Self-improving System Integration – Status and Challenges After Five Years of SISSY

Kirstie Bellman*, Jean Botev‡, Ada Diaconescu†, Lukas Esterle¶, Christian Gruhl§, Chris Landauer*,
Peter R. Lewis¶, Anthony Stein**, Sven Tomforde§††, and Rolf P. Würtz‖
*Topcy House Consulting, US, Email: bellmanhome@yahoo.com, topcycal@gmail.com
§Intelligent Embedded Systems, University of Kassel, Germany, Email: cgruhl@uni-kassel.de
‡University of Luxembourg, Email: jean.botev@uni.lu
†Telecom-Paris Tech, France, Email: ada.diaconescu@telecom-paristech.fr
¶Aston University, Birmingham, UK, Email: p.lewis@aston.ac.uk, l.esterle@aston.ac.uk
‖Ruhr Universität Bochum, Germany, Email: rolf.wuertz@ini.rub.de
**University of Augsburg, Germany, Email: anthony.stein@informatik.uni-augsburg.de
††University of Passau, Germany, Email: sven.tomforde@uni-passau.de

Abstract—**The self-improving system integration (SISSY) initiative has emerged in recent years in response to a systems engineering trend towards the organisation of open, interconnected systems integrating a large set of heterogeneous and autonomous subsystems. Based on the idea to equip subsystems with capabilities to assess and maintain their own integration status within the overall system composition, a variety of concepts, techniques, and contributions have been proposed and fruitfully discussed at the particular events of the underlying workshop series. In this article, we summarise and categorise these research efforts and derive a roadmap towards full-scale SISSY systems.**

I. Introduction

Information and communication technology (ICT) pervades every aspect of our daily lives. This inclusion changes our communities and all of our human interactions. It also presents a significant set of challenges to correctly designing and integrating our resulting technical systems. For instance, the embedding of ICT functionality in more and more devices (such as household appliances or thermostats) leads to novel interconnections and a changing structure of the overall system. Not only technical systems are increasingly coupled, a variety of previously isolated natural and human systems have consolidated into a kind of overall system of systems – an interwoven system structure [1].

This change of structure is fundamental and affects the entire production cycle of technical systems – standard system integration and testing is not feasible any more because it does not cope with the dynamic changes in a relatively open system of goals, behaviours, relationships, and even participating systems. Also, unlike many current complex systems, the integrating systems do not have any or little control or authority over each other, and indeed, there may be no central authority or control for most of the behaviours or goals of the system. There may be some standards, rules, or governing body for certain aspects of the system, but like the Internet, many goals and behaviours may be brought in an unregulated manner by participating users and systems. Furthermore, there may be limited or no knowledge of other participating systems available to other systems, as often is the case in the classical engineering of complex System of Systems (SoS).

The increasingly complex challenges of developing the right type of modelling, analysis, and infrastructure for designing and maintaining ICT infrastructures has continued to motivate research in the self-organising systems, Autonomic and Organic Computing systems communities – resulting in the vision of self-improving system integration (SISSY). The goal of the SISSY initiative is to study novel approaches to system of system integration and testing by applying self-* principles; specifically, approaches are investigated that allow for a continual process of self-integration among components and systems that is self-improving and evolving over time towards an optimised and stable solution.

Although research in self-organising systems – such as the Organic Computing (OC) [2] and Autonomic Computing (AC) [3] initiatives – has seen an exciting decade of development with considerable success in building individual systems, OC/AC is faced with the difficult challenge of integrating multiple self-organising systems, and integrating self-organising systems with traditionally engineered ones as well as naturally occurring human organisations. Meanwhile, although there has been important progress in system-of-systems methodologies (e.g. Service-oriented Architectures [4] and cloud technology [5]), many of these developments lack scalable methods for rapidly proving that new configurations of components/subsystems are correctly used or their changes verified or that these frameworks have pulled together the best possible context-sensitive configuration of resources for some purpose of a user or another system.

This article summarises the research activities in the context of SISSY over the last five years and aims at consolidating them into a preliminary blueprint for SISSY systems. We therefore categorise the different contributions from the past events of the SISSY workshop – 2014 at IEEE International Conference on Self-Adaptive and Self-Organising Systems (SASO14) [1] in London, UK, 2015 at IEEE/ACM Interna-

978-1-5386-5175-9/18/$31.00 ©2018 IEEE
DOI 10.1109/FAS-W.2018.00042

tional Conference on Autonomic Computing (ICAC15) [6] 2015 in Grenoble, France, 2016 at IEEE/ACM International Conference on Autonomic Computing (ICAC16) [7] 2016 in Würzburg, Germany, and 2017 at IEEE International Conference on Self-Adaptive and Self-Organising Systems (SASO17) [8] in Tuscon, US. We further take related contributions from the field into consideration to broaden the scope of the discussed research activities. Finally, we use the consolidated blueprint as a basis for deriving a research roadmap for SISSY by identifying the most urgent fields where insights are needed to close the gaps in SISSY technology.

The remainder of this article is organised as follows: Section II summarises the various contributions in the field of self-improving system integration from the previous workshops and categorises those research efforts. We consolidate these efforts into an aggregated blueprint for SISSY systems in Section III. Afterwards, Section IV identifies open issues and derives research challenges. Finally, Section V summarises the article and gives an outlook to future work.

