TASK QUARTERLY vol. 25, No 3, 2021, pp. 355–373

# THE ROLE OF KNOWLEDGE IN CYBER-PHYSICAL SYSTEMS OF SYSTEMS

RICARDO SANZ, JULITA BERMEJO, MANUEL RODRÍGUEZ AND ESTHER AGUADO

*The Autonomous Systems Laboratory*
*Polytechnic University of Madrid*
*28040 Madrid, Spain*

(received: 27 June 2021; revised: 21 July 2021; accepted: 14 August 2021; published online: 30 November 2021)

**Abstract:** Characterinsing the nature of cybephysical systems is not easy task. What are core aspects and what are not? This is especially tricky in systems-of-systems aggregates. Some EU-funded cyberphysical systems projects have performed a roadmapping exercise over the domain of Cyber-Physical Systems-of-Systems. In particlular, the EU-CPSoS project roadmap has identified t hree m ajor c hallenges a nd e leven r esearch a nd i nnovation p olicies t hat shall be addressed to solve the three challenges. The third core challenge addresses Cognitive Cyber-physical Systems of Systems. In this article we address the role that knowledge and cognition are to play in future cyber-physical systems of systems from a life-cycle perspective of high autonomy systems.

**Keywords:** Cyber-Physical Systems, Systems-of-Systems, Knowledge, Integration, Engineering, Autonomy

**DOI:** https://doi.org/10.34808/tq2021/25.3/e

## 1. Introduction

This article focuses on the role of knowledge in Cyber-Physical Systems-of-Systems (CPSoS)[1] construction and operation. Knowledge is widely recognised a valuable asset for performant people and organisations [1]. In fact, it has recently gained the status of critical asset for the competitive success for social groups, enterprises and countries. As we will see, it is even more critical for the success of autonomous systems —esp. in CPSoS— where their adequate operation, both as constituent systems and as whole systems-of-systems, depends on it.

1. CPSoS are Systems-of-Systems (SoS) where their constituent systems are cyberphysical systems (CPS).

It is commonly thought that knowledge is an exclusive affair of the human mind. However, as systems have grown in complexity, intelligence and autonomy, the knowledge they exercise to drive their activities becomes the core critical resource that can sustain long-term, adaptive and resilient autonomous operation [2]. System-embedded knowledge is the main enabler for SoS persistence and adaptivity, enabling meaningfully system-wide integration and interoperation [3]. In essence, knowledge constitutes a critical component in future CPSoS.

The EU-funded CPSOS project did identify three core long-term research challenges —management of CPSoS; engineering for the design-operation continuum; and cognition in CPSoS— that have strongly knowledge-dependent activities.

In this article we will analyse the role that knowledge plays in addressing these three challenges. Section 2 will clarify the interpretation of the terms CPS, SoS and CPSoS as used in this article. Section 3 will address the concept of knowledge in the intelligent autonomous systems engineering domain and its meaning in the scope of the EU-CPSOS challenges. Section 4 will introduce the role of knowledge as a sustaining element of the whole SoS life-cycle, as well as defining the knowledge and needs in this context. Section 5 focuses on the role played by knowledge within the four key subtopics defined in the third challenge of the EU-CPSOS roadmap — cognition in CPSoS. Finally, Section 6 points out some concluding remarks.

## 2. CPS, SoS and CPSoS

Some of the roadmapping projects funded by the EC in the system-of-systems domain, addressed the convergence points between cyber-physical systems (CPS) and systems-of-systems (SoS). These Cyber-Physical Systems of Systems (CPSoS) are complex systems that have two fundamental aspects that are intrinsically related:

- physical elements interact with and are controlled by a large number of distributed and networked computing elements and human users (the CPS aspect) and
- component subsystems may have independent purposes, authorities and life cycles (the SoS aspect).

The term Cyber-Physical System (CPS) is a modern fad to refer to what in the past were called distributed real-time embedded systems (DRES). Apparently, the CPS concept adds the idea of networked computing systems having a physical aspect. In this vein, The US NIST defines Cyber-Physical Systems as “smart” systems that are co-engineered interacting networks of physical and computational components. This physical aspect was also part of the DRES domain, where the term “embedded” implied the existence of a larger reality where the computers were situated. However, neither DRES research in the past, nor CPS research today have paid sufficient attention to the physical part of systems, being mostly restricted to the computation and communication problems (see for example the

CPS concept map available at cyberphysicalsystems.org). The larger picture is better captured, however, in the domain of distributed control systems (DCS), that fully adress the issues realted to physical dynamics, real-time computation and networking.

In contrast to the weak CPS term, the term ‘SoS’ aptly conveys a specific, concrete meaning not captured in other endeavours of systems engineering. Systems-of-systems [4] are aggregations of component systems (CSs) that provide some functionality at the system level. One central aspect of SoS is the relative idependence of component systems (they have their own, separate life-cycles). Maier [5] and the US DoD [6] identified four classes of systems considering their independent life-cycles and structures of control (see Table 1).

**Table 1.** Types of Systems of Systems identified by Maier and the US DoD

| | |
|---|---|
| Virtual | Lack both an agreed-upon-purpose and a central management authority for the SoS. Large-scale behaviour emerges – and maybe desirable or not – but this type of SoS must rely on implicit control to keep it working |
| Collaborative | The component systems accept interaction to fulfil agreed-upon-purposes. The commonly used example is the Internet. The IETF sets out standards but has no real power to enforce them |
| Acknowledged | Have recognized objectives as SoS, dedicated management, and resources. However, constituent systems retain their independent ownership, objectives, funding, development, and sustainment approaches. Changes in the systems are based on good-will collaboration |
| Directed | The SoS is built and managed to fulfil specific purposes. It is centrally managed during long-term operation to fulfil those purposes. Component systems may keep the capacity of operating independently, but their normal operational mode is subordinated to the SoS purpose |

It seems that the term “System of Systems” (SoS) —used since the 1950s— describes systems as they are (i.e. composed of independent constituent systems, that act jointly towards a common goal) but it essentially captures not what the system is but how and why it came to be and is used. In a very precise sense, it is an epistemological term related to the life-cycles of the systems involved —as seen by their builders and users. It is not an ontological term concerning the co-existence of the CSs and the SoS. As Leveson [7] says, “almost all systems are made up of existing subsystems”. Leveson indeed manifests herself against the use of the term SoS because it may lead to weaking safety analyses:

*“Safety is a system property. It must always be analysed top-down and for the system as a whole. When putting two or more existing components (‘systems’) together, the emergent properties must be analysed for the integrated system. Calling that larger system a ‘system of systems’ may be misleading by implying that emergent properties can be treated differently than any other system or different system engineering techniques can be used.”*

Emergence is key in SoS, but no more than in any other kind of system. Systems are always built in search of their emergent properties. The deep meaning of the SoS concept shall not be found in terms of what the systems are in mereological terms but in terms of the relations of their structures and functions with the engineering goals of the CSs and the SoS. The meaning of SoS shall be found in the higher connections of the systems engineering Vee models of both CSs and SoS [8]: the connection between requirements and the provision of value to system stakeholders.

