Frontiers of Information Technology & Electronic Engineering
www.jzus.zju.edu.cn; engineering.cae.cn; www.springerlink.com
ISSN 2095-9184 (print); ISSN 2095-9230 (online)
E-mail: jzus@zju.edu.cn

Perspective:

Emergence in cyber-physical systems: potential and risk*

Shmuel TYSZBEROWICZ‡1,2, David FAITELSON‡2

[1]*RISE-Centre for Research and Innovation in Software Engineering, Southwest University, Chongqing 400715, China*

[2]*Department of Software Engineering, Afeka Tel Aviv Academic College of Engineering, 6998812 Tel Aviv, Israel*

E-mail: tyshbe@tau.ac.il; davidf@afeka.ac.il

Received June 9, 2020; Revision accepted Oct. 11, 2020; Crosschecked Oct. 22, 2020

Abstract: Cyber-physical systems (CPSs) are distributed assemblages of computing, communicating, and physical components that sense their environment, algorithmically assess the incoming information, and affect their physical environment. Thus, they share a common structure with other complex adaptive systems, and therefore share both the possible benefits and the probable harmful effects of emergent phenomena. Emergence is an often unexpected pattern that arises from the interactions among the individual system components and the environment. In this paper we focus on three major problems concerning emergence in the context of CPSs: how to successfully exploit emergence, how to avoid its detrimental effects in a single CPS, and how to avoid harmful emergence that arises due to unexpected interaction among several independently developed CPSs that are operating in the same environment. We review the state of the research with regard to these problems and outline several approaches that could be used to address them.

Key words: Cyber-physical systems; Emergent behavior; Complex adaptive systems; Stigmergy; Subsumption; Digital twins

https://doi.org/10.1631/FITEE.2000279 **CLC number:** TP311

1 Introduction

Cyber-physical systems (CPSs) are engineered systems that are built from, and depend upon, the seamless integration of cyber (computation) and physical components. This combination aims at solving problems that cannot be solved by each part alone. Many researchers (e.g., Hu (2013) and Griffor et al. (2017)) believe that CPSs will improve scalability, safety, security, resiliency, and adaptability of systems. Recent initiatives such as Industry 4.0 and the Internet of Things (IoT) are examples of such systems. CPSs are often constructed as a network of autonomous communicating agents, whose behavior is governed by algorithms and software. It is probable that in the not too distant future, our entire world will be embedded with large CPSs. These systems will be developed and deployed by many organizations without any central design authority, yet they will continuously and frequently communicate and cooperate with each other.

We know that when we put together a large number of agents that communicate with each other, we often get emergent behaviors that we cannot predict when we examine each agent in isolation. Sometimes such emergent properties are beneficial, leading, for example, to the complex nests that termites and ants build (Khuong et al., 2016). However, sometimes emergent properties are harmful. The problem is that because of the speed at which computerized agents process information and communicate, any

‡ Corresponding authors

* Project supported by the Capacity Development Fund of Southwest University, China (No. SWU116007) and the National Natural Science Foundation of China (Nos. 61732019, 61672435, and 61811530327)

ORCID: Shmuel TYSZBEROWICZ, https://orcid.org/0000-0003-4937-8138; David FAITELSON, https://orcid.org/0000-0002-8332-7644

negative emergent phenomenon introduced by their interactions will manifest itself quickly, and its effect will be greatly increased. Even today we can see the harmful effect of such negative emergent phenomena in several isolated places where we have installed algorithmic agents.

For example, the Internet is arguably the largest cyber-physical system that we have deployed so far. In the last 20 years, it has changed our society beyond recognition. Few, if any, could have predicted the social and economic structures, both beneficial and harmful, that have emerged from the conceptually simple technical act of connecting many computers to each other.

As another example, stock markets are complex adaptive systems (Tesfatsion, 2002) where emergent phenomena such as crashes and bubbles develop from time to time. However, in recent years—with the introduction of algorithmic trading agents—there have been several catastrophic incidents (for example, on May 6, 2010, the Dow Jones Industrial average lost and regained over 1000 points within a space of 36 min (Clarke, 2014)) that were attributed to the behavior of such agents (Lenglet, 2011).

We may argue that a computerized stock exchange is a cyber-physical system, because it consists of a network of independent algorithmic agents that sample and affect the state of the world through their autonomous actions. The flash crashes are early examples of harmful emergence in a CPS. Another early example of harmful emergence in CPSs occurred due to the behavior of social traffic management systems. When a major road is blocked, in some cases the system would route all vehicles to a narrow road, which would quickly flood the new route and create an even bigger traffic jam. The system would then again reroute the vehicles to use yet another narrow road; as this pattern repeats, more and more vehicles become trapped in these narrow roads. The emerging result is often much worse than if all the vehicles would have just queued on the original road (GLOBES, 2018; Salem, 2018; Cabannes et al., 2019).

By analogy to the stock exchange, we fear that when the human drivers in the vehicles are replaced with computerized agents, the harmful emergent effects will be significantly amplified. What would be the equivalent of a flash crash in a future autonomous vehicle CPS?

Fortunately, because large CPSs are still in their infancy and have not yet been deployed at scale (this makes it difficult to find examples of emergence in existing CPSs), we still have time to get ready and perhaps find ways to prevent such harmful emergent phenomena.

CPSs present many challenges that we must address if we want them to achieve their potential benefits (Baheti and Gill, 2011; Zhang and He, 2011; Lee, 2015; Koutsoukos et al., 2018; Törngren and Grogan, 2018; Törngren and Sellgren, 2018; Neema et al., 2019). In this paper we will explore one particular challenge, namely that of emergence. We will focus on three major problems concerning emergence: First, how to exploit emergence to achieve beneficial properties; second, how to ensure that our CPSs will not create harmful emergent phenomena; and finally, and perhaps the most difficult challenge, how to ensure that when many unrelated organizations deploy their CPSs in the same environment, their combined interactions will not create harmful emergent phenomena.

Our contribution is threefold: (1) a description of several key properties of emergent phenomena that must be considered when designing and operating CPSs; (2) an investigation of three major challenges that we must address in order to benefit from emergence and avoid its dangers when implementing CPSs; (3) a review that collects in a single place important and recent research that is specifically relevant to the problem of emergence in the context of CPSs.

The rest of the paper is organized as follows. In Section 2 we explain CPSs. Then, in Section 3, we explain the nature and properties of emergence and illustrate emergent phenomena in the context of CPSs. We elaborate on the three major challenges that we have identified with regard to emergence in CPSs in Section 4. In Section 5, we review the work related to the problems of emergence in the context of CPSs. Finally, conclusions are drawn in Section 6.