II. Research Directions in SISSY

As outlined above, the SISSY initiative has resulted in a set of workshops. In the rest of this section, we categorise the contributions from these workshops from problem description to design concepts and to classes of techniques for realising SISSY technology. In particular, we propose the following categories of research: a) aspects of the underlying problem and a definition of SISSY, b) design concepts and architectures for systems with SISSY capabilities, c) collaboration mechanisms and concepts for building collective systems, d) concepts and techniques for autonomous learning and self-awareness, e) concepts and techniques for promoting computational trust and security, and f) applications of SISSY concepts.

A. Problem Description

A first set of contributions aimed at defining the class of systems that need novel technology from the SISSY domain to be able to appropriately engineer, master, and maintain them. As a result, the class of "Interwoven Systems" (IwS) [9], [10] has been defined with its various characteristics.

Based on Maier's definition of the term "System of Systems" (SoS) [11], IwS describes an SoS structure with specific properties. For instance, we typically find aspects such as heterogeneity of contained subsystems, real-time demands, (partly hidden) mutual influences among actions and performance of subsystems, or uncertainty of environment and observations, see [10].

In addition to this identification and definition of the class of IwS, contributions described specific examples for such systems, see [12] and [1]. Furthermore, it has been claimed [13] that, in order to rapidly learn to integrate and align with both humans and other technical systems in a changing environment, SISSY mechanisms and design concepts should embody socially-sensitive properties [14].

B. Design Concepts and Architectures

The underlying design concept provides the fundamental basis of realising SISSY capabilities in large-scale system organisations. Consequently, some of the contributions proposed novel architectures or variants of existing approaches.

In [12], the authors adapted the Observer/Controller framework [15] and the corresponding design process [16] as known from the Organic Computing domain [2]. The responsibilities of the components has been kept – but the concept has been extended by means of covering aspects such as detection of mutual influences and computational trust. Furthermore, the need of mechanisms for self-explanation of self-integration behaviour has been highlighted [12].

Also with roots in the ideas of OC, [17] presented a goal-oriented holonic architecture. The basic idea is that all systems provide the same kind of interfaces for status description and goal manipulation – and the mechanisms to realise the large-scale system composition are mainly based on conflict resolution for low-level goal conflicts and contradictory demands. Here, a particularly relevant capability is to plan and to adapt these plans continuously to changing requirements and dynamics. [18] described a concept and the resulting challenges for planning models in the context of self-modelling as performed by SISSY systems. This is accompanied by a process planning and self-improvement component especially dedicated to cyber-physical systems and their challenges [19].

As already visible in the description above, a key challenge for SISSY systems is to allow for an efficient and goal-oriented collaboration among the distributed heterogeneous subsystems. In this context, contributions dealt with the question of how to design cooperating self-improving systems [20] and how to realise a model-based cooperative system integration [21]. Furthermore, a design approach for a reflective service for SoS integration has been presented [22]. This has to be accompanied by ensuring appropriate use of computational resources, which has been discussed in [23].

A key component in all discussed architectural concepts is responsible for deciding about adaptations of the integration status. In this context, [24] provided an overview and an analysis of possible decision-making techniques suitable for this task. However, the decision horizon goes far beyond a purely reactive adaptation to changing observations. In contrast, [25] and [26] highlighted that such a self-improvement process has to be based on a long-term decision horizon in the sense that planing is accompanied by self-reflection capabilities of the SISSY systems. This, in turn, requires a collective approach and consequently has implications of the design of these systems – [26] presented an adapted variant of the Observer/Controller concept.

Finally, a fundamentally different way to design SISSY systems has been presented in [27]. Here, the idea is to establish a system-wide artificial DNA concept that is responsible for self-building embedded systems. Conceptually, this implies a design process that makes heavy use of modularisation and re-usage of components and code fragments.

C. Collaboration Mechanisms and Social Integration

As already outlined by the IwS design as extension of the Systems-of-Systems concept [11], the main idea is that subsystems do not act in an isolated manner any more. In contrast, collaboration and cooperation among the distributed entities is required for self-integration and self-improvement purposes. Several contributions picked up some important issues in this context.

For instance, [28] proposed a concept for managing a posteriori federations of self-integrating subsystems. This is closely related to mechanisms for negotiating behaviour restrictions as proposed by [29]. The reflective service, noted above, proposes to fill the gap between the goals of the SoS and that of a participating system; it assigns reflective agents to each new participating system and it assesses the incoming system's capabilities and behaviour through communication, observation and active experimentation and matches the system with the current needs or goals of the overall SoS.

Furthermore, researchers have argued that such a complex open system structure demands for novel concepts of governance – we should consider the overall system as a kind of digital society that requires complex control mechanisms like those available in human organisations [30].

D. Learning

In several papers, the desired capability of "learning" has been investigated. A continuous assessment and adaptation of the integration status as well as the need to self-improve this integration status requires that the autonomous systems are able to generate novel knowledge at runtime. Due to the sheer size of the possible input space, design-time knowledge is far from being sufficient and needs to be supported by mechanisms to derive insights from observations and experiences. In [31], a summary and in-depth comparison of approaches for self-improvement in Self-adaptive systems has been presented.

As a basis for realising these capabilities, specific techniques from the domain of machine learning have been proposed, including reinforcement learning, ensemble learning, and probabilistic generative modelling. One contribution presented a basic concept for self-improving autonomous subsystems and how these subsystems can dynamically learn the most appropriate action within a multi-agent constellation [32]. However, this mostly focused on the individual system and its own isolated adaptation and action-based learning behaviour. Furthermore, questions regarding the process of gathering knowledge at runtime and from various knowledge sources with different characteristics have been discussed.

