principles do not need reinterpretation or redefinition, the way of operationalizing them in the case of CPSs may be different from the way they are applied in the case of LCSs.

Principle P_2 can be directly (without adaptation) applied to CPSs due to the availability of the enabling technologies, such as sensing, monitoring, information processing, fault diagnosis, and failure prognosis algorithms [61]. The application process is essentially the same as in the case of LCSs, which typically involves the use of FMEA, FTA, HAZOP, Markov chain [62] and Petri-net [63] methods, and Bayesian models for failure analysis. Methods such as FTA and Petri-nets can also be used for failure propagation analysis [64]. Which signals are to be considered as indicators of faults, how they can be sensed in real time, and with which frequency they have to be sensed and evaluated has to be carefully determined. Similarly, the monitoring frequency for each component needs careful consideration and harmonization. It is, however, acknowledged in the literature that the introduction of a high-level of automation usually results in more complex and costly maintenance actions [65].

The process of applying principle P_5 to CPSs is practically the same as to LCSs. It involves analysing the criticality of component failures, as well as the urgency of response and repair actions. The objective of this analysis is to determine the components that are prone to failures, with the highest probability and causes the highest risk levels; and to make decisions on the strategy of redesigning and on better solutions. Decisions can also be made on which failures can be managed by the CPSs themselves, and which need immediate availability of maintenance plans and involvement of personnel. These depend on the forecasted occurrence frequency of component failures. Having considered these influencing factors, whether applying structural redundancy, more resilient components, functional re-configuration, or more robust system architecture can be a better solution can also be determined, taking into consideration the associated costs and extra efforts [2].

The application of principle P_8 can be considered as a design challenge. It concerns not only the design decisions and solutions in the design process of CPSs, but the preventive and corrective actions that can be taken during the operation of a system. In other words, the application of this principle requires concurrent elaboration of both a preventive maintenance strategy and a corrective maintenance plan.

In the context of CPSs, the objective of principle P_{10} is to detect and isolate faults through a collaborative strategy that involves the actions of both the maintenance experts and the self-adaptive system.

The information platform required by the latter can be generated by continuous monitoring of the system, reflective real-time modification of detection algorithms, introducing changes in the system arrangement, and planning the response actions. Once faults are known, they can be prioritized based on the probability of occurrence, as well as on their criticality with regards to system operation.

Finally, the reason principle P_{11} can be applied to CPSs without adaptation is that the system functionality and the technologies used can alert the operator if there is a fault. This involves diagnosis-based report generation, ubiquitous communication of failure information, decision making on the suspension or continuation of system operation, proposals for maintenance or repair, identification of replaceable parts, determination of resources and tool demands, and capacity and activity planning. The system should also determine what information should be delivered to which stakeholders. The three most important pieces of information that should be delivered to the operator are the description of the failure, its place in the system, its neighbourhood, and its criticality.

6 SOME SUGGESTIONS ON SPECIFIC MAINTENANCE PRINCIPLES FOR CPSs

Which new principles are needed for a particular family of CPS? It is obvious that due to the complex functionality, structure, and operation of CPSs, they need additional maintenance principles that are not necessary for LCSs. The sought after dedicated principles are especially important for high-end CPSs, which are open, dynamic, decentralized, intelligent and self- organizing systems. Their intense interaction with the natural and engineered environments and penetration into the social and cognitive domains of stakeholders require further investigations, because of the increasing exposure to the environment and humans. The primary CPS features that makes them require novel maintenance principles are: (i) non-linearity (interaction, circumstances, and behaviour), (ii) applications in dynamic and harsh environments, and (iii) growing level of automation.

The non-linearity of CPSs has many sources and forms of manifestation. Open decentralized systems may have the capability to dynamically change their system boundaries. They may also adapt their operation to the actual operation circumstances. In general, the change in the components and the change of behaviour of the components complicates both the forecasting and the correction of the failures. These systems are not predictable as they frequently and intensively move between many discrete states and transitions [66]. To cope with these characteristics, maintenance principles dedicated to dynamic complex systems are needed. It is imperative that they must address fault management and elimination in hardware, software and information systems in an integrated way. Evidently, eliminating the sharp boundary between analogue and discrete physical components and the software and information system components is a fundamental challenge. Currently, biological analogies, such as the human immune system, are dealt with in some research to understand which features, behaviours and architecture result in perpetual corrective behaviour that emerges from local detection and interventions. In terms of interoperating software, some researchers have dealt with the notion of the fractionated CPS that goes beyond the conventional definition of a software-controlled hardware system that is interacting with the physical world [31].

