CARDIFF UNIVERSITY
PRIFYSGOL CAERDYDD

ORCA – Online Research @ Cardiff

This is an Open Access document downloaded from ORCA, Cardiff University's institutional repository:https://orca.cardiff.ac.uk/id/eprint/171110/

This is the author's version of a work that was submitted to / accepted for publication.

Citation for final published version:

Puliafito, Carlo, Rana, Omer , Bittencourt, Luiz F. and Wu, Hao 2024. Serverless computing in the cloud-to-edge continuum. Future Generation Computer Systems 161 , pp. 514-517. 10.1016/j.future.2024.07.043

Publishers page: http://dx.doi.org/10.1016/j.future.2024.07.043

Please note:
Changes made as a result of publishing processes such as copy-editing, formatting and page numbers may not be reflected in this version. For the definitive version of this publication, please refer to the published source. You are advised to consult the publisher's version if you wish to cite this paper.

This version is being made available in accordance with publisher policies. See http://orca.cf.ac.uk/policies.html for usage policies. Copyright and moral rights for publications made available in ORCA are retained by the copyright holders.

Serverless Computing in the Cloud-to-Edge Continuum

Carlo Puliafito[a,*], Omer Rana[b], Luiz F. Bittencourt[c], Hao Wu[d]

[a]*Department of Information Engineering, University of Pisa, Pisa, Italy*
[b]*School of Computer Science and Informatics, Cardiff University, Cardiff, UK*
[c]*Instituto de Computação, Universidade Estadual de Campinas, Campinas, Brazil*
[d]*School of Artificial Intelligence, Beijing Normal University, Beijing, China*

Abstract

Serverless computing is establishing itself as a way to efficiently run cloud applications while abstracting the underlying infrastructure complexity away from application developers. At the same time, edge computing provides cloud-like facilities toward the edge of the network, in closer proximity to client devices. Integration of serverless and edge computing is promising. However, many research challenges still need to be solved to let this integration unleash its full potential. This special issue brings together works making high-quality and original contributions in this field, either proposing innovative strategies, implementing and testing novel solutions, or making new serverless datasets available to the community.

Keywords: Serverless, Function as a Service, FaaS, edge computing, continuum

1. Introduction

Serverless computing is revolutionizing the way to operate cloud systems and provision cloud applications. Through serverless, application developers do not have to manage the underlying infrastructure. Instead, they simply provide the application code, and the serverless platform takes care of resource management, service orchestration, and system monitoring. Moreover, serverless computing adopts function as a service (*FaaS*) as its main programming model, through which applications are decomposed into small pieces of code, known as functions, each running in stateless containers. Serverless introduces many benefits, such as the abstraction of infrastructure complexity, resource efficiency, high scalability, and a pure pay-per-use model.

Meanwhile, edge computing has established itself as a powerful extension of the cloud toward the network edge. Edge computing pervasively deploys servers along the continuum from the cloud to the network edge, bringing computation facilities closer to client devices. This proximity enables low latency and high throughput, hence paving the way to a plethora of emerging applications, spanning from augmented/virtual reality and smart vehicles to smart logistics and smart healthcare.

Operating the cloud-to-edge continuum through serverless computing seems promising. However, significant research challenges exist to effectively fill the gap between the cloud-oriented design of serverless and the distinguishing characteristics of edge computing (e.g., pervasive distribution, resource heterogeneity, stringent application requirements). Some of the research challenges in the field are: (i) orchestration of functions across the distributed continuum; (ii) management of application state; (iii) mitigation of the cold-start effect; and (iv) execution of long-running workloads such as federated learning.

This special issue brings together original and high-quality research contributions on the topic. All the articles in this special issue advance the current state of the art by either proposing innovative strategies, practically implementing and testing novel solutions, or making new serverless datasets available to the community. The next section provides details on the accepted manuscripts.

2. Contents of the Special Issue

All the manuscripts submitted to this special issue have undergone a rigorous blinded review process and have been analyzed by a Committee of internationally recognized experts in the field (see Section 3) based on coherence with the special-issue topics, originality and novelty, scientific soundness, methodological robustness, consistency and usefulness of any proposed use case, impact on the referenced scientific community, and overall readability. A total of 17 articles have been accepted, with an acceptance rate of 54.84%. The accepted articles investigate different aspects of serverless edge computing. Overall, we identified four subtopics, namely: (i) function orchestration and request scheduling; (ii) data management and data pipelines; (iii) hosting serverless functions on Internet of Things (IoT) resources; and (iv) other works on serverless edge computing. Figure 1 depicts the percentage of accepted manuscripts in each subtopic. As shown, function orchestration and request scheduling are the most investigated.