For instance, in [33], the authors described a concept for autonomously selecting knowledge sources at runtime and combining different learning paradigms that are closely connected to these various ways to integrate externally available knowledge. This extends work in [34] – there, the authors claim that collaborative interactive learning [35] mechanisms are needed for future smart technical systems in open environments. In particular, lifelong learning needs the freedom to identify and integrate novel sources of information or knowledge at all abstraction layers that range from individual human users to crowd sourcing technology and to highly uncertain sources such as the Internet.

In [36] and [37] multi-sensor systems have been investigated with an application in activity recognition. Here, probabilistic, generative models have been used for self-adaptation of sensor constellations – the integration status of sensors self-improves by learning.

In contrast, [38] described an approach to map learning problems to a standardised model base that contains representations of problem instances. By calculating a ranked list of similarities to these representatives the system can appropriately learn its generalisation capability, which is highly important for dealing with unknown observations in SISSY systems. Making use of related learning techniques, [39] focused on learning of system discovery mechanisms through an analog of neocortical architectures and experimental play.

Finally, ensemble methods have been investigated. Because anticipatory integration control requires forecasting of upcoming conditions and behaviour of the individual system itself as well as those in its vicinity, the authors of [40] proposed to make use of the different advantages of various techniques for the same purpose. They propose to additionally learn at a meta-level by means of deriving the best combination of (learning) techniques at runtime.

E. Computational Trust and Security

As outlined above, IwSs are typically open by design – autonomous systems are free to join and leave the overall system at any time. This inherently implies that subsystems are unknown and their behaviour can not be anticipated. In particular, this also implies that malicious elements may become part of the system and attempt to exploit it. As a countermeasure, two basic concepts have been discussed in detail: computational trust and security.

In the context of security, compliance with security requirements in multi-tenant applications [41] has been investigated. Furthermore, the need of mutually testing the correct functioning of subsystems has been highlighted in [42] – extending security mechanisms towards a continuous testing solution.

In the context of computational trust, a concept for establishing explicit communities of mutually trusted subsystems (i.e. isolating malicious or faulty subsystems) has been presented [43]. Based on this initial concept, several extensions have been investigated: Mechanisms for measuring trust and reputation [44], advanced attacks on such trust communities and corresponding countermeasures [45], accusation-based strategies to identify misbehaving subsystems and to establish forgiveness solutions [46], and robust self-monitoring mechanisms at runtime [47]. Furthermore, [48] demonstrated that computational trust and forgiveness techniques can result in improved reliability and reduced overhead.

F. Applications

The different approaches have been discussed and analysed in the context of different applications. As one example for

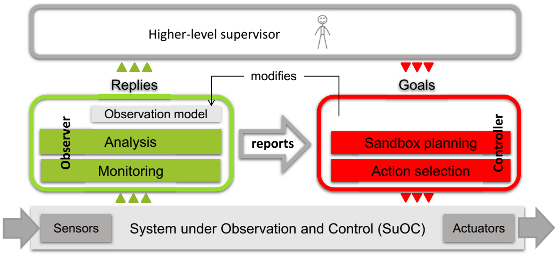

Fig. 1. Conceptual view of an individual SISSY system.

open systems with unknown cooperation partners, researchers have focused on the computational architecture for a volunteer grid system [47], [45], [46], [49]. There, agents provision and utilise computing resources of other agents. Based on this grid scenario, an extension for distributed image rendering has been part of the focus of research [49].

Based on the same techniques, protocols for robust communication in wireless sensor networks have been presented, see [48]. Closely related to this is work in the field of cloud computing, where provisioning and integration of computing resources is done using an anticipatory approach [50].

In [51], embedded many-core computing systems served as basic application scenario, where self-integration is done in the context of virtualisation. [38] presented computer vision as a domain where SISSY methods are beneficial. In contrast, [37] used embedded devices such as smartphones for activity recognition and developed novel methods for self-improving the corresponding classification systems.

Besides these basic ICT scenarios, several further application domains have been targeted. In [52], highly autonomous surgical training systems have been presented. [40] used urban traffic control and management (i.e. progressive signal systems and route recommendations) as examples for SISSY technology. Finally, [12] discussed smart grids, large-scale data centres, and traffic management as promising examples, concluding that especially applications in the context of interconnected infrastructure with strong ICT functionality have a high demand for SISSY technology.

III. A Blueprint for SISSY Systems

Figure 1 combines the properties of the OC-based view as defined in [12], [26] with those of the holonic system composition and structuring as proposed in [17], [2]. In particular, we assume a system element that is enriched with an Observer/Controller tandem responsible for assessing and maintaining the element's integration status.

In comparison to the standard OC design concept, each SISSY element possesses mechanisms for the following:

- **Mutual influences**: A major challenge of IwSs is the existence of other systems with direct and indirect influence on the system's status and performance. In order to be able to deal with such influences and dependencies, they have to be detected in the first place. Based on knowledge about available neighbours (i.e. those systems that might have a certain influence or impact on the system of interest), techniques to find correlations in behaviour and mechanisms to detect hidden effects among these systems and their behaviours are needed. An example of such techniques can be found in [53].
- **Emergence detection:** Systems that are based on self-organised behaviour are likely to have emergent effects. Since self-integration makes heavy use of self-organisation, emergence has to be considered using techniques such as those presented in [54]. This requires the creation and use of new language and representational mechanisms to describe new phenomena that emerge.
- **Self-reflection:** Critically assessing the system's own knowledge and experiences is deemed a crucial capability of SISSY systems. On the one hand, the system should be able to assess its own knowledge base and models. On the other hand, the system may also need to assess and model the knowledge, viewpoint, and models of other SISSY elements. Potentially, this can be done through a combination of communication, observation with reasoning, and the active experimentation as discussed in [55]. If a SISSY system is able to detect *knowledge gaps*, i.e., regions of the system's state space that are not covered at all, or only with knowledge of insufficient quality, within its knowledge base (cf. [56]), it is, in a subsequent step, also able to *proactively* 'bridge these gaps' by initiating appropriate countermeasures. Therefore, a variety of so-called oracles with different degrees of reliability and availability can be incorporated, as proposed in [33]. Another possibility to overcome identified knowledge gaps would be to *interpolate* missing information/knowledge from already existing elements in the systems' original knowledge bases, as proposed in [57]. Naturally, other SISSY elements can be requested to provide support at different abstraction levels, e.g., by providing sensor information, experiences, or abstracted knowledge. Here a collective approach is necessary based on exchange mechanisms for those kinds of information [25], [26].