The operation of CPSs in unpredictable and harsh environmental elements, such as chemical reagents and humidity, also imply the need for new maintenance principles. These and similar operating conditions invalidate the traditional forecasting models, as these conditions will most likely affect the hardware component’s lifecycle, and increase the chances of malfunctioning. Researchers are engaged in finding theories and technological solutions for inherently fault-tolerant dynamic architectures, as well as non-model-based zero-delay monitoring and proactive detection solutions. Another domain of research interest can be vague forecasting based on incomplete and localized bodies of knowledge.

Both CPSs and their components are reaching a high level of autonomy. This is enabled by their increasing smartness or intelligence, which is a result of wide-ranging information elicitation, reasoning and inference, and the “agentialization” of system operation. System intelligence also supports moving decision making and preparation of maintenance from the design phase to the runtime phase of the system’s lifecycle. The automation of maintenance not only has positive technical outcomes, but also reduces the required human efforts, intervention, costs and safety, and improves servicing capabilities [67]. It seems to be necessary to include maintenance-related aspects in model-based design of CPSs and to be able to detect near failure states in operation.

Finally, there is a need to develop self-maintenance principles for various families of CPSs. As discussed earlier in this paper, some of the current maintenance principles can be considered in the case of systems with self-detection (self-diagnosis) and failure prevention capabilities. State sensors built into physical components, smart materials, and emergent behaviour analysers are already used in current CPSs. These principles should thoroughly cover the maintenance process in such a way that human involvement is reduced considerably. It follows that currently used self-diagnosis and failure detection methods should be combined with new techniques for monitoring, changing and repairing parts during system operation. Lee et al. argue that self-maintenance techniques should enable awareness of the changing operational regimes to dynamically select prognostic models in order to ensure accurate prediction [68]. We can say that there are advantages of combining the implementation of this concept with the implementation of response actions through automated actuators.

7 DEMONSTRATION OF THE APPLICABILITY OF MAINTENANCE PRINCIPLES TO CYBER-PHYSICAL GREENHOUSES

We use a case of a cyber-physical greenhouse (CPGH) to demonstrate the applicability of the maintenance principles presented and discussed in the previous Sections. A CPGH is considered to be a cyber-physical augmentation of the traditional greenhouse in order to make it capable of providing new services. Actual examples are used to explain how a maintenance principle can or cannot be operationalized in a CPGH, and what types of adaptations may be necessary.

7.1 Non-Applicable Principles

- *Conduct maintenance actions once a failure has occurred*

Because CPGHs are naturally mission critical systems, it is necessary to prevent any system failure, rather than to eliminate the effects of failure, and recover from occurred failures and malfunctioning. What follows from this requirement is that the principles of the strategy of corrective maintenance simply cannot be applied in this case. Instead of these, the operationalization of an extended set of preventive maintenance principles is needed. This need is evident from the following practical challenge: in a CPGH, parameters such as temperature, humidity and CO_2 are typically controlled through ventilation. The plant-monitoring sub-system should be able to measure the transpiration and temperature of the plant. If, for example, due to the lack of maintenance, this sub-system fails, the actuators (such as heaters or fans) will not be able to react, or will erroneously respond, and this will cause a serious damage of the plant.

- *Conduct general maintenance once any of the system components fail*

This maintenance principle has no relevance in the context of mission critical CPGH systems. As in the above explained case, if maintenance activities are done only when a component fails, both the risk of plant damage and the hazard of the lack of availability of the entire system prevail. Consequently, only maintenance principles that stimulate preventive maintenance activities should be operationalized. As a practical situation, one can argue that some crops, such as roses, are highly sensitive to changes in temperature. Suppose that no failure has occurred in the entire system until a given point in time. If the above maintenance principle is applied, then even the critical system components are not maintained. Should there be lack of maintenance, for instance, not only may the temperature control sub-system break down, but also other critical components of the greenhouse, such as the boiler, and this may lead to a complete failure of the CPGH system. Likewise, the malfunctioning of the boiler may cause damage to roses during cold seasons, and this may not only seriously affect their quality, but may also cause losses to the grower.