2.1. Function orchestration and request scheduling

The article **"Load Balancing for Heterogeneous Serverless Edge Computing: A Performance-Driven and Empiri-**

*Corresponding author
Email addresses: carlo.puliafito@unipi.it (Carlo Puliafito), ranaof@cardiff.ac.uk (Omer Rana), bit@ic.unicamp.br (Luiz F. Bittencourt), wuhao@bnu.edu.cn (Hao Wu)

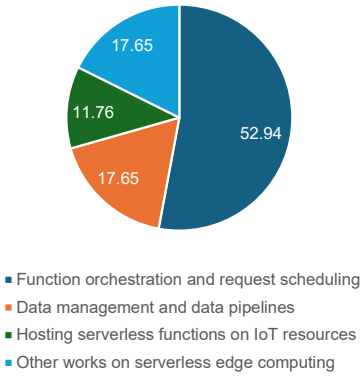

Figure 1: The percentage of accepted manuscripts addressing each of the covered subtopics.

cal Approach" [1] presents a performance-driven, weight-tuning approach to balance the load effectively when considering the heterogeneous characteristics and capabilities of nodes in the edge. The proposed approach uses extensive empirical profiling of nodes to understand their performance based on different metrics and acquire knowledge to weight a round-robin strategy to optimize performance metrics. Moreover, a multi-objective and a coordinated distributed approaches are introduced. A heterogeneous serverless edge architecture – Hedgi – is presented, enabling the proposed load balancing policies to be configured and utilized and allowing analyses of load balancing trade-offs in heterogeneous serverless edge infrastructures.

Filinis et al. [2] in **"Intent-driven orchestration of serverless applications in the computing continuum"** describe an intent-based approach for the orchestration of a function chain over an edge and cloud infrastructure. Intent-driven, in this context, implies that high-level objectives and requirements are specified by application developers rather than low-level resource configurations. Their approach makes use of a reinforcement learning-based autoscaling method that incorporates a reward function that aims to provide optimal utilization of resources and timely autoscaling of the functions, taking advantage of methods for workload prediction. The key message is the importance of efficient resource management and scheduling for serverless functions across edge and cloud environments to meet service level objectives (SLOs) while optimizing resource utilization. A two-stage approach combining reinforcement learning for autoscaling and an optimization-based scheduler for function placement is proposed as a novel contribution. The authors report lessons learned about the effectiveness of their approach in reducing SLO violations and improving resource efficiency. The key takeaway is that intent-driven orchestration can significantly enhance serverless application performance in distributed environments. Experimentally, the authors used a DQN (Deep Q-Network) algorithm to support autoscaling, and formulate an optimization problem for the scheduling component. They evaluated their framework using simulated workloads and compared it against baseline approaches, demonstrating improvements in meeting SLOs and resource utilization.

In **"QoS-aware offloading policies for serverless functions in the Cloud-to-Edge continuum"** Russo et al. [3] consider FaaS with heterogeneous functions and service classes, aiming to maximize utility given a monetary budget for resource usage. Their approach considers a two-level approach: (i) FaaS nodes rely on a randomized policy to schedule every incoming request according to a set of probability values, and (ii) periodically, a linear programming model is solved to determine the probabilities to use for scheduling. They provide a formulation of function offloading for each FaaS node, function and service class as an efficient linear programming problem. Evaluation of their approach is supported through simulation, with a proof-of-concept implementation of the QoS-aware policy developed in Serverledge. The use of QoS classes and resource allocation policies to identify different function types and resources, is also an interesting contribution of this work, which can have a wide adoption within the FaaS community.

In **"SD-SRF: An Intelligent Service Deployment Scheme for Serverless-operated Cloud-Edge Computing in 6G Network"**, Wang et al. [4] explore optimization of service deployment in serverless-operated 6G networks. The work presents two algorithms: (i) SD-SRF, an offline greedy algorithm with Service Rising and Falling phases; (ii) PPO-MSD, an online algorithm utilizing reinforcement learning. These algorithms aim at minimizing latency and deployment costs in multi-layer edge networks. Results show that SD-SRF reduces delay by 32.40% compared to existing approaches, while PPO-MSD achieves a 9.91% reduction, overall demonstrating significant performance improvements and cost-efficiency in cloud-edge service deployments.

