JETIR.ORG ISSN: 2349-5162 | ESTD Year : 2014 | Monthly Issue
JOURNAL OF EMERGING TECHNOLOGIES AND INNOVATIVE RESEARCH (JETIR)
An International Scholarly Open Access, Peer-reviewed, Refereed Journal

FAULT TOLERANCE TECHNIQUES IN INTERNET OF THINGS: A NARRATIVE LITERATURE REVIEW

[1]Lilian Cherotich Ronoh, [2]Jackson Muketo Mutua
[1]lronoh@kafu.ac.ke, [2]jacmuketo@gmail.com
[1]Assistant Lecturer [2]Part-Time Assistant Lecturer
[1]Department on Information Technology and Informatics, Kaimosi Friends University, Vihiga, Kenya
[2]Department on Information Technology and Informatics, Kaimosi Friends University, Vihiga, Kenya

Abstract: Fault tolerance is the ability of an IoT system to continue functioning correctly despite faults, is essential due to the distributed and resource-constrained nature of IoT deployments. The review presents current research on fault tolerance techniques across different layers of the IoT architecture – sensing, routing, and control as well as general fault tolerance mechanisms like redundancy, data replication, and consensus protocols. It also examines how architectural choices, such as distributed, layered, and hybrid approaches, influence resilience. Key theoretical concepts underpinning fault tolerance, including dependability and the Fault-Error-Failure Cycle, are discussed. The review highlights the emphasis on redundancy, the growing importance of data replication, and the evolution of consensus mechanisms for IoT. Challenges such as scalability, energy efficiency, and the need for standardization and real-world validation are identified. Future research directions include developing more scalable and energy-efficient techniques, exploring cross-layer optimization, and addressing Byzantine faults. This review serves as a valuable resource for understanding and implementing fault tolerance in IoT systems.

Keywords: *IoT, Fault tolerance, Failure, Faults, Techniques, Mechanisms, IoT architecture*

I. INTRODUCTION

The Internet of Things (IoT) has rapidly evolved into a pervasive technological landscape, characterized by a vast network of interconnected physical and virtual objects capable of collecting, exchanging, and acting upon data [1]. This interconnectedness promises enhanced efficiency, automation, and intelligence across diverse domains. However, the very nature of IoT systems, with their distributed, resource-constrained devices operating in potentially harsh environments, renders them susceptible to various faults and failures. Fault tolerance, the ability of an IoT system to continue operating correctly despite the occurrence of faults in its components, is therefore a critical design consideration to ensure the reliability, availability, and integrity of IoT applications [2].

This literature review aims to provide a comprehensive overview of fault tolerance techniques in the IoT, drawing upon a diverse set of recent research. The scope of this review encompasses various approaches to achieving fault tolerance across different layers of IoT architectures and for the main functionalities performed by IoT devices, namely, sensing, routing, and control. It also examines general fault tolerance mechanisms applicable to IoT systems and discusses the architectural considerations for building resilient IoT deployments.

The purpose of this review is threefold: first, to synthesize the existing body of knowledge on fault tolerance in IoT; second, to identify key trends, challenges, and gaps in the current research; and third, to highlight potential directions for future investigation in this increasingly important field. By consolidating the findings from various studies, this review aims to provide a valuable resource for researchers, practitioners, and developers seeking to understand and implement fault tolerance in their IoT systems.

This literature review addresses the following key research questions:

i. What are the primary types of faults and failures that affect IoT systems?
ii. What fault tolerance techniques are employed at different layers of the IoT architecture, including the sensing, network (routing), and control layers?
iii. What general fault tolerance mechanisms, such as redundancy, data replication, and consensus protocols, are utilized in IoT?
iv. How do architectural choices and styles influence the implementation and effectiveness of fault tolerance in IoT systems?
v. What are the current trends, challenges, and open research issues in the field of fault tolerance for the Internet of Things?

The remainder of this review is structured as follows: Section 2 discusses relevant theoretical concepts underpinning fault tolerance in distributed systems, which provide a foundation for understanding IoT resilience. Section 3 presents a detailed review of fault tolerance techniques, organized thematically based on their application in sensing, routing, control, and as general mechanisms. Section 4 offers a critical analysis and synthesis of the reviewed literature, highlighting gaps, agreements, and limitations. Finally, Section 5 concludes the

review by summarizing key findings, discussing their implications, and suggesting future research directions.