7.2 Exportable Principles

- *Support maintenance actions by monitoring activities*

This principle suggests a continuous monitoring of a system in order to be able to explore the need for maintenance, and to reduce the chance of failure over time. The principle is not only operationalizable, but also very useful in the context of CPGH systems, which should typically feature multiple wireless sensor networks. We can illustrate possible practical utilization in this regard. Let us consider, for example that definition of the so-called “set points” is currently done in CPGHs based on monitoring temperature, humidity and CO_2 levels. Any unexpected variation in these sensed parameters with respect to the “set points” may lead to improper operation. Furthermore, the observed variations of the parameters can be used as alerting signals of failure. It can also be the case that variations in the physiological parameters of plants may also cause failures in the sensors and/or actuators.

- *Redesign to avoid the causes of failure*

To discuss the reusability of this maintenance principle, let us use a practical example of the sensors used for measuring plant transpiration and temperature. It can occur in both traditional and cyber-physical greenhouses that sensors are recurrently suffering from failures due to the effects of humidity and chemical corrosion within the greenhouse. Considering the recurring nature of this important sub-system, this part of the whole system calls for redesign and replacement; otherwise, the grower will face substantial operational costs. To facilitate redesign, the most influential factors and the weak points have to be identified. This requires a comprehensive and systematic analysis because, in the case of CPGHs, failure can be caused by disruption of hardware, malfunction of software, the loss of cyber content, or all of these together.

- *Assure continuity of system operation despite the presence of faults*

One widely-used approach to assure the continuity of operation is the building of various types of redundancies into a system. Sub-system or component multiplication is seen as an effective approach to increasing the dependability of mission critical CPGHs. For example, the use of more than one fertilizer injector machine in an irrigation process can guarantee the availability of fertilizers even if one of them breaks down or malfunctions. If this principle is considered in the design process of CPGHs, the foreseeable operational deficiencies can be eliminated, or the number of their occurrence can be reduced. This principle also entails taking measures to make sure that a failure in the sub-system will not affect other sub-systems. For instance, failure of the reasoning engine of the irrigation sub-system will not influence the performance of the rest of the CPGH system if each of its intelligent sub-systems has a reasoning engine on its own.

- *Detect and isolate faults*

This principle can be applied straightaway in the case of CPGH systems. Owing to their component-based implementation, it can simply be the identification of which components have broken down, or are not working properly. Component-based implementation of CPGHs also facilitates the isolation of erroneous components and helps sustain the operation of the rest of the systems. For instance, the behaviour of a sensor that measures the temperature of plants can be tested by making control measures, by comparing the temperature needed locally in the greenhouse to the measurements taken on close to the plants. If the differences are above the margin of error, it can be concluded that a sensor fault is in development. These measurements should be incorporated in the troubleshooting algorithms of CPGHs and in any other post-processing procedures.

- *Alerting the operator in case of fault*

CPSs allow both direct (co-located) and indirect (dislocated or remote) interaction with users (both sub-systems and humans) through dedicated interfaces. Communication with external agent sub-systems can be used for alerting and requiring intervention beyond the level of reliability that is it usually achievable with human users and supervisors in the case of LCSs. For instance, automatically sending a message to a supervisory agent sub-system as well as to the greenhouse operator using mobile devices such as tablets and smartphones can shorten the reaction time, and may lead to more knowledge-intensive decision making. This duality in alerting is particularly necessary if any failure occurs that cannot be managed by the system.

7.3 Principles that Require Adaptation

- *Schedule maintenance actions*

CPGHs are dynamic systems subjected to unpredictable situations. A situation may have various influence on the life cycle of the involved individual physical components. This differs from the way of operation and from the operational situations that are typical in the maintenance of LCSs. The dynamic (task- and environment-influenced) operation of CPGHs makes the planning and execution of systematic maintenance somewhat difficult. Meanwhile, the increased opportunity of sensor- and smart reasoning-based automated monitoring makes it possible to combine the principle of scheduled maintenance with comprehensive, continuous monitoring. Efficient use of sensors in measuring the most informative parameters of plants can lead to a context-sensitive surveillance and the control of the CPGH system. Let us take the example of using artificial light. If lighting components are used less often, this can be taken into consideration in their scheduled replacement and, in addition, the planned visual/instrumented checking activities can also be done less frequently.