Conceptually, we assume that each self-integrating system element is connected to other system elements using standard communication networks. In addition, a SISSY overlay network is formed autonomously by the SISSY elements, and this overlay network reflects the current system organisation and realises the necessary middleware functionality. As a result, the overall system composition distinguishes between three system-wide layers: 1) the underlying physical network, 2) the overlay network connecting the individual SISSY systems, and 3) the individual system layer. Figure 2 illustrates the resulting layered system organisation.

The middleware solution is serving as a communication platform augmented with a set of basic services running in automated and distributed mode, i.e., without the need for external control and intervention. These basic services include the following techniques to be mutually available:

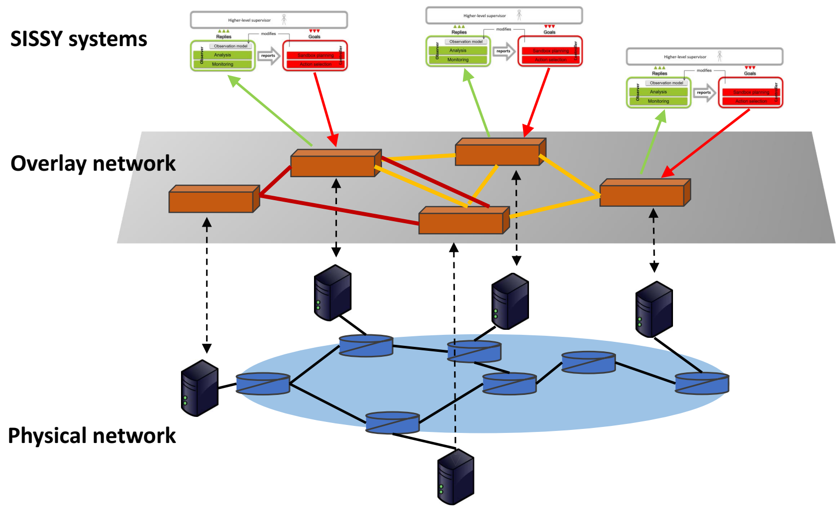

Fig. 2. Overall system structure containing several interconnected SISSY systems.

- **Neighbour discovery:** Mutual influence and emergence describe two phenomena that may result from the missing awareness of others' influence. A prerequisite for such an awareness is initial information about available other systems that join and leave. It is important to note that neighbourhood has a looser definition than that usually applied to geographically or ordered "nearness–farness" dimensions. The neighbourhood here has an emphasis on relevant or possible impacts (desirable or undesirable), even though such a determination is often formally undecideable. Hence, the middleware has to provide a neighbour discovery mechanism that updates a (local) neighbour cache. In order to allow for such a mechanism, existing techniques can be applied, either simply by using broadcast procedures or by more advanced neighbour discovery routines from the data networking domain (especially from mobile ad-hoc networks [58]).
- **Service discovery based on capability description:** Based on e.g. ontology-driven self-description of capabilities and status, a service discovery routine is needed to detect relevant interaction partners among the available SISSY elements. Appropriate and adaptive resource discovery is a key part of the approach to the reflective systems approach in [59]. Of course, with emergent behaviors, the ontologies will change, and the terms of cooperation among the systems will also have to change. This level of language creation and sharing has not been adequately studied.
- **Computational trust:** Trust and reliability information are created based on one's own experiences in the first place. In addition, distributed entities need to exchange information to be able to deal with unknown elements or those with limited experiences. Therefore, technical trust and reliability estimations need to be established in a distributed manner, potentially augmented by a reputation system. These assessments can be derived using existing work, see e.g. [43], [44].

The previously described prototypical SISSY element and its position within the overall system composition reflects our view on the state of the art within the last years of research. We use this as basis for analysing urgent research challenges in the next section. However, this is not meant to cover the entire spectrum of research efforts in the SISSY domain and probably leaves some of the current developments out of focus.

IV. Research Perspectives

In order to finally bring SISSY systems into operation, we need to address many research issues. The following section aims at presenting a roadmap for SISSY systems that characterises the required efforts in different paths that we have to follow.

As we noted at the beginning, the key SISSY challenge is to do continual integration improvement and optimisation *without* full knowledge and *without* full control or authority over the other SISSY systems. Below we start to unpack some of the research implications of this challenge.

A. Interactions benefit and not harm participating systems

This leads to another part of this formidable challenge which is what are the language processes and modelling capabilities we need to create in order to allow a SISSY system to make "first contact" with new systems, new components, and new situations. What does it need to model, negotiate and discover potential benefits and pitfalls of working with another system? Importantly, how does it build up or bootstrap its existing knowledge in order to decide on its level of integration with another system? This last point leads directly to another critical implication of the SISSY challenge.

B. Building up knowledge is critical

Above we noted some of the research challenges in building up knowledge. One was approaches for learning about the environment, which includes other entities who could be more or less smart through active experimentation and play, as well as the need for new types of negotiation and communication [60].