The article **"Security computing resource allocation based on deep reinforcement learning in serverless multi-cloud edge computing"** by Zhang et al. [5] proposes an optimized approach for secure resource allocation within serverless cloud-edge environments. The presented method is based on action-constrained deep reinforcement learning and enhances security while minimizing system costs and improving performance. Simulation results show that the proposed method reduces system costs by approximately 24.4% compared to existing methods, demonstrating superior performance and efficient resource allocation in serverless multi-cloud edge environments.

Wang et al. [6] in **"Deep Reinforcement Learning-based scheduling for optimizing system load and response time in edge and fog computing environments"** introduce DRLIS, an algorithm based on deep reinforcement learning that schedules IoT applications across cloud-edge environments operated through serverless. DRLIS aims at minimizing response time while balancing the load among edge servers. The authors implement DRLIS as a scheduler in the FogBus2 FaaS framework. Results show that the proposed algorithm significantly reduces the execution cost of IoT applications by up to 55%, 37%, and 50% in terms of load balancing, response time, and weighted cost respectively, compared to metaheuristic algorithms and other reinforcement learning techniques.

The article **"Rescheduling serverless workloads across the cloud-to-edge continuum"** by Risco et al. [7] focuses on the mitigation of server overload in serverless edge systems.

Specifically, that work proposes a strategy to detect when a server, or cluster of servers, is overloaded and let the server delegate part of its workload to other servers in the cloud-to-edge continuum. The proposed approach is implemented in the OSCAR open-source serverless platform. Results show that the presented mechanism effectively mitigates overloads, reduces execution time, and ensures high availability. In experiments involving a wildfire-detection use case, this strategy significantly improves performance and scalability by dynamically rescheduling jobs across different computational resources.

In the article titled **"Fluidity: Providing flexible deployment and adaptation policy experimentation for serverless and distributed applications spanning cloud–edge–mobile environments"**, Pournaropoulos et al. [8] present a framework for the adaptive deployment of serverless applications across an infrastructure comprising cloud, edge, and mobile nodes. Fluidity enables flexible, policy-driven deployments and runtime adaptations, abstracting infrastructure management from application developers. Performance evaluation, which was conducted on a lab testbed and involved a simulated drone, demonstrates that Fluidity supports flexible application execution with minimal overhead, efficiently handling dynamic adaptations.

Hudson et al. [9] in **"QoS-aware edge AI placement and scheduling with multiple implementations in FaaS-based edge computing"** address optimal function placement and request scheduling for artificial-intelligence services in serverless edge computing environments. Authors first cast the problem as an integer linear problem and then decompose it into two separate placement and scheduling subproblems. Then, they propose an algorithm for placement and request scheduling, which allows intercommunication among edge servers so that they can offload requests from one another. Besides, this work presents FLIES, an algorithm based on federated learning for predicting incoming requests and their QoS requirements, hence to inform placement decisions. Results show the effectiveness of both the proposed algorithms when compared against alternative approaches.

2.2. Data management and data pipelines

In **"StructMesh: A storage framework for serverless computing continuum"** [10], the authors present StructMesh, a storage framework aimed to be a transparent solution to users in a serverless computing continuum system composed of multiple distributed infrastructures. StructMesh is organized into processing and data planes to respectively manage functions as patterns and represent maps of available storage, enabling the integration of storage resources into a unified storage service. A prototype and a case study are used to showcase the harmonization of processing serverless and storage functions in a serverless pipeline, suggesting that the framework can enable continuum data exchange in serverless pipelines, efficiently managing data through a fog-cloud infrastructure.

Bauer et al. [11] in **"The globus compute dataset: An open function-as-a-service dataset from the edge to the cloud"** provide a dataset for machine learning and scientific computing applications which can be widely used with the FaaS community. This data set relates to the use of the Globus Compute platform, which allows users to deploy endpoints on arbitrary remote computers, from edge devices (e.g., Raspberry Pi, Jetson Nano, and personal laptops) to a high performance computing cluster. The dataset covers 31 weeks and includes 2M+ task submissions from 250+ users executed on 600 remote computing endpoints with 277,386 registered functions. Initial analysis of the data set is reported using statistical measures for the system performance, task inter-arrival times, task invocations, function properties (lines of code, cyclomatic complexity, number of imported libraries, etc) and user behavior. Similar to Google data sets available previously, we believe this data set also provides valuable insights into the execution and operation of FaaS systems. The authors indicate that this data set will be made public.