We cannot forget that real-world CPSoS will always have humans involved (users, engineers, operators, etc.). The human-system relation will be always shaped by what the human knows about the system and also by what the system knows about the human. This mutual understanding shall necessarily go much deeper than what the mere HMI provides. When humans act as system components —i.e. humans acting as CSs—, there is a critical functional dependence that relies on the operational semantics behind the interface.

Semantic interoperability is necessary for both CPS and SoS as a cornerstone for their integration [9] and system knowledge is what sustains it. Knowledge is the critical asset that deeply glues all CSs together -cyber, physical and human.

## 3. On Knowledge and Cyber-Technology

In a sense, knowledge is a quite overloaded word. Being a core topic of philosophy —the whole discipline of epistemology orbits around it— knowledge has also become the central issue in artificial cognitive systems.

### *3.1. What is Knowledge?*

A good, old fashioned philosopher would say that knowledge is justified true belief (even if considering the criticisms that this definition has received). In the domain of cognitive psychology, knowledge is seen as cognitive agent mental content. Newell defined knowledge as “Whatever can be ascribed to an agent, such that its behaviour can be computed according to the principle of rationality” [10].

In the cognitive CPSoS domain, the main interpretations of the concept come from artificial intelligence, where a fully pragmatic position towards knowledge is taken. In the domain of knowledge engineering, it is thought that knowledge must have a functional value for a program. For example, Schreiber at al. defined knowledge as the “Whole body of data and information that people bring to bear to practical use in action, in order to carry out tasks and create new information” [11].

From a cyber-physical, cognitive systems perspective, we consider that the nature of knowledge stands in the relation of the mental content of an epistemic agent with a part of the universe it relates to. Thus, knowledge is a model that an entity has of some other entity [12]. Knowledege is models and models are knowledge[2]. The main value of knowledge sits in its explanatory and predictive capabilities, that are helpful to improve agent behaviour. In this sense, Halladay and Milligan defined knowledge as "Conceptual models of systems and principles [that explain] functioning, causes and effects, form, features and may have a predictive nature. [...] As with any model, the more closely the model correlates to its target, the more capable the model is of explaining and projecting the behaviors of the model's target" [13]. Cognitive behaviour is model-driven behaviour.

These models/knowledge can be created from first principles, analysis and specification —as when we model the physics of a reactor or extract human knowledge using expert systems technology — or can be learnt from data. The dichotomy designed vs. learnt has shaped the whole field of artificial intelligence since its beginnings with strong defenders and harsh attacks from both sides. As it happens with most systems engineering decisions, there are trade-offs to make when choosing an specific approach to solve a system problem. Design-based approaches are usual when there is strong certainty about the problem and its solution and/or a need for system assurance. Learning-based approaches are preferred for uncertainty-laden tasks but they suffer from lack of explanability and non-statistical guarantees concerning the reliability of their learnt model predictions.

### *3.2. Technological Knowledge*

Engineering, Systems engineering, CPS engineering and SoS engineering are all knowledge-intensive activities. Engineering is both a knowledge-consuming and a knowledge-generating activity [14]. The two main sources of knowledge for engineering are general science and previous experience (i.e. engineering itself). The production of engineering knowledge has indeed a virtuous circle effect, where previous success stories can be leveraged in the design and construction of new systems (see for example the productivity effect of the software patterns movement [15]).

Engineering as activity produces two types of main products: systems and knowledge. This knowledge may refer to the systems themselves but also to the engineering processes used to build such systems [16]. From the "knowledge is models" perspective this means that engineering produces "models of systems" and "models of processes to build systems". Models, considered as system knowledge, are of particular importance for engineering.

The last years global drive towards model-driven engineering or model-based systems engineering (MDE, MBSE) is just the recognition of this dual model fact: that engineering activities shall be organised following "models of processes to build systems using models". The SysML profile for UML is an example of this

2. Note that knowledge can be internal to the agent but it can also be externalised. In this sense, we may wonder to what extent blackbox models can be considered knowledge. They have predictive capabilities but they cannot be externalised nor used to explain causes and effects. Maybe we shall restrict this coception of knowledge to white box models.

recognition, e.g. incorporating mechanisms for modelling requirements to expand the life-cycle coverage of models.

Systems' models typically use domain-specific modelling languages (DSML) and models of processes have been usually captured in text (e.g. [8]), but more formal languages have also been used in different domains [17]. These models of engineering processes can be found in all classes of disciplines, not only in software systems construction where they are a common trade. For example, the PISTEP Process Plant Engineering Activity Model [18] is a class of model that captures the life-cycle of the engineering process to build a continuous process plant (e.g. a refinery)[3]. In the context of CPS modelling and simulation, the TAMS4CPS project [19] separates system's models in two big categories: descriptive models, where the system is represented in abstracted form for purposes of communication; and experimental models, that are used for enquiry, to conduct experiments about the system.

It is worth noting that regardless of the kind of model or the language the model is implemented in, models are useless unless the agent owns cognitive mechanisms to exploit them. Models and model exercisers are the two essential components of a cognitive agent. As Merrill suggests, "Cognitive psychology suggests that a mental model consists of two major components: knowledge structures (schema) and processes for using this knowledge (mental operations)." [20]. Both aspects of models —descriptive and experimental— shall cohere in the domains of CPSoS: there will be multiple stakeholders —users, engineers, owners, operators, artificial intelligences, etc.— that shall exercise the models that they share by communication. Model coherence — the fact that different aspects/constituents/views of a model (like descriptive/experimental) must be coherent— is thus necessary to produce cognitive agreement in the actions and perceptions of the system as a whole.

