Connectors of smart design and smart systems

Horváth, Imre

DOI
10.1017/S0890060421000068

Publication date
2021

Document Version
Final published version

Published in
Artificial Intelligence for Engineering Design, Analysis and Manufacturing: AIEDAM

Citation (APA)
Horváth, I. (2021). Connectors of smart design and smart systems. Artificial Intelligence for Engineering Design, Analysis and Manufacturing: AIEDAM, 35(2), 132-150. https://doi.org/10.1017/S0890060421000068

Important note
To cite this publication, please use the final published version (if applicable). Please check the document version above.

Copyright
Other than for strictly personal use, it is not permitted to download, forward or distribute the text or part of it, without the consent of the author(s) and/or copyright holder(s), unless the work is under an open content license such as Creative Commons.

Takedown policy
Please contact us and provide details if you believe this document breaches copyrights. We will remove access to the work immediately and investigate your claim.
Connectors of smart design and smart systems

Imre Horváth
Faculty of Industrial Design Engineering, Delft University of Technology, Landbergstraat 15, 2628CE Delft, The Netherlands

Abstract

Though they can be traced back to different roots, both smart design and smart systems have to do with the recent developments of artificial intelligence. There are two major questions related to them: (i) What way are smart design and smart systems enabled by artificial narrow, general, or super intelligence? and (ii) How can smart design be used in the realization of smart systems? and How can smart systems contribute to smart design? A difficulty is that there are no exact definitions for these novel concepts in the literature. The endeavor to analyze the current situation and to answer the above questions stimulated an exploratory research whose first findings are summarized in this paper. Its first part elaborates on a plausible interpretation of the concept of smartness and provides an overview of the characteristics of smart design as a creative problem solving methodology supported by artificial intelligence. The second part exposes the paradigmatic features and system engineering issues of smart systems, which are equipped with application-specific synthetic system knowledge and reasoning mechanisms. The third part presents and elaborates on a conceptual model of AI-based couplings of smart design and smart systems. The couplings may manifest in various concrete forms in real life that are referred to as “connectors” in this paper. The principal types of connectors are exemplified and discussed. It has been found that smart design trends to manifest as a methodology of blue-printing smart systems and that smart systems will be intellectualized on the enablers of implementation of smart design. Understanding the affordances of and creating proper connectors between smart design and smart systems need further explorative research.

Introduction

The word “smart” is included twice in the title. Interestingly, the meanings of this word in the two contexts are somewhat different. This is a typical example of the ambiguities caused by the use of this word as a general adjective in largely different contexts. This also casts light on definitional issues that can be found in the professional literature, which often uses the term “smart” as a jolly joker. Therefore, let us elaborate further on the meanings of smartness in the context indicated by the title. However, it must be mentioned before everything that the practice of design, as well as the theories of design, goes through subsequent paradigmatic changes due to external influences and internal necessities. A fact of the matter is that the focus of value generation, artifactual paradigms, enabling technologies, applied methodologies, and business strategies of design are continually enriched (Fig. 1). While useful products and manageable processes were in the focus of industrial design engineering before and in the 1950, the attention shifted to multi-disciplinary systems and related services in the 1980s (Horváth, 2020a). In addition to paying attention to products and services (Biehl, 2017), the concepts of environments and experiences appeared as part of a paradigmatic shift at the turning of the present century. Neuhüttler et al. (2017) provided an example of considering value propositions in the development of smart service business models. Smartness can be associated with all forms of value propositions, but it will have a different flavor in each case. Unfortunately, there are only tentative definitions for smart experiences, smart environments, smart systems, and so forth.

The starting point of our discussion can be that, in human behavioral context, “smart” means (i) behaving with above-average logical reasoning skills, (ii) successfully applying them to solve technical, scientific, social, economic, and political problems, and (iii) having pragmatic, emotional, and ethical concerns. This circumscription has two direct implications: (i) smart humans/systems are better at solving problems than others and (ii) what helps better solve problems also makes the problem solver smarter. Thus, smartness is a measure of the ability to act in a situation based on innate competences and the knowledge and information known. This explains why smart is often associated with the actions and ability to navigate in an unknown place, or solving complicated puzzles, or convincing others about the truth of a claim.

Even if philosophical studies and speculations are also considered, the literature seems to be incomplete on the fundamentals of smartness in nonhuman contexts. Notwithstanding, there are some seminal and influential works that warrant more attention and exposition (Adams, 2012). Many authors put an equal sign between smartness and intelligence as a characteristic
of behavior and capabilities (Shang and You, 2019). They consider smartness and intelligence as synonyms and use these two terms interchangeably (Wang, 2019). However, we consider them different and demarcate the two notions in this paper, in particular in the context of intellectualized engineered systems (IESs). The bottom line is that we consider intelligence as an ultimate experiential manifestation of being and behaving as human (Herring, 1925). As a native gift, it is generic, global, and multifaceted. On the other hand, smartness is specific, localized, and purposeful. These two natural phenomena have different indicators and measures (Rosenblueth et al., 1943).

Rough thoughts on smartness versus intelligence identify discriminators such as awareness versus consciousness, reasoning versus thinking, acquiring versus learning, and self-enhancement versus self-adaptation. Personal intelligence is considered as a kind of given (IQ of individuals is supposed to be stable as they age), while personal smartness is accepted as changing with intellectualization, practicing, and experiencing. Intelligence is often considered a measure of the ability to learn (Meisenberg and Lynn, 2011). This suggests that intelligence is an intrinsic general property. Smartness is not about the probability but more about the rate of accumulation of competences, knowledge and information of an individual. A human or a system can become smarter by improving the competences through practicing and obtaining more relevant information and knowledge (Rindermann and Ceci, 2009). Eventually, it is the ability to survive through any means.

The reproduction of human smartness in IESs is a current hot issue. It is assumed that system smartness means that the access and extraction of information and knowledge for/by IESs is not static and that they may exhibit evolving problem solving capabilities. And this is where artificial narrow intelligence (ANI) comes to the scene (Alippi and Ozawa, 2019). Current artificial intelligence is an eclectic body of unintegrated knowledge. Several constituents and manifestations of ANI have been developed and studied, and artificial general intelligence (AGI) and artificial explainable intelligence (AEI) research have gained a lot of impetus in the last decade. Three basic and generic forms of artificial intelligence are distinguished: (i) assisted intelligence, (ii) augmented intelligence, and (iii) autonomous intelligence. They offer a rich and rapidly growing set of problem solving technologies. Enabling system-level smartness by applying ANI or AGI or AEI technologies is in the center of current cognitive systems research and development, but its future possibilities and implications are seen and judged differently (Ren and Chen, 2019). As shown in Figure 2, four conflicting worldviews and philosophical positions are competing. For this reason, the future situation does not seem to be clear in the present days, even if the technological trends and future affordances are considered.

We claim that ANI is not more and not less than a popularized and ambitious formulation of the name of a wide but the fragmented domain of research in reproducing various manifestations of human intelligence. In the framework of cybernetics, intellect is achieved by a subsequent integration and abstraction of data, information, and knowledge. We are treating data, information, and knowledge as three distinct tiers. Data are regarded as a collection of facts, information is the meaning of interrelated data, and knowledge is the problem solving power of integrated and abstracted information, while intellect is the ability of properly applying knowledge in various and varying contexts. Sensors gather signals and convert them to data. Information structures capture and encode relations among data and expose their meaning. Advanced reasoning mechanisms (such as artificial neural networks) discover relationships hiding in massive data streams and convert it into patterns of knowledge. Context management learns the conditions (meta-knowledge) of the proper application of problem solving knowledge.

![Fig. 1. Shifting artifact paradigms and value generation in design.](image)

![Fig. 2. Positions concerning the current and future status of artificial intelligence.](image)
Considering all the facts and issues, what does smart mean in the context of design processes and engineered systems, or in a combined context? The major questions about the essence and forms of relationships between smart design and smart systems are indicated in Figure 3 (Horváth, 2020a). What does smart design have to do with smart systems and what do smart systems have to do with smart design? What is smart design of smart systems? What does connect them? These questions will be addressed in the rest of this paper.

Part 1: Smart design

Dual nature of design smartness

The dual nature of design smartness mentioned in the title of this subsection originates in the word “design” itself, which is used here as an adjective, but it can refer either to a purposeful process or to a purposeful artifact. Both the process and the artifact can be smart, but differently. The fact of the matter is that we focus on the smartness of the design process, since the concept of “purposeful” artifacts eventually leads us to the domain of smart systems. How can the phenomenon (observable process and activities) of smart design be defined with simple terms? Perhaps the simplest definition would be: Smart design is a process enabled by artificial intelligence. However, this definition does not claim more than saying: a house is heated by gas or a vehicle is fueled by petrol. Therefore, for a better articulation, a more detailed definition is needed. Smart design has different relationships with learned human professionals. Some writers stated that any design method/methodology dealing with a part or the whole of a design process with a smart technology will constitute a smart designing process.