II. THEORETICAL FRAMEWORK

The design and implementation of fault-tolerant IoT systems often draw upon fundamental concepts and models from the field of distributed systems and dependability engineering [3]. Several key theoretical underpinnings guide the research and development of resilience mechanisms in IoT.

Dependability

It is defined as the ability of a computing system to be trusted such that the service it offers can be justifiably relied upon. Dependability encompasses several attributes, including availability (readiness for correct service), reliability (continuity of correct service), integrity (absence of improper system alterations), and safety (absence of catastrophic consequences on the user(s) and the environment) [4]. Fault tolerance is a crucial means of achieving these dependability attributes in the presence of faults.

Fault-Error-Failure Cycle

It provides a fundamental model for understanding how system failures occur. A fault is an imperfection or defect in the system (e.g., a hardware malfunction, a software bug) [5]. A fault can lead to an error, which is a deviation from the correct internal state of the system. An error, in turn, can propagate and eventually cause a failure, which is the inability of the system to deliver its intended service. Fault tolerance aims to break this cycle by preventing faults from leading to errors or by masking errors before they cause failures.

Redundancy

It is a core principle in fault tolerance, involving the provision of extra resources (hardware, software, information, or time) beyond what is strictly needed for normal operation. The rationale behind redundancy is that if one component fails, the redundant components can take over, ensuring continued service. Different forms of redundancy exist:

Hardware Redundancy: Involves replicating hardware components, such as sensors, actuators, processing units, or communication links. This can be active (hot) redundancy, where all redundant components operate simultaneously, or passive (cold/standby) redundancy, where backup components are activated only upon the failure of the primary component [6].

Software Redundancy: Employs multiple versions of software to perform the same task, often with different design or implementation approaches such as N-Version Programming [7].

Information Redundancy: Involves adding extra information, such as error-detecting or error-correcting codes, to data to detect and potentially correct errors that may occur during transmission or storage [5].

Time Redundancy: Repeats operations or tasks in time to mask transient faults or to allow for recovery.

Failure Models

Understanding failure models is also crucial for designing effective fault tolerance mechanisms. Common failure models include fail-stop, where a component simply ceases to function and signals its failure; omission failures, where a component fails to perform an action; timing failures, where a component's response is too early or too late; and byzantine failures, where a component behaves arbitrarily, possibly producing incorrect or conflicting outputs [8]. Byzantine fault tolerance is particularly challenging to address, often requiring sophisticated consensus mechanisms [9].

Layered Architectural Models

The model that is commonly used for IoT systems (e.g., perception, network, application) provide a framework for considering where and how fault tolerance mechanisms can be implemented most effectively. Faults can occur at any layer, and the choice of fault tolerance techniques often depends on the specific characteristics and failure modes of the components at each layer [10].

III. REVIEW OF LITERATURE

This section reviews the literature on fault tolerance in IoT, organized thematically based on the specific aspects of IoT systems and the types of fault tolerance techniques employed.

Fault Tolerance in Sensing

The sensing layer, comprising various sensor devices, is the foundation of data acquisition in IoT systems. These devices are often resource-constrained and deployed in diverse and potentially challenging environments, making them prone to failure [11]. Several techniques have been proposed to enhance the fault tolerance of sensing functionalities.

Redundant Sensor Deployment: Strategically deploying multiple sensors that measure the same physical phenomenon can provide hardware redundancy. If one sensor fails or provides erroneous readings, the data from other redundant sensors can be used to ensure data availability and accuracy [12].

Sensor Fusion Algorithms: These algorithms combine data from multiple sensors to improve the accuracy, reliability, and robustness of the sensed information. By fusing data from potentially faulty sensors, these algorithms can detect inconsistencies, filter out erroneous readings, and provide a more reliable estimate of the measured parameter [13].

Distributed Fault Detection: Tasks can be distributed across the network to detect faulty sensors based on inconsistencies in their readings or deviations from expected behaviour. Previous studies proposed a distributed protocol for detecting node replication attacks in wireless sensor networks [14].

Data Replication at Sensor Nodes: Data Recovery Approach or Fault Tolerant (DRAFT) IoT node algorithm, is a fully distributed approach where each IoT node maintains a redundant local database containing data from its neighbour nodes within the cluster [15]. If a node fails, its data can be retrieved from these redundant copies, enhancing data availability. The algorithm generates parity information for data recovery.