Figure 1 later, will summarily depict the different forms that this knowledge (and its associated exercisers) takes in engineering processes and CPSoS life-cycles. It is not surprising to see that this role analysis matches those done in the domain of the philosophy of technology. Mitcham [21] points out that technology can be approached from four basic perspectives: i) as a certain type of objects (artefacts), ii) as a specific class of knowledge (technological knowledge), iii) as a set of activities (producing and using artefacts) and, iv) as manifestation of a determined human will in relation to the world (technology as volition) [21].

This last aspect mentioned by Mitcham —volition— is of special importance in the case of CPSoS because it is human will —i.e. as captured in stakeholder's requirements— what specifically characterises the SoS aspect of these systems[4] —at both the CSs and the SoS levels. All these aspects of technology match

3. See also the standardised Plant Life-cycle Activity Model (ISO 15926 - the Lingua Franca of global interoperability).

4. And of any (designed) system in fact.

specific processes, assets and roles in the CPSoS life-cycle (See Figure 1). In particular the capture of human intentions at runtime —in the form of exercisable and verifiable requirement models— will be a cornerstone of solid CPSoS.

### *3.3. Knowledge and SoS Emergence*

An important issue in CPSoS engineering is emergence [22]. Emergence is a complex, somewhat philosophical concept, that has several interpretations around a central common understanding of "appearing as a result of interactions of subsystems". This appearance can be done purposefully, because this "systemic emergence" is frequently the raison d'étre for the existence of such systems. Another common use of emergent behaviour of a system is behaviour that arises out of the interactions between parts of a system and which cannot easily be predicted or extrapolated from the behavior of those individual parts. Obviously, unexpectedness may be a major problem for CPSoS if the emergent system behaviour happens to be detrimental [23]; but it is obviously not necessarily so, because systems are built because of what they emergently provide.

Behavioural emergence is just an epistemological issue, not an ontological one; it is strictly related to what is known about the system, i.e. with the available system models and exercisers. From the perspective of the CPSoS, 'emergent' means 'systemic', i.e. phenomena at the level of the higher SoS. It is only from the perspective of the knowledge about the system —as components, as a whole— that the term 'emergent' means 'unexpected' or 'unpredictable'. It is the model and the model exerciser what fails in this situation; hence the unexpectedness. Emergence is the beneficial and/or detrimental[5] effect of both a) the lack of knowledge at the CS and SoS levels, b) an insufficient knowledge flow, or c) the lack of cognitive capability —of model exercisers— to exploit the knowledge when available.

Scientific engineering strives for better knowledge of the CPSoS because "perfect" knowledge would entail the possibility of elimination of unexpected, detrimental emergence[6]. However, models may be useful even when they are not perfect. Tolerating modelling mismatches —uncertainty, vagueness, partiality, locality, etc— are critical capabilities for cognitive CPSoS. Conventional MBSE is not good at tolerating vagueness and uncertainty in its models; addressing them at the model level is a necessary capability for robust CPSoS construction.

---

5. Note that in complex systems there are always multiple goals that may de differently affected by emergence.

6. In fact, we can think that no model can indeed be perfect as we cannot have complete (infinite) knowledge; and also because, if perfect, models (and thus knowledge) would be the reality itself. Perfection, however, may be achievable on two gounds: i) when targeting a higher abstraction level without entering low level physical details; and ii) when the system modelled is in informational system and hence a model can be truly isomorphic to it.

# 4. Knowledge in Cyber-Physical Systems-of-systems

## *4.1. Knowledge in the CPSoS life-cycle*

The system life-cycle spans from the identification of a need for some specific stakeholders to system decommission and life-cycle closure. In this life-cycle a major event is the transition from the system engineering phase to system operation phase. However, in many situations the deployment of the system does not end the engineering activities. Examples abound: long-lived systems, adaptive systems, systems of systems, etc. The existence of a design-operation continuum —as described by the EU-CPSOS roadmap [24]— is the normal situation in CPSoS.

Wide-spectrum knowledge —multifacetted knowledge that addresses both the systems' and the systems engineering processes' aspects— becomes the critical asset that enables both system construction and system operation. In the case of CPSs, this knowledge shall necessarily include aspects of the physical side of the system. This has been partially addressed in the domains of embedded systems (that model the computing and networking platform), control systems (that model the plant under control) and user interfaces (that model the human behind the interface). In the case of SoS there are both models at the CS level and at the SoS level. However, the constituent system knowledge is usually available at the system-of-system level only as agreed-upon interfaces. This is in general not enough for certain classes of analyses at the SoS level; deeper behavioural knowledge about the CSs is needed.

There is also a need to extend the modelling of teleological aspects —the ConOps, the requirements— from the engineering to the operation stages. Teleological knowledge shall flow from the CS level to the SoS level to be able to address SoS goals without sacrificing CS goals (e.g. in acknowledged SoS). For SoS, there are stakeholders for both the SoS and for the CSs themselves. The stakeholders of the SoS may have limited knowledge of the constraints, development activities, and capabilities of the CSs. This will depend on the profile of the stakeholders and the type of the SoS (see Table 1).

The main depositaries of system knowledge are on one side systems and speciality engineers, and, on the other side, system participants. The main barrier for system knowledge integration and flow are the disciplinary profiles and cognitive capabilities of all these different stakeholders[25]. In autonomous CPSoS the system runtime artefacts are a third class of cognitive agents —besides builders and participants— that exploit knowledge about the system. Systems engineers of SoS find that they need to focus on those areas that are critical to the SoS success, usually leaving CS-level issues to their respective systems or domain engineers. The systems engineers at the CS level own the necessary expertise, the cognitive capability and the responsibility to exert the knowledge because they are in the best position —close to CS implementation details.