2 Cyber-physical systems

The term "cyber-physical systems" was coined in 2006 by Helen Gill at the National Science Foundation in the United States, and it refers to the integration of computational and physical processes. CPS research is characterized as "a new discipline at the intersection of physical, biological, engineering and information sciences" (Sztipanovits, 2007). For example, a smart grid (Fang et al., 2012) is a CPS that uses bidirectional electricity and information flow to provide efficient adaptive and robust delivery of electricity. Such a system could reroute the delivery of electricity around a failed transformer or use dynamic pricing to flatten the electricity demand curve. A smart healthcare system (Catarinucci et al., 2015) could improve the quality of service in a hospital, by using wireless sensor networks to track the state of patients in real time and alert staff when an emergency situation occurs. Smart transportation systems (Glancy, 2013) are CPSs that consist of vehicles, infrastructure, and geographic information system (GIS) navigation systems, all communicating with each other to improve the safety and efficiency of the transportation system. Other examples of CPSs are autonomous vehicles, entertainment systems (consumer electronics), industrial (factory) automation, dynamic control systems, industrial robots, smart buildings, weapon systems, hybrid gas-electric vehicles, etc. See, e.g., Chen (2017), for additional CPS applications.

In general, CPSs employ sensors and network connectivity to enable collection and exchange of data through wired and wireless networks: they receive data samples from sensors, perform certain computations, and then send commands to actuators in a distributed heterogeneous environment. The cyber component (algorithms in the system) is responsible for processing, communicating, and controlling information (Sanfelice, 2015; Tripakis, 2016). The physical components include systems that exist in the physical world (e.g., processes that are monitored or controlled); those components continuously interact with their environment. The cyber component is distributed throughout the system and is tightly coupled with the physical components through connected components such as sensors, actuators, converters, signal conditioners, and digital communication networks. Components of a CPS can run on various platforms and use functionalities provided by other components. Systems will implement features by collaborating with other systems that provide (part of) the required functionality, and this configuration occurs while being deployed in the field (Mosterman and Zander, 2016). Hence, CPS functionalities are emerging from the networked interaction of physical and computational processes (Broy et al., 2019).

To summarize, CPSs are hybrid engineered systems that dynamically combine autonomous heterogeneous independent, and many times distributed, components. These components constantly interact with each other by exchanging data. CPSs provide rich functionalities that could only be realized by these dynamic interactions and cannot be delivered otherwise. This emergent behavior is a distinguishing characteristic of CPSs (Maier, 1996; Keating et al., 2008; Ncube et al., 2013). CPSs also interact with humans through many modalities (Baheti and Gill, 2011), and operate under highly dynamic conditions (D'Angelo et al., 2017). These dynamics introduce uncertainty, which may harm the system and lead to incomplete, inaccurate, and unreliable results (Perez-Palacin and Mirandola, 2014).

Due to all these factors, the design and maintenance of CPSs is extremely complex. In the next section we will discuss emergence phenomena in some detail and illustrate how emergent behaviors may occur in natural and man-made systems and in particular in CPSs.

3 Emergence in cyber-physical systems

A tornado is a powerful natural phenomenon. Under the right atmospheric conditions the interactions between individual air molecules create a new phenomenon, a stable dynamic process that we recognize as a rotating column of air. A tornado is an example of emergence in nature. Other examples include waves of all kinds, the weather, the flocking of birds, the schooling of fish, the nests constructed by ants and termites, market economies, and traffic congestion (which is really a kind of wave). Thus, emergence is everywhere: it appears in physical systems, in biological systems, in computational systems, e.g., gliders in Conway's game of life (see, for example, Adamatzky (2010)), and in social systems (see, for instance, Sawyer (2005)). Sometimes it is beneficial (e.g., market economies) and sometimes it is harmful (e.g., traffic congestion (Liu et al., 2009) and locust swarms (Ariel and Ayali, 2015)).

Emergence is a loaded term and there is a lot of controversy regarding the exact definition of emergence. We hope that by considering these examples and the following definitions, the concept of emergence will naturally emerge in the mind of the reader (for many more definitions see Johnson and Padilla (2018)):

1. Emergent behavior is not explicitly encoded in the agents or components that make up the simuland or the model; rather, it emerges in reality or during a simulation from the interaction of agents or components with each other and their environment (Williams, 1997).

2. Emergents are recognizable, persistent, and recurring patterns that are not predicated due to the inherent difficulty of calculation and the size of the potential state space (Holland JH, 1999).

3. An emergent property is a property possessed by an assemblage of things that is not possessed by any member of the assemblage individually (Maier, 2015).

For our purposes, we will summarize these definitions as follows: An emergent phenomenon is a stable pattern of organization of the components of an ensemble that is a result of the interactions between the components. Because we discuss emergence in the context of CPSs, we will further focus on emergence in biological and man-made systems. In particular, beneficial emergence is common in biological systems; such systems typically have a particular structure known as a complex adaptive system (Mittal et al., 2018). A complex adaptive system consists of a large number of agents (individual active components) that interact to achieve a goal. However, no single agent is aware of the system's goal, but rather their combined behavior and interactions generate emerging patterns that achieve the system's goal (build a nest, regulate temperature, forage for food, etc.). The agents in a complex adaptive system interact locally, either directly with neighboring agents or indirectly through the environment using a stigmergic mechanism (Theraulaz and Bonabeau, 1999). In a stigmergic mechanism, the agents communicate indirectly by using the environment as a shared memory. For example, ants that are carrying food, deposit pheromones on the ground. Other ants detect these pheromones and use them as a guide to find the food source. Due to their structure, complex adaptive systems are extremely scalable, fault-tolerant, and adaptive (Holland JH, 2006; Mittal et al., 2018)—all important engineering qualities. Complex adaptive systems, however, are inherently unstable, and when they are stressed beyond their limits, they often break catastrophically (Tan et al., 2005). In addition, unless we know where to look, we cannot see any warning sign for the impending catastrophe, because the self-adapting system keeps regulating until the very last moment before its behavior becomes chaotic (Novak and Wilensky, 2006).

As we have seen previously, CPSs are often built from many active individual (and sometimes autonomous) components that interact with each other and with their environment (Monostori, 2018). Their structure and dynamics mean that, whether we like it or not, such systems are in many cases complex and adaptive, and may therefore exhibit emergent phenomena and behave chaotically (Kopetz et al., 2016). A dramatic example of a chaotic behavior in a man-made system is the previously mentioned flash crash event (Clarke, 2014).

CPS applications are planned for major critical infrastructure such as energy systems (Fang et al., 2012), health care, and transportation. Therefore, it is vital that we have enough theory and tools to design such systems with very high levels of assurance that they will not produce detrimental emergent behaviors and that we will know when they are dangerously approaching a point of catastrophic collapse. We need to guarantee the trustworthiness of both the expected and unexpected emergent behaviors of the CPS.