But what designers actually do when they do smart designing? Smartness makes it possible for designers to get additional intelligence, analyze on a wider basis, decide more objectively, and operate under uncertain circumstances, not just in dynamic circumstances. Schuh et al. (2019) discussed that smart products are based on digitized (or cyber-physical) products, which consist of physical, intelligent, and connected components, and are capable of a digital upgrading through internet-based services. In their view, the primary function(s) must remain in place in smartification. Vroom and Horváth (2014) used the term cyber-physical augmentation to describe the process of equipping conventional products and systems with smart functionality. Ko et al. (2012) also exemplified that cyber-physical augmentation can introduce novel primary functions. In another research project, the term “cognitive engineering” was adopted and adapted to describe the interconnected processes of (i) intellectualization of product/systems and (ii) designing physical, cognitive, and affective interaction of humans with smart products/systems (Horváth, 2020b).

It was discussed that the objective of the cognitive engineering of ANI-enabled smart products is to ideate surprisingly genuine and useful behavioral functions and behaviors, such as awareness building, abductive reasoning, operation forecasting, apobetic interaction, and conditioned self-adaptation. Other authors argued that the cognitive engineering also aims at optimizing “human-in-the-loop” and “system-in-the-loop” situations (Woods and Roth, 1988; Roth et al., 2002). We believe that, in addition to these, smart designing needs a novel design thinking, which takes counts on the changing role of the subject of design in terms of goal definition and process implementation. It has to be mentioned that we do not have a specific model of design thinking yet that would simultaneously enable researchers, developers, and designers to implement computational support systems, facilitate research, guide education, or support interdisciplinary collaboration in the field of smart design of smart systems.

On the other side of the coin are smart products, services, and experiences as the manifestation of smart designs. This area is still facing an even more intrinsic definitional challenge. The issue is what establishes a smart design (as a value generator)? In the industrial practice, information and communication technologies have been merged into common domestic products and they were called smart. Some say that the products affected mainly by the Internet of Things are smart products (Bello and Zeadally, 2019). In this view, various goal-driven combinations of hardware, software, and cyberware technologies endow products with smart functionality. Obviously, this kind of simplifications fail to provide a proper characterization of smart products. The following concise overview of the related publications casts lights
on more articulated definitions that consider important functional, implementation, and operational features too.

Bonner (1999) addressed the implications of using intelligence in consumer products. Gutierrez et al. (2013) tried to provide a consensus definition for the term “smart product,” but ended up with a conclusion that the distinguishing characteristics of smart products and intelligent products are similar. This conclusion that the term “intelligent product” is a synonym of “smart product” is also proposed by the paper of Filho et al. (2017). Actually, are there any features (functions, arrangements, behavior, etc.), which a design should necessarily exhibit to qualify as a smart product? Wuest et al. (2018) proposed pro-activeness as a necessary characteristic to arrive at intelligent products. Rijswijk and Hultink (2009) proposed a set of abilities (dimensions) to collectively characterize smartness of current and future smart products, namely (i) autonomy, (ii) adaptability, (iii) reactivity, (iv) multifunctionality, (v) ability to cooperate, (vi) human-like interaction, and (vii) personality. Having these, smart products show a range of capabilities that can only be found in nonsmart products to a limited extent. In their study, they measured the reflections of stakeholders on smart products in terms of innovation attributes such as (i) relative advantage, (ii) compatibility, (iii) observability, (iv) complexity, and (v) perceived risk. According to the technology-oriented definition of Abramovici (2014), smart products are cyber-physical systems (CPSs) defined as intelligent mechatronic products capable of communicating and interacting with other CPSs by using different means such as Internet or wireless LAN. In addition, Abramovici et al. (2016) argued that virtual twins are integrative components of smart products.

Hicking et al. (2018) claimed that, in order to maintain or improve their competitiveness, SMEs must transform their existing business models considering new smart products and/or the smartification of already existing products. Chowdhury et al. (2018) discussed that the use of smart technologies and components such as sensors, wireless connectivity, control systems, and machine-embedded software has led to a new generation of industrial product-service systems that are called smart product-service systems (S-PSSs). They reviewed the state of progress in the field of S-PSSs and found that S-PSSs are characterized by digital resource-driven value creation processes and business models and offer novel value-creating features such as (i) boundary spanning with digital boundary objects, (ii) intelligent dynamic functional capabilities, (iii) active value creation with customers through digital resource integration, (iv) using digital platform for value co-creation, (v) employing new business models based on S-PSS packages and smart automation services, and (vi) using product generated digital data for improved customer relationship and PSS redesign.

Smart products need to exploit more far-reaching self-organization approaches beyond service composition as investigated in “mainstream” (semantic web) service-oriented software research (Gold et al., 2004). Furthermore, as Mühläuser (2008) argued, smart products can be conceived as “services” from the IT perspective and can leverage off intensive research on “service composition” and sub-issues like service (self-)description and discovery, orchestration and choreography this way. Based on these considerations, he proposed the following “early” definition: “A smart product is an entity (tangible object, software, or service) designed and made for self-organized embedding into different (smart) environments in the course of its lifecycle, providing improved simplicity and openness through improved product-to-user and product-to-product interaction by means of context-awareness, semantic self-description, proactive behavior, multimodal natural interfaces, AI planning, and machine learning.” Artificial intelligence technologies also contributed to the development of personalized interface agents for high functionality interfaces that can operate over a broad spectrum of interaction tasks (Handosa et al., 2020).

The necessity of a more detailed definition is underpinned by the need to exactly specify: (i) who executes or what establishes a smart design process? (ii) what is the subject of a smart design process? (iii) what is the objective of a smart design process? (iv) how is a smart design process conducted? (v) what intellect is used to enable a smart design process? and (vi) what are the affordances and limitations of smart design processes? Like conventional and computer-aided design, smart design also includes creative and reflective design actions. There is a difference between smart design and knowledge-intensive computer-aided design, the latter being supported by (i) lifecycle information management (Yang et al., 2007), (ii) multi-disciplinary collaboration of teams (Zha et al., 2003), and (iii) using knowledge warehouses and design ontologies (Sun et al., 2010). In a smart design process, the role of humans is changed. It is also an issue of smart design processes to what extent the decision-making and inference mechanism are intervened by humans. Current smart design approaches intend to make designing smarter by including means that support increasing and automating problem solving competences (Parasuraman et al., 2000). A smart design process often implies that designers are assisted by some sort of tools or platforms, which can offer some good/better outcomes/options, for example, when designers do not necessarily have to be involved in the decision making (Bucklin et al., 1998). The resources (enablers) of a smart design process are AI-means, knowledge bases, inference engines, validation mechanism, etc. As a bottom line, smart design is closely related to (and is dependent on) structured design thinking and system thinking (Adams et al., 2014).

**Essence of smart designing**

The very essence of smart designing is smartification, which may work toward many different objectives (Luis-Ferreira et al., 2019). As defined by Schuh et al. (2019), smartification is understood as the digital refinement of an existing product by embedding digital technologies and smart services. In this intentional framework, theories, methods, technologies, competences, and applications play an equal role. On the one hand, independent from the form of value generation, smartification intends to realize several distinctive characteristics of (smart) value carriers, such as (i) personalization (customization according to the unique characteristics and needs of the stakeholders), (ii) connectedness (opportunity of communicating, integrating, and bundling with other systems), (iii) situatedness (monitoring dynamic internal/external circumstances and influential effects in order to maintain optimal operation), (iv) awareness (consideration of changing and emerging objectives, resources, situation, and constraints), (v) adaptiveness (changing operational goals according to new conditions and affordances), and (vi) proactivity (anticipation of stakeholders’ intentions and plans and new affordances). On the other hand, smart design should cope with many challenging paradigmatic features of value carriers, such as (i) heterogeneity (designing hardware, software, and cyberware constituents in synergy), (ii) compositionality (holism of top-level system operation), (iii) timeliness (operation without time delay (zero-time) or working in near-zero time), (iv) controlling (implementing supervisory
and self-control as human-in-the-loop and system-in-the-loop), (v) automation (augmenting interactive operation with autonomous task execution, and (vi) dependability (ensuring security, safety, reliability, resilience, and other operational characteristics).

There is a wide range of enablers for smart design (Zheng et al., 2019). They include analog and digital hardware means, software and media tools, as well as learning/reasoning mechanisms, and cyberware including (big) data, synthesized information, and human and system knowledge. As concrete resources, smart designing blends: (i) human creativity, intents, abilities, competencies, and experiences, (ii) networked physical and virtualized computing environments (such as edge, fog, and cloud computing), (iii) penetration into real-life processes and generation of lifecycle-related big data, information, knowledge, and meta-knowledge, (iv) integrated warehouse of manual and digital ideation, conceptualization, architecting, modeling, analysis, simulation, tools, etc., and (v) tailored problem solving intellect offered by artificial intelligence approaches. In combination with traditional design methods, the framework of generative design utilizes ANI and knowledge representation techniques to generate better designs in a shorter time (Akinsolu et al., 2019).