- *Avoid failures in the system by preventing the occurrence of faults*

In the case of LCSs, the operating conditions are usually known in the design stage. The system operation and behaviour can be pre-tested based on virtual or testable physical prototypes. However, this is hardly possible in the case of CPGHs working in unforeseeable dynamic circumstances. If the deficiencies in behaviour cannot be explored and eliminated through prototype testing, then the objective is to prevent the occurrence of faults. To this end, the development of new testing approaches that are able to evaluate the capability of the sub-systems in unexpected situations, considering the negotiation processes between the subsystems as well, is necessary. For instance, if the automated cooling/heating sub-system decides that natural airing is to be done (i.e. windows should be opened) due to a sudden change in the climate situation in the greenhouse (plant's temperature), this decision may also affect the CO_2 regulation but the direction of opening of the vents may not be appropriate. In this particular situation, a negotiation process between the reasoning engines of both systems is required. This means that the negotiation capability of the concerned sub-systems should be tested during the design and prototyping stage.

- *Reduce the amounts of faults and their severity*

CPGHs are more complex systems than traditional greenhouses; consequently, many more different possibilities are there for both component failures and system break downs. The number of faults and reducing their severity requires redefinition and reinterpretation of the above maintenance principle. While the overall goal should be kept, the way of achieving it should be adapted to the complexity of CPGH systems and the multitude of functional interactions among the components. Consider the fact that the natural horticultural system constituents (such as plants and climate, CO_2, humidity, and lighting sub-systems) and the constituents of cyber-physical augmentation (including sensing technologies, reasoning engines, data transmitters, and smart actuators) should be seamlessly blended and operating. Formulation of all the relevant and most appropriate maintenance principles requires further research, in particular if reducing the severity and impacts of the failures is also a major objective.

- *Predicting failures on the system*

Typically, the system models currently used in the analysis and operation simulation of LCSs do not take into account the occurrence of the unexpected situations to which CPSs are often subjected. Therefore, the results of these traditional prediction models and software tools may not be entirely reliable in the context of CPSs. In the case of model-based maintenance, sufficiently comprehensive (modelling the dynamics of the CPGH systems as well as the dynamics of the embedding environment) and articulated (covering both the natural horticultural system constituents and the cyber-physical augmentation constituents) prediction models are needed. Such models would capture information about the growth of the plants and their effect on the system performance. In fact, the ultimate objective of developing such kind of prediction tools is to reduce the amount of unexpected situations through a deep analysis of the effects of variations of the internal and external parameters of CPGH systems. Research in this direction is still at its infancy; therefore, these desirable new maintenance principles are not yet known.

8 CONCLUSIONS

We have reviewed the maintenance principles currently applied in LCSs with the intention of determining if they are relevant to the maintenance of CPSs. Due to the proliferation of CPSs and their applications, including in mission critical areas, there has been a growing need to analyse how the maintenance of these systems should be conducted and to identify maintenance principles that can be successfully applied to them. CPSs are complicated complex systems, which nevertheless have some similarities with LCSs. For instance, both integrate information technologies into physical devices, are geographically distributed, have multiple energy sources, functional units, and intense interactions with human stakeholders and the embedded environment. High-end CPSs are, however, non-linear systems, which feature a multitude of functional connections among the components, exhibit a high level of automation and intelligence, and are developed to operate in dynamic or harsh environments. There is also a great dissimilarity between their system features. These facts inspired us to analyse which generic maintenance principles of LCSs could be transferred to CPSs.

In the work presented in this paper, we identified the four groups of principles presented and discussed in the previous sections. We have argued and explained why certain principles can be applied directly, and why certain principles need adaptation. In our analysis, we established that some features of CPSs cannot be addressed by exportable maintenance principles. Novel maintenance principles should, therefore, be developed for these features. The reported work, however, is just the first step, and further research is expected to provide a deeper understanding through more accurate and focused analyses aimed at identifying the appropriate maintenance principles for CPSs. Future study will also include the identification of influential factors and causalities.

We should probably not expect to develop overly generic maintenance principles that are equally and broadly applicable to all CPSs, including high-end CPSs that are, for example, capable of reorganizing themselves. The maintainability of high-end CPSs depends on multiple external factors that dynamically influence their operation. The aforementioned examples illustrate how the known maintenance principles of LCSs can be considered in the case of CPGH systems. Also indicated is the need for extensive further research as well as for real-world environment-based studies to reveal what new maintenance principles are needed, and how they can be operationalized in future cyber-physical greenhouse systems. Therefore, in addition to defining new maintenance principles, our future research will also concentrate on the development of a complex troubleshooting model and a maintenance advisory system for CPGHs.