The heterogeneity of these classes of knowledge is only apparent and related to the model exercisers more that to the system itself. The integrated models

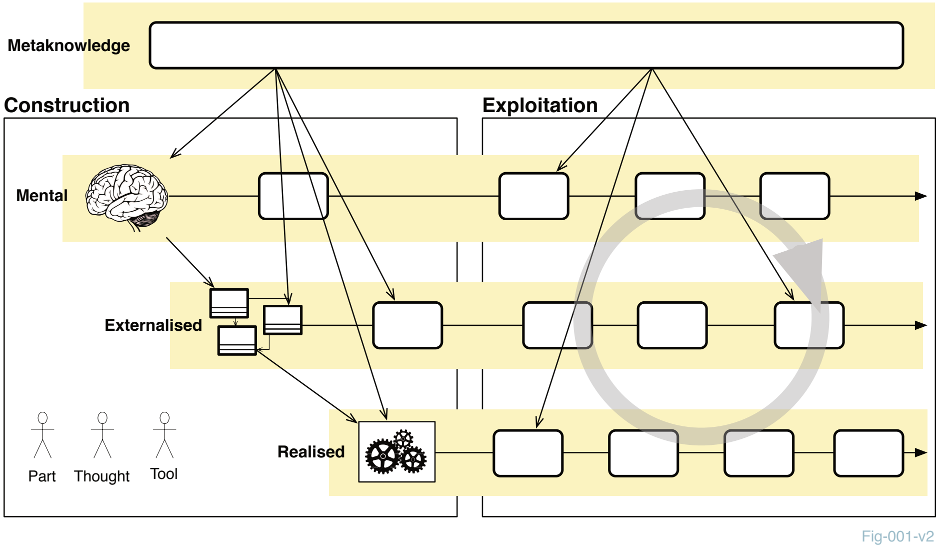


**Figure 1.** The CPSoS life-cycle knowledge entities and relations. Knowledge intensive approaches to CPSoS construction and operation recognise the vertebrating and dynamising roles that this knowledge has. It is knowledge —in human heads, in externalised models, realised in entities— what underlies the design-operation continuum that enables the runtime adaptation of the cyber-physical system and its seamless incorporation into systems-of-systems. Unified metaknowledge bridges the construction-exploitation gap, enabling runtime knowledge loops that sustain system adaptation. The system includes humans that manipulate models -thought-, transform models -tool- and physically act -part

postulated by the MBSE doctrine point in this direction. Note however that we must change the focus of the model from the CPSoS as such, to the larger view of modelling the CPSoS + its lifecycle. MBSE models become "live" models in the sense that they do not represent an static CPSoS but also its evolution, so an initial static model would not be sufficient; it is needed to have a model that adapts to how the CPSoS evolve and to the new knowledge that can be acquired during its lifetime.

Vincenti [26] proposed a six part taxonomy of the knowledge that aeronautical engineers did use: 1) fundamental system design concepts (e.g. what are the components of an airplane), 2) design criteria and specifications (e.g. limits on temperature in the fuel tank), 3) theoretical tools (e.g. Navier-Stokes equations for fluid dynamics calculations), 4) quantitative data (e.g. drag coefficients of a wing), 5) practical considerations (e.g. doing tradeoffs between cost and quality) and 6) design instrumentalities (e.g. knowing how to trace the root requirements for an engine). All these are aspects of engineering knowledge that will appear in the life-cycle perspective succinctly shown in Figure 4.1.

The knowledge that sustains this process models both the CPSoS and the goals, requirements and methods that directed its construction. System configuration management includes also intentional items that are necessary to properly manage the SoS at runtime. "Once rolled out, operating and maintaining a system of systems requires a good knowledge of the 'as-deployed-and-configured' system's physical, functional and behavioural configuration"[27]. Here the aviation industry has great experience, but needs to be extended in relation to cognitive aspects[7].

The CPS+SoS focus of the EU-CPSOS project implies that the deep models of the systems (computation+physics) shall be shared across constituent system boundaries. A particular case of this situation is when humans play a role as components of a larger system. Obviously humans should have life-cycles that are independent of the SoS life-cycle. Human-in-the-loop CPS (HiLCPS) or Socio-cyber-physical systems (SCPS) are terms used to refer to these classes of SoS. In the general case, there will be a collection of heterogeneous knowledge-based agents that share knowledge about themselves —as components and as CSs— to produce a synergetic effect at the SoS level. The needed transition is a move from information-based integration to knowledge-based integration. CSs shall interchange not just information about their state —e.g. through CORBA or REST interfaces— but deep knowledge —i.e. models— about themselves. For example, [28] shows a prototype collaborative maintenance planning system for a machine tool -a core component of Industry 4.0 environments. This system links machine provider and machine exploiter knowledge management systems, including operating machine tools —constituting a CPSoS. This knowledge flow provides advantages over the integration of traditional engineering information systems —e.g. CAD, PLM and ERP[8]— in managing machine tool maintenance and service information including dynamic and unstructured knowledge.

To achieve this objective, however, we need a common mental infrastructure in all the CSs of the CPSoS. Agents shall share a common ontology to be able to exercise the models they interchange. This could become a daunting task, because in many cases such referential framework is implicit (e.g. in relation to the knowledge of physical aspects in a CPS system). Partial knowledge needs context to be properly interpreted. In this vein, Hayes and Walsham [29] address the problem of knowledge management from a dual content/relational perspective. The content perspective —the vision of knowledge as representation of facts— implies both clear semantics and an interpretational context, hence simplifying the problem of knowledge codification and retrieval. On the opposite side, the relational perspective highlights that the contextual and relational aspects of knowledge pose enormous problems for interpretation unless the factual relations and the interpretational context are stored and retrieved with it. This amounts to enormous difficulties for knowledge sharing and exercising in scopes differing from where it was created. This implies a serious problem of SoS, where interpretation agents may not have a shared architecture and ontology, and where

7. Artificial intelligence AI has found no comfortable accommodation in aerospace due to the inherent difficulties in certifying AI behaviour.

8. Computer aided engineering, product data and lifecycle management, and enterprise resource planning systems.

their evolutions follow the paths set by the peculiar evolution of the requirements of each CS and not the SoS at large.

### *4.2. Knowledge Needs in CPSoS*

Models at different levels of abstraction will help to formalize the CPSoSs requirements, allowing their traceability from design to implementation [30]. Model-based design or development is a key enabler to cope with the complexity of CPSs, as it allows both to early validate requirements and to detect integration issues based on the models of the subsystems and the defined system architecture [31], [32]. Ontologies and domain-specific languages are useful mechanisms to obtain machine-processable models of relevant application domains. Ontologies are shared specifications of a conceptualisation. They could be used to obtain CPS domain models, as they define the fundamental concepts and their relationships in such particular domain. Domain-specific languages, based on ontologies, would provide a consistent grammar to specify the CPS [33], [34]. The underlying idea is to obtain a consistent and formal description of CPS, easing the interaction among different components, subsystems or domains by sharing a common terminology and description.