Industrial smart design is not separable from the downstream activities of production flows, and smart design cannot neglect smart manufacturing (SM) processes (Subramanyam and Lu, 1991). According to Wang et al. (2018), SM refers to a new manufacturing paradigm, where manufacturing machines are fully connected through wireless networks, monitored by sensors, and controlled by advanced computational intelligence using advanced data analytics to complement physical science to improve product quality, system productivity, performance, decision making, and sustainability while reducing costs. SM is an immediate objective in the consumer goods industries, which are driven by fast-changing customer demands and, in some cases, tight regulatory frameworks, and their progress strongly depends on digitalization, digital transformation, and the use of large datasets together with predictive models and solution-finding algorithms (Dai et al., 2015). The use of data, models, algorithms, and computer control is supposed to optimize the whole supply chain in the production of manufactured products (Litster and Bogle, 2019). The core element is the data-driven smartness.

**Problematics of a smart design methodology**

Some researchers expose smartification as the doctrine of “smart designing of anything.” It is possible, in principle, but actually the doctrine largely depends on the artifactual paradigm and the practice of value generation. Smart design is not computational design or automated design or axiomatic design, but a sophisticated design process completed by a synergistic cooperation of human experts and intellectualized systems. As mentioned earlier, it is enabled by an extensive utilization of AI-based system resources (Smithers et al., 1990) and system-generated synthetic knowledge, which augments intuitive human knowledge (Horváth, 2020b). Though not exclusively, it has effects on the methodology of designing smart products, systems, services, and experiences too. However, this variety of value manifestations is demanding and calls for a directed articulation. For instance, Loizou et al. (2019) brought up that traditional requirement engineering approaches may not be sufficient for creating a robust information platform for the conceptualization of smart systems and proposed co-design as an alternate. Others proposed the concepts of digital twins as an approach of lifetime information aggregation in the case of connected designs (Tao et al., 2019). Nevertheless, it has been recognized that the approaches proposed so far do not provide an integrated and comprehensive digital-twin approach to support the complete smart product lifecycle from the stages of requirements elicitation, product design, customization, and production monitoring (Boschert et al., 2018).

A full-value methodology of smart designing should be built around and upon the recognition that smart design and smart systems are inseparable. In the language of mathematics, they have a bijective relationship, that is, one-to-one correspondence between the various elements of the two domains. It means that each smart system needs a dedicated smart design methodology, and each smart design methodology is tailored to a particular (family of) smart systems. There are no unpaired elements. But it implies that, most probably, there is no generic all-embracing smart design methodology. Having claimed this, it has to be mentioned that this does not exclude the possibility of abstracting a meta-methodology that represents a genotype-methodology from which many different phenotype-methodologies and eventually prototype-methodologies could be derived. Current research still has only a limited contribution to understanding and explanation of this issue. Our proposition is that a systematic methodology of smart design should have five pillars, as shown in Figure 4. Specifically, it should (i) be underpinned by a harmonized composition of fundamental theories, laws, and principles, (ii) have application context-dependent adaptable and sharable procedural scenarios and workflows, (iii) have a pool of application-independent and application-dependent conventional (manual), computational (digital), and intellect-driven (semantic) methods, (iv) exploit a related enabling technological infrastructure and intellectualized toolboxes, tools, and instrument, and (v) a set of consistent and self-decidable applicability and performance criteria.

Revisiting the issue of the above-mentioned bijective relationship, smart designing is the core process of creating smart systems, whereas smart systems provide the front and back ends to smart designing in that the environment states are acquired to trigger the smart designing process, and the design decisions will be executed by the smart system. Intellectualized systems are deemed to contribute to smart design as self-contained designing agents, which are able (i) to sense, model, reason, learn, and adapt according to their objectives, states, and environments; (ii) to help address complexity, heterogeneity, interoperability, communication, productivity, etc.; and (iii) to identify major conflicts in these and produce solutions to resolve the conflict. Our literature study explored that there are no comprehensive, tested, and practical underpinning theories documented in the literature yet. Notwithstanding, there are a few of incomprehensive, sketchy, and untested partial methodologies and collections of methods proposed.

A fundamental question related to the methodology of smart design is that how much smart design can be independent of the specificities of the designed smart system? In the near future, smart designing mechanisms must be the core of a self-organizing smart system. With the absence of fully automatic smart designing algorithms, one can look into the design methodology/methods that are currently aiming at human designers and analyze how much dependence they have on human designers and how these dependences can be further automated? From the perspective of companies developing consumer durables, Schuh et al. (2019) proposed a strategic methodology for smartification, which does not extend to the technical issues of realization.
product-as-a-service or product-as-an-experience. If ANI (as well as AGI and AEl) plays a central role in the systematic methodology of smart design, then the primary issue is its role in theoretical underpinning, procedural execution, methodological support, instrument provisioning, and assessment. This was found rather under-elaborated. There are some progressive steps and approaches, for instance, in the field of electronics and software design, which are deemed to be application-constrained-specific smart design methodologies. As arbitrary examples, the works of Tsukuda et al. (1993) and Lin et al. (2017) can be mentioned.

The methodology of artificial intelligence-driven smart design is variously and restrictively approached in the literature. Pessôa et al., 2017 can be mentioned. Mattern (2018) interpreted it as calculative composition and addressed a number of related ethical issues. From the technological side, Tang (1997) proposed a knowledge-based architecture for intelligent design support. The co-creation aspect of smart design was addressed in the paper of McCormack et al. (2020), which dealt with real-time collaboration with creative artificial intelligence, and the paper of Fu and Zhou (2020), which focused on human and AI co-creation. Quanz et al. (2020) proposed a machine learning-based co-creative design framework. Various application opportunities of AI-based design methodology were studied. Fisher (1986) considered an AI-based methodology for factory design, while Chien and Morris (2014) discussed space application approaches of artificial intelligence. Oh et al. (2019) introduced the concept of deep generative design.

Part 2: Smart systems

Smartness of systems

In the literature, a “cognitive system” is seen as one that performs cognitive work via cognitive functions such as communicating, deciding, planning, and problem solving. The term “smart system” is not congruent with this definition. It emerged some 60 years ago and has gone through at least three metamorphoses. The first mentioning can be traced back to the very beginning of 1970s when the Texas Instruments company developed its first microprocessor. Twenty years later, the term was interpreted when the Internet was realized and created the basis of semantic content repositories. In the mid of the first decade of this century, the concept of CPSS was introduced. Usually, they do not only rely on the computer network but are also equipped with reasoning capabilities, which are needed for human-supervised or automated smart problem solving. Some years ago, the concept of systelligence (self-managed system intelligence) emerged. It is supposed to make intellectualized engineering systems capable to sense, stream data, build awareness, infer and reason, monitor operation, plan self-adaptation, and enhance their performance as focused smart problem solvers (Horváth, 2020b). Thus, rapid technological advances may be one of the reasons why system smartness means a different thing for everyone dealing with it!

Bures et al. (2020) discussed that smart systems manifest as a heterogeneous, interconnected landscape of various applications of Internet of things, CPSS, and/or smart sensing systems. Furthermore, they saw a typical smart system application as the compositions of autonomous yet inherently cooperating components, including hardware units running upon specific networks and associated software components, achieving smartness by sensing and operation, both in an autonomous and in a collaborative manner. Components proactively sense the environment and provide their knowledge to other components to allow them to take smart and well-founded decisions. A typical conceptualization of a smart system is that it can (i) change its reasoning strategy and activate problem solving agents accordingly and (ii) learn new models to process changing and growing set of (sensor) input data or knowledge base contents. Their computing mechanisms are preprogrammed for doing this, though the needed adaptation and computational resources are determined at run-time. Systems adapt themselves within an anticipated envelope of changes. Smartness does not mean that all decisions are made by the reasoning mechanisms, but with the involvement of humans in the system operation loops (Schirner et al., 2013). This can be done by mixed-initiative reasoning (shared decision making and action taking) (Dautenhahn, 1998).

Intellectualized engineered systems present (i) materiality, (ii) agentivity, (iii) intellectuality, (iv) purposiveness, and (v) transformativity. These all are needed for the realization of smartness from a functionality perspective. Behavioral smartness arises when intelligence meets the context. It implies that system
Smartness is also a judgment from a nonfunctional (experiential) perspective. This judgment may come from the actors of the application environment, including humans and other systems. This duality is shown in Figure 5, which interprets system smartness as an evolving phenomenon. This is enabled by the train of evolving computational problem solving mechanisms and functions. A nonsubjective judgment of the experienced system’s performance requires objective criteria and proper measures. These are not available in the literature yet. Nevertheless, based on the analogy of experiencing smartness in human problem solving, we regarded (i) ingenuity (inventiveness in the human context), (ii) dexterity (agility), (iii) convincingness (proficiency), and (iv) dependability (reliability) as observable nonfunctional characteristics in the context of impressionable problem solving performance of smart CPSs (Horváth, 2020a). These characteristics interconnect the experienced quality of smart problem solving and sophistication of system functionalities, as well as the stakeholders of the system.