9 REFERENCES

[1] Chun, I., Kim, J., Kim, W.T., Lee, E. (2011). Self-managed system development method for cyber-physical systems. *Control and Automation, and Energy System Engineering*, vol. 256, p. 191-194, DOI:10.1007/978-3-642-26010-0_23.

[2] Colnaric, M., Verber, D., Halang, W. A. (2008). *Real-Time Characteristics And Safety Of Embedded Systems. Distributed Embedded Control Systems*, Springer, London, p. 3-28.

[3] Sierla, S., O'Halloran, B.M., Karhela, T., Papakonstantinou, N., Tumer, I.Y. (2013). Common cause failure analysis of cyber–physical systems situated in constructed environments. *Research in Engineering Design*, vol. 24, no. 4, p 375-394, DOI:10.1007/s00163-013-0156-2.

[4] Frazzon, E.M., Hartmann, J., Makuschewitz, T., Scholz-Reiter, B. (2013). Towards socio-cyber-physical systems in production networks. *Procedia CIRP*, vol. 7, p. 49-54, DOI:10.1016/j.procir.2013.05.009.

[5] Parvin, S., Hussain, F.K., Hussain, O.K., Thein, T., Park, J.S. (2012). Multi-cyber framework for availability enhancement of cyber physical systems. *Computing*, vol. 95, no. 10-11, p 927-948, DOI:10.1007/s00607-012-0227-7.

[6] Sha, L., Gopalakrishnan, S., Liu, X., Wang, Q. (2009). *Cyber-Physical Systems: A New Frontier.* Tsai, J.J.P., Yu, P.S. (eds.), *Machine Learning in Cyber Trust,* Springer, p. 3-13, DOI:10.1007/978-0-387-88735-7_1.

[7] Yagan, O., Qian, D., Zhang, J., Cochran, D. (2012). Optimal Allocation of Interconnecting Links in Cyber-Physical Systems: Interdependence, Cascading Failures, and Robustness. *IEEE Transactions On Parallel And Distributed Systems*, vol. 23, no. 9, p. 1708-1721, DOI:10.1109/TPDS.2012.62.

[8] Miclea, L., Sanislav, T. (2011). About dependability in cyber-physical systems. *9th East-West Design & Test Symposium*, Sevastopol.

[9] Érdi, P. (2008). Complex Systems: The Intellectual Landscape. *Complexity Explained*. Springer, Berlin, Heidelber, p. 1-23.

[10] Qian, K., den Haring, D., Cao, L. (2009). Introduction to Embedded Systems. *Embedded Software Development*, Springer, p. 1-37, DOI:10.1007/978-1-4419-0606-9_1.

[11] Marwedel, P. (2011). Introduction. *Embedded System Design*. Springer Netherlands, Amsterdam, p. 1-19, DOI:10.1007/978-94-007-0257-8_1.

[12] Reddy, P.M. (2002). Embedded systems. *Resonance*, vol. 7, no. 12, p. 20-30, DOI:10.1007/BF02834526.

[13] Barolli, L., Takizawa, M., Hussain, F. (2011). Special issue on emerging trends in cyber-physical systems. *Journal of Ambient Intelligence and Humanized Computing*, vol. 2, no. 4, p. 249-250, DOI:10.1007/s12652-011-0062-2.

[14] Horváth, I., Gerritsen, B. (2012). Cyber-Physical Systems: Concepts, technologies and implementation principles. *9th Tools and Methods of Competitive Engineering*, Karlsruhe.

[15] Kumara, S., Cui, L., Zhang, J. (2011). Sensors, networks and internet of things: research challenges in health care. *8th International Workshop on Information Integration on the Web*, New York.

[16] Karsai, G., Sztipanovits, J. (2008). Model-Integrated Development of Cyber-Physical Systems. Brinkschulte, U., Givargis, T., Russo, S. (eds.), *Software Technologies for Embedded and Ubiquitous Systems,* Springer, Berlin, Heidelberg, p. 46-54, DOI:10.1007/978-3-540-87785-1_5.

[17] Wu, F.J., Kao, Y.F., Tseng, Y.C. (2011). From wireless sensor networks towards cyber physical systems. *Pervasive and Mobile Computing*, vol. 7, no. 4, p. 397-413, DOI:10.1016/j.pmcj.2011.03.003.

