



Investigating the Interaction between Energy Consumption, Quality of Service, Reliability, Security, and Maintainability of Computer Systems and Networks

Dionysios Kehagias¹ · Marija Jankovic¹ · Miltiadis Siavvas¹ · Erol Gelenbe^{2,3}

Received: 1 July 2020 / Accepted: 13 November 2020 / Published online: 8 January 2021
© Springer Nature Singapore Pte Ltd 2021

Abstract

The Conference on Energy Consumption, Quality of Service, Reliability, Security, and Maintainability of Computer Systems and Networks (EQSEM) was held as a virtual conference on October 20–21, 2020. This paper summarises the objectives and proceedings of this conference. It then briefly presents the keynotes and other papers which were presented. Then, in the context of the EU-funded research project SDK4ED which motivated this conference, we outline several solutions that are being developed for managing the potential inter-dependencies and corresponding trade-offs between design-time and runtime qualities in software applications, and review the key functionalities that have been implemented within the SDK4ED integrated platform as of this writing. Then, we briefly introduce four papers among those presented at the EQSEM Conference that are included in this issue of the journal SN Computer Science, presenting relevant research achievements of the SDK4ED Project on software maintainability, reliability, and energy efficiency.

Keywords Software quality · Trade-offs · Energy efficiency · Maintainability · Reliability

Introduction

This special issue of the SN Computer Science journal consists of a selection of papers from the *Symposium on the Sustainability of ICT and the Interaction between Energy Consumption, Quality of Service, Reliability and Security, Maintainability of Computer Systems and Networks*

(EQSEM), held virtually between October 20 and October 21, 2020, organized by the *Paris Scientific Center of the Polish Academy of Sciences* and the *Institute of Theoretical and Applied Informatics of the Polish Academy of Sciences (IITIS-PAN)*, with the participation of the *H2020 SDK4ED Research and Innovation Project*. The purpose of the symposium was to bring together practitioners and researchers from both industry and academia to discuss progress on software quality engineering processes, methods, and tools, with specific emphasis on the inter-relationships and trade-offs between various metrics related to energy consumption, quality of service, maintainability, reliability, and security. Moreover, the symposium marked an important milestone in successful completion of the SDK4ED project and offered an opportunity to communicate the latest research results.

The overall objective of the SDK4ED project is to provide solutions that will allow project managers and software engineers to monitor and improve the quality of their embedded software, with respect to three important quality attributes, namely Maintainability, Dependability, and Energy Consumption. The project also provides solutions for identifying trade-offs among these three quality attributes. It does so, by delivering appropriate tools, which are meant to support software engineers on deciding which source code

This article is part of the topical collection “Interaction between Energy Consumption, Quality of Service, Reliability and Security, Maintainability of Computer Systems and Network” guest edited by Erol Gelenbe.

✉ Dionysios Kehagias
diok@iti.gr

Marija Jankovic
jankovicm@iti.gr

Miltiadis Siavvas
siavvasm@iti.gr

¹ Centre for Research and Technology Hellas, Information Technologies Institute, Thessaloniki, Greece

² Institute of Theoretical and Applied Informatics, Polish Academy of Sciences, Gliwice, Poland

³ Université Côte d’Azur, Laboratoire I3S, Nice, France

refactorings are necessary for improving a specific quality attribute, weighting their impact on the other two. The aforementioned solutions are integrated into a unified cloud-based platform. To the best of our knowledge, SDK4ED comprises the first platform that focuses on trade-offs between conflicting quality criteria by considering complex inter-relationships between design-time and runtime software qualities [22, 30, 31]. The conceptual architecture of the SDK4ED platform is presented in [13].

In what follows, Sect. 2 summarizes the content and presentations of the EQSEM Conference, while Sect. 3 provides an overview of the SDK4ED Platform, emphasizing its most recent implementation. Section 4 highlights the contributions of the papers that are included in this special issue. Finally, Sect. 5 provides brief conclusions and discusses future work.