Energy-Aware Approaches: Since sensor nodes often rely on limited battery power, energy-aware fault tolerance techniques aim to balance resilience with energy efficiency to prolong network lifetime. An Energy Efficient Message scheduling algorithm (EAAFTMS) that incorporates backup data storage to avoid re-clustering overhead and increase availability while being energy efficient was proposed [16].

Fault Tolerance in Routing

Ensuring reliable and efficient data delivery across the IoT network in the presence of node failures and link disruptions is crucial. Fault-tolerant routing protocols play a vital role in achieving this.

Redundant Paths and Multipath Routing: These protocols establish multiple paths between source and destination nodes. If the primary path fails due to a node or link failure, data can be rerouted through an alternative path, enhancing resilience. Data backup on different virtual machines to maintain connectivity in case of network failures [10].

Randomized Routing: By introducing randomness in the path selection process, these protocols can avoid congested or faulty areas of the network and improve the probability of successful data delivery [4].

Fault Detection and Recovery Mechanisms in Routing Protocols: Some routing protocols incorporate mechanisms to detect faulty nodes or links and dynamically adapt the routing paths to avoid them. [9] proposed an efficient fault-tolerant routing in IoT Wireless Sensor Networks based on bipartite-flow graph modelling.

Clustering-Based Approaches: In clustered IoT networks, the failure of a cluster head can disrupt communication. Fault tolerance can be achieved by having backup cluster heads or mechanisms for electing a new cluster head in case of failure.

Machine Learning for Adaptive Routing: Machine learning techniques are being explored to develop routing protocols that can learn from network conditions, predict potential failures, and dynamically adjust routing decisions to maintain connectivity and performance [16].

Fault Tolerance in Control

Control devices in IoT systems often implement state machines that make critical decisions based on sensor data and may trigger actions through actuators. Ensuring the fault tolerance of these control functions is essential for the correct operation of the entire system. This is achieved through:

State Machine Replication: This is a traditional approach to fault tolerance for stateful systems, where multiple replicas of the control device execute the same operations and maintain consistent states. Consensus protocols are typically used to ensure that all replicas agree on the sequence of operations, even in the presence of failures [17]

Consensus Protocols: Classical consensus algorithms, such as Paxos and Raft, enable a distributed set of nodes to agree on a single value or a sequence of actions, even if some nodes fail. However, [4] note that traditional consensus protocols developed in the 80s and 90s may have limitations for the dynamic and resource-constrained nature of IoT environments.

Blockchain-Based Consensus: The emergence of blockchain technology has introduced new approaches to achieving distributed consensus. Blockchain-based protocols can provide fault tolerance and data integrity in decentralized systems. Modifications are needed to apply standard blockchain protocols effectively for fault tolerance in IoT control devices [10].

Distributed State Machines: The ability for control devices to implement distributed state machines that can change in size and capabilities over time offers a way to achieve more dynamic fault tolerance in IoT control.

General Fault Tolerance Mechanisms in IoT

Beyond the layer-specific techniques, several general mechanisms contribute to the overall fault tolerance of IoT systems:

Redundancy: Various forms of redundancy (hardware, software, information, time) are widely employed to enhance fault tolerance at different levels of IoT systems. Previous studies proposed a hybrid architecture that simultaneously uses proactive and reactive policies, likely leveraging redundancy in its structure [5].

Failure Detection and Diagnosis: Timely and accurate detection of faults is the first step towards achieving fault tolerance. Various techniques are used for fault detection, including self-detection, neighbour monitoring, and dedicated fault detection nodes.

Fault Isolation and Containment: Once a fault is detected, it is crucial to isolate the affected component or subsystem to prevent the error from propagating to other parts of the system.

Fault Recovery: Recovery mechanisms aim to restore the system to a correct operating state after a fault has occurred. This can involve techniques like checkpointing, rollback, and the activation of redundant components.

Proactive Fault Tolerance: This approach focuses on predicting potential faults before they occur and taking preventive actions, such as migrating critical tasks to healthier nodes or performing pre-emptive maintenance. [18] presented a smart home architecture using neural networks to predict sensor data based on correlations, providing redundancy and job migration.

Reactive Fault Tolerance: This approach involves responding to faults after they have been detected, typically through mechanisms like fault detection, isolation, and recovery. The hybrid architecture proposed by [19] utilizes all three types of reactive policies simultaneously

Virtualization: Virtualization technologies can provide an abstraction layer that enhances fault tolerance by enabling the rapid replacement or failover of virtualized components in case of underlying hardware failures. [20] discussed fault-tolerant embedding using virtualization in wireless sensor networks.