The article **"Evaluating NiFi and MQTT based serverless data pipelines in fog computing environments"** by Mirampalli et al. [12] explores the use of serverless computing to run data pipelines for IoT applications in the cloud-to-edge continuum. Along the journey from data source to a data lake or data warehouse, data may flow through a set of intermediate tasks (e.g., filtering, transformation, aggregation). Authors in that work propose running such intermediate tasks as serverless functions. The study considers two serverless approaches, one based on Apache NiFi and the other on message queuing telemetry transport (MQTT), and compares their performance when processing image streaming data. The main results indicate that while Apache NiFi consumes more CPU resources than MQTT, it performs better in terms of pipeline execution time and memory utilization. The NiFi-based pipeline is recommended for more intensive workloads due to its superior performance and stability.

2.3. Hosting serverless functions on IoT resources

In the article **"FaaS for IoT: Evolving Serverless towards Deviceless in I/Oclouds"**, Merlino et al. [13] introduce a framework based on OpenStack, extending serverless computing to IoT devices at the edge. In other words, the framework allows serverless functions to be hosted directly on IoT devices. Besides, the solution features a dashboard providing a graphical interface for an integrated development environment to aid IoT developers in designing data pipelines. Results demonstrate significant improvements in application setup, management, and latency reduction, validated through experiments showing enhanced resource utilization and system efficiency.

Loconte et al. [14] in **"Expanding the cloud-to-edge continuum to the IoT in serverless federated learning"** present a novel serverless architecture to run artificial intelligence and machine learning applications across the continuum comprising IoT, edge, and cloud nodes. Authors define and implement a framework based on the proposed architecture, wherein IoT devices may run both federated training and inference tasks, without any need for code modifications. The framework is evaluated over a testbed comprising an Amazon AWS cloud, Raspberry Pis and STM32 microcontrollers. Experimental tests on communication latencies, training and prediction time, and accuracy improvement across the federated architecture suggest the feasibility and usefulness of the proposed approach.

2.4. Other works on serverless edge computing

The work **"EneA-FL: Energy-aware Orchestration for Serverless Federated Learning"** [15] presents a scheme to promote smart energy management in highly heterogeneous serverless computing scenarios where Federated Learning clients have energy budgets and different computing capabilities. The interaction between clients and the central aggregator is dynamically adapted based on the energy budget, computational capabilities and target accuracy to promote a client selection for the next training rounds. Experimental analysis shows that EneA-FL is able to reduce energy consumption yet yielding faster convergence of models to the target accuracy.

The authors in **"Exploiting Microservices and Serverless for Digital Twins in the Cloud-to-Edge Continuum"** [16] explore the cloud-to-edge continuum to support digital twins (DTs), introducing a serverless-ready model for digital twins that enables the orchestration and execution of DTs in a cost-effective manner using a microservices environment. The proposed model is implemented in a microservices and two serverless environments considering a cloud provider framework and an on-premises edge deployment. The evaluation of the implemented model allows understanding the quality of the cyber-physical entanglement and also analyzing the performance results of different model setups qualitatively and quantitatively. In detail, the microservices implementation presented in the paper is an order of magnitude more entangled than the serverless deployment, showing reduced exchange of information between DT and the physical system.

In [17] **"Towards providing a priority-based vital sign offloading in healthcare with serverless computing and a fog-cloud architecture"** by Cassel et al. an approach to prioritizing vital sign offloading in healthcare applications using serverless computing and a fog-cloud architecture is presented. In this way individuals with high-priority vital signs achieve short response times, so people with comorbidity can have health problems identified as soon as possible. The proposed system combines fog computing for local processing with cloud computing for handling larger workloads. The key innovation is the use of heuristics to prioritize urgent vital signs and make intelligent offloading decisions based on resource usage – such as CPU utilization, network latency, and the criticality of the patient's condition. A number of experiments are reported to evaluate the system, with the most comprehensive scenario involving 80,000 vital signs, multiple fog nodes, and two healthcare services. Key findings include: (i) The system successfully prioritizes urgent vital signs, with smaller response times for critical cases; (ii) the Ranking Heuristic effectively offloads more data for healthier individuals, demonstrating the efficacy of the prioritization scheme; (iii) in high-load situations, the system offloads more tasks to the cloud; (iv) the architecture shows scalability in handling large volumes of vital signs across different priority levels. Overall, the results suggest that the proposed fog-cloud architecture with serverless computing offers a promising solution for efficient and priority-based processing of healthcare data in smart city environments.

