The Organic Computing Doctoral Dissertation Colloquium: Status and Overview in 2019

Christian Krupitzer[1] Sven Tomforde[2]

Abstract: Today, people are surrounded by smart and connected devices. *Gartner Inc.* estimated that 8.4 billion devices were connected in the Internet-of-Things worldwide in 2017, reaching 20.4 billion by 2020. The growing number of mobile and embedded devices in combination with the omnipresence of (wireless) network connections facilitates new applications, such as autonomous driving, ambient assisted living / smart home, or Industry 4.0. However, it requires integration of all available, highly specialised, and heterogeneous devices, ranging from embedded sensor nodes to servers in the cloud. Further, the inclusion of data streams with sensor data and web data leads to an increasing complexity in system development. Additionally, as these systems are mobile, changing environmental conditions increase the complexity even further.

Organic Computing systems are able to deal with such systems and lower the complexity for their development by introducing methods for integrating adaptiveness and intelligence to computing systems. This workshops brings together PhD students from different research streams related to Organic Computing with the objective to represent the wide field of research in the domain of adaptive and intelligence systems.

Keywords: Organic Computing; Autonomic Computing; Self-adaptive Systems; Self-organising Systems; Self-Aware Computing Systems

1 Introduction

The *Organic Computing Doctoral Dissertation Colloquium* (OC-DDC) series is part of the Organic Computing initiative [MSSU11, MST17] which is steered by a Special Interest Group of the German Society of Computer Science (Gesellschaft für Informatik e.V.). It provides an environment for PhD students who have their research focus within the broader Organic Computing (OC) community to discuss their PhD project and current trends in the corresponding research areas, including, for instance, research in the context of Autonomic Computing [KC03], Proactive Computing [Te00], Complex Adaptive Systems [KST11], Self-Organisation [Ca03], Self-Aware Computing [Ko17], and related topics.

The main goal of the colloquium is to foster excellence in OC-related research by providing feedback and advice that are particularly relevant to the students' studies and career

[1] Software Engineering Group, Universität Würzburg christian.krupitzer@uni-wuerzburg.de
[2] Intelligent Embedded Systems, Universität Kassel stomforde@uni-kassel.de

doi:10.18420/inf2019_ws57

development. Thereby, participants are involved in all stages of the colloquium organisation and the review process to gain experience.

The OC-DDC 2019 takes place in Kassel, Germany in September 2019 as part of the annual meeting of the German Society of Computer Science (Gesellschaft für Informatik, GI): the Informatik 2019[3]. It is organised by the *Special Interest Group "Organic Computing"* within the GI in cooperation with the *Software Engineering* group at the University of Würzburg and the *Intelligent Embedded Systems* group at University of Kassel. It continued the series of successful OC-DDC events:

- The 2013 OC-DDC took place in Augsburg, Germany [To13].
- The 2014 OC-DDC took place in Kassel, Germany [TS14].
- The 2015 OC-DDC took place in Augsburg, Germany (together with the TSOS Spring School) [TS15].
- The 2016 OC-DDC took place in Duisburg, Germany [TS16].
- The 2017 OC-DDC took place in Bochum, Germany [TS18].
- The 2018 OC-DDC took place in Würzburg, Germany [TS19].

The goal of this paper is to outline the scope of the OC-DDC including the focus of the OC initiative and related concepts that fit into the frame of OC. Furthermore, we summarise the contributions of this year's papers accepted for the OC-DDC 2019 in Kassel and highlight current developments in the OC community. The authors of the accepted papers prepared an initial extended abstract of their PhD project, helped to perform a sophisticated review process, and finally came up with articles summarising their project and their current work in the context of OC. Hence, the six accepted contributions and the keynote given by Christian Krupitzer from the University of Würzburg reflect the diversity of the aspects of OC.

The remainder of this article is organised as follows: Section 2 briefly introduces the term "Organic Computing" and describes the scope of the initiative. Afterwards, Section 3 gives an overview of the contributions accepted for the OC-DDC 2019. Finally, Section 4 summarises the article and gives an outlook to future developments.