The classical approach to the design of reliable engineering systems is to develop a general scientific theory of the phenomena relevant to the system in question, and then use theory to guide the system's structure and behavior to ensure that it achieves its goals. Thus, for example, the theory of aerodynamics is a basis for the design of all kinds of flying machines, the theory of material strength is used by designers of a vast range of different products, and the theory of software complexity is used to analyze the asymptotic execution time of algorithms regardless of their problem domain. It would be very helpful if we could develop a similar general theory of emergence that would help us predict, for any kind of system, if it will develop an emergent phenomenon and of what kind. However, no such theory currently exists, and we argue that no such theory can exist. To see why, let us look at Conway's game of life, one of the most commonly used examples of emergence (Adamatzky, 2010; Kopetz et al., 2016; Schaff, 2018). Conway's game of life is a cellular automaton that is based on a small set of simple rules, yet it produces complex patterns that cannot be predicted just by analyzing the rules. Conway has shown that this system is Turing complete (Rendell, 2002). If we had an algorithm that could decide if a given pattern would emerge in the game of life, then we could use this algorithm to solve the halting problem by assembling a Turing machine in the game and asking the algorithm if a halting pattern would eventually appear. Therefore, the question of predicting emergence is fundamentally not answerable in the general case. See Culik et al. (1990) for a more rigorous treatment. However, like the halting problem, emergence is semi-decidable. That is, we can simulate the game of life and wait for the pattern to emerge; this is the best we can do in the general case.

Even though we cannot hope to produce a general theory of emergence, we can nevertheless point to several properties that are characteristics of emergent phenomena:

1. Physical emergence requires a constant supply of energy (Johnson et al., 2018, Chapter 8). Tornadoes, waves, and Jupiter's giant red spot, all emerge only when enough energy accumulates in the system, and disappear when that energy dissipates. Indeed, such phenomena are called dissipative structures in the modern theory of thermodynamics (Kondepudi and Prigogine, 2014).

2. Similarly, in agent based systems, emergence requires a lot of interaction (information exchange). A few ants cannot survive, a few birds cannot create a flock, and a few grasshoppers cannot form a locust swarm. The high speed and high frequency of transactions made by algorithmic trading was one of the primary contributing factors to the flash crash (Johnson and Padilla, 2018).

3. Emergence often involves feedback loops where the agents affect their environment and in turn are affected by changes in their environment (Clack and Carlos-Sandberg, 2018, Chapter 20). For example, social traffic systems create a dynamic traffic map from the reports sent by the system's agents (the drivers) which use the very same map to guide their future navigation decisions.

There is a very strong link between the first two properties. An important theorem of information theory shows that there is a global minimal energetic price for the transmission of one bit of information, regardless of the implementation details of the agents that transmit and receive the information (Pierce, 2012, Chapter 10). Therefore, as the number of messages and the frequency of information exchange increase, so does the energy in the system; and with it, the chances of creating emergent phenomena increase.

These properties are particularly important for CPS designers because as modern computers communicate and process information extremely fast, harmful emergence could occur so quickly that its effects become severe before we have a chance to counter them. For example, a power grid CPS with active algorithmic agents that constantly balance its nodes could fail catastrophically, domino style, when some nodes fail and others become overloaded as they try to compensate for the failure (Buldyrev et al., 2010).

In addition, we believe that we can exploit our understanding of these principles to reduce the risk of catastrophic emergence in a CPS. If the risk of emergence rises as the amount of energy and the communication rate increase, then we can reduce this risk by deliberately limiting the amount of energy and the information exchange rate in the system. For example, if a social navigation system restricts knowledge about the availability of narrow roads based on their capacity, it can avoid flooding them with vehicles. This is similar to how bees transmit information to their nest mates. When a bee returns with information about the availability of a food source, it dances to recruit members, but it cannot recruit the entire nest because only bees that are relatively close to it can see the dance. This prevents the bees from traveling en masse to the same location. However, this idea requires further research to test its effectiveness.

4 Major challenges

We have identified three major challenges with respect to emergence in the context of implementing CPSs. We list them here in increasing levels of difficulty:

(1) Designing a CPS to exploit emergence as beneficial phenomena;

(2) Verifying that a CPS does not produce harmful emergent phenomena;

(3) Ensuring that a collection of independently developed CPSs do not produce harmful emergent

phenomena when they are deployed in the same environment.

We will now discuss each of these challenges.

4.1 Design for emergence

The first challenge is how to use the benefits of emergence to design robust and efficient CPSs. By observing nature, we know that it is possible to design extremely robust and effective systems by exploiting emergence. Important examples are social insects, such as ants and termites, whose systems consist of simple (relatively speaking) agents, but nevertheless create highly complicated and adaptive structures. For example, termites build enormous nests (at a typical size of 8 m, these structures are 1000 times larger than the body size of a termite) that house millions of individuals, and maintain the temperature inside the nest between 33 and 36 °C, while the temperatures outside range from below freezing at night to above 45 °C by day (Korb, 2003; Jones and Oldroyd, 2006). Systems built in this engineering style have many benefits: they are fault-tolerant, because there is no central authority and therefore no single point of failure; they are adaptive, because the individual agents always respond to the concrete information in their immediate environment; they are extremely scalable, because they are based on local communication patterns; and they are simple to engineer, because the agents are simple and the communication rules connecting the agents are simple and local.

In recent years there have been several examples of successful systems that were designed in this style. For instance, the Eastgate Centre mall in Harare, Zimbabwe, was designed to passively regulate its temperature based on ideas taken from the way termites build their nests (Baird, 2003). Other examples are the various peer-to-peer networks for transferring large files over the Internet (Nwogugu, 2016, Chapter 4.1.1). Open markets are another example of systems that successfully exploit emergence. Indeed, this was noted a long time ago by Adam Smith, who called this beneficial emergent phenomenon “the invisible hand” (Smith, 2002, Chapter 2). The operating mechanisms of open markets were later investigated using game theory, and more recently extended to algorithmic game theory (Nisan et al., 2007). Such techniques have been successfully employed in practical resource allocation problems; for example, a resource exchange was used to efficiently allocate the resources on the Cassini spacecraft among the 13 different scientific instruments aboard the vessel and the five different countries that built them (Ledyard et al., 1994). These techniques are specific to resource allocation problems, but whether they can be adapted to design other forms of emergent systems is an interesting open problem. Thus, with the exception of (the very problem-specific) market-based systems, there is not yet a coherent body of knowledge describing how we can systematically design and develop emergent systems.