Architectural Considerations for Fault Tolerance

The architecture of an IoT system significantly influences its ability to tolerate faults. Different architectural patterns and styles offer varying degrees of resilience:

Distributed Architectures: The inherently distributed nature of IoT can be leveraged for fault tolerance by distributing critical functionalities across multiple independent nodes. The failure of a single node is less likely to bring down the entire system [21].

Layered Architectures: The typical layered models of IoT (e.g., perception, network, application) allow for the implementation of fault tolerance mechanisms at each layer, tailored to the specific failure modes of the components at that layer. Previous studies focused on fault tolerance at the actuate and sense layers [16].

Hybrid Architectures: Combining centralized and decentralized elements can offer a balance between performance and fault tolerance. [19] proposed a hybrid architecture using both proactive and reactive fault tolerance policies [14].

Service-Oriented Architectures (SOA) and Microservices: These architectural styles, where applications are composed of loosely coupled services, can enhance fault isolation and resilience. The failure of one service is less likely to impact other services, and failed services can potentially be restarted or replaced independently [16].

Cloud and Edge Computing Integration: Offloading some processing and control functions to the cloud or edge can provide access to more powerful and potentially more resilient infrastructure. [24] presented a Cloud and Edge Fault-Tolerant IoT (CEFIoT) layered design, a fault-tolerant IoT architecture for edge and cloud.

IV. ANALYSIS & SYNTHESIS

The reviewed literature highlights significant research efforts in addressing fault tolerance in the Internet of Things. Several key observations and insights emerge from this analysis:

Emphasis on Redundancy: Redundancy in various forms (hardware, data, paths) remains a fundamental approach to achieving fault tolerance across different layers of IoT systems.

Data Replication and Recovery: With the increasing volume and value of data generated by IoT devices, ensuring data availability in the face of node failures is a critical concern, leading to the development of distributed data replication and recovery techniques like DRAFT [14].

Consensus Mechanisms: While traditional consensus protocols provide robust fault tolerance for control functions, their applicability in dynamic and resource-constrained IoT

environments is being challenged, leading to exploration of blockchain-based and other more flexible approaches [23].

Hybrid and Layered Approaches: Recognizing that faults can occur at any layer, a growing emphasis is placed on designing fault tolerance mechanisms that are integrated across different layers of the IoT architecture and leverage a combination of proactive and reactive strategies [19].

Machine Learning: Machine learning techniques are increasingly being used for fault detection, prediction, and adaptive decision-making in routing and resource management, offering the potential for more intelligent and proactive fault tolerance [21].

Energy Efficiency as a Key Constraint: Given that many IoT devices are battery-powered, balancing fault tolerance with energy efficiency is a persistent challenge. Research continues to explore energy-aware fault tolerance mechanisms to prolong network lifetime [10].

Limited Focus on Cross-Layer Fault Tolerance in Specific Domains: Previous studies specifically noted a lack of research covering cross-layer fault tolerance in Underwater Sensor Networks (USNs) [16].

Despite the advancements, several gaps and limitations in the existing literature can be identified:

Scalability of Fault Tolerance Mechanisms: Ensuring that fault tolerance techniques remain effective and efficient as IoT deployments scale to include massive numbers of devices remains a significant challenge.

Complexity and Management Overhead: Implementing redundancy and other fault tolerance mechanisms often increases the complexity of IoT systems and introduces management overhead. Striking the right balance between resilience and complexity is crucial [10].

Standardization and Interoperability: The lack of widely adopted standards for fault tolerance in IoT can hinder interoperability and make it difficult to integrate fault-tolerant solutions from different vendors [23].

Real-World Validation: Many proposed fault tolerance techniques are evaluated through simulations. More real-world deployments and experimental validations are needed to assess their effectiveness in practical scenarios.

Addressing Byzantine Faults: While some research touches upon security aspects and attacks, more focused work is needed on effectively addressing Byzantine faults, where compromised IoT devices may exhibit arbitrary and potentially malicious behaviour.

V. CONCLUSION

This literature review has provided a comprehensive overview of fault tolerance techniques in the Internet of Things, highlighting the diverse strategies employed across sensing, routing, control, and as general mechanisms. The findings underscore the critical importance of fault tolerance in ensuring the dependability of IoT systems and the significant research efforts dedicated to this domain. Redundancy, data replication, and evolving consensus protocols form the bedrock of many fault tolerance approaches, while layered and hybrid architectures provide frameworks for integrating resilience mechanisms. The emerging role of machine learning offers promising avenues for more intelligent and adaptive fault tolerance.