The EQSEM Conference on 20–21 October 2021

The EQSEM Conference was organized by the Paris Scientific Center and the Institute of Theoretical and Applied Informatics (IITIS-PAN), both of the Polish Academy of Sciences, in collaboration with the H2020 SDK4ED Research and Innovation Project coordinated by the Information Technologies Institute of the Center for Research and Technology Hellas (ITI-CERTH). To this effect, the Organizing Committee composed of Tadeusz Czachòrski and Erol Gelenbe of IITIS-PAN and Dr. Magdalena Sajdak (Paris Scientific Center, Polish Academy of Sciences), worked with the Scientific Committee chaired by Erol Gelenbe (Chair, IITIS-PAN) and comprising Miltiadis Siavvas (ITI-CERTH, Scientific Secretary), Tadeusz Czachòrski, Joanna Domańska (IITIS-PAN), Krzysztof Grochla (IITIS-PAN), Jean-Michel Fourneau (Univ. Versailles Saint Quentin, France), Marija Jankovic (ITI-CERTH), Alain Jean-Marie (INRIA, France), and Dionysios Kehagias (ITI-CERTH). Initially meant to be held in Paris, it was finally held online due to the constraints imposed by the COVID-19 pandemic.

The objective of the conference was to investigate how metrics related to Energy Consumption, Quality of Service, Reliability, Security, and Maintainability are closely linked together, and how trade-offs can be achieved to optimise these different considerations within a given system. Two keynotes set the tone for the meeting, developing a complementary exchange of views about the role of energy in ICT, when Dr. Michel Schmitt of the High Council for the Economy, Ministry of the Economy, Finance and the Recovery, France spoke about “Recent trends and data concerning the energy consumption of ICT”, and Professor Fabrice Flipo, Institut Mines-Telecom Business School, Evry., France discussed “The worrisome trajectory of energy consumption by

digital systems”. (L’inquiétante trajectoire de la consommation énergétique du numérique)

In other presentations during the first day, Charalampos Marantos, Konstantinos Salapas, Lazaros Papadopoulos, and Dimitrios Soudris of the ICCS National Technical University of Athens, spoke about “A flexible Tool for Applications Performance and Energy consumption estimation through Static analysis”. Then, Erol Gelenbe, Pawel Boryszko, Miltiadis Siavvas, and Joanna Domanska, from the Institute of Theoretical and Applied Informatics, Polish Academy of Sciences and the Center for Research and Technology Hellas, discussed “Optimum Checkpointing for Time and Energy”. Finally that day, Tadek Czachorski, Erol Gelenbe (IITIS-PAN) and Dariusz Marek (Polytechnic University of Gliwice), discussed “Optimum Energy for ICT and Mission Accomplishment for Mobile Platforms”.

During the second day of the conference a team from the University of Macedonia (Greece), and the Technical University of Cluj (Romania) composed of Georgios Digkas, Apostolos Ampatzoglou, Alexander Chatzigeorgiou, Paris Avgeriou, Oliviu Matei, and Robert Heb presented the paper about “Calculating the Technical Debt Interest Generation Risk: A Case Study on New Code”, this was followed by the work of by Darius Sas, Paris Avgeriou, and Ruben Scheedler on “Exploring the relation between co-changes and architectural smells”, from the University of Groningen (The Netherlands).