3. Acknowledgments to the Committee of Reviewers

Timely and valuable contribution from Reviewers has been essential for the success of this special issue. We therefore acknowledge the considerable time and effort devoted by Reviewers at assessing manuscripts and providing useful comments for significantly improving quality of submitted works. It is a true pleasure to thank the 35 Reviewers, who we report in alphabetical order:

- Leonardo Bacciottini (*University of Pisa, Italy*)
- Hamza Baniata (*University of Szeged, Hungary*)
- Paolo Bellavista (*University of Bologna, Italy*)
- Victor Chang (*Aston University, UK*)
- Claudia Melania Chituc (*Leibniz Institute for Research and Information in Education, Germany*)
- Claudio Cicconetti (*IIT-CNR, Italy*)
- Chinmaya Kumar Dehury (*University of Tartu, Estonia*)
- Flavia Delicato (*Federal Fluminense University, Brazil*)
- Alireza Ghasempour (*Utah State University, USA*)
- Hadi Gholami (*University of Kansas, USA*)
- Brij Gupta (*National Institute of Technology Kurukshetra, India*)
- Riyaz Ahamed Ariyaluran Habeeb (*University of Malaya, Malaysia*)
- Joan Garcia-Haro (*Universidad Politecnica de Cartagena, Spain*)
- Kannan Govindarajan (*Indiana University Purdue University Indianapolis, USA*)
- Bahman Javadi (*Western Sydney University, Australia*)
- Weiwei Jiang (*Beijing University of Posts and Telecommunications, China*)
- Amin Keshavarzi (*Islamic Azad University, Iran*)
- Francesco Longo (*University of Messina, Italy*)
- Pietro Manzoni (*Universitat Politecnica de Valencia, Spain*)
- Rashid Mehmood (*King Abdulaziz University, Saudi Arabia*)
- Giovanni Merlino (*University of Messina, Italy*)
- Symeon Papavassiliou (*National Technical University of Athens, Greece*)
- Al-Sakib Khan Pathan (*United International University, Bangladesh*)

- Kedar Potdar (*NVIDIA, USA*)
- Antonio Puliafito (*University of Messina, Italy*)
- Praveen Kumar Rayani (*National Institute of Technology Durgapur, India*)
- Roberto Rodrigues Filho (*Federal University of Santa Catarina, Brazil*)
- Zhishu Shen (*Wuhan University of Technology, China*)
- Satish Narayana Srirama (*University of Hyderabad, India*)
- Alex Suja (*St Xavier's Catholic College of Engineering, India*)
- Carlo Vallati (*University of Pisa, Italy*)
- Bo Wang (*Zhengzhou University of Light Industry, China*)
- Kewen Wang (*University of Connecticut, USA*)
- Huan Zhou (*China Three Gorges University, China*)
- Eugenio Zimeo (*University of Sannio, Italy*)

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

[1] M. S. Aslanpour, A. N. Toosi, M. A. Cheema, M. B. Chhetri, M. A. Salehi, Load balancing for heterogeneous serverless edge computing: A performance-driven and empirical approach, Future Generation Computer Systems 154 (2024) 266–280. doi:10.1016/j.future.2024.01.020.

[2] N. Filinis, I. Tzanettis, D. Spatharakis, E. Fotopoulou, I. Dimolitsas, A. Zafeiropoulos, C. Vassilakis, S. Papavassiliou, Intent-driven orchestration of serverless applications in the computing continuum, Future Generation Computer Systems 154 (2024) 72–86. doi:10.1016/j.future.2023.12.032.

[3] G. Russo Russo, D. Ferrarelli, D. Pasquali, V. Cardellini, F. Lo Presti, Qos-aware offloading policies for serverless functions in the cloud-to-edge continuum, Future Generation Computer Systems 156 (2024) 1–15. doi:10.1016/j.future.2024.02.019.