2 Organic Computing

The term "Organic Computing" (OC) has been coined in the German technical computer science community in 2003 to describe a novel paradigm in systems engineering by equipping technical systems with "life-like" properties [Al03]. Technically, this means to move traditional design-time decisions to runtime and into the responsibility of systems

[3] See `http://www.informatik2019.de` (last accessed: 21/06/2019)

themselves. As a result, systems have a dramatically increased decision freedom that leads to highly autonomous behaviour. The goal of this process is to allow for self-adaptation and self-improvement [Kr16a] of system behaviour at runtime [To11]. Especially since conditions that occur at runtime can only be anticipated to a certain degree, efficient mechanisms are required that guide the system's behaviour even in cases of missing knowledge or uncertain environmental status. Consequently, Organic Computing summarises a variety of aspects and techniques that are required to finally develop such mechanisms [TMS14]. For instance, a major challenge for Organic Computing and related research initiatives is the increasing interconnectedness of autonomous systems [THS14]. As a result, we face open, distributed systems consisting of autonomous elements belonging to different authorities [Ha15, Di16]. Autonomous entities as part of large-scale distributed system compositions require secure interaction schemes [Ha13].

Typically, an OC system is defined as follows: An OC system is a computer system that acts without or with only limited manual intervention. It thereby achieves and maintains a certain performance or utility even in time-variate environments and disturbed situations. In response to the dynamics, it adapts and improves its behaviour over time and interacts with other systems to achieve an individual and/or system-wide goal [TSM17].

OC is not the only research initiative which addresses the threatening collapse of ICT systems due to their inherent complexity—and it has never been the claim of OC. In contrast, OC is embedded in a broader research community sharing parts of the motivation or even parts of the concepts. One example for the basic motivation is the ubiquitous and pervasive computing initiative—which is a direct successor of the vision formulated by Marc Weiser in [We91]—addressing the abundance of intelligent devices and the penetration of almost every aspect of our lives with information technology. In the following paragraphs, we outline the—at least from our point of view—most prominent related research streams.

A) Proactive Computing (PAC): The term PAC reaches back to the year 2000, when David Tennenhouse stated that—as he called it—'human-in-the-loop computing' has its limits [Te00]. Consequently, the main driving force was the question of how to move from human-centred to human-supervised computing, which implies to transfer responsibilities to the systems themselves. In that sense, PAC can be seen as one of the major starting points within the ongoing process to move traditional design-time decisions to runtime.

B) Autonomic Computing (AC): About 3 years after Tennenhouse article introducing PAC, IBM came up with the vision of AC [KC03]. The initial motivation was that complexity in the management of large data centres is constantly increasing and consequently computing systems need an automated backbone structure similar to the autonomic nervous system of humans or animals. The initial idea by the authors Kephart and Chess was to equip this autonomic structure with capabilities to relieve the designer from specifying all possibly occurring situations and configurations already during the design process. In turn, the system to be developed should become responsible for identifying appropriate reactions on perceived changes in the environmental conditions. As a result, this also relieves the administrator from configuration and maintenance tasks—especially by finding optimised settings for all considered resources [LMD13]. Driven by a strong product perspective and fuelled by economic needs of IBM, the initial vision of AC had a strong focus on developing automated solutions for large-scale IT infrastructures—mainly with respect to configuration and self-management decisions. After a full decade of research in the field, AC has become one of the synonyms for autonomous and self-managed systems and is used synonymous with the term OC.

C) Self-Adaptive and Self-Organising Systems (SASO): The term SASO comprises all kinds of activities where technical systems autonomously vary their own behaviour (self-adaptation) and their internal functional structure (self-organisation) [Kr15b]. Consequently, it can be considered as an overarching umbrella for the previous terms. Furthermore, the SASO community shifted its focus within the last decade towards emphasising the necessity to integrate aspects of social communities into large-scale technical systems [Be17], e.g., by establishing enduring institutions [Di16, Pi14] and making use of social concepts such as trust [Ka15].