A model of a CPS constituent system will include models of the different elements belonging to the CPS, such as the physical processes, the software, the computation platforms, the networks (and in last instance of the humans interacting with it). Modelling these systems is challenging as a multi- and interdisciplinary approach to consider their inherent heterogeneity will be needed [35]. Creating models of the domain, the participants, the objectives, the requirements, the available services as well as the tasks will allow addressing the adaptive behaviour required for these systems (esp. when integrated into a SoS).

To obtain these models, it would be needed some kind of semantic foundations to integrate different heterogeneous models and modelling languages. This is specially important when merging CSs into heterogeneous SoS. The lack of a common definition and languages to describe large and complex CPSoS makes it difficult to deal with their heterogeneity. Tasks such as defining an ontology of model types, developing a CPS model paradigm to construct CPS reference models will result in obtaining CPS modelling ontologies or a standard set of modelling practices to obtain model-designed CPSs [36].

Due to the inherent heterogeneity in a SoS, its modelling process would require an interdisciplinary approach to consider the different components in the CSs (computational processes, networking, and human actors). Multi-domain modelling, understood as the necessity to establish a body of knowledge to model all relevant features of CPS, would be a key element to engineering CPS [37]. In practice, abstract models about data and knowledge need to be combined with those from the physical elements in the CPS, and the human stakeholders interacting with the system. For the latter, new disciplines such as cognitive psychology and sociology will be needed to develop models of human perception, interaction, knowledge, thought processes and problem solving [38]. At the end the model has to have three dimensions: i) Vertical: Top - down or abstract to specific; ii) Horizontal: Multi-domain aspects; and iii) Circular: Life-cycle (from requirements to decomission). All the three need to have a common —or at least integrated and coherent— representation (core model) that guarantees model consistency and seamless flow of knowledge.

As CPS work on the physical world, they are subject to environment changes and unexpected conditions. However, they are asked to be robust and react to system failures. A possible approach is to design at each level within the overall system, components predictable and reliable, as long as it is possible. There is a need to address how small changes do not dramatically change the CPS expected outcome. Hence, aspects such as uncertainty at the cyber level, uncertain data or reconfigurations should be considered as part of CPS design [39]. However, current methods to characterise and evaluate uncertainty during the design and development phases are limited and inadequate [40].

### 4.3. Knowledge Flow in CPSoS

Cyber-physical systems-of-systems are characterised by the flow of matter, energy and information. At the cyber layer, we must make a distinction between flow of data vs. flow of knowledge[41]. Flow of data is what we have today in our CPSs and it is not enough[9] for the long term objectives identified by EU-CPSOS. Within a robust, adaptive CPSoS there is a need of integration at the knowledge level [10] to be able to attain SoS and CSs objectives dynamically.

In knowledge-based cyber-physical systems there are three core processes that deal with knowledge: perception, the generation of knowledge from sensor data; manipulation, the transformation of knowledge; and action, the generation of actuator data from models. In essence the instantiation and federation of these processes at the CS and SoS levels is what defines the DoD taxonomy of SoS (See Table 2).

**Table 2.** Knowledge and control flows in Systems of Systems

| | |
|---|---|
| Virtual | Virtual SoS do not have knowledge flows |
| Collaborative | In collaborative SoS, there are flows of knowledge but there are no action mechanisms at the SoS level |
| Acknowledged | Acknowledged SoS have flows of knowledge and heterarchical action mechanisms at the SoS and CSs level |
| Directed | Directed SoS have flows of knowledge and hierarchical action mechanisms at the SoS and CSs level |

We must acknowledge the need of making these flows reflect the needs coming from the merging of engineering and operation workflows. In the design-operation continuum, the flow of models between all cognitive entities in the CPSoS —both humans and AIs— will enable the realization of the necessary advances.

9. See for example that the DoD uses the term net-centricity to refer to flow of data [6].

Note that the questions of property, authority and control that specifically identify SoS aspects will be also part of the models themselves. This is a crucial step concerning the second challenge proposed by EU-CPSOS.

## 5. Knowledge in the EU-CPSOS Roadmap

The EU-CPSoS has identified three core challenges [24]:

- Core Challenge 1: Distributed, Reliable and Efficient Management of Cyber-physical Systems of Systems
- Core Challenge 2: Engineering Support for the Design-operation Continuum of Cyber-physical Systems of Systems
- Core Challenge 3: Towards Cognitive Cyber-physical Systems of Systems

All these three challenges have been examined in the previous sections in relation with the role of system knowledge in the construction and operation of CPSoS. In this final section we will comment on some issues concerning cognitive aspects of CPSoS that have not always been explored. We will specifically focus on four key subtopics of challenge 3 that the EU-CPSOS roadmap has identified.

### *5.1. Situational awareness*

There is a perceivable need of improving situational awareness in large-scale and complex cyber-physical systems. The pervasive use of low power distributed sensing seems to be the key to this possibility. Obviously the availability of suitable infrastructure —sensors, networks, formats, etc.— is a necessary step to achieve situational awareness, but it is not enough.

CPSoS need also architectural advances in the integration and evaluation of these data across the SoS hierarchy. There is a growing need of creating mechanisms for the engineering of emergent perception, i.e. the perception of emergent phenomena at higher layers of the cognitive pyramid that governs a CPSoS.

In this sense, the use of the so-called "cognitive"[10] technologies [42] may be of help but will not be enough. There will always be a need of merging the data-driven approaches of these technologies with the model-driven techniques of more assurable engineering approaches, as it is the model-driven the modeling that will provide an explanation of the data-driven in order to "discover" and "assimilate" any emergent knowledge.

### *5.2. Handling large amounts of data*

The plethora of sensing systems described before will produce a huge amount of data that will require new mechanisms. Real-time parallel information processing at the exascale may be necessary in some circumstances. The previously mentioned "cognitive" technologies will be useful in the extraction of information

10. We use quotation marks to signal the misuse of the term "cognitive" to refer to a particular class of IT approaches used in the management of big data. Cognitive technology is a much broader field than these technologies seem to suggest.

and knowledge —i.e. models— from such a stream of data but there will be new needs for attention and forgetting.