C. Modelling

A key aspect of research is to come up with a SISSY-tailored *system model* that captures and integrates challenges already well known in related disciplines such as machine learning, optimisation, conflict resolution, service-oriented architectures, etc. Such a unified system model with well-defined subproblems to be solved, paves the way for straightforward application of existing techniques and algorithms from strongly related domains and prevents SISSY research from re-invention of concepts.

D. Definitions, Metrics, and Instrumentation

In classical system engineering, a great deal of time is spent by designers on carefully selecting what components can be interfaced, how they should be constrained in the ways that they are interfaced, and analysing the resulting behaviour at many different levels of the complex system. Gradually, designers have recognised that integration is not a one-time activity and that, especially in complex, interconnected, dynamic and fairly open systems, integration must be an ongoing process. Furthermore, since SISSY systems are also self-adaptive and dynamically changing systems, this means that there are potentially many new goals, new situations to adapt to, and a rapidly changing environment (that includes the other self-aware systems). This also means that integration for a SISSY system, regardless of how good the initial design and integrated SoS was, must somehow be continually re-accomplished by the SISSY system at runtime. There are three major types of critical integration tasks that must be guaranteed by the SISSY system. First, that the systems are able to perform their priority functions with sufficient performance when integrated with other systems. Of course, sufficient means that given a situation and certain agreements a system may be willing to underperform for a specified time. Priority means that what is being performed may change depending on the timing and state of the participating systems. Second, that the systems are not being harmed (they are safe). This again needs to be defined by the systems involved and the context. Third, that the systems can potentially gain benefit from the participation of the other systems – from the integration. Let's look at this last very carefully. Classic integration emphasised to some extent the notion that we need all the component systems for some reason and that integration meant that 1) we could gain all their functionality; 2) without harming the overall system or the other component systems. But with SISSY we are raising the bar; we are defining improvement as in part both being an improvement of the performance or situation of the individual component *and* of gaining new benefits because of the integration.

E. Self-Assessment of Integration Status

Assessing the current state of the integration by each individual subsystem becomes critical as incomplete integration might lead to faulty overall system behaviour. The question arises on how to evaluate such an integration status as *a priori* knowledge might either not be available or out-of-date due to the changing environment and a changing system. For successful integration both, the integrating system and the running system it is integrating into, need to be aware about how to integrate. Since this needs to be established at runtime, there are two fundamental approaches: a white-box approach and a black-box approach. In a white-box approach, both systems have access to their respective interfaces and potentially even their source-codes. Both know mutually how they have to operate in certain conditions. In the black-box approach, one of the systems is not sharing this information and the state of integration needs to be established in another way. The aforementioned experimental play or active experimentation could be one way to ensure the integration of all systems. However, it is vital to each individual component to perform this experimentation in a semi-controlled environment to ensure that the system is safe throughout the integration process. We can also imagine elements that are responsible for continuously monitoring and maintaining the safety of the combined system integration, see [61].

F. Proactive Learning Behaviour

Instead of equipping future SISSY systems with self-adaptation techniques that only react in response to observed system demands, we should think of methodologies that extend such algorithms towards proactive ones explicitly seeking for interesting states where the system was not able to gather sufficient experiences so far. By acquiring knowledge before it is actually needed, we assume that negative effects due to disturbances or other unforeseen situations and intensified by random guessing or simply not responding to change, clearly impacting the overall system utility, can be alleviated. Challenges such as deciding on which direction to explore or judging the usefulness of currently pursued knowledge need to be overcome. Directions of future research regarding this aspect could be, for instance, *Computational Curiosity* [62], *Intrinsically Motivated Learning* [63]. It will also be important to understand how to manage the dynamic knowledge structures to maintain system efficiencies, as in [64].

G. The Social Dimension

Lastly, integration always involves integrating the human in the technical systems. This goes well beyond user interface aspects or reporting out from the system. A human has to be able to trust, to understand, and to control in critical cases the behaviour of the system. Part of the motivation for reflective systems has always been the ability of the system to report out to the human for monitoring what its state, goals, and activities are. In the case of SISSY, we not only need this, but entirely new means of communications and paths into the reasoning and operation of the collective system in order to better predict upcoming behaviours and potential problems. This includes making any dedicatedly engineered socio-technical system or subsystem as, e.g., described in [65], [66], interface accordingly.

V. Outlook

Looking back on five years of discussion of how to build systems that take the burden of integration from the developers we have identified a set of results and further challenges. In the future, these need to be concreted on several levels.

Despite the formidable challenges, there are lots of small useful steps that we can take *now*. The need for continually self-integrating and self-improving systems is more apparent than ever. And with new computational technologies and new self* approaches and applications, we have many promising ways of approaching the SISSY challenges. Even though the challenges are formidable indeed, we end this brief overview of the SISSY work on an optimistic note by pointing out three areas where one can make immediate gains and have the benefits apparent.

1) A SISSY-aware system designer and developer can start now to make a system more compatible to the ideals of SISSY systems that continually improve their internal and external integration and performance by doing a more comprehensive job of explicitly defining the goals for desired integration and improvement within different potential operational environments and system modes. Explicit goals will lead to better defined requirements and specifications. As discussed briefly above, part of this will be a shift in thinking about interfaces and integration in a more dynamic manner and asking questions such as "What is good enough for a particular system function?"; "What is the duration needed for a given integration? What critical behaviours must be preserved under all circumstances?" Designers can also start putting in more of the instrumentation and feedback paths that are so critical to adaptation and improvement.