[4] L. Wang, A. Liu, N. N. Xiong, S. Zhang, T. Wang, M. Dong, SD-SRF: An intelligent service deployment scheme for serverless-operated cloud-edge computing in 6G networks, Future Generation Computer Systems 151 (2024) 242–259. doi:10.1016/j.future.2023.09.027.

[5] H. Zhang, J. Wang, H. Zhang, C. Bu, Security computing resource allocation based on deep reinforcement learning in serverless multi-cloud edge computing, Future Generation Computer Systems 151 (2024) 152–161. doi:10.1016/j.future.2023.09.016.

[6] Z. Wang, M. Goudarzi, M. Gong, R. Buyya, Deep reinforcement learning-based scheduling for optimizing system load and response time in edge and fog computing environments, Future Generation Computer Systems 152 (2024) 55–69. doi:10.1016/j.future.2023.10.012.

[7] S. Risco, C. Alarcón, S. Langarita, M. Caballer, G. Moltó, Rescheduling serverless workloads across the cloud-to-edge continuum, Future Generation Computer Systems 153 (2024) 457–466. doi:10.1016/j.future.2023.12.015.

[8] F. Pournaropoulos, A. Patras, C. D. Antonopoulos, N. Bellas, S. Lalis, Fluidity: Providing flexible deployment and adaptation policy experimentation for serverless and distributed applications spanning cloud–edge–mobile environments, Future Generation Computer Systems 157 (2024) 210–225. doi:10.1016/j.future.2024.03.031.

[9] N. Hudson, H. Khamfroush, M. Baughman, D. E. Lucani, K. Chard, I. Foster, Qos-aware edge AI placement and scheduling with multiple implementations in faas-based edge computing, Future Generation Computer Systems 157 (2024) 250–263. doi:10.1016/j.future.2024.03.035.

[10] D. Carrizales-Espinoza, D. D. Sanchez-Gallegos, J. Gonzalez-Compean, J. Carretero, StructMesh: A storage framework for serverless computing continuum, Future Generation Computer Systems 159 (2024) 353–369. doi:10.1016/j.future.2024.05.033.

[11] A. Bauer, H. Pan, R. Chard, Y. Babuji, J. Bryan, D. Tiwari, I. Foster, K. Chard, The globus compute dataset: An open function-as-a-service dataset from the edge to the cloud, Future Generation Computer Systems 153 (2024) 558–574. doi:10.1016/j.future.2023.12.007.

[12] S. Mirampalli, R. Wankar, S. N. Srirama, Evaluating NiFi and MQTT based serverless data pipelines in fog computing environments, Future Generation Computer Systems 150 (2024) 341–353. doi:10.1016/j.future.2023.09.014.

[13] G. Merlino, G. Tricomi, L. D'Agati, Z. Benomar, F. Longo, A. Puliafito, FaaS for IoT: Evolving serverless towards deviceless in I/Oclouds, Future Generation Computer Systems 154 (2024) 189–205. doi:10.1016/j.future.2023.12.029.

[14] D. Loconte, S. Ieva, A. Pinto, G. Loseto, F. Scioscia, M. Ruta, Expanding the cloud-to-edge continuum to the IoT in serverless federated learning, Future Generation Computer Systems 155 (2024) 447–462. doi:10.1016/j.future.2024.02.024.

[15] A. Agiollo, P. Bellavista, M. Mendula, A. Omicini, EneA-FL: Energy-aware orchestration for serverless federated learning, Future Generation Computer Systems 154 (2024) 219–234. doi:10.1016/j.future.2024.01.007.

[16] P. Bellavista, N. Bicocchi, M. Fogli, C. Giannelli, M. Mamei, M. Picone, Exploiting microservices and serverless for digital twins in the cloud-to-edge continuum, Future Generation Computer Systems 157 (2024) 275–287. doi:10.1016/j.future.2024.03.052.

[17] G. André Setti Cassel, R. da Rosa Righi, C. André da Costa, M. Rosecler Bez, M. Pasin, Towards providing a priority-based vital sign offloading in healthcare with serverless computing and a fog-cloud architecture, Future Generation Computer Systems 157 (2024) 51–66. doi:10.1016/j.future.2024.03.032.