A system based on emergence must still have a well-specified functionality and goals, but its components are not designed to explicitly achieve its goals. Instead, what we explicitly encode in the system is a set of simple behavioral rules for each agent, and the desired goal should emerge from the common actions of the agents that follow the rules (Petty, 2018). This is in contrast to the standard approach to system design where we encode its functionality explicitly.

To illustrate the difference between a standard system design and an emergent system design (We are not suggesting the following system as a realistic design, but merely as a sketch that illustrates how emergent systems typically operate), consider the problem of regulating the temperature of a large building. A standard system design might consist of a small set of temperature sensors and air conditioning units, all connected to a central controller that measures the temperature in the building and activates the air conditioners to heat or cool as necessary. In contrast, inspired by the Daisyworld model (Novak and Wilensky, 2006), an emergent system design might consist of a large collection of individual units evenly arranged on the roof of the building. Each such unit has a temperature sensor and a disk pointed to the sky. The unit controls the disk’s color. If the temperature around the unit is too high, it changes the color of the disk to white, and if the temperature is too low, it changes the color to black. The units do not communicate with each other directly, but they do communicate with each other through a stigmergic effect, because when they cool or heat their surroundings, they affect the neighboring units, which sense these changes in temperature. Provided that we have correctly determined the system parameters (number of units, size of disks, etc.) and in a suitable environment, the overall (emerging) effect of all the units will be to keep the building at a constant temperature. Compared to the traditional system design, this emergent design is fault-tolerant, because failure of a single unit will have a small local effect on the operation of the system, whereas a failure of the controller will disable the entire system. It is also scalable, because to support larger buildings, we can simply add more units. Finally, it is very adaptable because it will automatically compensate for changes in ambient temperature and will also regulate different areas of the building according to the specific temperature of those areas.

One particular architecture that is built to exploit emergence and is therefore worth exploring is subsumption (Brooks, 1990). Brooks developed this style in the late 1980s and used it to build highly adaptable robots that used modest hardware, but performed exceptionally well in practice. The robots that Brooks built were designed to operate in uncontrolled environments. They constantly sampled their environment, computed responses to what they sensed, and changed their state and behavior to affect the environment to achieve their goals. Thus, they are good CPS models.

In the subsumption style, a CPS is developed in layers of increasing behavioral sophistication, but in contrast to the traditional layered style, the layers are composed in parallel. For example, in many of Brooks's robots the lowest layer implemented a simple obstacle-avoidance behavior. After this layer was developed and tested, a new layer was added to make the robot move about and explore its surroundings. Both layers run in parallel; therefore, if while moving about, the obstacle-avoidance layer detects an obstacle, it changes the direction of the robot to avoid the obstacle. Each layer implements a new kind of behavior. Thus, to extend the system with a new behavior, one only has to add a new layer, which is installed and run in parallel with the others. Conflict resolution is implemented using inhibitory signals that prevent messages from passing along the inhibited channel for a short period of time. For example, Connell (1989) programmed one of Brooks's robots to steal empty soda cans from people's desks. The robot had an arm and an object recognition system, each implemented using separate behavioral layers. The object recognition system caused the robot to keep moving until it detected a soda can, at which point it would stop. The arm behavioral layer would initiate a grasping movement when the robot stopped moving (this is a simplification; for the full description see Connell (1989)). Thus, the system was implemented as a collection of simple individual agents, each of which implemented a simple behavior. The robot's functional goal emerged from the interaction and combined behavior of these agents.

Brooks's team has successfully developed many kinds of robots using this architecture, which makes subsumption the only architecture of which we are aware, that could claim to be a successful systematic technique for designing non-trivial engineered-emergence systems.

Successful emergence-based systems are fault-tolerant, scalable, and adaptable. These are important benefits that we would like to achieve. Unfortunately, unlike traditional design, such systems can also exhibit undesirable chaotic behaviors. We would like to avoid these kinds of negative phenomena, which brings us to our next challenge.

4.2 Avoiding harmful intra-CPS emergence

The second challenge is to ensure that a system does not produce undesirable emergent phenomena. This challenge is more difficult than the previous one, because we do not know, a priori, all the factors that can create emergent phenomena. We can create models and simulate the system to explore its behavior and to study how it behaves with different parameters. Such an analysis can help us find undesirable emergent phenomena that are hidden in our system design, but there may be properties that we have either decided not to add to our model or of which we are not aware, and these may lead to emergent phenomena that we will not be able to see in our simulations (Mittal et al., 2018).

One possible approach that we believe is worth investigating, is to borrow a well-known idea from fault-tolerant systems (Pullum, 2001) and assign the modeling task to several independent teams. Analyzing the different models and comparing them to each other can broaden the scope of the system under investigation and may improve our chances of detecting hidden emergent phenomena.

Another approach is to create a catalog of emergent design patterns (we can start with stealing Nature's patterns) that come with a large body of knowledge regarding their beneficial and detrimental emergent phenomena. By basing our designs on familiar patterns, we can minimize the risk of unpleasant surprises. Moreover, any such surprise that we will find will benefit all the other systems that are using this pattern. For example, researchers have mimicked the behavior of flocking birds to successfully coordinate large numbers of drones (Vásárhelyi et al., 2018), while other researchers have mimicked the communication patterns of ants to optimize routing of TCP/IP communication packets (Gadomska and Pacut, 2007). We could use the architecture of such systems as templates or starting points for designing CPSs.

Ecological systems are frequently very diverse, with many different species interacting, cooperating, and competing in the same environment. There is very strong evidence (Peterson et al., 1998; Gunderson, 2000) that biodiversity is a major factor in the resilience of ecological systems. We can see this phenomenon very clearly in a simulation of predator-prey dynamics (Wilensky and Reisman, 2006). The model consists of two variants. In the first variant, there are just two species: wolves that prey on sheep. This version is unstable; in many simulations either the entire sheep population or the entire wolf population completely dies and the system collapses. The second version adds grass to the model, and then it becomes very stable, making extinction events very rare. Accordingly, it could be beneficial to design CPSs using various agents, some of them performing similar functions but in different ways and with different characteristics. This redundancy can reduce the sensitivity of the system, making it less likely to develop chaotic oscillations, and therefore increase its resilience. This achieves the same purpose as using redundant components in traditional safety-critical systems.

Looking for harmful emergent phenomena in our system is a difficult task, but at least we have some control over the system because we are the designers of this system.