Ulsoy (2019) gave an explanatory example by viewing mechatronic systems as the representative of synergistic integration of mechanics, electronics, and computer science principles in their operation. The role of mechanisms is even more influential in the case of smart CPSs expected to implement complex and interdependent reasoning, learning, and adaptation processes (Tavčar and Horváth, 2018). Obviously, for this reason, their overall design is more challenging than designing their hardware, software, and cyberware components even in the application cases of moderate complexity. This is evidenced by many studies in the literature (Lieberman et al., 2014; Huang, 2016; Mallikarjuna et al., 2020).

Parallel with the technological and functional advancement, the holistic operational mechanisms of complex intellectualized engineered systems are getting bigger attention. Computation-based operation mechanisms are seen not only as abstract principles but also as features of the created reality that can be used as an implementation concept of complicated synergic (nonadditive) operations. This is line with Mario Bunge’s ontological claim about the mechanisms of the experimented physical reality, which explains why it is doing and what it does (Bunge, 1997). This view regards mechanisms as the highest level functional organization capacity of systems. From a practical perspective, a mechanism synergistically integrates operational principles. Implemented mechanisms lend themselves to central physical and computational processes in concrete systems. A particular dependable mechanism is inseparable from practical systems, since it integrates natural, technological, cognitive, temporal, and social dimensions. According to this mental model, smartness is an outcome of a successful and evidenced implementation of the underpinning mechanisms. There is a probability that this mechanism-oriented thinking becomes more widespread and influential in the context of intellectualized engineered systems. As suggested by Bunge (2004), the bottom line question is: what constructed mechanism makes a smart system purposefully working?

Many researchers think that a computational intelligence-assisted design framework is strongly needed for smart systems. Their operational and behavioral self-adaptation would need some sort of dedicated system intelligence (Ashby, 1947). In other words, the ability to extend the knowledge base and enhance the reasoning mechanisms would be a measure of the intelligence of systems. In a simplified form, the smartness of intellectualized engineering systems could be measured in terms of their potentiality to solve a range of challenging real-life application problems, while their intelligence could be measured in terms of their abilities to extend their knowledge household and problem solving mechanisms. In the language of system engineering, it is directly related to the innate self-adapting capabilities and possibilities (Sabatucci et al., 2018). Adaptation can be based on external supervision and on update/upgrade agents, whereas self-adaptation assumes internal supervision and agents to establish a more beneficial modus operandi. It assumes, for instance, reflexive goal, situation, and state awareness, supervised and unsupervised learning, operation and performance planning, and functional, structural, and behavioral transformations.

Paradigmatic features of smart systems

What are the signatures of a smart system? This reads as a benign question, but it is not. The reason is that intellectualized engineered systems ontologically (paradigmatically) change over...
time. This recognition lent itself to a reasoning model that, considering the continuous increase of the level of self-intelligence and the level of self-organization, identified five generations of systems in the context of CPSs (Horváth et al., 2017). This model shown in Figure 6 interprets smart CPSs as second-generation systems and identifies self-awareness and self-adaptation as their distinguishing paradigmatic features. According to this model, an intelligent system is supposed to have system-level consciousness and should be able (i) to make (critical) decisions autonomously (without supervision or human intervention) based on (a) novel, (b) abstract, (c) uncertain, and/or (d) incomplete information; (ii) to create, propose, maintain, and devote values and perform value-based decisions that it has or other systems have created; (iii) to define new objectives, reprogram itself, acquire proper knowledge, and resolve its operational conflict even if only imperfect information is available; and (iv) to reproduce itself and survive in situations that were not foreseen at the initial implementation of the system and thus were not part of the original design intentions. Besides this model, various sets of distinguishing features and classifications have been proposed. As the aspects of classification, (i) implemented technological functions, (ii) services for application domains, and/or (iii) engineering performance characteristics have been considered. Based on the overall engineering performance characteristics, (i) passive systems, (ii) reactive systems, (iii) active systems, and (vi) proactive systems have been distinguished. The model proposed by Schuh et al. (2019) arranged the digital features of products into eight categories according to (i) the type of data collection, (ii) the type of interaction, (iii) the place of product intelligence, (iv) the place of data retention, (v) the type of interconnectedness, (vi) the type of connectivity, (vii) the degree of product intelligence, and (viii) the degree of independence. Some publications claim that the key factors or distinguishing paradigmatic features of smart systems are (i) connectivity and networking capabilities, (ii) operation under self-control or autonomously, (iii) context-sensitive interaction with users/devices, (iv) low-energy consumption and environment friendliness, (v) relying on cloud infrastructures and services, and (vi) using techniques of artificial intelligence. As most frequently identified ones in the literature, we propose (i) multi-level cooperative openness (Trokhimchuck, 2017), (ii) system-level reasoning and learning capabilities (Akbar et al., 2017), (iii) system operation in dynamic contexts (Alegre et al., 2016), (iv) semantic, pragmatic, and apobetic interaction (Jameson, 2007), (v) self-supervised planning and adaptation (Seilonen et al., 2003), and (vi) ensuring multi-aspect dependability (Kamal Kaur et al., 2018; Fig. 7).

The concept of multi-level cooperative open-ended systems has significance from the viewpoints of (i) organizing heterogeneous systems into a system of systems, (ii) the independent development of conceptually diverse subsystems, and (iii) facilitating continual incremental evolution (Anders et al., 2016). Multi-level cooperative openness offers flexible interconnection and interoperation among all constituents that incidentally joining or leaving the system ensemble. Open systems are characterized by intrinsic incompleteness and decisional locality (Hewitt and De Jong, 1984) and imply the need for self-organization potentials (Gershenson et al., 2018). The cooperative open-ended smart system should be able to amplitively integrate diverse, multi-source, data streams (Saracco et al., 1990). It means that they are supposed not only to semantically merge data streams but also to extract or synthesize additional intellect that is not conveyed in the original data streams. This creates an opportunity to dynamically manage the context information derived from multiple sources.

System-level reasoning (SLR) capabilities enable systems to make decisions similar to those of humans and manage...
themselves or other linked systems. Two fundamental assumptions are that (i) no learning is possible without the application of prior domain knowledge and (ii) learning is possible without being explicitly programmed. Mathematical logic and ANI development have offered many “standard” reasoning mechanisms, which proved to suffer limitations when applied in specific problem-solving contexts. Addressing real-life generic application problems (e.g., control of a self-driving car) needs multiple, inter-operating, task-specific problem-solving mechanisms, instead of one monolithic one (Geng and Cassandras, 2011). This is deemed to be one of the most determining features of smart systems (Wu et al., 2012). The principle of divide-and-conquer is applied to overcome developmental and application complexities and to increase reusability and adaptability. Most of the process-based reasoning mechanisms utilize cyber-physical intelligence and implement inductive inferencing. They may be deterministic (scenario-driven) and nondeterministic (knowledge driven). Typical examples for the former are the scenario-based causal decision-making mechanism proposed by Conrado and De Oude (2014) and the adaptive scenario-based reasoning mechanism of Cheng and Wang (2012). The latter is exemplified, for instance, by the semantic similarity-driven query intent discovery mechanism proposed by Fariha and Meliou (2019) and the procedural abduction mechanism synthesized for run-time adaptation by Horváth (2019). The computational elements of procedural abduction are shown in Figure 8.

Crowder et al. (2020) discussed the foundations of system-level thinking for artificial intelligent systems. Bijlsma et al. (2019) approached the issue of SLR from the perspective of the knowledge domain. Khan et al. (2014) discussed a typical example of advanced vehicle-level reasoning in an aircraft system. As discussed by Reed and Pease (2017), algorithmic reasoning mechanisms confront obstacles when they operate on ambiguous, conditional, contradictory, fragmented, inert, uncertain, or imperfect knowledge. Due to the recent trends of artificial intelligence research/development, system-level learning (SLL) has been synergistically interconnected with SLR. One example is deep learning that connects computational reasoning to nonprogrammed computational learning. The theory of computational SLL intends to explain and optimize (i) the inference principles of algorithmic and nonalgorithmic learning, (ii) the design and analysis of machine learning algorithms, and (iii) the sorts of computationally learnable problems. Among others, Bayesian, supervised, unsupervised, deep, and adversarial learning strategies are applied. The optimization of the approaches is a concern. For instance, deep convolutional generative adversarial networks apply certain architectural (topological) constraints in unsupervised learning (Gao et al., 2018). Fiser and Lengyel (2019) presented a probabilistic framework for perceptual and statistical learning. In the theoretical work of Kinouchi and Kato (2013), SLL activity was considered a primitive implementation of consciousness. Kinouchi and Mackin (2018) proposed an SLL consciousness model that modifies the configuration and states of a humanoid robot by self-action-decision functions toward autonomous adaptation. Kitagawa et al. (2018) investigated multi-stage learning in the context of robot application. Darling et al. (2016) introduced the concept of emergent learning as a framework for the whole-system strategy of learning and adaptation.