D) Multi-Agent Systems (MAS): The basic idea of an MAS is to describe several interacting, autonomous entities as a unified system. The term 'agent' is used to highlight that a software unit is acting autonomously by processing tasks on behalf of a user or administrator. However, this may also refer to robots, humans, or teams containing a mixture of both [Wo09]. MAS and the contained individual agents are used to model or solve problems that cannot be handled in a standard monolithic way due to parallelism and/or high-complexity reasons. In this sense, the concept is close to the ideas of OC and AC. However, the scope differs since MAS usually constitute some kind of heuristic approach for a hardly solvable or very complex problem [Je00], and they are mainly used for modelling purposes. In real-world applications, where SASO technology is required, the contained concepts (e.g. logic-based approaches) are sometimes less applicable in comparison to AC/OC approaches.

E) Complex Adaptive Systems (CAS): The term CAS is used to describe a collection of heterogeneous entities that have autonomous goals and behaviours but cooperate with each other. Furthermore, these contained entities are meant to adapt their behaviour in a collective manner in order to accomplish their own tasks or a common task and reach their individual goals or a common goal in an efficient and effective way [KLT03]. This is very close to the description of MAS, but CAS may be considered as the next development step and consequently as logical continuation from context-awareness and self-adaptation of a single application or system to a unified complex acting as a unified whole [No06].

F) Distributed Systems (DS): The basic idea of OC and the overarching SASO community is to develop individual entities that act with a local scope and collaborate to achieve system-wide goals. By construction, this incorporates a distributed nature of the overall system structure—and consequently defines a DS. A DS is considered to be an overall system composition that contains individual components that are located on different networked computers. These individual components communicate and coordinate their actions through message passing [TVS07] in order to achieve a common or system-wide goal. Typically, three significant characteristics of DS are defined: the concurrency of components, the lack of a global clock, and possible independent failures of components [CDK05].

G) Collaborative Interactive Learning Systems (CIL): The CIL initiative is embedded in the broader SASO community and aims at creating the foundation for new kinds of autonomous systems that act in dynamic, unconstrained real-world environments [Si18]. Such a CIL system is intended to be effective, understandable, and pleasant in its interaction and collaboration with humans. Consequently, the CIL initiative considers not just machine-machine collaboration but also the questions how the learning process of digital systems can be enhanced through seamless interaction with humans and the other way round: how autonomous inter-active systems can enhance human collaborative learning [Ha19].

H) Interwoven Systems (IWS): Recently, the OC community defined a specific subclass of ICT that is characterised by demanding challenges for system development such as openness, heterogeneity, mutual influences, or real-time constraints: IWS [TMS14, Ha15]. In order to be able to master IWS, self-improving system integration has been identified as a key enabler [To16, Be18]. Consequently, it addresses a part of the scope of the overall SASO initiative and makes use of some fundamental OC concepts.

I) Self-Aware Computing Systems (SeAC): The understanding of SeAC systems was fostered by the 2015 Dagstuhl Seminar 15041 on *model-driven algorithms and architectures for self-aware computing systems*. As proposed by the seminar and documented in a recent Springer book on the topic [Ko17], SeAC systems have two main properties. In accordance with high-level goals, they (i) learn models, capturing knowledge about themselves and their environment (such as their structure, design, state, possible actions, and runtime behaviour) on an ongoing basis; and (ii) reason using the models (to predict, analyse, consider, or plan), which enables them to act based on their knowledge.

In conclusion, the term Organic Computing comprises efforts to develop intelligent systems, i.e., to investigate techniques and concepts that allow for a runtime self-adaptation and self-organisation of typically distributed subsystems. Therefore, approaches from different domains are needed: distributed systems (e.g., self-organised communication, election and consensus algorithms), multi-agent systems (e.g., self-organised system organisation, negotiation), machine learning, and systems engineering.

3 The OC-DDC 2019

In the following, we will see that most of the fields mentioned above are covered by the contributions of this year's OC-DDC. We briefly summarise the six accepted contributions.

Wenzel Pilar von Pilchau from the Organic Computing group at the University of Augsburg presents a concept in the area of deep reinforcement learning based on interpolation for getting information of data points in an area where only neighbouring samples are known.

A first approach of averaging rewards in a setting with unstable transition function and very low exploration is implemented and shows promising results.

Eric Hutter from the Embedded System group at the University of Frankfurt addresses the complexity of automotive embedded systems with bio-inspired techniques by proposing an organic concept based on artificial DNA and an artificial hormone system that can be used to realise highly reliable, robust, and flexible automotive systems. The mechanisms are applied to a statically configured Classic AUTOSAR environment.