The implications of the reviewed literature are significant for the design and deployment of reliable IoT applications. Understanding the trade-offs between different fault tolerance techniques is crucial for selecting the most appropriate solutions for specific use cases.

Future research directions in fault tolerance for the Internet of Things are manifold:

- Developing more scalable and energy-efficient fault tolerance mechanisms to support large-scale IoT deployments with resource-constrained devices.
- Investigating cross-layer optimization of fault tolerance techniques to leverage synergies between different layers and improve overall system resilience.
- Addressing the challenges of Byzantine fault tolerance in IoT environments where malicious nodes may be present.
- Developing standardized frameworks and methodologies for designing, implementing, and evaluating fault-tolerant IoT systems to enhance interoperability and facilitate wider adoption.
- Conducting more extensive real-world evaluations and deployments of proposed fault tolerance techniques to validate their effectiveness in practical settings.
- Investigating lightweight and resource-aware fault tolerance solutions tailored for specific IoT application domains, such as industrial IoT (IIoT), healthcare, and environmental monitoring.

By addressing these future research directions, the field of fault tolerance in the Internet of Things can continue to advance, paving the way for the development of increasingly reliable, available, and trustworthy IoT systems that can deliver their full potential across various critical applications.

REFERENCES

[1] Dhawan, D., Ahamad, F., & Tripathi, M. M. (2022). A System Model of Fault Tolerance Technique in the Distributed and Scalable System: A Review. International Journal of Innovative Research in Science Engineering and Technology. doi:10.55524/ijircst.2022.9.1.14

[2] Melo, M., & Aquino, G. (2021). FaTEMa: A Framework for Multi-Layer Fault Tolerance in IoT Systems. Sensors, 1-27. doi:https://doi.org/10.3390/s21217181

[3] Chakraborty, R. S., Mathew, J., & Vasilakos, A. V. (Eds.). (2019). Security and Fault Tolerance in Internet of Things. Springer International Publishing. https://doi.org/10.1007/978-3-030-02807-7

[4] Rullo, A., Serra, E., & Lobo, J. (2019). Redundancy as a Measure of Fault-Tolerance for the Internet of Things: A Review. In S. Calo, E. Bertino, & D. Verma (Eds.), Policy-Based Autonomic Data Governance (Vol. 11550, pp. 202–226). Springer International Publishing. https://doi.org/10.1007/978-3-030-17277-0_11

[5] Agrawal, A., & Toshniwal, D. (2021). Fault Tolerance in IoT: Techniques and Comparative Study. ASIAN JOURNAL OF CONVERGENCE IN TECHNOLOGY, 7(1), 49–52. https://doi.org/10.33130/AJCT.2021v07i01.011

[6] Chandra, P. (2025). Building Fault-Tolerant Systems with Redundancy and Recovery Mechanisms in Distributed Environments. International Journal of Scientific Research in Computer Science, Engineering and Information Technology, 2616-2625. doi:doi : https://doi.org/10.32628/CSEIT251112251

[7] Kozar, A., Del Monte, B., Zeuch, S., & Markl, V. (2024). Fault Tolerance Placement in the Internet of Things. Proceedings of the ACM on Management of Data, 2(3), 1–29. https://doi.org/10.1145/3654941

[8] Vedavalli, P., & Deepak, C. (2020). Enhancing Reliability and Fault Tolerance in IoT. 2020 International Conference on Artificial Intelligence and Signal Processing (AISP), 1–6. https://doi.org/10.1109/AISP48273.2020.9073174

[9] Lin, J.-W., Chelliah, P. R., Hsu, M.-C., & Hou, J.-X. (2021). Efficient Fault-Tolerant Routing in IoT Wireless Sensor Networks Based on Bipartite-Flow Graph

Modeling. IEEE Access, 7, 14022–14034. https://doi.org/10.1109/ACCESS.2019.2894002

[10] Kumar, S., Ranjan, P., Singh, P., & Tripathy, M. R. (2020). Design and Implementation of Fault Tolerance Technique for Internet of Things (IoT). 2020 12th International Conference on Computational Intelligence and Communication Networks (CICN), 154–159. https://doi.org/10.1109/CICN49253.2020.9242553