2) Designers can start to build in reflective processes. One of the lessons learned in the self* community is that a little explicit knowledge and a bit of reasoning and processing of monitored input can go a long way to enabling robust and adaptive behaviours in complex systems. Having explicit goals and tracking on the use of resources means that all of the system's behaviour potentially can be reasoned about and communicated to other entities for coordination, cooperation, integration, and even for coaching and teaching (having an external system view another system and help it do better in some behaviour).

3) Building lots of small monitoring and adaptation mechanisms aggregate into surprisingly powerfully responsive and adaptive system. Again a lesson learned from our community wide work is that it is important to start small and to do small, focused, helpful activities within large complex systems. This after all is very much like biological systems that have thousands of small adaptive mechanisms at many different levels of the system and of many kinds. As these small adaptive methods aggregate, an important next step is to start building modest methods for monitoring the parallel adaptive methods and to integrate them.

References

[1] K. L. Bellman, S. Tomforde, and R. P. Würtz, "Interwoven Systems: Self-Improving Systems Integration," in SASOWS2014 [67], pp. 123–127.

[2] C. Müller-Schloer and S. Tomforde, *Organic Computing – Technical Systems for Survival in the Real World*. Autonomic Systems, Birkhäuser Verlag, October 2017. ISBN: 978-3-319-68476-5.

[3] J. Kephart and D. Chess, "The Vision of Autonomic Computing," *IEEE Computer*, vol. 36, no. 1, pp. 41–50, 2003.

[4] M. P. Papazoglou and W.-J. Van Den Heuvel, "Service oriented architectures: approaches, technologies and research issues," *The VLDB journal*, vol. 16, no. 3, pp. 389–415, 2007.

[5] Q. Zhang, L. Cheng, and R. Boutaba, "Cloud computing: state-of-the-art and research challenges," *Journal of internet services and applications*, vol. 1, no. 1, pp. 7–18, 2010.

[6] P. Lalanda, A. Diaconescu, L. Cherkasova, and S. Kounev, eds., *2015 IEEE International Conference on Autonomic Computing, Grenoble, France, July 7-10, 2015*, IEEE Computer Society, 2015.

[7] K. L. Bellman, S. Tomforde, and R. P. Würtz, "Preface to SISSY 2016," in Kounev *et al.* [68], p. 275.

[8] *2nd IEEE International Workshops on Foundations and Applications of Self* Systems, FAS-STARW@SASO/ICCAC 2017, Tucson, AZ, USA, September 18-22, 2017*, IEEE Computer Society, 2017.

[9] S. Tomforde, J. Hähner, H. Seebach, W. Reif, B. Sick, A. Wacker, and I. Scholtes, "Engineering and Mastering Interwoven Systems," in *ARCS 2014 - 27th International Conference on Architecture of Computing Systems, Workshop Proceedings, February 25-28, 2014, Luebeck, Germany, University of Luebeck, Institute of Computer Engineering*, pp. 1–8, 2014.

[10] S. Tomforde, J. Hähner, and B. Sick, "Interwoven Systems," *Informatik-Spektrum*, vol. 37, no. 5, pp. 483–487, 2014. Aktuelles Schlagwort.

[11] M. W. Maier, "Architecting principles for systems-of-systems," *Systems Engineering*, vol. 1, no. 4, pp. 267–284, 1998.

[12] S. Tomforde, S. Rudolph, K. Bellman, and R. Würtz, "An Organic Computing Perspective on Self-Improving System Interweaving at Runtime," in Kounev *et al.* [68], pp. 276 – 284.

[13] P. R. Lewis, "Self-integrating and self-improving systems must be socially sensitive," in FAS*2017 [8], pp. 148–148.

[14] K. Bellman, J. Botev, H. Hildmann, P. R. Lewis, S. Marsh, J. Pitt, I. Scholtes, and S. Tomforde, "Socially-sensitive systems design," *IEEE Technology & Society Magazine, Special Issue on Social Concepts in Self-Organising Systems*, vol. 36, no. 3, pp. 72–80, 2017.

[15] S. Tomforde, H. Prothmann, J. Branke, J. Hähner, M. Mnif, C. Müller-Schloer, U. Richter, and H. Schmeck, "Observation and Control of Organic Systems," in *Organic Computing – A Paradigm Shift for Complex Systems* (C. Müller-Schloer, H. Schmeck, and T. Ungerer, eds.), Autonomic Systems, pp. 325–338, Birkhäuser Verlag, 2011.

[16] S. Tomforde and C. Müller-Schloer, "Incremental Design of Adaptive Systems," *Journal of Ambient Intelligence and Smart Environments*, vol. 6, pp. 179–198, 2014.

[17] A. Diaconescu, S. Frey, C. Müller-Schloer, J. Pitt, and S. Tomforde, "Goal-oriented Holonics for Complex System (Self-)Integration: Concepts and Case Studies," in *Proceedings of the 10th IEEE International Conference on Self-Adaptive and Self-Organising Systems, held September 12 – 16, 2016 in Augsburg, Germany*, pp. 100–109, IEEE, 2016.

[18] J. Frank, "Reflecting on planning models: A challenge for self-modeling systems," in Lalanda *et al.* [6], pp. 255–260.

[19] C. Landauer and K. Bellman, "Process planning and self-improvement in cyber-physical systems," in SASOWS2014 [67], pp. 144–149.

[20] C. Landauer and K. Bellman, "Designing cooperating self-improving systems," in Kounev *et al.* [68], pp. 273–278.

[21] C. Landauer and K. L. Bellman, "Model-based cooperative system engineering and integration," in Kounev *et al.* [68], pp. 334–341.