As Kuipers aptly described, the sensing system of highly cognitive systems will have so much bandwidth that trying to read it all will be like trying to drink from a firehose [43]. Making sense of this caudal of information will require the use of sound mechanisms of filtering meaningful information. This is relatively easy when system objectives are simple, but CPSoS will have different requirements in the different CSs and at the SoS level, thus requiring protocol for the flow of dynamic attention mechanisms across the system.

In the same direction, the amount of petabytes of information to be stored for dynamic analysis and learning from past events will be a challenge. Systems will have a need for meaningful forgetting; and, as it was for attention, this shall be done in a distributed fashion and across layers of emergence.

### *5.3. Learning good operational patterns*

These systems will be able to help operators by using learning capabilities over these shared data to generate good operational patterns to support decision making process. While this will be helpful, there are two issues that shall be solved to make it operational.

The first one is about what can be learnt. Systems will be most of the time at their nominal settings or close to them, thus limiting the possibilities of learning. The introduction of perturbations to perform identification will not be generally accepted because they may upset the users or produce detrimental emergent effects (as it happened in Chernobyl) especially in CPSoS environments. Use of simulators may be a possibility, but the problem is similar: simulators are not always good at simulating abnormal situations. The integration of first-principles models from engineering and data-driven models from runtime may improve the quality of these simulators.

The second issue is related to the possibility of explanation. Artificial intelligence systems may not be good enough at explaining why they reached some conclusion. This may be necessary in some situations of shared autonomy —humans and artefacts making collaborative decisions— and may be impossible if the decision-making mechanism is fully data-driven.

### *5.4. Analysis of user behaviour*

The analysis of user behaviour will be of major utility to prevent misuse of the system. This has already been demonstrated in many socio-technical systems (e.g. in car driving). Note however that this may unnecessarily overconstrain the behaviour of the human-in-the-loop, hence sacrificing one of the major values that humans-as-components do have: resilience. Note that the issue here is one of deep understanding as mentioned before: knowledge of the behaviour at the level of the interface —the HMI in this case— is not enough to determine systemic properties at the level of the SoS. Cognitive artificial agents shall extend their theory-of-mind competences to reach a deep level of integration with humans.

The human-machine interaction challenge has as objective a deep and seamless integration between humans and CPSs. This requires a better understanding on the strengths and weaknesses of humans, in terms of situational awareness, to manage risks and safety [36, 40]. Models of users, where their intentions, emotions, plans, roles and objectives are defined, should be used during the engineering process of CPSoSs. It would be necessary to investigate which models, methods, and interdisciplinary research efforts are required to understand the discerning changes in human-system interactions due to the data, communication technologies, and networking capabilities of the CPSs at the CS an at the SoS levels [44]. However, including humans in the system is not exempt of challenges: 1) to characterise the entire range of human-in-the-loop control applications; 2) to capture and to model human behaviour; and 3) to determine the methods to integrate human models into the system [45].

## 6. Conclusions

In this article we have analysed the roles that knowledge plays in the engineering and operation of cyber-physical systems of systems. However, no matter how critical it is, knowledge has not received the unifying treatment that the construction of robust and resilient cyber-physical systems-of-systems requieres.

There are several reasons that justify the necessity for a more systemic, knowledge-centric approach to the engineering of CPSoS. Firstly, the interdisciplinary features of these systems would require integrating knowledge from different domains (computer science, engineering, physical sciences, cognitive sciences, etc) [38, 32]. The many aspects involved in CPSoS require establishing a multidisciplinary collaboration among disciplines to tackle research and development with a comprehensive view [27]. Therefore, foundational theories should be developed bearing in mind the cross-domain and cross-discipline aspects in CPS, which could only be achieved by integrating existing systems theories in a common one. Secondly, the socio-technical character that has been identified in these systems. CPSoS should be developed with a systematic treatment on how human actors are involved. Note that it is not only the case that humans use, interact and influence the CPSoS —i.e. as users or operators, in a coupling affected by their emotions, desires and intentions. It is also the case that humans are the primary depositaries of the knowledge that defines, vertebrates and enables dynamic and adaptive CPSoS operation. Engineering knowledge shall be seamlessly integrated and operationalised into the runtime CPSoS.

Therefore, the engineering of CPSoSs requires a new way of thinking, merging and integrating existing disciplines from computer science to cognitive sciences to cater for the functionalities devised for this kind of systems. It is necessary a systematic, transdisciplinary, scientific approach that integrates the myriad of paradigms (ontologies, foundational theories, modelling approaches) belonging to the different domains and disciplines covered by CPSoSs.

The knowledge-centric CPSoS will be more adaptable and cognisant of its very own capabilities and possibilities, opening a new world for adaptive, resilient operation of robust autonomous cyber-physical systems-of-systems [46].

## References


[1] Spender J-C and Grant R M 1996 *Knowledge and the Firm: Overview*, *Strategic Management Journal* **17** 5

[2] Bermejo J and Hernández C and Sanz R 2016 *Model-based Engineering of Autonomous Systems using Ontologies and Metamodels*

[3] Sanz R., Bermejo J, Escasany J, Chinchilla R and Garcia C A 2003 *Meaning generation and artificial wisdom* 401

[4] *System of Systems Engineering: Innovations for the Twenty-First Century* 2008, Wiley

[5] Maier M W 1998 *Architecting principles for systems-of-systems*, *Systems Engineering*, John Wiley  Sons, Inc., **1** (4) 2674

[6] DoD 2008 *Systems Engineering Guide for Systems of Systems*, *Technical Report Version 1.0.*

[7] Leveson N 2013 *The Drawbacks in Using The Term 'System of Systems'*, *Biomedical Instrumentation & Technology* **March/April** 115

[8] Walden D D, Roedler G J, Forsberg K J, Hamelin R D and Shortell T M 2015 *Systems Engineering Handbook. A Guide for System Life Cycle Processes and Activities*, Wiley