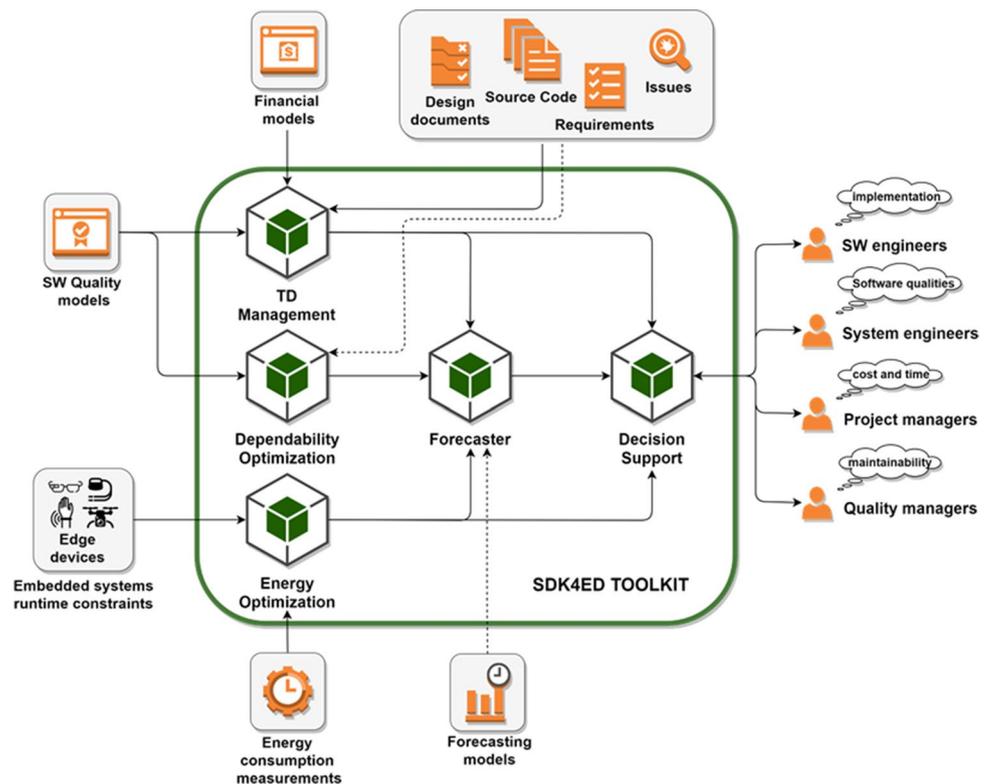
Then, the team composed of Dimitrios Tsoukalas, Maria Mathioudaki, Miltiadis Siavvas, Dionysios Kehagias, and Alexander Chatzigeorgiou summarized their work on “A Clustering Approach towards Cross-project Technical Debt Forecasting”, which was carried out at the University of Macedonia and ITI, Center for Research and Technology Hellas. Finally, Katia Filus, Erol Gelenbe, Joanna Domanska, and Miltiadis Siavvas discussed their work on “Software vulnerability prediction with the Random Neural Network”, which was jointly carried out at IITIS-PAN and ITI-CERTH.

These presentations were concluded with an exchange of questions and discussions between speakers and participants.

Overview of the SDK4ED Platform

The high-level overview of the SDK4ED Platform is presented in Fig. 1. As shown in Fig. 1, the platform comprises five core toolboxes, which are: (i) the Technical Debt (TD) Management Toolbox, (ii) the Energy Optimisation Toolbox, (iii) the Dependability Optimisation Toolbox, (iv) the Forecaster Toolbox, and (v) the Decision Support Toolbox. The first three toolboxes are responsible for monitoring and optimising the Maintainability, Energy Consumption, and Dependability of embedded software products, respectively. The other two are meant to further support decision-making

Fig. 1 The overall architecture of the SDK4ED Platform



during software development, through the provision of quality forecasts and analysis of potential trade-offs. The SDK4ED Platform provides solutions for monitoring and improving the quality of software applications written in Java, C, and C++ programming languages, covering, but not limited to, a wide range of embedded software applications.

The different toolboxes of the SDK4ED Platform have been implemented in the form of Microservices. The main features of the five core toolboxes that constitute the SDK4ED Platform are described below, in an attempt to help the reader gain a deeper understanding of the SDK4ED Platform, as well as of the novel functionalities that it provides.