[11] Jammalamadaka, K. R., Bhupati, C., Bhanu, J., & Balakrishna, K. D. (2025). Making IoT Networks Highly Fault-Tolerant Through Power Fault Prediction, Isolation and Composite Networking in the Device Layer. Journal of Sensor and Actuator Networks. doi:https://doi.org/10.3390/jsan14020024

[12] Power, A., & Kotonya, G. (2019). Complex Patterns of Failure: Fault Tolerance via Complex Event Processing for IoT Systems. 2019 International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), 986–993. https://doi.org/10.1109/iThings/GreenCom/CPSCom/SmartData.2019.00173

[13] Desikan, J., Sushil, K. S., Jayanthiladevi. D, A., Shashi, B., Vinay, R., & Manish, K. (2025). Hybrid Machine Learning-Based Fault-Tolerant Sensor Data Fusion and Anomaly Detection for Fire Risk Mitigation in IIoT Environment. Sensors. doi:https://doi.org/10.3390/s25072146

[14] Frohlich, A. A., Scheffel, R. M., Kozhaya, D., & Verissimo, P. E. (2019). Byzantine Resilient Protocol for the IoT. IEEE Internet of Things Journal, 6(2), 2506–2517. https://doi.org/10.1109/JIOT.2018.2871157

[15] Mehanovic, A., Rasmussen, T. H., & Kjargaard, M. B. (2018). Brume—A Horizontally Scalable and Fault Tolerant Building Operating System. 2018 IEEE/ACM Third International Conference on Internet-of-Things Design and Implementation (IoTDI), 84–95. https://doi.org/10.1109/IoTDI.2018.00018

[16] Bakhshi Kiadehi, K., Rahmani, A. M., & Sabbagh Molahosseini, A. (2021). A fault-tolerant architecture for internet-of-things based on software-defined networks. Telecommunication Systems, 77(1), 155–169. https://doi.org/10.1007/s11235-020-00750-1

[17] Zou, Y., Li, Y., Guanlin, J., Ruirui, Z., Zhenzhen, X., Huiqun, L., & Yu, D. (2024). A survey of fault tolerant consensus in wireless networks. High-Confidence Computing, doi:https://doi.org/10.1016/j.hcc.2024.100202

[18] Choubey, P. K., Pateria, S., Saxena, A., Vaisakh Punnekkattu Chirayil Sb, Jha, K. K., & Sharana Basaiah Pm. (2015). Power efficient, bandwidth optimized and fault tolerant sensor management for IOT in Smart Home. 2015 IEEE International Advance Computing Conference (IACC), 366–370. https://doi.org/10.1109/IADCC.2015.7154732

[19] Nazari Cheraghlou, M., Khadem-Zadeh, A., & Haghparast, M. (2019). A New Hybrid Fault Tolerance Approach for Internet of Things. Electronics, 8(5), 518. https://doi.org/10.3390/electronics8050518

[20] Kaiwartya, O., Abdullah, A. H., Cao, Y., Lloret, J., Kumar, S., Shah, R. R., Prasad, M., & Prakash, S. (2018). Virtualization in Wireless Sensor Networks: Fault Tolerant Embedding for Internet of Things. 5(2), 571–580. https://doi.org/10.1109/JIOT.2017.2717704

[**21**] Muccini, H., & Moghaddam, M. T. (2018). IoT Architectural Styles: A Systematic Mapping Study. In C. E. Cuesta, D. Garlan, & J. Pérez (Eds.), Software Architecture (Vol. 11048, pp. 68–85). Springer International Publishing. https://doi.org/10.1007/978-3-030-00761-4_5

[22] Zhou, S., Lin, K.-J., Na, J., Chuang, C.-C., & Shih, C.-S. (2015). Supporting Service Adaptation in Fault Tolerant Internet of Things. 2015 IEEE 8th International Conference on Service-Oriented Computing and Applications (SOCA), 65–72. https://doi.org/10.1109/SOCA.2015.38

[23] Grover, J., & Garimella, R. M. (2018). Reliable and Fault-Tolerant IoT-Edge Architecture. 2018 IEEE SENSORS, 1–4. https://doi.org/10.1109/ICSENS.2018.8589624

[24] Asad, J., Robert, J., Heljanko, K., & Framling, K. (2020). IoTEF: A Federated Edge-Cloud Architecture for Fault-Tolerant IoT Applications. Journal of Grid Computing. doi::10.1007/s10723-019-09498-8