Building context awareness (Alegre et al., 2016) and situation awareness are often mentioned functions not only of humans but also of self-regulating systems (Endsley, 2018). Dey et al. (2001) have defined the term context-awareness of systems as the capability of using the context to provide relevant information and/or services to the user, where relevance depends on the tasks of the users. Perera et al. (2013) argued that context awareness is an ingredient of the smartness of pervasive computing systems and proposed a comprehensive taxonomy of the context-information-related functionalities, which include (i) context acquisition, (ii) context modeling, (iii) context reasoning, (iv) context distribution, and (v) context processing functions. The idea of dynamic context inferring grew out from the concepts of context-aware computing (Li et al., 2020) and context-aware applications (Koch et al., 2019). The last years have witnessed the proliferation of context-aware recommender systems (Haruna et al., 2017) and context-aware self-managing systems (Shishkov et al., 2018). Measuring awareness is not only a technological but also a cognitive metrological problem due to its abstractness and the lack of reference (Endsley, 1995).

Establishing model-based system self-awareness is a challenging information engineering task that spreads over time and cannot be reduced to elicitation and management of context information (Matthews et al., 2001). It is made complicated by its many facets such as identity, goal, state, spatial, temporal, behavioral, context, and social awareness. They together have led to the issue of all-inclusive awareness and, eventually, to mimicked consciousness of smart systems (da Silva and Gudwin, 2010).

Apart and together, semantics, pragmatics, and apobetics play a key role not only in interpersonal communication and interaction but also in collaboration with smart systems (Horváth, 2012). The bottom line is achieving sufficient informing and successful execution (Cena et al., 2019). From a design perspective, they work with different theories, criteria, principles, and approaches. Semantics, pragmatics, and apobetics consider the achievable objectives, the ways of achieving, and the implications and reflections, respectively (Horváth et al., 2014). Semantics supports sharing meaning and understanding in mental (interpretative) and physical (manipulative) actions (Endert et al., 2011), whereas pragmatics extends it with the concern about the success of achieving the intended or expected goals of the conducted actions. Apobetics concentrates on the relations between the way and the effects of how actions are made and on the implications caused by the results of actions on the stakeholders, as well as on their reflections. It investigates the quality of triggered cognitive and emotional reflections. Though the need for new methods of human
interaction with smart systems and environments is widely recognized (Jameson, 2007), the progress is hindered by the lack of theories of apobetics. Computational sentience is almost neglected in the current literature of interaction research. As Mayer et al. (2014) posited, the interaction with smart systems (things) requires not only different interaction modalities and technological enablers but also a different mentality. Smartness opens up numerous new interaction possibilities in particular in the cognitive, perceptive, and the affective domains (Luyten and Coninx, 2005). System design should differentiate between the modes of executing actions, which can be (i) without any degree of freedom, (ii) with a limited degree of freedom, and (iii) with the maximum degree of freedom. Furthermore, it should take both the purpose of actions and the expected reflections into account (Bujnowski et al., 2011). Apobetic-level investigations should consider these as problem solving constraints (Yang et al., 2018).

Involving simultaneous and rapid changes in operational objectives, functionality, architecture, computation, interaction, and security, behavioral adaptation is a challenging paradigmatic feature of smart systems (Dobson et al., 2019). Systems equipped with this ability have been variously called self-adaptive systems, self-managing systems, or self-organizing systems. Often self-healing systems and self-optimizing systems are also sorted in this category (Sabatucci et al., 2018) proposed a meta-model that identifies four types of self-adaptive systems: (i) Type 1 (anticipating changes and possible reactions at design-time), (ii) Type 2 (equipped with alternative strategies for reacting to changes), (iii) Type 3 (aware of its objectives and operates with uncertain knowledge), and (iv) Type 4 (able of self-modifying its specification as biological systems do). To adapt themselves, smart systems need to use: (i) sensory perception (detecting and anticipating changes in the environment), (ii) cognition (reasoning about perceived changes and deciding on the best action), (iii) execution (controlling the implementation of cognitive decisions), and (iv) provisioning assurances (De Lemos et al., 2017). The major issue is that there is no general theory to explain self-adaptation in all contexts and there is no clear view on how to realize self-adaptation in a self-supervised manner at run-time (Weyns, 2019). Self-planning of adaptation is a complicated data-driven task that should consider not only the states of operations and the environmental changes but also the affordances of the system and the availability, inclusion, and exclusion of resources (Yamanobe et al., 2017). Run-time planning of adaptation is supposed to happen in a proactive manner (Muccini and Vaidyanathan, 2019). Reactive self-adaptation cannot avoid negative events and cannot derive better operation modes. Ideally, proactive self-adaptation may resolve these drawbacks, if it is able to detect the need for adaptation by online testing and define an adaptation plan in a narrow time window. However, current research cannot offer solutions for these yet. In the last years, the MAPE-K methods have been used successfully for the adaptation of software (da Silva et al., 2020). Figure 9 shows the conceptual workflow of the proactive self-adaptation of CPSs. Currently, (i) no formal proofs of reaching good functional and architectural solutions are known and (ii) the behavior (operation under application circumstances) of a system cannot be validated without deploying it in a real environment. Many authors identified adaptation as a configurable interoperability problem and emphasized that self-management is inseparable from autonomic computing (Rutten et al., 2017).

In the case of smart systems, the criteria of dependability appear in a more complicated form than in the case of nonintellectualized but mission-critical systems (Kamal Kaur et al., 2018). In the conventional interpretation, dependability integrates such system attributes as availability, reliability, security, safety, integrity, survivability, and maintainability (Avizienis et al., 2001). Mainly focusing on hardware and software dependability (Bernardi et al., 2013), early works proposed redundancy as means of increasing their dependability and model-based analysis and simulation as useful methods (Sharvia et al., 2016). In the
case of smart systems, dependability also concerns the dependability of reasoning mechanisms, system knowledge, and the decisions made, and entails the need for multi-concern intellect analysis and the adoption of novel certification schemes (Biggs et al., 2011). It is assumed that intelligence provides more robustness as well as a comprehensive self-management of faults. The current research results still show an existing limit line at designing smart embedded systems for dependability (Srivastava and Singh, 2009). The issues related to the evaluation of the dependability of CPSs during the design period are recognized (Gheraibia et al., 2019), but providing any generic solution is difficult due to the increasing aggregate complexity, the operational and environmental dynamics, and the consequences of self-adaptation (Sondermann-Wölke et al., 2010). The dependability of complex smart systems is ontologically emergent and compositionality also plays a crucial role in it (Hartmann, 2014).

Part 3: Coupling of smart design and smart systems

Introducing the concept of connectors

Let us start from what has been discussed in the previous subsections of this paper. The question addressed in this subsection is: does AI create couplings between smart design and smart systems? And, if the answer is affirmative, then what creates what type of relationships? The above discussion of smart design and smart systems revealed that what kind of results of cognitive science, artificial intelligence, computing technologies, and cognitive engineering appear. There are several cognitive enablers that are utilized equally well in smart design and smart systems. The fact of the matter is that these cognitive enablers can facilitate the realization of the concept of partially self-designing smart systems. For this reason, they will be referred to as “connectors” in the rest of this paper. They have a special role in equipping smart systems with design functions and enhancing the methodology of smart designing. As an example of the latter, we may consider (i) helping the exploration of complex solution spaces and affordances, (ii) establishing extended cyber-physical-social environment, (iii) offering application problem-specific reasoning mechanisms, (iv) playing the role of “big brother” in problem solving processes, (v) facilitating transdisciplinary concept synthesis approaches, and (vi) creating the basis for functional design automats.

With some simplification, we can argue that paradigmatic features of smart systems are (i) multi-level cooperative openness, (ii) SLR and learning capabilities, (iii) system operation in dynamic contexts, (iv) semantic, pragmatic, and apobetic interaction, (v) self-supervised planning and adaptation, and (vi) ensuring multi-aspect dependability. The realization of these system features requires the use of AI connectors in the smart design process. Not considering the forerunning activities (fuzzy front-end) and the successive activities (back-end) of the system design process, a smart design process involves (i) functional, (ii) architectural, (iii) hardware, (iv) software, (v) cyberware, (vi) cognitive, (vii) workflow, (viii) interface, (ix) production, and (x) installation design activities. The artificial intelligence-based connectors can establish M to N enabling the relationships between the features of smart systems and the activities of system design. They may facilitate the process of smart designing, but they can also be embedded in smart systems as operation enablers. This explains how AI-enablers serves as connectors. It is visualized in Figure 10.