Technical Debt Management Toolbox The main responsibility of the TD Management Toolbox is to monitor and optimise the *Maintainability* of software products through the notion of TD [6]. In particular, its main goal is to provide efficient support for TD measurement, prioritisation, and handling in Embedded Software applications. The TD Toolbox provides solutions in the form of web services for: (i) *TD Analysis*, which is responsible for analysing and evaluating the TD aspects of a software application including TD Principal, TD Interest, and TD Interest Probability using the SonarQube¹ platform and FITTED framework [2], (ii) *TD Evaluation of New Code*, which is responsible for assessing the TD of the newly added code and prevent the new code

from being committed to the software repository if its value is high, and (iii) *TD Refactorings Identification*, which is responsible for suggesting code-, architecture-, and design-level refactorings that are expected to improve the TD of the analysed application. A multitude of novel features have been developed within the course of the SDK4ED project with respect to the aforementioned solutions (i.e., services) [1, 2, 4, 5, 8, 9, 16, 23, 35, 37, 39].

Energy Optimisation Toolbox The purpose of the Energy Optimisation Toolbox is to monitor and optimise the *Energy Consumption* of a given software program. More specifically, this module provides solutions for (i) estimating the energy consumption of a software program and identifying energy-hungry parts, and (ii) providing recommendations for potential energy optimisations. These solutions are implemented in the form of web services, which are the *Energy Consumption Analysis* web service and the *Energy Estimation and Optimisation* web service of the SDK4ED ecosystem. More specifically, the former is responsible for analysing a given software program with respect to its energy consumption using popular dynamic analysis tools including Linux Perf² and Valgrid³. with the purpose to detect energy-hungry parts (i.e., energy hotspots). The latter is responsible for suggesting code- and design-level optimisations that are

¹ <https://www.sonarqube.org/>

² https://perf.wiki.kernel.org/index.php/Main_Page.

³ <http://www.valgrind.org/>

expected to reduce the energy consumption of the identified hotspots, and, in turn, of the broader software program. These services are based on novel energy estimation techniques that have been developed within the context of the SDK4ED project [17–20, 36, 44].

Dependability Optimisation Toolbox The Dependability Optimisation Toolbox is responsible for assessing and optimising the Dependability of software products. In particular, this module provides solutions for evaluating and improving Software Security and Software Reliability, which are two important facets of Software Dependability. The toolbox provides solutions for monitoring and improving those quality attributes in the form of individual web services. With respect to software security, the toolbox provides: (i) the *Quantitative Security Assessment* web service, which is responsible for assessing the internal security level of software products in a quantitative manner, using a novel static analysis-based *Security Assessment Model* that has been developed within the course of the SDK4ED project [28, 29], utilising novel concepts from the field of software quality [33], and (ii) the *Vulnerability Prediction* web service, which is responsible for identifying potentially vulnerable software components that reside in the analysed software [15], based on novel text mining-based deep learning models. With respect to software reliability, novel mathematical models for suggesting the optimum checkpoint interval that optimises important Quality-of-Service (QoS) parameters [10–12, 14, 25–27], with emphasis on energy consumption and performance were developed and used as part of the *Optimum Checkpoint Interval Recommendation* web service of the Dependability Toolbox.

Forecaster Toolbox The purpose of the Forecaster Toolbox is to predict the evolution of the three quality objectives of choice, which are the Maintainability, the Energy Consumption, and the Dependability of software products. This is achieved through the provision of advanced time series and machine-learning (ML) models, which are built based on the results of the three aforementioned modules (see Fig. 1). Hence, the Forecaster Toolbox provides three individual web services, namely *TD Forecaster*, *Energy Forecaster*, and *Dependability Forecaster*, which provide ML models for predicting the evolution of TD, Energy Consumption, and Dependability, respectively. Novel models have been developed for this purpose, especially for the case of TD forecasting [40, 41, 43].

Decision Support Toolbox The main objective of the Decision Support Toolbox is to facilitate decision-making during the development and maintenance of embedded software products. In particular, the purpose of this toolbox is to provide information about the impact that refactorings for improving a specific quality attribute (e.g., Dependability) will have on the other two attributes that the platform focuses on (i.e., Energy Consumption and Maintainability).