[9] Iarovyi S and Mohammed W M and Lobov A and Ferrer B R and Lastra J L M 2016 *Cyber-Physical Systems for Open Knowledge-Driven Manufacturing Execution Systems*, *Proceedings of the IEEE* **104** (5) 1142

[10] Newell A 1982 *The Knowledge Level*, *Artificial Intelligence* **18** 87

[11] Schreiber G T and Akkermans H 2000 *Knowledge Engineering and Management: The CommonKADS Methodology*, MIT Press

[12] Craik K J W 1943 *The Nature of Explanation*, Cambridge University Press

[13] Halladay S and Milligan C 2004 *The application of network science principles to knowledge simulation*

[14] Fong P S.W. 2003 *Knowledge creation in multidisciplinary project teams: an empirical study of the processes and their dynamic interrelationships*, *International Journal of Project Management* **21** (7) 497

[15] Gamma E, Helm R, Johnson R and Vlissides J 1995 *Design Patterns: Elements of Reusable Object-Oriented Software*, Addison-Wesley Publishing Company

[16] Bermejo-Alonso J, Sanz R, Rodriguez M and Hernandez C 2010 *Ontology-Based Engineering of Autonomous Systems* 47

[17] OMG 2011 *Business Process Model and Notation (BPMN) Version 2.0*

[18] Hawtin J W and Chung P W H

[19] TAMS4CPS 2016 *Draft Strategic Research Agenda for Collaboration. Modelling and Simulation for Cyber-Physical-Systems*

[20] Merrill M D 2000 *Knowledge objects and mental models* 244

[21] Mitcham C 1994 *Thinking through technology. The path between engineering and philosophy*, The University of Chicago Press

[22] Leveson N 2012 *Engineering a safer world: systems thinking applied to safety*, The MIT Press

[23] AMADEOS 2016 *AMADEOS Conceptual Model - Revised* **D2.3**

[24] CPSOS 2016 *Proposal of a European Research and Innovation Agenda on Cyber-physical Systems of Systems — 2016-2025* **D3.3**

[25] Ratcheva V 2009 *Integrating diverse knowledge through boundary spanning processes – The case of multidisciplinary project teams*, *International Journal of Project Management* **27** (3) 206

[26] Vincenti W G 1993 *What Engineers Know and How They Know It*, Johns Hopkins University Press

[27] CPSOS 2016 *Analysis of the State-of-the-Art and Future Challenges in Cyber-physical Systems of Systems* **D2.4**

[28] Wan S, Li D, Gao J, Roy R and Tong Y 2017 *Process and knowledge management in a collaborative maintenance planning system for high value machine tools*, *Computers in Industry* **84** 14

[29] Hayes M. and Walsham G 2003 *Knowledge sharing and ICTs: A relational perspective*, Blackwell 54

[30] CyPhERS 2013 *Characteristics, capabilities, potential application of cyber-physical systems: a preliminary analysis* **D2.1**

[31] Karsai G and Sztipanovits J 2008 *Model-Integrated Development of Cyber-Physical Systems*, Springer Berlin Heidelberg 46

[32] CyPhERS 2014 *CPS Methods and Techniques* **D4.1**

[33] Schatz B 2014 *The role of models in engineering of Cyber-Physical Systems: challenges and possibilities*

[34] Smirnov A, Levashova T, Shilov N and Sandkuhl K 2014 *COntology for cyber-physical-social systems self-organisation* 1017

[35] Derler P, Lee E A and Sangiovanni-Vincentelli A 2012 *Modeling Cyber-Physical Systems*, *Proceedings of the IEEE (special issue on CPS)* **100** (1) 13

[36] NIST Steering Committee for Foundations in Innovation for Cyber-Physical Systems 2013 *Strategic R&D Opportunities for 21st Century Cyber-Physical Systems*

[37] CyPhERS 2015 *Research Agenda and Recommendation for Actions* **D6.1+2**

[38] Geisberger E and Broy M 2015 *Living in a networked world. Integrated research agenda Cyber-Physical Systems (agendaCPS). ACATECH Study.*

[39] CPS Summit 2008 *Cyber-physical Systems Summit*

[40] NIST 2013 *Strategic Vision and Business Drivers for 21st Century Cyber-Physical Systems*

[41] Floridi L 2005 *Is Semantic Information Meaningful Data?*, *Philosophy and Phenomenological Research* 351

[42] Schatsky D, Muraskin C and Gurumurthy R 2014 *Demystifying artificial intelligence. What business leaders need to know about cognitive technologies*

[43] Kuipers B 2005 *Consciousness: Drinking from the Firehose of Experience* **20** 1298

[44] Broy M and Schmidt A 2014 *Challenges in Engineering Cyber-Physical Systems*, *Computer*, IEEE Computer Society Press, **47** (2) 70

[45] Munir S, Stankovic J A, Chieh-Jan M L and Shan L 2013 *Cyber Physical System Challenges for Human-in-the-Loop Control*, USENIX

[46] Hernández C, Fernández J L, Guadalupe S-E, Bermejo-Alonso J and Sanz R 2015 *Model-Based Metacontrol for Self-adaptation*, Springer, **I** 643

[47] Williams T 2002 *Knowledge and Its Limits*, Oxford University Press

[48] CyPhERS 2014 *CPS: State of the Art* **D5.1**

[49] Kopetz H A 2014 *A Conceptual Model for the Information Transfer in Systems of Systems*

[50] Jakus G, Milutinović V, Omerović S and Tomazžić S 2013 *Concepts, Ontologies, and Knowledge Representation*, Springer

[51] Reed S K and Pease A 2017 *Reasoning from imperfect knowledge*, *Cognitive Systems Research* **41** 56

[52] Pezzillo Iacono P, Marcello M, Marcello M, Gianluigi M and Caterina G 2012 *Knowledge creation and inter-organizational relationships: the development of innovation in the railway industry*, *Journal of Knowledge Management* **16** (4) 6046

[53] Canonico P and De Nito E and Mangia G 2012 *Control mechanisms and knowledge integration in exploitative project teams: a case study from the coal fired power plant industry*, *Journal of Knowledge Management* **16** (4) 5389

[54] Noor S and Minhas H N 2014 *Context-Aware Perception for Cyber-Physical Systems*, Springer Singapore, **540** 149

[55] Meijers A W M and de Vries M J 2009 *Technological Knowledge*, John Wiley & Sons Ltd 70