Primary types of connectors

Intuitively, the AI-based connectors can be sorted into three categories: (i) conceptual (platforms, frameworks, protocols, and experience), (ii) functional (models, mechanisms, and knowledge), and (iii) instrumental (environments, tools, and methods). The connectors in the conceptual category support the blueprinting of smart systems and the cognitive engineering of smartness, while those listed in the functional category support the realization of functionalities. The connectors in the instrumental category are the external enablers of the cognitive design of systems. These will be discussed below. Note that new connectors and/or categories may need to be considered as additional results of artificial intelligence research emerge.

AI-platforms are predesigned modular technological architectures, composed of a core and a periphery, for the seamless integration of cognitive system resources with the potential to easily

Fig. 9. Operational framework of proactively self-adaptive smart systems.
plugin as many hardware, software, and cyberware components as possible or needed (Mucha and Seppala, 2020). They provide most of the interfaces that are needed for their interoperation of components but also allow offer a straightforward approach to customization for purpose. AI-platforms are designed to help connect and integrate components not only physically but also from workflow integration and information sharing points of view (Epstein et al., 2018). As templates, they support building multiple smart application systems within the same functional and technological framework and by utilizing cloud services. Typical representatives are such as (i) Algorithmia, (ii) DeepMind, (iii) CloudCV, (iv) FairML, (v) PsychLab, (vi) OpenML, (vii) Themis-ML, (viii) ParAI, and (ix) TuringBox. Different AI-platforms provide different opportunities and restrictions. Customized add-on and plug-in components are used to complement the standard functionality provided by some platforms. Therefore, they can be seen both as a constraint on the intellect development process and as a facilitator of tailoring problem solvers. The advantage of using AI-platforms is that application systems can be developed faster and cheaper than when they are built standalone (Venkataramani et al., 2020).

AI-frameworks are the conceptual constructs of the cognitive parts of smart systems including logical, functional, architectural, computational, and interaction aspects. Typically, they are extracted, abstracted, or constructed based on implemented smart systems or parts thereof (Torres and Penman, 2020). In general, they allow for the easier and faster creation of applications by purposefully arranging and interrelating operational concepts. Cena et al. (2019) sketched up a framework for incorporating multi-dimensional intelligence in smart physical objects. The generic functionality offered by AI-frameworks can be selectively changed by additional, purposely developed constituents, thus providing application-specific features and services. Using an AI-framework supports intellectual effectiveness, consideration of the requirements, as well as the needed functionality, its feasibility, and the decision on system-level features. In addition, an AI-framework captures the relationships of the cognitive elements and facilitates their analysis. Opposing formal quantitative models, which provide a theoretical explanation, conceptual frameworks provide insights and understanding. There are also different interpretations of AI-frameworks. For instance, Zysman and Nitzberg (2020) proposed a framework for discussing the limits, possibilities, and risks of AI.

AI-protocols are constructed to support the organization of system-internal cognitive workflows or system-external interaction and supervision workflows (Kesavan et al., 2008). As typical
Artificial Intelligence for Engineering Design, Analysis and Manufacturing

in computer science, protocols include a standardized set of rules for preparation, processing, and communicating data, instructions, commands, activities, constraints, acknowledgements, etc. for cognitive problem solving. AI-protocols also enable the collaboration and networking of software agents and the proper scaling of artificial intelligence lifecycle on demand (Kuwabara et al., 1995). AI-protocols are application-specific and propagate different methods of scenario development or process organization. A current concern is ontological modeling of AI-protocols (Zhou et al., 2006), and the development of scalable networking and security protocols (Shu and Lee, 2007).

AI-experience represents the tacit knowledge of designing cognitive capabilities (Simon, 1986). It is often called the know-howlow of artificial intelligence. The subjects can be varied, ranging from insights in efficient problem solving approaches, through deploying AI-models and best practices in a given context, to the limitations and affordances of AI-tools. The development of repositories and warehouses for aggregation and availing of AI-experience is challenging due to intangibility and abstractness of knowledge and subjectivity of experiences. It is also not clarified how to convert experiential know how into teachable and learnable design principles (Liao et al., 2020). Smith (2019) discussed the paradox of being flooded by information about the all-mightiness of artificial intelligence as well as the obvious dreads and lasting uncertainties. While positive experience is seen as a success factor of adopting AI in design, Chen et al. (2020) reported on the lack of knowledge about the success factors of AI adoption in the telecommunication industry.

AI-models cognitively represent some real-world object or phenomenon as a set of logical arrangement, mathematical equations, taxonomical relationships, and computational concepts (Rajaee and Jafari, 2020). In general, AI-models are generated in three stages, including (i) model generation, (ii) model interpretation, and (iii) model validation. Two subcategories can be identified: (i) explicit models and (ii) implicit models. Explicit models rely on some descriptive or explanatory theories, are preprogrammed based on algorithms, and are represented using computational languages. Examples of explicit AI-models are decision trees, parameterized simulation models, support vector machines, ontology structures, taxonomical constructs, and probabilistic schemas (Hu et al., 2019). Implicit models are pattern-type constructs for which no explicit algorithm is preprogrammed. These are typical in machine and deep learning, where mathematical algorithms are trained to generate models using data and human expert input. They attempt to replicate specific decision processes and replicate decisions that (a team of) experts would make if they could review all available data when the same information is provided. It is expected that implicit models reveal the rationale behind the pattern and decision to help interpret the decision process (Gade et al., 2019). This stimulates research in domain-specific explainable artificial intelligence (Jia et al., 2020).

AI-mechanisms are physical, software, and/or cyberware entities and related computational activities that produce certain problem solving behavior (Ojo et al., 2019). The idea of mechanism is a central part of the concept of smart systems, which may include multiple various nonampliative (NARMs) and ampliative reasoning mechanisms (ARMs). NARMs are as such as (i) classification, (ii) searching/looking up, and (iii) contextualization. ARMs are as such as (i) fusion, (ii) inferring, (iii) reasoning, (iv) abstraction, (v) learning, (vi) decision making, and (vii) adaptation (of knowledge). They can be application-neutral and application-specific. Application-neutral ARMs are needed for intelligent and creative reasoning in intellect-intensive applications, which could not be performed by mathematical logic or (i) inductive, (ii) deductive, (iii) abductive, and (iv) retrospective reasoning, or any combination of them. Historically, five major families of ARMs have been developed, such as (i) symbolic, (ii) analogical, (iii) probabilistic, (iv) evolutionist, and (v) connectionist. Their interoperation is restricted by large differences in representational syntaxes and computational approaches (Varshney et al., 2019). A current issue is designing awareness, inferring/reasoning, and adaptation mechanisms toward an optimal problem solving potential. Artificial neural network architectures are the most widespread representatives of AI-mechanisms, producing nontransparent implicit models (Seidel et al., 2018b). Two main targets for current research are (i) dynamic ANN model training and (ii) dynamic architecture generation (layer and node addition or removal).

AI-knowledge primarily means problem solving knowledge, rather than system development knowledge (Horváth, 2020). The two main constituents of AI-knowledge are (i) codified human knowledge and (ii) self-generated illative knowledge. Accordingly, AI-knowledge is partly generated by knowledge engineering of human knowledge in development time and partly by the AI inferring, reasoning, and learning mechanisms run-time (Feng et al., 2021). Knowledge engineering includes the codification (aggregation, structuring, representation, and validation) of knowledge (AlGhanem et al., 2020). The reasoning mechanisms of a system process codified human knowledge and append it with synthetic knowledge patterns that are not part, but derivatives of codified knowledge. Thus, AI-knowledge can be explicit (explainable) and implicit (unexplainable). Explainable knowledge is conclusions derived based on the results of numerical or symbolic computations by a system, and not explainable knowledge is the abstract patterns constructed by machine learning algorithms (Zatsman, 2020). AI-knowledge is of extreme heterogeneity not only with respect to its cognitive contents but also to its representation, storage, and processing (Kasabov, 2019). A recognized issue is the verification and validation of AI-knowledge in varying application contexts. The machine readability of system knowledge has become one of the recent targets (Baiguess, 2018).