The ability to withstand disturbances while remaining functioning in a desired way is regarded as a crucial element in the field of OC. However, current approaches to self-healing and robustness fail in considering hardware-related breakdowns. Markus Görlich-Bucher from the Organic Computing group at the University of Augsburg presents a problem statement and various requirements an OC system must fulfil in order to increase its robustness against hardware-related disturbances.

Thomas Becker from the Computer Architecture group at the Karlsruhe Institute of Technology describes a mechanism based on rule-based machine learning which combines self-optimisation techniques with task-based runtime systems for reducing system complexity, efficiently using resources as well as the ability to dynamically adapt to new situations using symptom-based fault detection. This enables the consideration of dependability as an optimisation goal.

Simon Reichhuber from the Intelligent Systems group at the University of Passau presents a concept to combine the fields of Multi-Agent Learning, OC and Online Learning. The focus of the work is on the real-time evaluation of knowledge sources. The practicability of the approach is shown in two use cases: collaborative crawling and machine park collaboration.

Additive manufacturing machines following the Fused Deposition Modelling (FDM) process can rapidly produce wide varieties of parts. A 3D computer model is divided into instructions the FDM machine uses to produce the part layer by layer. Michael Heider from the Organic Computing group at the University of Augsburg uses early automated detection of parts not inside tolerances using machine learning techniques such as XCS to adaptively change instructions during printing that return the part into tolerances.

Furthermore, in his keynote entitled "Next Generation MAPE Control - Current Research Trends in Adaptive Systems" Christian Krupitzer from the Software Engineering group at the University of Würzburg presents different current research trends that affects the MAPE functionality for reasoning on adaptation, such as black-box monitoring, predictive analysis [Zü19], self-learning model-free optimisation for adaptation reasoning [Fr19] as well as interference management for decentralised autonomous systems [LKT19, Ma13] and their application in use cases such as platooning [Kr18b, Kr19] or wearable-based fall detection [Kr18a]. He links these topics to his work on software engineering for reusability in adaptive systems [Kr15a, Kr16b, Kr17a, Kr17b].

4 Conclusion

This paper discussed the Organic Computing Doctoral Dissertation Colloquium (OC-DDC) series and gave an overview on current activities in 2019. Based on an initial overview of the history of the OC-DDC, we briefly summarised the term Organic Computing (OC) and defined it. Afterwards, we mentioned related fields that are also addressed under the umbrella of OC.

The major part of this paper focused on the contributions accepted for this year's OC-DDC held in Kassel. These contributions were outlined and the keynote was summarised. We thank all authors for their contributions and we are looking forward to seeing again very interesting OC research at the next OC-DDC.

References

[Al03] Allrutz, Ralf; Cap, Clemens; Eilers, Stefan; Fey, Dietmar; Haase, Helmut; Hochberger, Christian; Karl, Wolfgang; Kolpatzik, Bernd; Krebs, Jochen; Langhammer, Falk et al.: Organic computing-computer-und systemarchitektur im jahr 2010. Vde/itg/gi-positionspapier, Gesellschaft für Informatik eV, 2003.

[Be17] Bellman, Kirstie; Botev, Jean; Hildmann, Hanno; Lewis, Peter R.; Marsh, Stephen; Pitt, Jeremy; Scholtes, Ingo; Tomforde, Sven: Socially-Sensitive Systems Design. IEEE Technology & Society Magazine, Special Issue on Social Concepts in Self-Organising Systems, p. to appear, 2017.

[Be18] Bellman, Kirstie L.; Botev, Jean; Diaconescu, Ada; Esterle, Lukas; Gruhl, Christian; Landauer, Christopher; Lewis, Peter R.; Stein, Anthony; Tomforde, Sven; Würtz, Rolf P.: Self-Improving System Integration - Status and Challenges after Five Years of SISSY. In: 2018 IEEE 3rd International Workshops on Foundations and Applications of Self* Systems (FAS*W), Trento, Italy, September 3-7, 2018. pp. 160–167, 2018.