4.3 Avoiding harmful inter-CPS emergence

The final, and most difficult, challenge that we have identified is how to avoid harmful emergent phenomena that arise as a result of the interaction between CPSs that have been developed independently and obliviously of each other, and then deployed in the same environment. For example, a future smart transportation system may interact in a non-trivial manner with a future smart power grid system. When several CPSs are operating in the same environment, we can consider each CPS as an agent, sensing and modifying its environment and interacting with other CPS agents indirectly through (unintentional but nevertheless real) stigmergic effects. As far as we know, such a scenario has never occurred so far, but we know that systems of agents that interact indirectly through stigmergic effects often create emergent phenomena. Therefore, there is no reason to believe that we will be spared from such emergence when many CPSs operate in the same environment. This is a difficult challenge, because it is not merely a technological problem but rather a techno-social problem. It requires regulatory and political mechanisms to control and certify the introduction of major CPSs into critical environments such as transportation systems and energy systems. This problem resembles the case of a patient who is prescribed drugs by several doctors, each treating a different disease. Two drugs that when taken individually can cure each ailment, may produce a harmful result when taken together. Yet, our poor patient may not be alerted to this harmful situation because each doctor, treating her/him individually and unaware of the other treatments, will not have any reason to suspect such an interaction. This problem is addressed by regulation (Tannenbaum and Sheehan, 2014).

5 Related work

There is a lot of discussion about the nature of emergence: whether it is a shallow phenomenon (weak emergence) that simply represents our ignorance or inability to find an explicit connection between the components and the overall behavior, or whether there is something else that causes the emergence, something that cannot be deduced even in principle (strong emergence) (Wildman and Shults, 2018). We do not think this question is important for CPS design, because the end result is the same regardless of the reason.

Although there are many works that discuss emergence in the context of systems engineering and CPSs, there are relatively few works that offer practical advice regarding the construction of such systems in the presence of emergence. Keating et al.

(2008) noted that the requirements for dealing with emergence should be established to effectively address them at design time, but did not explain how this could be done. Wassyng et al. (2016) said that the complexity and distributed nature of CPSs make it difficult to identify hazards that arise as a result of emergent behaviors, but they did not offer design techniques to deal with this problem. Similarly, Monostori (2018) addressed the importance of forecasting the emergent behavior of CPSs but without detailed discussion. Törngren and Grogan (2018) highlighted the challenges involved in building CPSs as opposed to existing systems, but did not discuss emergence apart from briefly mentioning it as a major challenge. Maier (1996) offered guidelines for the design of large systems of systems but left the question of how to design for emergent behavior open, only suggesting that using pseudo-economic mechanisms (also known as mechanism design and algorithmic game theory) could be a fruitful research direction. In a later work Maier discussed emergence in the context of systems of systems (SoS) (Maier, 2015). He classified emergence into four categories based on the difficulty of predicting it, and gave examples of each category (except for the last one, spooky emergence, which represents emergent system phenomena that are inconsistent with the behavior of its parts). He noted that often we fail to predict harmful emergent phenomena but can understand it after it manifests itself.

One work that does provide practical design guidelines was presented by Kopetz et al. (2016), who suggested equipping the CPS with an independent monitor component that will monitor its state, and bring it to a safe state once it identifies that a harmful emergent phenomenon is about to occur. Such a monitor could be extended to a digital twin (DT). A DT is a software virtual model of a physical system that continuously mirrors its current state and runtime behavior. It is an elaboration of existing virtual representations of machines and products, such as computer-aided design and computer-aided manufacturing, taking advantages of the current state of the art of networking. The DT uses highly detailed simulations that exploit the data coming from the field (Macchi et al., 2018). It can be used to predict the future states of the system (Gabor et al., 2016).

However, for such an approach to work, we must know what to put in our model. We must know the critical parameters that cause the emergent phenomena to occur. For example, in the Daisyworld simulation (Novak and Wilensky, 2006), it is not effective to monitor the system's temperature, because the system will maintain it without any noticeable irregularity right until the point where it collapses and starts oscillating chaotically. Instead, we must keep track of the number of black and white daisies and sound the alarm when one of these populations becomes dangerously low. Thus, we could use simulations to push our system to its boundaries and study what parameters correlate with the onslaught of its destabilization. We can then design our systems to report and measure these parameters at runtime.

Another highly relevant work was proposed by Zurita and Tumer (2017). They first reviewed the notion of emergence as it is explored in complexity science, then described several important misconceptions about emergence, and finally, discussed how we can employ emergence to design systems using multi-agent systems. They proposed several approaches for constructing such systems, but did not discuss how we can ensure that these systems will not possess harmful emergent phenomena.

Jun et al. (2008) took an interesting view of emergence by analyzing emergence in a formal method setting. They investigated emergence as a refinement relationship between descriptions of the same system at different abstraction levels. We believe that this line of work is important in particular for safety-critical applications that often use formal methods to ensure their correct operation. A formal method approach to emergence could help in designing safety-critical systems that use beneficial emergence and avoid harmful emergence.

Many researchers have noted the importance of using simulations to study emergence (e.g., Holland JH (1992), Wilensky and Reisman (2006), Welch et al. (2012), Holland T (2018), and Zeigler (2018)). However, simulations raise two major problems. First, because simulations use numerical algorithms, we can never be sure if the phenomena we observed are real or an artifact of the inaccuracies in the simulation. Even the famous Lorenz attractor was only relatively recently proven to be real (Tucker, 1999). Second, as noted by several researchers (Ferris, 2018; Mittal et al., 2018), we can never be certain that we have included in our model, all the parameters that cover all the possible emergent phenomena that the system can create.

Large CPSs, such as a smart power grid or a smart transportation system, are complex entities that consist of many components, each of which is itself a complex entity; thus, they are systems of systems. SoS engineering is a sub-field of systems engineering that focuses on the boundaries and interactions among independent, distributed, and evolving constituent systems and their stakeholders. There is a large body of knowledge concerning the design and architecture of SoS that is relevant to the design and architecture of CPSs. In particular, Maier (1996) and Maier and Rechtin (1997) have identified four important SoS design principles, which we summarize below:

1. Stable intermediate forms

It will be much easier for a system of systems to evolve if it is designed in a series of increments, where each increment is a functioning system. This idea is very similar to agile techniques such as continuous deployment and delta modeling (Haber et al., 2011). The Internet is an excellent example of this principle, because in the last 50 years it has successfully evolved through stable intermediate forms to reach its current state.

2. Policy triage

Systems of systems often communicate with other systems and their components are often created by many different vendors. Therefore, it is important not to over-control them, but instead to focus control on the means for achieving reliable interoperability—leaving the rest as the responsibility of the individual component makers.

3. Leverage at the interfaces

The most important leverage point for affecting a large CPS is at the component interfaces. The best example is perhaps the Internet Protocol (IP). If you know how to speak IP, you become a member of the Internet. There is nothing else that is required.

