Design Challenges in Hardware Development of Time-Sensitive Networking: A Research Plan

Adnan Ghaderi, Masoud Daneshtalab, Mohammad Ashjaei, Mohammad Loni, Saad Mubeen, and Mikael Sjödin

Mälardalen University, Västerås, Sweden
{adnan.ghaderi, masoud.daneshtalab, mohammad.ashjaei, mohammad.loni, saad.mubeen, mikael.sjodin}@mdh.se

Abstract. Time-Sensitive Networking (TSN) is a set of ongoing projects within the IEEE standardization to guarantee timeliness and low-latency communication based on switched Ethernet for industrial applications. The huge demand is mainly coming from industries where intensive data transmission is required, such as in the modern vehicles where cameras, lidars and high-bandwidth modern sensors are connected. The TSN standards are evolving over time, hence the hardware needs to change depending upon the modifications. In addition, high performance hardware is required to obtain a full benefit from the standards. In this paper, we present a research plan for developing novel techniques to support a parameterized and modular hardware IP core of the multi-stage TSN switch fabric in VHSIC (Very High Speed Integrated Circuit) Hardware Description Language (VHDL), which can be deployed in any Field-Programmable-Gate-Array (FPGA) devices. We present the challenges on the way towards the mentioned goal.

Keywords: Time-Sensitive Network · FPGA · Predictability · Memory Management

1 Introduction

Nowadays, modern vehicles demand a more complex computing system in order to monitor and control different vehicular systems. In particular, there is high load on Electronic Control Units (ECUs) to process intensive data coming from high-resolution sensors, such as radar, lidar, cameras and ultrasonic sensors. This complexity is not only on the computation power, but also on the communication infrastructure to transfer the high amount of data. Conventional communication infrastructure, e.g., Controller Area Networks, cannot support this demand, mainly due to limited communication bandwidth support. Therefore, the research community already recommends using high-bandwidth on-board communication networks with solid ECUs [1, 2]. Switched Ethernet is suggested to become a backbone network for on-board vehicular communication networks. However, lack of real-time capabilities in Ethernet prevented the use of this protocol in industrial domains, including the vehicular domain. In a recent

effort to support high-bandwidth industrial communications, the IEEE Time Sensitive Networking task group has developed a set of standards [3–5] obtaining many features to be used in real-time applications. TSN-based architectures are obtaining more attractions thanks to the features of TSN, such as precise time synchronization, low-latency transmission and supporting several real-time traffic classes. Current research works can not take full advantage of the TSN for a model-based software development environment and hardware realization. Full-fledged TSN-based development could be useful for the vehicular industry regarding on-board communication.

1.1 Motivation

The appearing of the TSN standards has sparked a huge interest among vehicular industries which is evident in their activities and research towards obtaining high performance communication design. Several Tier-1 suppliers already provide TSN hardware, while the big challenges are in how to utilize TSN in products and how to efficiently design such hardware. The TSN standards are evolving over time, which means that the tools and hardware should be adapted time to time. This leads to provide a solution based on a fully modular and parameterized hardware. To this end, not much work has been done regarding such activity to design and develop a fully parameterized modular TSN hardware, which is the main goal of this research plan.

1.2 Research plan

For taking full advantage of TSN, we are going to benefit new techniques for timing analysis to verify predictability, designing hardware switch fabric to match the evolving TSN standards, deploying and executing TSN-based vehicular applications. Fig. 1 illustrates the envisioned workflow in our research and the proposed scientific techniques, prototypes and industrial demonstrators. In this workflow we will perform the following:

- First, new analysis techniques are required to allow modelling of software functions that use TSN and support verification of timing properties of these functions. This environment should be complemented with a tool chain that implements the techniques and a proof-of-concept validator to demonstrate its usability in industrial settings.
- In the second part, we are going to deal with the hardware implementation of the TSN switch fabric. To match with the evolving nature of the TSN standards we target using component-off-the-shelf (COTS) FPGA which is a re-programmable hardware device. FPGAs are well established in a variety of specialized applications, such as high-throughput communications and real-time image processing, as they can profit from long pipelines, on-chip data-reuse and customized scheduling algorithms. The hardware IP design should be modular and parameterized, which bring reusability and facilitate the upgrade of the implementation as demanded. Moreover, parameterized hardware for various applications in which the hardware is used.