[22] K. L. Bellman and C. A. Landauer, "Early work on the brain patch, a reflective service for system of systems integration," in Lalanda *et al.* [6], pp. 249–254.

[23] P. R. Nelson, "Ensuring appropriate use of computational resources in interwoven systems," in FAS*2017 [8], pp. 139–143.

[24] P. Idziak and S. Clarke, "An analysis of decision-making techniques in dynamic, self-adaptive systems," in SASOWS2014 [67], pp. 137–143.

[25] S. Tomforde, J. Hähner, S. von Mammen, C. Gruhl, B. Sick, and K. Geihs, ""Know Thyself" – Computational Self-Reflection in Intelligent Technical Systems," in SASOWS2014 [67], pp. 150–159.

[26] J. Hähner, S. von Mammen, S. Timpf, S. Tomforde, B. Sick, K. Geihs, G. Stumme, G. Hornung, and T. Goeble, "'Know Thyselves – Computational Self-Reflection in Collective Technical Systems," in *Proceedings of the 29th GI/ITG International Conference on Architecture of Computing Systems – ARCS Workshops, held 4 – 7 April 2016 in Nuremberg, Germany, Workshop on Self-Optimisation in Autonomic and Organic Computing Systems (SAOS16)*, pp. 1–8, VDE Verlag, 2016.

[27] U. Brinkschulte, M. Pacher, and B. Betting, "A simulator to validate the concept of artificial dna for self-building embedded systems," in SASOWS2014 [67], pp. 160–169.

[28] C. A. Lee, "Managing a posteriori federations," in FAS*2017 [8], pp. 130–131.

[29] C. Landauer and K. L. Bellman, "Integration by negotiated behavior restrictions," in FAS*2017 [8], pp. 117–121.

[30] J. Pitt and A. Diaconescu, "Structure and governance of communities for the digital society," in Lalanda *et al.* [6], pp. 279–284.

[31] C. Krupitzer, F. M. Roth, M. Pfannemüller, and C. Becker, "Comparison of approaches for self-improvement in self-adaptive systems," in Kounev *et al.* [68], pp. 308–314.

[32] S. Wang, "Towards dynamic epistemic learning of actions for self-improving agents and multi-agent systems," in Kounev *et al.* [68], pp. 292–299.

[33] A. Calma, D. Kottke, B. Sick, and S. Tomforde, "Learning to learn: Dynamic runtime exploitation of various knowledge sources and machine learning paradigms," in FAS*2017 [8], pp. 109–116.

[34] G. Bahle, A. Calma, J. M. Leimeister, P. Lukowicz, S. Oeste-Reiß, T. Reitmaier, A. Schmidt, B. Sick, G. Stumme, and K. A. Zweig, "Lifelong learning and collaboration of smart technical systems in open-ended environments–opportunistic collaborative interactive learning," in Kounev *et al.* [68], pp. 315–324.

[35] B. Sick, S. Oeste-Reiß, A. Schmidt, S. Tomforde, and A. K. Zweig, "Collaborative Interactive Learning (Aktuelles Schlagwort)," *Informatik Spektrum*, vol. 41, no. 1, pp. 1–4, 2018.

[36] M. Jänicke, B. Sick, P. Lukowicz, and D. Bannach, "Self-adapting multi-sensor systems: A concept for self-improvement and self-healing techniques," in SASOWS2014 [67], pp. 128–136.

[37] M. Jänicke, S. Tomforde, and B. Sick, "Towards self-improving activity recognition systems based on probabilistic, generative models," in Kounev *et al.* [68], pp. 285–291.

[38] R. Grieben and R. P. Würtz, "Cue integration by similarity rank list coding-application to invariant object recognition," in FAS*2017 [8], pp. 132–137.

[39] J. S. Everist, "System discovery through neocortical architectures and experimental play," in FAS*2017 [8], pp. 138–138.

[40] M. Sommer, S. Tomforde, J. Hahner, and D. Auer, "Learning a dynamic re-combination strategy of forecast techniques at runtime," in Kounev *et al.* [68], pp. 261–266.

[41] S. Alqahtani, X. He, and R. Gamble, "Adapting compliance of security requirements in multi-tenant applications," in FAS*2017 [8], pp. 122–129.

[42] H. Heck, S. Rudolph, C. Gruhl, A. Wacker, J. Hähner, B. Sick, and S. Tomforde, "Towards autonomous self-tests at runtime," in *2016 IEEE 1st International Workshops on Foundations and Applications of Self* Systems (FAS*W), Augsburg, Germany, September 12-16, 2016*, pp. 98–99, 2016.

[43] S. Edenhofer, S. Tomforde, J. Kantert, L. Klejnowski, Y. Bernard, J. Hähner, and C. Müller-Schloer, "Trust communities: An open, self-organised social infrastructure of autonomous agents," in *Trustworthy Open Self-Organising Systems* (W. Reif, G. Anders, H. Seebach, J.-P. Steghöfer, E. André, J. Hähner, C. Müller-Schloer, and T. Ungerer, eds.), pp. 127–152, Springer, 2015.

[44] J. Kantert, S. Edenhofer, S. Tomforde, and C. Müller-Schloer, "Representation of trust and reputation in self-managed computing systems," in *13th IEEE International Conference on Dependable, Autonomic and Secure Computing, DASC 2015, Liverpool, UK, October 26-28, 2015*, pp. 1827–1834, 2015.

[45] S. Edenhofer, J. Kantert, L. Klejnowski, S. Tomforde, J. Hähner, and C. Müller-Schloer, "Advanced attacks to trusted communities in multi-agent systems," in SASOWS2014 [67], pp. 186–191.