[Ca03] Camazine, Scott; Deneubourg, Jean-Louis; Franks, Nigel R.; James Sneyd and, Guy Theraulaz; Bonabeau, Eric: Self-Organization in Biological Systems. Princeton Univers. Press, 2003.

[CDK05] Coulouris, George F; Dollimore, Jean; Kindberg, Tim: Distributed systems: concepts and design. pearson education, 2005.

[Di16] Diaconescu, Ada; Frey, Sylvain; Müller-Schloer, Christian; Pitt, Jeremy; Tomforde, Sven: 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. pp. 100 – 109, 2016.

[Fr19] Fredericks, Erik M.; Gerostathopoulos, Ilias; Krupitzer, Christian; Vogel, Thomas: Planning as Optimization: Dynamically Discovering Optimal Configurations for Runtime Situations. In: Proceedings of the 13th IEEE International Conference on Self-Adaptive and Self-Organizing Systems, SASO 2019. IEEE, June 2019.

[Ha13] Haehner, J.; Rudolph, S.; Tomforde, S.; Fisch, D.; Sick, B.; Kopal, N.; Wacker, A.: A Concept for Securing Cyber-Physical Systems with Organic Computing Techniques. In: Workshop Proceedings of the 2013 International Conference on Architectures of Computing Systems (ARCS13) – Workshop on Self-optimisation in Organic and Autonomic Computing Systems (SAOS13). VDE Verlag, 2013.

[Ha15] Haehner, J.; Brinkschulte, U.; Lukowicz, P.; Mostaghim, S.; Sick, B.; Tomforde, S.: Runtime Self-Integration as Key Challenge for Mastering Interwoven Systems. In: Proceedings of the 28th GI/ITG International Conference on Architecture of Computing Systems – ARCS Workshop, Workshop on Self-Optimisation in Organic and Autonomic Computing Systems (SAOS15). VDE Verlag, pp. 1 – 8, 2015.

[Ha19] Hanika, Tom; Herde, Marek; Kuhn, Jochen; Leimeister, Jan Marco; Lukowicz, Paul; Oeste-Reiß, Sarah; Schmidt, Albrecht; Sick, Bernhard; Stumme, Gerd; Tomforde, Sven; Zweig, Katharina Anna: Collaborative Interactive Learning - A clarification of terms and a differentiation from other research fields. CoRR, abs/1905.07264, 2019.

[Je00] Jennings, N. R.: Building Complex Software Systems: The Case for an Agent-Based Approach. Communications of the ACM, 44:12–23, 2000.

[Ka15] Kantert, Jan; Edenhofer, Sarah; Tomforde, Sven; Müller-Schloer, Christian: Representation of Trust and Reputation in Self-Managed Computing Systems. In: 15th IEEE International Conference on Computer and Information Technology, CIT 2015; 14th IEEE International Conference on Ubiquitous Computing and Communications, IUCC 2015; 13th IEEE International Conference on Dependable, Autonomic and Secure Computing, DASC 2015; 13th IEEE International Conference on Pervasive Intelligence and Computing, PICom 2015, Liverpool, United Kingdom, October 26-28, 2015. pp. 1827–1834, 2015.

[KC03] Kephart, Jeffrey O.; Chess, David M.: The Vision of Autonomic Computing. IEEE Computer, 36(1):41–50, 2003.

[KLT03] Kleinmann, Karl; Lazarus, Richard; Tomlinson, Ray: An infrastructure for adaptive control of multi-agent systems. In: IEMC'03 Proceedings. Managing Technologically Driven Organizations: The Human Side of Innovation and Change (IEEE Cat. No. 03CH37502). IEEE, pp. 230–236, 2003.

[Ko17] Kounev, Samuel; Kephart, Jeffrey O.; Milenkoski, Aleksandar; Zhu, Xiaoyun, eds. Self-Aware Computing Systems. Springer International Publishing, Cham, 2017.

[Kr15a] Krupitzer, Christian; Roth, Felix Maximilian; VanSyckel, Sebastian; Becker, Christian: Towards Reusability in Autonomic Computing. In: Proceedings of the International Conference on Autonomic Computing (ICAC). IEEE, pp. 115–120, 2015.