AI-environments are combined hardware, software, and cyberware utilities that allow smart system development, though they themselves are not necessarily smart (Malik and Singh, 2020). Therefore, they are seen as integrated tool complexes and often referred to as integrated development toolboxes, if they are extended with resource selection and recommendation functions. Among others, they manage source codes, component libraries, and standard components. Knowledge ontologies are included or linked to AI-environments. Recently, environments have been specialized to particular AI tasks such as deep learning, image processing or speech generation, and optimized for performance (Hechler et al., 2020). Among many others, the commercialized representatives of AI-environments are (in alphabetical order): (i) AI Platform (Google), (ii) Azure (Microsoft), (iii) Dialogflow (Google), (iv) Holmes AIA (Wipro), (v) MindMeld (Cisco), (vi) Nia (Infosys), (vii) Rainbow (Rainbird Technologies), (viii) Symphony Ayasdi.AI (Ayasdi), (ix) Tensorflow (Google), and (x) Watson Studio (IBM). These environments offer a wide range of cognitive functions that the human mind uses to perform problem solving, learning, reasoning, motion, manipulation, perception, speech, vision, cooperation, and behavior. AI-environments make it easier to move from the ideas through implementation to...
AI-tools have been developed for handling a particular problem with a proper set of computational functions. As Pichai (2018) argued, AI tools have the potential to unlock new realms of scientific research and knowledge in critical domains like biology, chemistry, medicine, and environmental sciences. Thus, the arsenal of AI tools is immense (Bawack et al., 2019). They can be sorted into five general categories: (i) sensing tools (for computer vision, biometrics analysis, health diagnoses, speech recognition, space navigation, etc.) (ii) understanding (for language interpretation, document extraction, automated translation, text analysis, pattern recognition, etc.), (iii) manipulation (logical infering, decision making, process planning, agent development, robot control, collaboration management, contextual recommendation, etc.), (iv) knowledge engineering (for semantic merging, ontologization, semantic merging, automated verification, context processing, etc.), (v) problem solving (for task decomposition, scenario-driven reasoning, artifact classification, solution space exploration, creative composition, etc.), and computational learning (for situation probmodeling, genetic evolution, machine learning, deep learning, natural mimicry, etc.) (Minton, 2017). AI-tools are already used in knowledge-intensive design, complementing cognitive capabilities of designers (Karan and Asadi, 2019). Altavilla and Blanco (2020) posited that using AI-tools can reach the automation level where the tool alone generates or selects the final design outcome and presents the result to the stakeholders at the end of the process.

AI-methods are contemplated as implementation enablers, rather than the principles of operations of smart systems. They provide principles for the effective utilization of artificial intelligence enablers in a context-sensitive manner as well as for the organization of applications. Agré (1997) argued that, in the first 15 years, AI researchers developed a series of technical methods that provide interesting, technically precise accounts of a wide range of human phenomena and cognitive tasks. A characteristic set of methods has been dedicated to algorithmic compositions (Papadopoulos and Wiggins, 1999). AI-methods imply novel design practices. Seidel et al. (2018a) gave an example by investigating a triple-loop model of design activities that is entailed by the application of autonomous AI-tools. Among the road pavers, Haase (1990) discussed that how AI-methods can be used in real-time software products, while Newman et al. (2020) presented a systematic approach to using AI-methods in civilian applications and education. Begler and Gavrilova (2018) reviewed AI-methods for knowledge management systems.

The strength of the interrelationship created by a particular connector determines the applicable design strategy. If the interrelationships among the connectors are zero (or weak), then a traditional sequential design approach can be applied. If the relationships are strong, then a co-design approach is to be applied. The measures of interrelationships vary in every application case.

Part 4: Reflections, conclusions, and future research

Reflections

The concepts or smart design and smart systems are getting more and more attention in research, development, and education. Both of them intends to exploit the recent results of artificial intelligence research, no matter if problem solving methods, intellectualized tools, or application-specific knowledge processing are concerned. Apart from this commodity, they also have an intricate mutual relationship as enablers. On the one hand, tailored to the innovation of smart systems, smart design can change the traditional methodologies of systems engineering and make it more synergistic, efficient, and reliable. On the other hand, smart systems can be used as empowering resources of smart design that not only extend the cognitive space of designing but also offer new affordances. Ultimately, a part of design can be delegated to self-supervised, self-adaptive, smart systems in the near future. The current trends are pointing toward this end, though there are still many known and unknown technological issues and obvious knowledge deficits.

This article has made an attempt to contribute to answering some of the common issues. The intention could not be else by casting light on a possible interpretation and approach, without enforcing strict definitions and other formal specification. Recognizing the fact that research in the contexts of the manifestation and interrelationship of smart design and smart systems is still in an early stage, the content discussed in the paper must be seen as possible ingredients of a conceptual framework. It has been concluded that, toward a comprehensive theory and methodology development, integration and abstraction of the knowledge conveyed by the individual research efforts is needed on a short notice. Furthermore, due to the transdisciplinary nature of the related research phenomenon, experts of the related disciplines (cognitive science, system theory, advance computation, system engineering, artificial intelligence, knowledge engineering, and so forth) need to collaborate and formulate not mono-disciplinary research questions. These are deemed indispensable, but also instrumental, to answering the questions related to their very nature, functionality, implementation, utilization, and impacts of smart designing of smarts systems and employing smart systems in smart design.

Some propositions

What the reader may know or do differently in the light of the above (incomplete) literature analysis and critical systems thinking driven argumentation? Though the lack of undisputable definitions and exhausting explanations is obvious, several propositions can be made. They are as follows:

(a) Smart design is a viable concept in the age of intelligence revolution, since it can provide significant cognitive support to designers to cope with complexities, heterogeneities, compositionality, affordances, and dependability of smart systems.

(b) Smart design should not be seen as axiomatization-based automated design by specialized systems. It is based on an extensive cognitive and creative interaction between humans and systems.

(c) Considering the premature stage of development/evolution, smart design raises many more issues than it can address and solve, but it is rapidly gaining potential as the literature reflects it.

(d) It can be prognosticated without a larger risk that smart design and smart systems will become inseparable in the near future.

(e) Self-supervised (or only partially human-supervised) runtime self-design of smart systems is a realistic idea, though more knowledge is needed about the theoretical fundamentals and practical implementation of self-adaption of complex) is and will be needed.
(f) A theory of smart systems is supposed to provide a unified set of propositions made with the aim of achieving some form of understanding that provides an explanatory power and predictive ability.

(g) Application dependence of smart systems and measuring their performance in various application contexts need further in-depth studies.

(h) It seems that a proper realization of self-adaptation cannot be separated from the assessment of the goals, state monitoring, observing the environment, building awareness, SLR and SLL, and proactive planning.

(i) Self-adaptation plans should be verified before execution and the outcome of adaptation should be validated run-time without suspending system operation.

(j) It is not enough for a smart system to perform smart problem solving operation, but it should also raise the impression that its behavior and the solution are smart.

**Future research opportunities**

Since smart design and smart systems are in an early stage, an extremely large number of phenomena and related issues need to be studied. Figure 11 shows the major research issues in the domain of self-adaptive smart systems. However, research should extend not only to the technological and cognitive domains but also to human, social, business, and visionary domains. Instead of going into technical details concerning manifestations, causalities, implications, dependences, affordances, constraints, etc., we propose the following for consideration.

(1) Systematic blending of the specific bodies and chunks of knowledge (including the foundational theories) available related to smart design, smart systems, and smart exploitation.

(2) Development of strategic roadmaps that consider both the digital transformation and the proliferation of intellectualization with regards to changing human roles and system affordances and that can be used as templates by both the academia and the industry.

(3) Moving toward transdisciplinary research and development activities that synthesize and amplify the knowledge, methods, competences, and experiences of the stakeholders of smart design and smart systems.

(4) Fully fledged manifestation of the concept of computational mechanisms may lead to the situation when intellectualized engineering systems are built from self-adaptive modules of general or application-specific computational mechanisms, rather than from discrete components, fulfilling not only composability but also compositionality specifications and principles.

**References**

Aframović M (2014) Smart products. In Chatti S, Laperrière L, Reinhart G and Tolho T (eds), CIRP Encyclopedia of Production Engineering, 2nd ed. Berlin-Heidelberg: The International Academy for Product Engineering, Springer, pp. 1–5.

Aframović M, Gobel JC and Savarino P (2016) Virtual twins as integrative components of smart products. Proceedings of the IFIP International Conference on Product Lifecycle Management. Cham: Springer, pp. 217–226.

Adams KM (2012) Systems theory: a formal construct for understanding systems. International Journal of System of Systems Engineering 3, 209–224.

Adams KM, Hester PT, Bradley JM, Meyers TJ and Keating CB (2014) Systems theory as the foundation for understanding systems. Systems Engineering 17, 112–123.

Agre PE (1997) Toward a critical technical practice: lessons learned in trying to reform AI. In Agre PE, Bowker G, Gasser L, Star L and Turner B (eds), Bridging the Great Divide: Social Science, Technical Systems, and Cooperative Work. Hillsdale, NJ: Lawrence Erlbaum Associates, Inc., pp. 1–17.

Akbar A, Khan A, Carrer F and Moessner K (2017) Predictive analytics for complex IoT data streams. IEEE Internet of Things Journal 4, 1571–1582.

Akinsolu MO, Danjuma IM, Mistry KK, Liu B, Abd-Alhameed RA, Lazaridis PI and Excell P (2019) Efficient AI-driven design of microwave antennas using PSADEA. Proceedings of the 2nd IEEE Middle East and North Africa Communications Conference. IEEE, Manama, Bahrain, pp. 1–5.

Alegre U, Augusto JC and Clark T (2016) Engineering context-aware systems and applications: a survey. Journal of Systems and Software 117, 55–83.

Alghanem H, Shanaa M, Salloum S and Shaalan K (2020) The role of KM in enhancing AI algorithms and systems. Advances in Science, Technology and Engineering Systems Journal 5, 388–396.