[46] S. Edenhofer, C. Stifter, U. Jänen, J. Kantert, S. Tomforde, J. Hähner, and C. Müller-Schloer, "An Accusation-based Strategy to Handle Undesirable Behaviour in Multi-Agent Systems," in Lalanda *et al.* [6], pp. 243 – 248.

[47] J. Kantert, S. Edenhofer, S. Tomforde, J. Hähner, and C. Müller-Schloer, "Robust Self-Monitoring in Trusted Desktop Grid for Self-Configuration at Runtime," in SASOWS2014 [67], pp. 178 – 185. DOI 10.1109/SASOW.2014.28.

[48] J. Kantert, S. Tomforde, G. von Zengen, S. Weber, L. Wolf, and C. Müller-Schloer, "Improving Reliability and Reducing Overhead in Low-Power Sensor Networks using Trust and Foregiveness," in Kounev *et al.* [68], pp. 325–333.

[49] J. Kantert, H. Spiegelberg, S. Tomforde, J. Hähner, and C. Müller-Schloer, "Distributed Rendering in an Open Self-Organised Trusted Desktop Grid," in Lalanda *et al.* [6], pp. 267–272.

[50] M. Sommer, M. Klink, S. Tomforde, and J. Hähner, "Predictive load balancing in cloud computing environments based on ensemble forecasting," in Kounev *et al.* [68], pp. 300–307.

[51] S. Wildermann and J. Teich, "Self-integration for virtualization of embedded many-core systems," in SASOWS2014 [67], pp. 170–177.

[52] J. W. Rozenblit, "Models and methods for a high autonomy, self-improving surgical training system," in FAS*2017 [8], pp. 144–145.

[53] S. Rudolph, S. Tomforde, B. Sick, and J. Hähner, "A Mutual Influence Detection Algorithm for Systems with Local Performance Measurement," in *Proceedings of the 9th IEEE International Conference on Self-adapting and Self-organising Systems (SASO15), held September 21st to September 25th in Boston, USA*, pp. 144–150, 2015.

[54] D. Fisch, M. Jänicke, B. Sick, and C. Müller-Schloer, "Quantitative Emergence – A Refined Approach Based on Divergence Measures," in *Proceedings of IEEE Conference on Self-Adaptive and Self-Organising Systems (SASO10), held in Budapest, Hungary, September 27-October 1, 2010*, (Budapest, HU), pp. 94–103, 2010.

[55] K. Bellman, "Self-reflection and a version of structured "playing" may be critical for the verification and validation of complex system of systems," *CSDM 2013*, 2013.

[56] A. Stein, R. Maier, and J. Hähner, "Toward curious learning classifier systems: Combining xcs with active learning concepts," in *Proceedings of the Genetic and Evolutionary Computation Conference Companion*, GECCO '17, (New York, NY, USA), pp. 1349–1356, ACM, 2017.

[57] A. Stein, S. Tomforde, D. Rauh, and J. Hähner, "Dealing with unforeseen situations in the context of self-adaptive urban traffic control: How to bridge the gap?," in Kounev *et al.* [68], pp. 167–172.

[58] J. Loo, J. L. Mauri, and J. H. Ortiz, *Mobile ad hoc networks: current status and future trends*. CRC Press, 2016.

[59] K. L. Bellman and C. Landauer, "Towards an integration science: The influence of Richard Bellman on our research," *Journal of Mathematical Analysis and Applications*, vol. 249, no. 1, pp. 3–31, 2000.

[60] S. T. Powers, A. Ekárt, and P. R. Lewis, "Modelling enduring institutions: The complementarity of evolutionary and agent-based approaches," *Cognitive Systems Research*, 2018.

[61] C. Landauer, "Active system integrity fences," in *Proc. of the 12th Int. Works. on Models@run.time, 18 September 2017, Austin, Texas, held together with 20th ACM/IEEE Int. Conf. on Model Driven Engineering Languages and Systems*, 2017.

[62] Q. Wu and C. Miao, "Curiosity: From psychology to computation," *ACM Comput. Surv.*, vol. 46, pp. 18:1–18:26, Dec. 2013.

[63] A. G. Barto, S. Singh, and N. Chentanez, "Intrinsically motivated learning of hierarchical collections of skills," in *Proceedings of the 3rd International Conference on Development and Learning*, pp. 112–19, Citeseer, 2004.

[64] C. Landauer, "Mitigating the inevitable failure of knowledge representation," in *Proceedings the 2nd International Workshop on Models@run.time for Self-aware Computing Systems*, 2017.

[65] J. Botev, S. Rothkugel, and J. Klein, "Socio-Inspired Design Approaches for Self-Adaptive and Self-Organizing Collaborative Systems," in *Proceedings of the 3rd International Workshop on Self-Adaptive and Self-Organising Socio-Technical Systems (SASOŜT)*, pp. 19 – 24, 2015.

[66] A. J. I. Jones, A. Artikis, and J. Pitt, "The Design of Intelligent Socio-Technical Systems," *Artificial Intelligence Review*, vol. 39, no. 1, pp. 5 – 20, 2013.

[67] *Eighth IEEE International Conference on Self-Adaptive and Self-Organizing Systems Workshops, SASOW 2014, London, United Kingdom, September 8-12, 2014*, IEEE Computer Society, 2014.

[68] S. Kounev, H. Giese, and J. Liu, eds., *2016 IEEE International Conference on Autonomic Computing, ICAC 2016, Wuerzburg, Germany, July 17-22, 2016*, IEEE Computer Society, 2016.