[Kr15b] Krupitzer, Christian; Roth, Felix Maximilian; VanSyckel, Sebastian; Schiele, Gregor; Becker, Christian: A survey on engineering approaches for self-adaptive systems. Pervasive and Mobile Computing, 17:184–206, 2015.

[Kr16a] Krupitzer, C.; Roth, F. M.; Pfannemüller, M.; Becker, C.: Comparison of Approaches for Self-Improvement in Self-Adaptive Systems. In: 2016 IEEE International Conference on Autonomic Computing (ICAC). pp. 308–314, July 2016.

[Kr16b] Krupitzer, Christian; Roth, Felix Maximilian; Becker, Christian; Weckesser, M.; Lochau, Malte; Schürr, Andy: FESAS IDE: An Integrated Development Environment for Autonomic Computing. In: Proceedings of the International Conference on Autonomic Computing (ICAC). IEEE, pp. 15–24, 2016.

[Kr17a] Krupitzer, C; Breitbach, M; Saal, J; Becker, C; Segata, M; Lo Cigno, R.: RoCoSys: A framework for coordination of mobile IoT devices. In: Proceedings of PerCom Workshops. pp. 485–490, 2017.

[Kr17b] Krupitzer, Christian; Otto, Julian; Roth, Felix Maximilian; Frommgen, Alexander; Becker, Christian: Adding Self-Improvement to an Autonomic Traffic Management System. In: Proceedings of the International Conference on Autonomic Computing (ICAC). 2017.

[Kr18a] Krupitzer, C.; Sztyler, T.; Edinger, J.; Breitbach, M.; Stuckenschmidt, H.; Becker, C.: Hips Do Lie! A Position-Aware Mobile Fall Detection System. In: Proceedings of the International Conference on Pervasive Computing and Communications (PerCom). 2018.

[Kr18b] Krupitzer, Christian; Segata, Michele; Breitbach, Martin; El-Tawab, Samy S.; Tomforde, Sven; Becker, Christian: Towards Infrastructure-Aided Self-Organized Hybrid Platooning. In: Proceedings of the IEEE Global Conference on Internet of Things. 2018.

[Kr19] Krupitzer, Christian; Lesch, Veronika; Pfannemüller, Martin; Becker, Christian; Segata, Michele: A Modular Simulation Framework for Analyzing Platooning Coordination. In: Proceedings of the 1st ACM MobiHoc Workshop on Technologies, mOdels, and Protocols for Cooperative Connected Cars. TOP-Cars '19. ACM, pp. 25–30, 2019.

[KST11] Kernbach, S.; Schmickl, T.; Timmis, J.: Collective adaptive systems: Challenges beyond evolvability. ACM Computing Research Repository (CoRR), 2011. last access: 07/14/2014.

[LKT19] Lesch, Veronika; Krupitzer, Christian; Tomforde, Sven: Multi-objective Optimisation in Hybrid Collaborating Adaptive Systems. In: Proceedings of the 7th edition in the Series on Autonomously Learning and Optimising Systems (SAOS), co-located with 32nd GI/ITG ARCS 2019. Gesellschaft fuer Informatik (GI), May 2019.

[LMD13] Lalanda, Philippe; McCann, Julie A; Diaconescu, Ada: Autonomic computing. In: Principles, Design and Implementation. Springer, 2013.

[Ma13] Majuntke, V.; VanSyckel, S.; Schäfer, D.; Krupitzer, C.; Schiele, G.; Becker, C.: COMITY: Coordinated application adaptation in multi-platform pervasive systems. In: IEEE International Conference on Pervasive Computing and Communications. 2013.

[MSSU11] Müller-Schloer, Christian; Schmeck, Hartmut; Ungerer, Theo, eds. Organic Computing - A Paradigm Shift for Complex Systems. Autonomic Systems. Birkhäuser Verlag, 2011.

[MST17] Müller-Schloer, Christian; Tomforde, Sven: Organic Computing – Techncial Systems for Survival in the Real World. Autonomic Systems. Birkhäuser Verlag, October 2017. ISBN: 978-3-319-68476-5.

[No06] Northrop, Linda; Feiler, Peter; Gabriel, Richard P; Goodenough, John; Linger, Rick; Longstaff, Tom; Kazman, Rick; Klein, Mark; Schmidt, Douglas; Sullivan, Kevin et al.: Ultra-large-scale systems: The software challenge of the future. Technical report, 2006.