In fact, trade-off analysis is employed by the Decision Support Toolbox (based on the outputs of the other modules, as shown in Fig. 1) to compute these impacts and provide useful recommendations. Throughout the SDK4ED project, several empirical studies have been conducted for reaching conclusions with respect to potential trade-offs among the three quality attributes of choice [22, 30–32, 34]. Based on the identified trade-offs, a multi-criteria decision-making (MCDM) model that leverages concepts from fuzzy logic has been developed [38], providing information about the impacts of the suggested refactorings. Similarly to the other toolboxes, the MCDM model is implemented in the form of a web service.

Special Issue Paper Highlights

The first paper in the special issue presents the latest research work done in the context of the SDK4ED Energy Optimisation toolbox. Marantos et al. in [21] propose a flexible tool for estimating performance and energy in embedded platforms by applying static analysis techniques. Moreover, they define a complete methodology with clearly defined steps to easily add estimation models for various platforms. Whereas state-of-the-art approaches for energy estimation rely on dynamic instrumentation, which results in significant time overheads, the proposed approach takes advantage of static analysis applied at the instruction level. The notable benefits for the end-users include increased speed and the possibility of performing energy estimation during the Software Development Life Cycle development phase instead of measuring energy after the application deployment.

The second paper [7] presents a case study to assess artefacts that pose a high risk of generating TD interest. Digkas et al. highlight that TD interest generation represents a risk for software maintainability [3, 5]. Thus, they propose a novel Interest Generation Risk Importance (IGRI) measure for risk quantification. Furthermore, they conduct an empirical case study in an industrial, automotive setting to validate its ability to (i) prioritize TD artefacts and (ii) investigate the consequences of adding new code in the software system. The results of this study validate the hypothesis that the IGRI metric could be used to prioritize software artefacts effectively and that new code is less risky for generating TD interest. Overall, the proposed metric could be a valuable addition to existing software quality frameworks, such as the SDK4ED framework, with built-in support for TD interest monitoring and optimization.

The third paper [24] investigates the complex interplay between maintainability as a design time and reliability as a runtime software quality using two indicators: co-changes and architectural smells. Sas et al. stress the importance of architectural debt (ATD) and its negative impact on

design-time system quality such as maintainability. They identify three important research questions and conduct an explanatory multiple-case study on 14 open source Java projects. Both state-of-the-art and newly developed algorithms are used to detect co-changes. Data exploration and statistical analysis show that a low maintainability level negatively impacts reliability, even at the architecture design level.

The last paper [42] proposes a clustering-based approach that aims to improve cross-project TD forecasting models. Tsoukalas et al. highlight the issue of performing TD forecasting for new software projects lacking a reliable and sufficient number of past commits [41]. Thus, authors investigate whether the consideration of TD-related similarities could lead to more accurate cross-project forecasting. The results of the experimental study suggest that the adoption of clustering leads to more accurate cross-project TD forecasting.

Conclusions

In this paper, we have outlined the contents of the EQSEM Conference, and provided an overview of the main outcomes of the SDK4ED project. We have outlined the SDK4ED platform that takes a major step beyond state-of-art in the field of embedded software systems by providing novel concepts and approaches for managing qualities related to maintainability, dependability, and energy efficiency software. To the best of our knowledge, the SDK4ED platform is the first to handle the non-trivial relations and trade-offs between these qualities. Ongoing investment on improving these software-level qualities is gaining interest among industry practitioners, since hardware improvements are usually more challenging and costlier to implement.

The contributed papers presented in this special issue demonstrate mature progress towards the implementation of a fully operational platform for monitoring, optimisation, and trade-off calculation between the aforementioned software qualities. Relevant innovations that have been produced so far, as described in the original publications of the project, address a wide variety of software quality engineering issues, such as static analysis-based energy estimation, TD risk estimation, trade-offs between reliability and maintainability at the architectural level, reliable quantitative security assessment, optimum checkpointing, etc.