[18] Song, Z., Chen, Y., Sastry, C. R., Tas, N.C. (2009). *Optimal Observation for Cyber-physical Systems.* Springer, London, p. 1-25, DOI:10.1007/978-1-84882-656-4_1.

[19] Beck, A.C.S., Lisbôa, C.A.L., Carro, L., Nazar, G.L., Pereira, M.M., Ferreira, R.R. (2013). Adaptability: the key for future embedded systems. in *Adaptable Embedded Systems*, Beck, A.C.S., Lisbôa, C.A.L., Carro, L. (eds.). Springer, New York, p. 1-12, DOI:10.1007/978-1-4614-1746-0_1.

[20] Santos, R.M., Santos, J., Orozco, J.D. (2009). Power saving and fault-tolerance in real-time critical embedded

systems. *Journal of Systems Architecture*, vol. 55, no. 2, p. 90-101, DOI:10.1016/j.sysarc.2008.09.001.

[21] Baheti, R., Gill, H. (2011). Cyber-physical systems, Samad, T., Annaswamy. A. (eds.), *The Impact of Control Technology*, IEEE Control System Society, Munich, p. 161-166.

[22] Hu, L., Xie, N., Kuang, Z., Zhao, K. (2012). Review of Cyber-Physical System Architecture. *IEEE 15th International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing Workshops*, p. 25-30, DOI:10.1109/ISORCW.2012.15.

[23] Gerritsen, B., Horváth, I. (2012). Current drivers and obstacles of synergy in Cyber-Physical Systems Design. *ASME International Design Engineering Technical Conference & Computer and Information in Engineering Conference,* Chicago.

[24] Guinand, F., Pigné, Y. (2006). Problem Solving and Complex Systems. Aziz-Alaoui, P.M.A., Bertelle, P.C. (eds.), *Emergent Properties in Natural and Artificial Dynamical Systems*, *Understanding Complex Systems.* Springer, Berlin, Heidelberg, p. 53-85, DOI:10.1007/3-540-34824-7_3.

[25] Wang, Y., Tan, G., Wang, Y., Yin, Y. (2012). Perceptual control architecture for cyber–physical systems in traffic incident management. *Journal of Systems Architecture*, vol. 58, no. 10, p. 398-411, DOI:10.1016/j.sysarc.2012.06.004.

[26] Lun, Y., Cheng, L. (2011). The research on the model of the context-aware for reliable sensing and explanation in cyber-physical system, *Procedia Engineering*, vol. 15, p. 1753-1757. DOI:10.1016/j.proeng.2011.08.327.

[27] Talcott, C. (2008). Cyber-physical systems and events. Wirsing, M., Banatre, J.-P., Hölzl, M., Rauschmayer, A. (eds.), *Software-Intensive Systems and New Computing Paradigms*, vol. 5380, Springer, Berlin, Heidelberg, p. 101-115, DOI:10.1007/978-3-540-89437-7_6.

[28] Rajkumar, R. Lee, I., Sha, L., Stankovic, J. (2010). Cyber-physical systems: the next computing revolution. *47th Design Automation Conference*, New York, DOI:10.1145/1837274.1837461.

[29] Li, P., Xu, Q., Wang, N. (2013). Evolution course and analysis of internet of things. *International Conference on Information Engineering and Applications*, Springer, London, p. 221-228.

[30] Perraju, T.S., Uma, G. (2005). Mission Critical Intelligent Systems. Leondes, C.T. (ed.), *Intelligent Knowledge-Based Systems*. Springer, p. 1184-1210, DOI:10.1007/978-1-4020-7829-3_34.

[31] Stehr, M.O., Talcott, C., Rushby, J., Lincoln, P., Kim, M., Cheung, S., Poggio, A. (2011). Fractionated Software for Networked Cyber-Physical Systems: Research Directions and Long-Term Vision. Agha, G., Danvy, O., Meseguer, J. (eds.), *Formal Modeling: Actors, Open Systems, Biological Systems,* Springer, Berlin, Heidelberg, p. 110-143.

[32] Wedde, H.F., Lind, J. A. (1997). Building Large, Complex, Distributed Safety-Critical Operating Systems. *Real-Time Systems*, vol. 13, no. 3, p. 277-302, DOI:10.1023/A:1007915628098.