Alippi C and Ozawa S (2019) Computational intelligence in the time of cyber-physical systems and the Internet of Things. In Kozma R, Alippi C, Choe Y and Morabito FC (eds), Artificial Intelligence in the Age of Neural Networks and Brain Computing, Cambridge, MA: Academic Press, pp. 245–263.
Malikarjuna BK, Chronopoulos C and Kozin I (2020) The concept of smartness in cyber-physical systems and connection to urban environment. *Annual Reviews in Control*. doi:10.1016/j.arcontrol.2020.10.0091367-5788/

Mattern S (2018) Calculative composition: the ethics of automating design. In Dubber MD, Pasquale F and Das S (eds), *The Oxford Handbook of Ethics of AI*. Oxford: Oxford University Press, pp. 1–20.

Matthews ML, Bryant DJ, Webb RD and Harbluk JL (2001) Model for situation awareness and driving: application to analysis and research for intelligent transportation systems. *Transportation Research Record 1779*, 26–32.

Mayer S, Tschofen A, Dey AK and Mattern F (2020) Autonomous tools and design: a triple-loop approach to human-machine learning. *Communications of the ACM 62*, 50–57.

Mühlhäuser M (2008) Smart products: an introduction. In Mühlhäuser M, Ferscha A and Aitenbichler E (eds), *Communications in Computer and Information Science, Constructing Ambient Intelligence*. Berlin, Heidelberg: Springer, pp. 158–164.

Neuhüttler J, Woyke IC and Ganz W (2017) Applying value proposition design for developing smart service business models in manufacturing firms. *Proceedings of the International Conference on Software Architecture Companion*. IEEE, pp. 242–245.

Mucha T and Seppala T (2020) Artificial Intelligence Platforms – A New Research Agenda for Digital Platform Economy. *Research Report No. 76, The Research Institute of the Finnish Economy*, p. 141.

Mühlhäuser M, Ferscha A and Aitenbichler E (eds), *The Research Institute of the Finnish Economy, p. 141.*

Oh S, Jung Y, Kim S, Lee I and Kang N (2019) Deep generative design: integration of topology optimization and generative models. *Journal of Mechanical Design 141*.

Ojo A, Mellouli S and Ahmadi Zeleti F (2019) A realist perspective on AI-era public management. *Proceedings of the 20th Annual International Conference on Digital Government Research*, pp. 159–170.

Papadopoulos G and Wiggins G (1999) AI methods for algorithmic composition: a survey, a critical view and future prospects. *Proceedings of the AISB Symposium on Musical Creativity*, Vol. 124, Edinburgh, UK, pp. 110–117.

Parasuraman R, Sheridan TB and Wickens CD (2000) *Human in the Loop: ... Cybernetics*. Berlin, Heidelberg: Springer-Verlag, pp. 158–164.

Pessoa MV and Becker JM (2020) Smart design engineering: a literature review of the impact of the 4th industrial revolution on product design and development. *Research in Engineering Design*, 31, 1–21.

Picchi S (2018) AI at Google: our principles. The *Keywords 7*, 1–3.

Quanz B, Sun W, Deshpande A, Shah D and Park JE (2020) Machine learning based co-creative design framework. arXiv preprint arXiv:2001.08791.

Rajase T and Jafari H (2020) Two decades on the artificial intelligence models advancement for modeling river sediment concentration: state-of-the-art. *Journal of Hydrology 588*, 1–13.

Reed SK and Pease A (2017) Reasoning from imperfect knowledge. *Cognitive Systems Research 41*, 56–72.

Ren X and Chen Y (2019) How can artificial intelligence help with space missions – a case study: computational intelligence-assisted design of space tether for payload orbital transfer under uncertainties. *IEEE Access 7*, 161449–161458.
Subramanyam S and Lu SC (1991) The impact of an AI-based design environment for simultaneous engineering on process planning. International Journal of Computer Integrated Manufacturing 4, 71–82.

Sun W, Ma QY, Gao TY and Chen S (2010) Knowledge-intensive support for product design with an ontology-based approach. The International Journal of Advanced Manufacturing Technology 48, 421–434.

Tang MX (1997) A knowledge-based architecture for intelligent design support. The Knowledge Engineering Review 12, 387–406.

Tao F, Sui F, Liu A, Qi Q, Zhang M, Song B and Nee AY (2019) Digital twin-driven product design framework. International Journal of Production Research 57, 3935–3953.

Tavčar J and Horváth I (2018) A review of the principles of designing smart cyber-physical systems for run-time adaptation: learned lessons and open issues. IEEE Transactions on Systems, Man, and Cybernetics: Systems 49, 145–158.

Torres E and Penman W (2020) An emerging AI mainstream: deepening our comparisons of AI frameworks through rhetorical analysis. AI & Society, 1–12 (Published online: 22 October 2020).

Trokhimchuk PP (2017) Theories of open systems: realities and perspectives. International Journal of Innovative Science and Research Technology 2, 51–60.

Tsukuda M, Arimoto K, Asakura M, Hidaka H and Fujishima K (1993) A smart design methodology with distributed extra gate-arrays for advanced ULSI memories. IEEE Transactions on Electronics 76, 1589–1594.

Ulsoy AG (2019) Smart product design for automotive systems. Frontiers of Mechanical Engineering 14, 102–112.

Varshney LR, Keskar NS and Socher R (2019) Pretrained AI models: performativity, mobility, and change. arXiv preprint arXiv:1909.03290.

Venkataramani S, Sun X, Wang N, Chen CY, Choi J, Kang M and Gopalakrishnan K (2020) Efficient AI system design with cross-layer approximate computing. Proceedings of the IEEE 108, 2232–2250.

Vroom RW and Horváth I (2014) Cyber-physical augmentation: an exploration. Proceedings of the 10th International Tools and Methods of Competitive Engineering Symposium.

Wang L (2019) From intelligence science to intelligent manufacturing. Engineering 5, 615–618.

Wang J, Ma Y, Zhang L, Gao RX and Wu D (2018) Deep learning for smart manufacturing: methods and applications. Journal of Manufacturing Systems 48, 144–156.

Wens D (2019) Software engineering of self-adaptive systems. In Cha S, Taylor RN and Kang K (eds), Handbook of Software Engineering. Cham: Springer, pp. 399–443.

Woods DD and Roth EM (1988) Cognitive engineering: human problem solving with tools. Human Factors 30, 415–430.

Wu Y, Li G, Wang L, Ma Y, Kolodziej J and Khan SU (2012) A review of data intensive computing. Proceedings of the 12th International Conference on Scalable Computing and Communications, Changzhou, China, pp. 1–6.

Wuest T, Schmidt T, Wei W and Romero D (2018) Towards (pro-)active intelligent products. International Journal of Product Lifecycle Management 11, 154–158.

Yamanobe N, Wan W, Ramirez-Alpizar IG, Petit D, Tsuji T, Akizuki S, Hashimoto M, Nagata K and Harada K (2017) A brief review of affordance in robotic manipulation research. Advanced Robotics 31, 1086–1101.

Yang X, Moore PR, Wong CB, Pu JS and Chong SK (2007) Product lifecycle information acquisition and management for consumer products. Industrial Management & Data Systems 107, 936–953.

Yang Q, Scuito A, Zimmerman J, Forlizzi J and Steinfeld A (2018) Investigating how experienced UX designers effectively work with machine learning. Proceedings of the Conference on Designing Interactive Systems, Hong Kong, China. Denver, CO, USA: ACM Press, pp. 585–596.

Zatsman I (2020) Three-dimensional encoding of emerging meanings in AI-systems. Proceedings of the 21st European Conference on Knowledge Management, pp. 44–42.

Zha XF, Sriram RD and Lu WF (2003) Knowledge intensive collaborative decision support for design process. Proceedings of the International Design Engineering Technical Conferences, Vol. 37009, pp. 425–438.

Zheng P, Wang Z and Chen CH (2019) Industrial smart product-service systems solution design via hybrid concerns. Procedia CIRP 83, 187–192.

Zhou L, Pung HK, Ngho LH and Gu T (2006) Ontology modeling of a dynamic protocol stack. Proceedings of the Conference on Local Computer Networks. IEEE, pp. 353–360.

Zysman J and Nitzberg M (2020) Governing AI: understanding the limits, possibility, and risks of AI in an era of intelligent tools and systems. BRIE Working Paper # 2020-5, pp. 1–28.

Dr. Imre Horváth is an emeritus professor of the Delft University of Technology, where he focused on cognitive engineering of smart cyber-physical systems. He promoted more than 20 PhD students and has more than 430 publications. He is a fellow of ASME and a member of the Royal Dutch Institute of Engineers. He received two honorary doctor titles, the ASME lifetime achievement award, and the Pahl-Betz ICONN award. He has served several international journals as an editor. He is an initiator of the International Tools and Methods of Competitive Engineering (TMCE) Symposia. His current research interest is knowledge science of intellectualized systems.