As the SDK4ED project, funded by the European Commission, is approaching its end in December of 2020, the project's next steps will mostly focus on making a stable version of the platform publicly available to all developers as a web-based open source platform. Also the enterprise partners of the project are investigating further commercial exploitation opportunities of the SDK4ED platform in the software development industry. We expect that this work will materialise to the widest possible extent, our goal of

increasing awareness of TD in the open-source and embedded software communities, and to provide useful insights on the impact that software refactoring can have on energy efficiency.

Acknowledgements The authors would like to thank Dr. Magdalena Sajdak, Prof. Tadek Czàchorski, and Dr Krzysztof Grochla, and all Scientific Programme Committee members, for their contributions to the organization of the EQSEM Conference. This work was funded by the European Union's Horizon 2020 Research and Innovation Programme through SDK4ED project under Grant Agreement No. 780572.

References

1. Amanatidis T, Mittas N, Chatzigeorgiou A, Ampatzoglou A, Angelis L. The developer's dilemma: factors affecting the decision to repay code debt. In: Proceedings of the 2018 International Conference on technical debt, 2018; pp. 62–6.
2. Ampatzoglou A, Michailidis A, Sarikyriakidis C, Ampatzoglou A, Chatzigeorgiou A, Avgeriou P. A framework for managing interest in technical debt: an industrial validation. In: Proceedings of the 2018 International Conference on Technical Debt, 2018; pp. 115–24.
3. Ampatzoglou A, Michailidis A, Sarikyriakidis C, Ampatzoglou A, Chatzigeorgiou A, Avgeriou P. A framework for managing interest in technical debt: an industrial validation. In: Proceedings of the 2018 International Conference on Technical Debt, TechDebt '18, 2018; pp. 115–24. Association for Computing Machinery, New York, NY, USA. <https://doi.org/10.1145/3194164.3194175>.
4. Arvanitou EM, Ampatzoglou A, Bibi S, Chatzigeorgiou A, Stamelos I. Monitoring technical debt in an industrial setting. In: Proceedings of the Evaluation and Assessment on Software Engineering, 2019; pp. 123–32.
5. Charalampidou S, Arvanitou EM, Ampatzoglou A, Avgeriou P, Chatzigeorgiou A, Stamelos I. Structural quality metrics as indicators of the long method bad smell: An empirical study. In: 2018 44th Euromicro Conference on software engineering and advanced applications (SEAA), 2018; pp. 234–38. IEEE.
6. Cunningham W. The wycash portfolio management system. ACM SIGPLAN OOPS Mess. 1993;4(2):29–30.
7. Digkas G, Ampatzoglou A, Chatzigeorgiou A, Avgeriou P, Matei O, Heb R. The risk of generating technical debt interest: a case study. SN Comput Sci. 2020. (**this issue**)
8. Digkas G, Lungu M, Avgeriou P, Chatzigeorgiou A, Ampatzoglou A. How do developers fix issues and pay back technical debt in the apache ecosystem? In: 2018 IEEE 25th International Conference on software analysis, evolution and reengineering (SANER), 2018; pp. 153–63.
9. Digkas G, Nikolaidis N, Ampatzoglou A, Chatzigeorgiou A.. Reusing code from stackoverflow: the effect on technical debt. In: 2019 45th Euromicro Conference on Software Engineering and Advanced Applications (SEAA), 2019; pp. 87–91.
10. Gelenbe E, Boryszko P, Siavvas M, Domanska J. Optimum checkpoints for time and energy. In: MASCOTS 2020, IEEE 28th International Symposium on modeling, analysis, and simulation of computer and telecommunication systems. IEEE Xpress, 2020.
11. Gelenbe E, Zhang Y. Performance optimization with energy packets. IEEE Syst J 13(4):3770–780.
12. Gelenbe E, Zhang Y.. Sharing energy for optimal edge performance. In: SOFSEM 2020: theory and practice of computer science. Cham: Springer International Publishing; 2020. p. 24–36.
13. Jankovic M, Kehagias D, Siavvas M, Tsoukalas D, Chatzigeorgiou A.. The SDK4ED approach to software quality optimization and