[33] Ponsard, C., Massonet, P., Rifaut, A., Molderez, J.F., van Lamsweerde, A., Van Tran, H. (2005). Early verification and validation of mission critical systems. *Electronic Notes in Theoretical Computer Science*, vol. 133, p. 237-254, DOI:10.1007/s10703-006-0028-8.

[34] Waeyenbergh, G., Pintelon, L. (2004). Maintenance concept development: A case study. *International Journal of Production Economics*, vol. 89, no. 3, p. 395-405, DOI:10.1016/j.ijpe.2003.09.008.

[35] Moesgaard, M., Froholdt, M., Poulfelt, F. (2010). *Return on Strategy: How to Achieve it!*, Routledge, New York, London.

[36] Stevenson, A. (2010). *Oxford Dictionary of English*, 3rd ed., Oxford University Press, Oxford.

[37] Xie, F., Yang, G., Song, X. (2006). Component-based hardware/software co-verification. *4th ACM and IEEE International Conference on Formal Methods and Models for Co-Design, Proceedings*, p. 27-36.

[38] Kelly, A. (2006). *Strategic Maintenance Planning.* Butterworth-Heinemann, Oxford.

[39] Starr, A., Al-Najjar, B., Holmberg, K., Jantunen, E., Bellew, J., Albarbar, A. (2010). Maintenance Today and Future Trends. Holmberg, K., Adgar, A., Arnaiz, A., Jantunen, E., Mascolo, J., Mekid, S. (eds.), *E-maintenance*. Springer, London, p. 5-37, DOI:10.1007/978-1-84996-205-6_2.

[40] Burhanuddin, M.A., Halawani, S.M., Ahmad, A.R. (2011). An efficient failure-based maintenance decision support system for smalland medium industries. Jao, C. (ed.), *Efficient Decision Support Systems - Practice and Challenges From Current to Future*. InTech, Rijeka, p. 195-211.

[41] Pintelon L.M., Gelders, L.F. (1992). Maintenance management decision making. *European Journal of operational Research*, vol. 58, no. 3, p. 301-317, DOI:10.1016/0377-2217(92)90062-E.

[42] Zhou, X., Xi, L., Lee, J. (2009). Opportunistic preventive maintenance scheduling for a multi-unit series system based on dynamic programming. *International Journal of Production Economics*, vol. 118, no. 2, p. 361-366, DOI:10.1016/j.ijpe.2008.09.012.

[43] Pintelon L., Muchiri, P.N. (2009). Safety and maintenance. Ben-Daya, M., Duffuaa, S.O., Raouf, A., Knezevic, J., Ait-Kadi, D. (eds.), *Handbook of Maintenance Management and Engineering*, Springer, London, p. 613-648, DOI:10.1007/978-1-84882-472-0_22.

[44] Labib A.W., Yuniarto, M.N. (2009). Maintenance strategies for changeable manufacturing. ElMaraghy, H.A. (ed.), *Changeable and Reconfigurable Manufacturing Systems*. Springer, London, p. 337-351, DOI:10.1007/978-1-84882-067-8_19.

[45] Johnson, D.M. (1996). A review of fault management techniques used in safety-critical avionic systems. *Progress in Aerospace Sciences*, vol. 32, no. 5, p. 415-431. DOI:10.1016/0376-0421(96)82785-0.

[46] Xu, S. (1999). On dependability of computing systems. *Journal of Computer Science and Technology*, vol. 14, no. 2, p. 116-128, DOI:10.1007/BF02946517.

[47] Dubrova, E. (2013). Fundamentals of Dependability. *Fault-Tolerant Design*, Springer, New York, p. 5-20, DOI:10.1007/978-1-4614-2113-9_2.

[48] Avižienis, A., Laprie, J.C., Randell, B. (2004). Dependability and its threats: a taxonomy. Jacquart, R. (ed.), *Building the Information Society*, Springer, New York, p. 91-120, DOI:10.1007/978-1-4020-8157-6_13.

[49] Jhumka, A. (2010). Dependability in service-oriented computing. Griffiths, N., Chao, K.M. (eds.), *Agent-Based Service-Oriented Computing*. Springer, London, p. 141-160, DOI:10.1007/978-1-84996-041-0_6.

[50] Kontoleon, J. (2008). Status and trends in the performance assessment of fault tolerant systems. Misra, P.K.B. (ed.), *Handbook of Performability Engineering*, Springer, London, p. 1087-1106, DOI:10.1007/978-1-84800-131-2_66.