4. Ensuring cooperation

Cooperation and collaboration are good. The more of them the better. When designing a system of systems that requires voluntary collaboration, the mechanisms and incentives for collaboration must be explicitly designed into the system. This principle is clearly visible in such systems as Facebook and the social navigation system Waze.

We have included these guidelines here because we believe that while not directly relevant to emergence, they are very important for the design of complex CPSs.

6 Conclusions

We have described the concept of emergence in the context of CPS design. We have seen that emergence can be either beneficial or harmful, but cannot be ignored. We have outlined three major problems that we believe are essential to solve if we are to succeed in developing large CPSs: (1) how to exploit emergence, (2) how to avoid harmful emergence in a single CPS, and (3) how to avoid harmful emergence in a CPS ecology. We have discussed some approaches for tackling these problems and reviewed the state of the research with regard to these problems. We have seen that emergence can be examined through many different scientific disciplines: modern thermodynamics, complexity theory, cellular automata, algorithmic game theory, biology, and chaos theory. Each of these fields is a relatively young field that has its own large body of knowledge. Thus, a fourth challenge we may add to our paper is the challenge of being able to understand enough about each field to shed more light on the phenomena of emergence and to distill this knowledge into practical insights with regard to construction of CPSs that will benefit from emergence and avoid its dangers.

Contributors

David FAITELSON and Shmuel TYSZBEROWICZ designed the research, processed the data, drafted the manuscript, and revised and finalized the paper.

Compliance with ethics guidelines

Shmuel TYSZBEROWICZ and David FAITELSON declare that they have no conflict of interest.

References

Adamatzky A, 2010. Game of Life Cellular Automata. Springer, London, UK. https://doi.org/10.1007/978-1-84996-217-9

Ariel G, Ayali A, 2015. Locust collective motion and its modeling. *PLoS Comput Biol*, 11(12):e1004522. https://doi.org/10.1371/journal.pcbi.1004522

Baheti R, Gill H, 2011. Cyber-physical systems. In: Samad T, Annaswamy AM (Eds.), The Impact of Control Technology. IEEE Control Systems Sociery.

Baird G, 2003. Eastgate Centre, Harare, Zimbabwe. In: The Architectural Expression of Environmental Control Systems. Taylor & Francis.

Brooks RA, 1990. Elephants don't play chess. *Robot Auton Syst*, 6(1-2):3-15.
https://doi.org/10.1016/S0921-8890(05)80025-9

Broy M, Daembkes H, Sztipanovits J, 2019. Editorial to the theme section on model-based design of cyber-physical systems. *Softw Syst Model*, 18(3):1575-1576.
https://doi.org/10.1007/s10270-018-0670-9

Buldyrev SV, Parshani R, Paul G, et al., 2010. Catastrophic cascade of failures in interdependent networks. *Nature*, 464(7291):1025-1028.
https://doi.org/10.1038/nature08932

Cabannes T, Sangiovanni M, Keimer A, et al., 2019. Regrets in routing networks: measuring the impact of routing apps in traffic. *ACM Trans Spat Algor Syst*, 5(2):9.
https://doi.org/10.1145/3325916

Catarinucci L, de Donno D, Mainetti L, et al., 2015. An IoT-aware architecture for smart healthcare systems. *IEEE Int Things J*, 2(6):515-526.

Chen H, 2017. Applications of cyber-physical system: a literature review. *J Ind Integr Manag*, 2(3):1750012.
https://doi.org/10.1142/S2424862217500129

Clack CD, Carlos-Sandberg L, 2018. InterDyne: a simulation method for exploring emergent behavior deriving from interaction dynamics. In: Rainey LB, Jamshidi M (Eds.), Engineering Emergence: a Modeling and Simulation Approach. CRC Press, Boca Raton.
https://doi.org/10.1201/9781138046412-20

Clarke T, 2014. High-frequency trading and dark pools: sharks never sleep. *Law Finan Mark Rev*, 8(4):342-351.
https://doi.org/10.5235/17521440.8.4.342

Connell JH, 1989. A Colony Architecture for an Artificial Creature. PhD Dissemination, Massachusetts Institute of Technology, Cambridge, USA.

Culik IIK, Hurd LP, Yu S, 1990. Computation theoretic aspects of cellular automata. *Phys D Nonl Phenom*, 45(1-3):357-378.
https://doi.org/10.1016/0167-2789(90)90194-T

D'Angelo M, Caporuscio M, Napolitano A, 2017. Model-driven engineering of decentralized control in cyber-physical systems. IEEE 2nd Int Workshops on Foundations and Applications of Self* Systems, p.7-12.
https://doi.org/10.1109/FAS-W.2017.113

Fang X, Misra S, Xue GL, et al., 2012. Smart grid—the new and improved power grid: a survey. *IEEE Commun Surv Tutor*, 14(4):944-980.
https://doi.org/10.1109/SURV.2011.101911.00087

Ferris TLJ, 2018. The challenge of performing research which will contribute helpful engineering knowledge concerning emergence. In: Rainey LB, Jamshidi M (Eds.), Engineering Emergence: a Modeling and Simulation Approach. CRC Press, Boca Raton.
https://doi.org/10.1201/9781138046412-14

Gabor T, Belzner L, Kiermeier M, et al., 2016. A simulation-based architecture for smart cyber-physical systems. IEEE Int Conf on Autonomic Computing, p.374-379.
https://doi.org/10.1109/ICAC.2016.29

Gadomska M, Pacut A, 2007. Performance of ant routing algorithms when using TCP. In: Giacobini M (Ed.), Applications of Evolutionary Computing. Springer, Berlin, Heidelberg, p.1-10.
https://doi.org/10.1007/978-3-540-71805-5_1

Glancy DJ, 2013. Sharing the road: smart transportation infrastructure. *Fordham Urban Law J*, 41(5):1617-1664.

GLOBES, 2018. Waze Makes the Jams Worse. Newspaper. https://en.globes.co.il/en/article-waze-makes-the-jams-worse-1001248309 [Accessed on Sept. 13, 2020].

Griffor ER, Greer C, Wollman DA, et al., 2017. Framework for Cyber-Physical Systems: Volume 1, Overview. Technical Report, No. 1500-201. National Institute of Standards and Technology, USA.

Gunderson LH, 2000. Ecological resilience—in theory and application. *Ann Rev Ecol Syst*, 31(1):425-439.
https://doi.org/10.1146/annurev.ecolsys.31.1.425

Haber A, Rendel H, Rumpe B, et al., 2011. Delta modeling for software architectures. Modellbasierte Entwicklung eingebetteter Systeme, p.1-10.

Holland JH, 1992. Complex adaptive systems. *Daedalus*, 121(1):17-30.