- interplay calculation. In: 15th China-Europe International Symposium on software engineering education, 2019.
14. Kadioglu YM, Gelenbe E. Product-form solution for cascade networks with intermittent energy. *IEEE Syst J.* 2018;13(1):918–27.
 15. Kalouptsoglou I, Siavvas M, Tsoukalas D, Kehagias D. Cross-project vulnerability prediction based on software metrics and deep learning. In: *Computational Science and Its Applications—ICCSA 2020*. Cham: Springer International Publishing; 2020. p. 877–93.
 16. Kouros P, Chaikalas T, Arvanitou EM, Chatzigeorgiou A., Ampatzoglou A., Amanatidis T. Jcaliper: search-based technical debt management. In: *Proceedings of the 34th ACM/SIGAPP Symposium on applied computing*, 2019; pp. 1721–730.
 17. Leon V, Asimakopoulos K, Xydis S, Soudris D, Pekmestzi K. Cooperative arithmetic-aware approximation techniques for energy-efficient multipliers. In: *Proceedings of the 56th Annual Design Automation Conference*, 2019; pp. 1–6.
 18. Maragos K, Lentaris G, Soudris D. In-the-field mitigation of process variability for improved fpga performance. *IEEE Trans Comput.* 2019; 68(7):1049–63.
 19. Maragos K, Taka E, Lentaris G, Stratakos I, Soudris D. Analysis of performance variation in 16nm finfet fpga devices. In: *2019 29th International Conference on Field Programmable Logic and Applications (FPL)*, 2019; pp. 38–44.
 20. Marantos C, Lamprakos CP, Tsoutsouras V, Siozios K, Soudris D. Towards plug&play smart thermostats inspired by reinforcement learning. In: *Proceedings of the Workshop on INTElligent Embedded Systems Architectures and Applications*, 2018; pp. 39–44.
 21. Marantos C, Salapas K, Papadopoulos L, Soudris D. A flexible tool for estimating applications performance and energy consumption through static analysis. *SN Comput. Sci.* 2020. **(this issue)**
 22. Papadopoulos L, Marantos C, Digkas G, Ampatzoglou A, Chatzigeorgiou A, Soudris D. Interrelations between software quality metrics, performance and energy consumption in embedded applications. In: *Proceedings of the 21st International Workshop on software and compilers for embedded systems*, 2018; pp. 62–65.
 23. Sas D, Avgeriou P, Fontana FA. Investigating instability architectural smells evolution: an exploratory case study. In: *2019 IEEE International Conference on software maintenance and evolution (ICSME)*, 2019; pp. 557–567. IEEE.
 24. Sas D, Avgeriou P, Kruizinga R, Scheedler R. Exploring the relation between co-changes and architectural smells. *SN Comput. Sci.* 2020 **(this issue 2020)**
 25. Siavvas M, Gelenbe E. Optimum checkpointing for long-running programs. In: *15th China-Europe International Symposium on software engineering education*. 2019.
 26. Siavvas M, Gelenbe E. Optimum checkpoints for programs with loops. *Simul Model Pract Theory.* 2019; 97:101951.
 27. Siavvas M, Gelenbe E. Optimum interval for application-level checkpoints. In: *2019 6th IEEE International Conference on Cyber Security and Cloud Computing (CSCloud)/2019 5th IEEE International Conference on Edge Computing and Scalable Cloud (EdgeCom)*, 2019; pp. 145–150. IEEE.
 28. Siavvas M, Gelenbe E, Kehagias D, Tzovaras D. Static analysis-based approaches for secure software development. In: *Security in computer and information sciences*. Cham: Springer International Publishing; 2018. p. 142–57.
 29. Siavvas M, Jankovic M, Kehagias D, Tzovaras D. Is popularity an indicator of software security? In: *2018 International Conference on intelligent systems (IS)*, 2018; p. 692–97. IEEE.