[51] Birman K.P., Glade, B.B. (1995). Reliability through consistency. *IEEE Software*, vol. 12, no. 3, p. 29-41. DOI:10.1109/52.382182.

[52] Verma, A. Srividya, A. Ramesh, P. (2010). A systemic approach to integrated e-maintenance of large engineering plants. *International Journal of Automation and Computing*, vol. 7, no. 2, p. 173-179, DOI:10.1007/s11633-010-0173-9.

[53] Sobh, T.S., Mostafa, W.M. (2011). A cooperative immunological approach for detecting network anomaly. *Applied Soft Computing,* vol. 11, no. 1, p. 1275-1283, DOI: 10.1016/j.asoc.2010.03.004.

[54] La, H.J., Kim, S.D. (2010). A Service-Based Approach to Designing Cyber Physical Systems. *IEEE/ACIS 9th International Conference on Computer and Information Science*, p. 895-900.

[55] Ben-Daya, M. (2009). Failure mode and effect analysis. Ben-Daya, M.S., Duffuaa, O., Raouf, A., Knezevic, J., Ait-Kadi, D. (eds.), *Handbook of Maintenance Management and Engineering*, Springer, London, p. 75-90, DOI:10.1007/978-1-84882-472-0_4.

[56] Čepin, M. (2011). Fault Tree Analysis. *Assessment of Power System Reliability*. Springer, London, p. 61-87, DOI:10.1007/978-0-85729-688-7_5.

[57] Stapelberg, R.F. (2009). Safety and risk in engineering design. *Handbook of Reliability, Availability, Maintainability and Safety in Engineering Design*. Springer, London, p. 529-798.

[58] Xu, T., Liu, Z., Tang, T., Zheng, W., Zhao, L. (2012). Component based design of fault tolerant devices in cyber physical system. *15th IEEE International Symposium on Object/Component/Service-Oriented Real-Time Distributed Computing Workshops*- p. 37-42, DOI:10.1109/ISORCW.2012.17.

[59] Garg, H., Sharma, S.P. (2012). Stochastic behavior analysis of complex repairable industrial systems utilizing uncertain data. *ISA Transactions*, vol. 51, no. 6, p. 752-762, DOI:10.1016/j.isatra.2012.06.012.

[60] Bencsik, A.L., Gati, J., Kartyas, G. (2012). Maintenance of complex automated systems. *7th IEEE International Symposium on Applied Computational Intelligence and Informatics*, p. 223-227.

[61] Chen, C., Brown, D., Sconyers, C., Zhang, B., Vachtsevanos, G., Orchard, M.E. (2012). An integrated architecture for fault diagnosis and failure prognosis of complex engineering systems. *Expert Systems with Applications*, vol. 39, no. 10, p. 9031-9040, DOI:10.1016/j.eswa.2012.02.050.

[62] Ching ,W.K., Ng, M.K. (2006). Markov Chains: Models, Algorithms and Applications. Springer, New Yourk.

[63] Narahari, Y. (1996). Petri nets. *Resonance*, vol. 4, no. 8, p. 58-69, DOI:10.1007/BF02837068.

[64] Cao, R., Chen, Y., Kang, R. (2012). Critical review of system failure behavior analysis method. *IEEE Conference on Prognostics and System Health Management*, p. 1-10.

[65] Pinjala, S.K., Pintelon, L., Vereecke, A. (2006). An empirical investigation on the relationship between business and maintenance strategies. *International Journal of Production Economics*, vol. 104, no. 1, p. 214-229, DOI:10.1016/j.ijpe.2004.12.024.

[66] McDermid, J., Thomas, M., Redmill, F. (2009). Professional issues in system safety engineering. Dale, C., Anderson, T. (eds.) *Safety-Critical Systems: Problems, Process and Practice*. Springer, London, p. 135-145.

[67] Hanemann, A., Sailer, M., Schmitz, D. (2004). Assured service quality by improved fault management. *2nd International Conference on Service Oriented Computing*, New York, p. 183-192.

[68] Lee, J., Ghaffari, M., Elmeligy, S. (2011). Self-maintenance and engineering immune systems: Towards smarter machines and manufacturing systems. *Annual Reviews in Control*, vol. 35, no. 1, p. 111-122. DOI:10.1016/j.arcontrol.2011.03.007.