[56] Cupani A 2006 *La peculiaridad del conocimiento tecnológico*, *Scientia Studia Sao Paulo* **4** (3) 353

[57] Boulding K E 1956 *General Systems Theory—The Skeleton of Science*, *Management Science* **2** (3) 1978

[58] Cheng P, Chong J C and Chen S C, Tarek I E and Millar C J M 2004 *Knowledge repositories in knowledge cities: institutions, conventions and knowledge subnetworks*, *Journal of Knowledge Management* **8** (5) 96

[59] Mugellesi Dow, Siegmar R and Siegmar P 2010 *Managing knowledge for spacecraft operations at ESOC*, *Journal of Knowledge Management* **14** (5) 659

[60] Francalanza E, Borg J and Constantinescu C 2017 *A knowledge-based tool for designing cyber physical production systems*, *Computers in Industry* **84** 39

[61] Howard C, Di Eugenio B, Jordan P and Katz S 2016 *Exploring Initiative as a Signal of Knowledge Co-Construction During Collaborative Problem Solving*, *Cognitive Science*

[62] Lyu G, Chu X and Xue D 2017 *Product modeling from knowledge, distributed computing and lifecycle perspectives: A literature review*, *Computers in Industry* **84** 1

[63] Millar C.J.M. and Ju Choi C 2010 *Development and knowledge resources: a conceptual analysis*, *Journal of Knowledge Management* **14** (5) 759

[64] Nielsen C B, Larsen P G, Fitzgerald J, Woodcock J and Peleska J 2015 *Systems of Systems Engineering: Basic Concepts, Model-Based Techniques, and Research Directions*, *ACM Computing Surveys* **48** (2) 18

[65] Bures T, Nicola R, Gerostathopoulos I, Hoch N, Kit M, Koch N, Monreale G V, Montanari U, Pugliese R, Serbedzija N, Wirsing M and Zambonelli F 2013 *A Life Cycle for the Development of Autonomic Systems: The e-Mobility Showcase*, *IEEE Computer Society Press*, IEEE Conference on Self-Adaptive and Self-Organizing Systems (SASO)

[66] Lind M 2009 *Challenges to Cognitive Systems Engineering: Understanding Qualitative Aspects of Control Actions*

[67] Conant R C and Ross A W 1970 *Every Good Regulator of a System Must Be a Model of that System*, *International Journal of Systems Science*, Taylor & Francis, **1** (2) 89

[68] Simon H A and Newell A 1956 *Models: their uses and limitations*, University of Chicago Press 66

[69] Newell A 1980 *Physical Symbol Systems*, *Cognitive Science* **4** 1353

[70] Landauer R 1992 *Information is Physical* 1

[71] Landauer C and Bellman K L 2001 *New Architectures for Constructed Complex Systems*, *Applied Mathematics and Computation* **120** 149

[72] *Computational Intelligence for Decision Support in Cyber-Physical Systems* 2014, Springer Singapore, **540**

[73] Vogel T and Giese H 2012 *A language for feedback loops in self-adaptive systems: Executable runtime megamodels* 129

[74] Clark J O 2008 *System of systems engineering and family of systems engineering from a standards perspective*, *2008 IEEE International Conference on System of Systems Engineering* 1

[75] Xiong J, Ge B-f, Zhang X-k, Yang K-w and Chen Y-w 2010 *Evaluation method of system-of-systems architecture using knowledge-based executable model* 1417

[76] Bongard J C 2013 *Evolutionary Robotics*, *Commun. ACM*, ACM, **56** (8) 74

[77] Vogel T and Giese H 2014 *Model-Driven Engineering of Self-Adaptive Software with EUREMA*, *ACM Trans. Auton. Adapt. Syst.*, ACM, **8** (4) 18

**Ricardo Sanz** is professor in Automatic Control and Systems Engineering at the Universidad Politécnica de Madrid (UPM). He got a degree in Electrical Engineering with a specialization in Automatic Control and Electronics in 1987 and a PhD in Robotics and Artificial Intelligence in 1990. He is the founder and coordinator of the Autonomous Systems Laboratory of UPM. He has been associated editor of several journals in the intersection between control, artificial intelligence and robotics. He has acted as expert for the European Commission in the fields of real-time embedded systems, cognitive systems and robotics, dependability, complex systems, emerging technologies and cyber-physical systems. He has been chairman of the IFAC TC 3.1 Computers for Control and the Object Management Group Control Systems working group. Since 2019 is the Spanish representative in the Council of Europe Ad-hoc Committee on Artificial Intelligence.

**Julita Bermejo-Alonso** holds a PhD Degree on Robotics by Universidad Politécnica de Madrid (Spain) and a Master of Science by the University of Oxford (UK). She has been lecturer in Calculus, Algebra, Control Systems and Robotics. She has pursued her research career in the domains of Ontologies, Knowledge Science, Manufacturing CPS and Robotics. She has published several papers and chapters in technical and scientific journals. She has participated in the definition of the forthcoming IEEE Autonomous Robotics (AuR) Ontology Standard as well as the IEEE Robot Task Representation standard under development. She has co-organised several Workshops and Conference on the topics of Ontologies and Robotics.

**Manuel Rodriguez** holds a MSc in Chemical Engineering and a PhD in Artificial Intelligence and Intelligent Control both from the Universidad Politecnica de Madrid and he did a postdoctoral fellowship at the Massachusetts Institute of Technology working on a modelling language for hybrid systems. Since then, he has been a professor in the Chemical Engineering Department at the Technical University of Madrid. His research is carried out in the Autonomous Systems Lab and it is focused on modelling and control of complex and large systems. He has been researcher and managed national and international projects. He is member of the International Society of Automation and a reviewer in several international journals. He has authored over 135 research papers published in peer-reviewed international journals and conferences.

**Esther Aguado** received the BEng and MEng degrees in Electrical, Control, and Electronics Industrial Engineering from the Technical University of Madrid, Spain in 2017 and 2019, respectively. In 2020, she received a MSc in Robotics and Automation in the same university, where she is currently pursuing a PhD degree in Robotics and Artificial Intelligence. Currently, her research targets the existing difficulty in engineering for increasingly autonomous and resilient systems. Her research interests include intelligent control, augmented autonomy, and robotic consciousness and self-awareness.