- Finally, in the third part, we provide a proof of concept by implementing the proposed techniques in an industrial tool chain and demonstrating their availability on industrial use-cases.

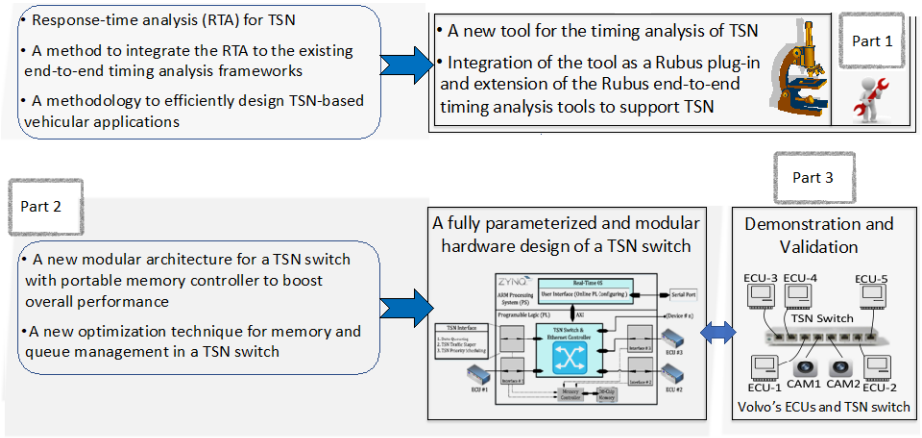

Fig. 1. Envisioned workflow: new scientific techniques and methods to be developed are depicted on the left; whereas the new tools, models and prototypes to be developed are shown in the right.

1.3 Outline

The paper is organized as follows. Section 2 describes the background. Section 3 describes the method and ongoing works. Finally, the summary of the research plan is presented in Section 4.

2 Background

In this section, we present the background and prior work related to the research plan, including TSN descriptions and features, predictability and hardware implementation.

2.1 Time Sensitive Networking

More than a decade, Ethernet has been used in many applications [6–9]. Increasing bandwidth and reducing costs put Ethernet in a better situation compared with traditional communication protocols, e.g., Controller Area Network (CAN) and FlexRay. The IEEE 802.3 Ethernet Working Group announced few amendments to the standards in response to improving industry requirements for

emerging Ethernet applications in recent years. However, new applications demand more bandwidth and real-time communication, thus the TSN task group aimed to extend standard Ethernet to guarantee the required bandwidth and real-time Communication. The IEEE TSN task group has been working on an extension of the IEEE 802.3 Ethernet for real-time capability since 2012. The TSN standards benefit from continuing improvements in Ethernet security, bandwidth and other capabilities. Among several capabilities, the task group developed time-aware traffic shaper (IEEE 802.1Qbv) [4], frame preemption support (IEEE 802.1Qbu) [5], clock synchronization (IEEE 802.1AS) [10] and path control and reservation (IEEE 802.1Qca) [11]. These capabilities improve Quality of Service (QoS) of Ethernet to guarantee frame transmission through switched networks, providing the inherent robustness and reliability. Based on these improvements, TSN obtained remarkable attention from the research community and the industry. Therefore, many works have recently investigated different aspects of TSN [12–16].

2.2 Predictability and timing Analysis

Distributed safety-critical applications in industry to achieve the proper realization of time behavior, needs to analyze a worst-case end-to-end latency. By using TSN, distributed applications without sacrificing real-time features can be built on top of standard Ethernet technologies. Several schedulability analysis techniques are proposed, such as [17] and [18], to compute the worst-case delay of traffic when only credit-based shaper is in place. The very recent work [19] proposed the notion of Eligible Interval to calculate the bounded delay per message, which delivers a tighter bound on the delay compared to the previous techniques. The TSN standards introduce other shapers than the credit-based shaper and new schedulability analysis techniques have been proposed, such as [20] and [21] based on the network calculus method. The main challenges with respect to predictability analysis are to consider all TSN features and shapers, which is not sufficiently studied.