Holland JH, 1999. Emergence: from Chaos to Order. Perseus Publishing.

Holland JH, 2006. Studying complex adaptive systems. *J Syst Sci Compl*, 19(1):1-8.
https://doi.org/10.1007/s11424-006-0001-z

Holland T, 2018. Foundations for the modeling and simulation of emergent behavior systems. In: Rainey LB, Jamshidi M (Eds.), Engineering Emergence: a Modeling and Simulation Approach. CRC Press, Boca Raton.
https://doi.org/10.1201/9781138046412-10

Hu F, 2013. Cyber-Physical Systems: Integrated Computing and Engineering Design. CRC Press, Boca Raton.
https://doi.org/10.1201/b15552

Johnson JJ, Padilla JJ, 2018. An ontology of emergence. In: Rainey LB, Jamshidi M (Eds.), Engineering Emergence: a Modeling and Simulation Approach. CRC Press, Boca Raton. https://doi.org/10.1201/9781138046412-8

Johnson JJ, Padilla JJ, Sousa-Poza A, 2018. Modeling emergence in systems of systems using thermodynamic concepts. In: Mittal S, Diallo S, Tolk A (Eds.), Emergent Behavior in Complex Systems Engineering: a Modeling and Simulation Approach. John Wiley & Sons, Inc.
https://doi.org/10.1002/9781119378952.ch8

Jones JC, Oldroyd BP, 2006. Nest thermoregulation in social insects. *Adv Insect Physiol*, 33:153-191.
https://doi.org/10.1016/S0065-2806(06)33003-2

Jun H, Liu ZM, Reed GM, et al., 2008. Ensemble engineering and emergence. In: Wirsing M, Banâtre JP, Hölzl M, et al. (Eds.), Software-Intensive Systems and New Computing Paradigms: Challenges and Visions. Springer, Berlin, Heidelberg, p.162-178.
https://doi.org/10.1007/978-3-540-89437-7_11

Keating CB, Padilla JJ, Adams K, 2008. System of systems engineering requirements: challenges and guidelines. *Eng Manag J*, 20(4):24-31.
https://doi.org/10.1080/10429247.2008.11431785

Khuong A, Gautrais J, Perna A, et al., 2016. Stigmergic construction and topochemical information shape ant nest architecture. *Proc Natl Acad Sci USA*, 113(5):1303-1308. https://doi.org/10.1073/pnas.1509829113

Kondepudi D, Prigogine I, 2014. Modern Thermodynamics: from Heat Engines to Dissipative Structures (2nd Ed.). John Wiley & Sons, New York.
https://doi.org/10.1002/9781118698723

Kopetz H, Bondavalli A, Brancati F, et al., 2016. Emergence in cyber-physical systems-of-systems (CPSoSs). In: Bondavalli A, Bouchenak S, Kopetz H (Eds.), Cyber-Physical Systems of Systems. Springer, Cham, p.73-96. https://doi.org/10.1007/978-3-319-47590-5_3

Korb J, 2003. Thermoregulation and ventilation of termite mounds. *Naturwissenschaften*, 90(5):212-219. https://doi.org/10.1007/s00114-002-0401-4

Koutsoukos X, Karsai G, Laszka A, et al., 2018. SURE: a modeling and simulation integration platform for evaluation of secure and resilient cyber-physical systems. *Proc IEEE*, 106(1):93-112. https://doi.org/10.1109/JPROC.2017.2731741

Ledyard JO, Porter D, Rangel A, 1994. Using computerized exchange systems to solve an allocation problem in project management. *J Organ Comput*, 4(3):271-296. https://doi.org/10.1080/10919399409540228

Lee EA, 2015. The past, present and future of cyber-physical systems: a focus on models. *Sensors*, 15(3):4837-4869. https://doi.org/10.3390/s150304837

Lenglet M, 2011. Conflicting codes and codings: how algorithmic trading is reshaping financial regulation. *Theory Cult Soc*, 28(6):44-66. https://doi.org/10.1177/0263276411417444

Liu C, Zhang QP, Zhang X, 2009. Emergence and disappearance of traffic congestion in weight-evolving networks. *Simul Modell Pract Theory*, 17(10):1566-1574. https://doi.org/10.1016/j.simpat.2009.06.001

Macchi M, Roda I, Negri E, et al., 2018. Exploring the role of digital twin for asset lifecycle management. *IFAC-PapersOnLine*, 51(11):790-795. https://doi.org/10.1016/j.ifacol.2018.08.415

Maier MW, 1996. Architecting principles for systems-of-systems. *INCOSE Int Symp*, 6(1):565-573. https://doi.org/10.1002/j.2334-5837.1996.tb02054.x

Maier MW, 2015. The role of modeling and simulation in system of systems development. In: Rainey LB, Tolk A (Eds.), Modeling and Simulation Support for System of Systems Engineering Applications. Wiley & Sons, Hoboken. https://doi.org/10.1002/9781118501757.ch2

Maier MW, Rechtin E, 1997. The Art of Systems Architecting. CRC Press, Inc.

Mittal S, Diallo S, Tolk A, 2018. Emergent Behavior in Complex Systems Engineering: a Modeling and Simulation Approach. John Wiley & Sons, Inc. https://doi.org/10.1002/9781119378952

Monostori L, 2018. Cyber-physical systems. In: Chatti S, Tolio T (Eds.), CIRP Encyclopedia of Production Engineering. Springer, Berlin. https://doi.org/10.1007/978-3-642-35950-7_16790-1

Mosterman PJ, Zander J, 2016. Cyber-physical systems challenges: a needs analysis for collaborating embedded software systems. *Softw Syst Model*, 15(1):5-16. https://doi.org/10.1007/s10270-015-0469-x

Ncube C, Lim SL, Dogan H, 2013. Identifying top challenges for international research on requirements engineering for systems of systems engineering. 21st IEEE Int Requirements Engineering Conf, p.342-344. https://doi.org/10.1109/RE.2013.6636746

Neema H, Sztipanovits J, Steinbrink C, et al., 2019. Simulation integration platforms for cyber-physical systems. Proc Workshop on Design Automation for CPS and IoT, p.10-19. https://doi.org/10.1145/3313151.3313169

Nisan N, Roughgarden T, Tardos É, et al., 2007. Algorithmic Game Theory. Cambridge University Press, Cambridge. https://doi.org/10.1017/CBO9780511800481

Novak M, Wilensky U, 2006. Netlogo Daisyworld Model (Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL). http://ccl.northwestern.edu/netlogo/models/Daisyworld

Nwogugu MIC, 2016. Illegal Online File Sharing, Decision-Analysis, and the Pricing of Digital Goods. CRC Press, Boca Raton. https://doi.org/10.1201/9781315383149

Perez-Palacin D, Mirandola R, 2014. Uncertainties in the modeling of self-adaptive systems: a taxonomy and an example of availability evaluation. Proc 5th ACM/SPEC Int Conf on Performance Engineering, p.3-14. https://doi.org/10.1145/2568088.2568095

Peterson G, Allen CR, Holling CS, 1998. Ecological resilience, biodiversity, and scale. *Ecosystems*, 1(1):6-18. https://doi.org/10.1007/s100219900002

Petty MD, 2018. Modeling and validation challenges for complex systems. In: Rainey LB, Jamshidi M (Eds.), Engineering Emergence: a Modeling and Simulation Approach. CRC Press, Boca Raton. https://doi.org/10.1201/9781138046412-9

Pierce JR, 2012. An Introduction to Information Theory: Symbols, Signals and Noise. Dover Publications.