[Pi14] Pitt, Jeremy: The computer after me: awareness and self-awareness in autonomic systems. World Scientific, 2014.

[Si18] Sick, Bernhard; Oeste-Reiß, Sarah; Schmidt, Albrecht; Tomforde, Sven; Zweig, Katharina Anna: Collaborative Interactive Learning. Informatik Spektrum, 41(1):52–55, 2018.

[Te00] Tennenhouse, David: Proactive computing. Communications of the ACM, 43(5):43–50, 2000.

[THS14] Tomforde, S.; Hähner, J.; Sick, B.: Interwoven Systems. Informatik-Spektrum, 37(5):483–487, 2014. Aktuelles Schlagwort.

[TMS14] Tomforde, S.; Müller-Schloer, C.: Incremental Design of Adaptive Systems. Journal of Ambient Intelligence and Smart Environments, 6:179 – 198, 2014.

[To11] Tomforde, S.; Prothmann, H.; Branke, J.; Haehner, J.; Mnif, M.; Müller-Schloer, C.; Richter, U.; Schmeck, H.: Observation and Control of Organic Systems. In (Müller-Schloer, C.; Schmeck, H.; Ungerer, T., eds): Organic Computing - A Paradigm Shift for Complex Systems, Autonomic Systems, pp. 325 – 338. Birkhäuser Verlag, 2011.

[To13] Tomforde, Sven: Proceedings of the First Organic Computing Doctoral Dissertation Colloquium (OC-DDC'13). Technical Report TR 2013-06, Fakultät für Informatik, Universität Augsburg, 2013.

[To16] Tomforde, Sven; Rudolph, Stefan; Bellman, Kirstie; Würtz, Rolf: An Organic Computing Perspective on Self-Improving System Interweaving at Runtime. In: Proceedings of the 13th IEEE International Conference on Autonomic Computing. 2016.

[TS14] Tomforde, Sven; Sick, Bernhard, eds. Organic Computing Doctoral Dissertation Colloquium – Proceedings of the OC-DDC14. kassel university press, Kassel, DE, 2014. ISBN 978-3-7376-0028-6.

[TS15] Tomforde, Sven; Sick, Bernhard, eds. Organic Computing Doctoral Dissertation Colloquium – Proceedings of the OC-DDC15. kassel university press, Kassel, DE, 2015. ISBN 978-3-86219-832-0.

[TS16] Tomforde, Sven; Sick, Bernhard, eds. Organic Computing Doctoral Dissertation Colloquium – Proceedings of the OC-DDC16. kassel university press, Kassel, DE, 2016. ISBN 978-3-7376-0300-3.

[TS18] Tomforde, Sven; Sick, Bernhard, eds. Organic Computing Doctoral Dissertation Colloquium – Proceedings of the OC-DDC17. kassel university press, Kassel, DE, 2018. ISBN 978-3-7376-0514-4.

[TS19] Tomforde, Sven; Sick, Bernhard, eds. Organic Computing Doctoral Dissertation Colloquium – Proceedings of the OC-DDC18. kassel university press, Kassel, DE, 2019. ISBN: 978-3-7376-0696-7.

[TSM17] Tomforde, Sven; Sick, Bernhard; Müller-Schloer, Christian: Organic Computing in the Spotlight. CoRR, abs/1701.08125, 2017.

[TVS07] Tanenbaum, Andrew S; Van Steen, Maarten: Distributed systems: principles and paradigms. Prentice-Hall, 2007.

[We91] Weiser, M.: The computer for the 21st century. Scientific American, 265(3):66–75, September 1991.

[Wo09] Wooldridge, M. J.: An Introduction to MultiAgent Systems. John Wiley & Sons Publishers, Hoboken, NJ, US, 2nd edition, 2009.

[Zü19] Züfle, Marwin; Bauer, André; Lesch, Veronika; Krupitzer, Christian; Herbst, Nikolas; Kounev, Samuel; Curtef, Valentin: Autonomic Forecasting Method Selection: Examination and Ways Ahead. In: Proceedings of the ICAC. 2019.