 30. Siavvas M, Marantos C, Papadopoulos L, Kehagias D, Soudris D, Tzovaras D. On the relationship between software security and energy consumption. In: *15th China-Europe International Symposium on software engineering education*, 2019.
 31. Siavvas M, Tsoukalas D, Jankovic M, Kehagias D, Chatzigeorgiou A, Tzovaras D, Anicic N, Gelenbe E. An empirical evaluation of the relationship between technical debt and software security. In: *9th International Conference on Information society and technology (ICIST)*, vol. 2019, 2019.
 32. Siavvas M, Tsoukalas D, Jankovic M, Kehagias D, Tzovaras D. Technical debt as an indicator of software security risk: a machine learning approach for software development enterprises. *Enterp Inf Syst.* 2020. <https://doi.org/10.1080/17517575.2020.1824017>.
 33. Siavvas MG, Chatzidimitriou KC, Symeonidis AL. Qatch—an adaptive framework for software product quality assessment. *Expert Syst Appl.* 2017;86:350–66. <http://www.sciencedirect.com/science/article/pii/S0957417417303883>.
 34. Siddiqi MA, Strydis C. Imd security vs. energy: are we tilting at windmills? poster. In: *Proceedings of the 16th ACM international conference on computing frontiers*, 2019; p. 283–85.
 35. Skiada P, Ampatzoglou A, Arvanitou EM, Chatzigeorgiou A, Stamelos I. Exploring the relationship between software modularity and technical debt. In: *2018 44th Euromicro Conference on software engineering and advanced applications (SEAA)*, 2018; pp. 404–7. IEEE.
 36. Stratakos I, Gourounas D, Tsoutsouras V, Economopoulos T, Matsopoulos G, Soudris D. Hardware acceleration of image registration algorithm on fpga-based systems on chip. In: *Proceedings of the International Conference on omni-layer intelligent systems*, 2019; pp. 92–7.
 37. Tsantalas N, Chaikalas T, Chatzigeorgiou A. Ten years of jdeodorant: Lessons learned from the hunt for smells. In: *2018 IEEE 25th International Conference on Software Analysis, Evolution and Reengineering (SANER)*, 2018; p. 4–14. IEEE.
 38. Tsimpourlas F, Papadopoulos L, Bartsokas A, Soudris D. A design space exploration framework for convolutional neural networks implemented on edge devices. *IEEE Trans Comput Aided Des Integr Circ Syst.* 2018; 37(11): 2212–221.
 39. Tsintzira AA, Ampatzoglou A, Matei O, Ampatzoglou A, Chatzigeorgiou A, Heb R. Technical debt quantification through metrics: an industrial validation. In: *15th China-Europe International Symposium on software engineering education*, 2019.
 40. Tsoukalas D, Jankovic M, Siavvas M, Kehagias D, Chatzigeorgiou A, Tzovaras D: On the applicability of time series models for technical debt forecasting. In: *15th China-Europe International Symposium on software engineering education*. 2019.
 41. Tsoukalas D, Kehagias D, Siavvas M, Chatzigeorgiou A. Technical debt forecasting: an empirical study on open-source repositories. *J Syst Softw.* 2020;170:110777. <http://www.sciencedirect.com/science/article/pii/S0164121220301904>.
 42. Tsoukalas D, Mathioudaki M, Siavvas M, Kehagias D, Chatzigeorgiou A. A clustering approach towards cross-project technical debt forecasting. *SN Comput. Sci.* 2020. **(this issue)**
 43. Tsoukalas D, Siavvas M, Jankovic M, Kehagias D, Chatzigeorgiou A, Tzovaras D. Methods and tools for td estimation and forecasting: a state-of-the-art survey. In: *2018 International Conference on intelligent systems (IS)*, 2018' p. 698–705. IEEE.
 44. Xygkis A, Soudris D, Papadopoulos L, Yous S, Moloney D. Efficient winograd-based convolution kernel implementation on edge devices. In: *2018 55th ACM/ESDA/IEEE Design Automation Conference (DAC)*, 2018; p. 1–6. IEEE.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.