Pullum LL, 2001. Software Fault Tolerance Techniques and Implementation. Artech House, Inc., Norwood, USA.

Rendell P, 2002. Turing universality of the game of life. In: Adamatzky A (Ed.), Collision-Based Computing. Springer, London, p.513-539. https://doi.org/10.1007/978-1-4471-0129-1_18

Salem T, 2018. Why Some Cities Have Had Enough of Waze. Newspaper. https://www.usnews.com/news/national-news/articles/2018-05-07/why-some-cities-have-had-enough-of-waze [Accessed on Sept. 13, 2020].

Sanfelice RG, 2015. Analysis and design of cyber-physical systems: a hybrid control systems approach. In: Rawat DB, Rodrigues JJPC, Stojmenovic I (Eds.), Cyber-Physical Systems: from Theory to Practice. CRC Press, p.3-13.

Sawyer RK, 2005. Social Emergence: Societies as Complex Systems. Cambridge University Press, Cambridge. https://doi.org/10.1007/s11577-007-0008-z

Schaff J, 2018. Leveraging deterministic chaos to mitigate combinatorial explosions. In: Rainey LB, Jamshidi M (Eds.), Engineering Emergence: a Modeling and Simulation Approach. CRC Press, Boca Raton. https://doi.org/10.1201/9781138046412-5

Smith A, 2002. An Inquiry into the Nature and Causes of the Wealth of Nations. In: Bullock CJ (Ed.), Wealth of Nations. Cosimo Classics. https://www.amazon.com/Wealth-Nations-Cosimo-Classics-Economics/dp/1602069409/ref=sr_1_1?dchild=1&keywords=wealth+of+nations+cosimo&qid=1604936887&sr=8-1

Sztipanovits J, 2007. Composition of cyber-physical systems. Proc 14th Annual IEEE Int Conf and Workshops on the Engineering of Computer, p.3-6. https://doi.org/10.1109/ECBS.2007.25

Tan J, Wen HJ, Awad N, 2005. Health care and services delivery systems as complex adaptive systems. *Commun ACM*, 48(5):36-44. https://doi.org/10.1145/1060710.1060737

Tannenbaum C, Sheehan NL, 2014. Understanding and preventing drug-drug and drug-gene interactions. *Exp Rev Clin Pharmacol*, 7(4):533-544. https://doi.org/10.1586/17512433.2014.910111

Tesfatsion L, 2002. Economic agents and markets as emergent phenomena. *Proc Natl Acad Sci USA*, 99(S3):7191-7192. https://doi.org/10.1073/pnas.072079199

Theraulaz G, Bonabeau E, 1999. A brief history of stigmergy. *Artif Life*, 5(2):97-116. https://doi.org/10.1162/106454699568700

Törngren M, Grogan PT, 2018. How to deal with the complexity of future cyber-physical systems? *Designs*, 2(4):40. https://doi.org/10.3390/designs2040040

Törngren M, Sellgren U, 2018. Complexity challenges in development of cyber-physical systems. In: Lohstroh M, Derler P, Sirjani M (Eds.), Principles of Modeling, p.478-503. https://doi.org/10.1007/978-3-319-95246-8_27

Tripakis S, 2016. Compositional model-based system design and other foundations for mastering change. In: Steffen B (Ed.), Transactions on Foundations for Mastering Change I. Springer, Cham, p.113-129. https://doi.org/10.1007/978-3-319-46508-1_7

Tucker W, 1999. The Lorenz attractor exists. *Compt Rend l'Acad Sci-Ser I-Math*, 328(12):1197-1202. https://doi.org/10.1016/S0764-4442(99)80439-X

Vásárhelyi G, Virágh C, Somorjai G, et al., 2018. Optimized flocking of autonomous drones in confined environments. *Sci Robot*, 3(20):eaat3536. https://doi.org/10.1126/scirobotics.aat3536

Wassyng A, Joannou P, Lawford M, et al., 2016. New standards for trustworthy cyber-physical systems. In: Romanovsky A, Ishikawa F (Eds.), Trustworthy Cyber-Physical Systems Engineering. CRC Press, New York. https://doi.org/10.1201/9781315369105

Welch PH, Wallnau K, Sampson AT, et al., 2012. To boldly go: an occam-π mission to engineer emergence. *Nat Comput*, 11(3):449-474. https://doi.org/10.1007/s11047-012-9304-2

Wildman WJ, Shults FL, 2018. Emergence: what does it mean and how is it relevant to computer engineering? In: Mittal S, Diallo S, Tolk A (Eds.), Emergent Behavior in Complex Systems Engineering: a Modeling and Simulation Approach. Wiley & Sons, p.21-34. https://doi.org/10.1002/9781119378952

Wilensky U, Reisman K, 2006. Thinking like a wolf, a sheep, or a firefly: learning biology through constructing and testing computational theories—an embodied modeling approach. *Cogn Instr*, 24(2):171-209. https://doi.org/10.1207/s1532690xci2402_1

Williams GP, 1997. Chaos Theory Tamed. Joseph Henry Press, Washington.

Zeigler BP, 2018. DEVS-based modeling and simulation framework for emergence in system of systems. In: Rainey LB, Jamshidi M (Eds.), Engineering Emergence: a Modeling and Simulation Approach. CRC Press, Boca Raton. https://doi.org/10.1201/9781138046412-3

Zhang LC, He JF, 2011. Aspect-oriented QoS specification for cyber-physical systems. Proc 5th Int Conf on Convergence and Hybrid Information Technology, p.399-406. https://doi.org/10.1007/978-3-642-24106-2_51

Zurita NFS, Tumer IY, 2017. A survey: towards understanding emergent behavior in complex engineered systems. ASME Int Design Engineering Technical Conf and Computers and Information in Engineering Conf. https://doi.org/10.1115/DETC2017-67453