2.3 Hardware implementation

FPGA is a promising platform in many industrial applications because of providing low energy consumption, re-programmability and/or re-configurability and higher flexibility. In addition, it offers better time-to-market compared to Application-Specific-Integrated-Circuit (ASIC) platforms [22]. FPGAs could be leveraged for implementing TSNs due to their extensive benefits including:

1. **Re-configurability**: The TSN protocols can be implemented in a customizable manner thanks to the FPGA reconfigurability. The design can be easily upgraded and/or replaced with the new TSN protocols. This feature highly diminishes the cost of design when it is upgraded.
2. **Supporting parallel designs**: FPGAs can be used to implement multiple data processing pipelines in parallel. Therefore, it supports high-throughput

designs. In addition, FPGAs can be used for developing time predictable designs due to supporting the Worst Case Execution Time (WCET) analysis.

3. **Security**: Nowadays, considering security issues is vital in communication protocols. FPGA provides hardware level security for TSN designs due to integrated security features such as differential power analysis protection, cryptography IP-cores and advanced encryption standards [23].
4. **Energy efficiency**: FPGA is an energy efficient solution compared to many COTS processing platforms such as Graphic Processing Units (GPUs) and CPUs [24]. This feature makes FPGAs a potential platform specially for low-power implementations of TSNs.
5. **Workload acceleration**: Recently, FPGA has become an interesting platform for accelerating specific-purpose applications such as machine learning [25], database processing [26] and biological sequence alignment [27].

FPGAs are widely used in Ethernet network [28–30]. However, there are only few works that have addressed the hardware realization of TSN. Groß et al. [31] have proposed a TSN node architecture design where the time-sensitive and computation-intensive network functions are implemented in dedicated hardware modules to reduce the CPU load. Their hardware/software co-design approach flexibly allocates network function to be executed completely in hardware, completely in software, or in both based on the dynamic load. The performance evaluations from a prototype implementation shows a significant reduction in the CPU load compared to a software. Liß et al. [32] introduce TrustNode, the architecture of a novel networking device that provides low-latency switching and routing. It integrates an FPGA with a standard x86-64 processor, which also targets TSN. It features frequency synchronization across networks and is easily extendable. There are also some commercial solutions for TSN hardware design[1], with not much information about the design.

These solutions lack the scalability and do not take advantage of modern memory technologies. Moreover, the existing solutions suffer from the inflexibility with respect to the evolving TSN standards. In our research we implement our design in an SoC FPGA that merges both the processor and the FPGA subsystems into a single device. The design will be analyzed and evaluated in a multi-port Zynq-based TSN board using both synthetic and real benchmarks. The idea for using Zynq-based is to use the collaboration of hardware and software(hardware/software co-design) to optimize cost, performance and power of design [33]. There are several benefits to use HW/SW co-design approach including:

1. **Better integration**: Reducing design time and product cost.
2. **Faster integration**: More reliability and performance.
3. **Verified integration**: Fewer errors compared to using separate design leading to faster verification.

[1] https://soc-e.com/mtsn-kit-a-comprehensive-multiport-tsn-setup/

3 Methods and ongoing Work

This paper presents a research plan for developing novel techniques to support timing predictability analysis and hardware realization that use TSN as the backbone for onboard network communication. Using TSN guarantees higher bandwidth and real-time communication for new applications. Moreover, TSN benefits from continuing improvements in Ethernet capabilities which mentioned in section 2.1. However, for taking full advantage of the TSN in our research the following challenges need to be addressed.

Real-time: The TSN standards present several traffic scheduling mechanism, such as credit-based shaping, time-aware shaping and preemption mechanisms, to support different latency guarantees. When developing the hardware the scheduling mechanisms should be considered for each port with the performance as high as possible.

Reliability: To improve the reliability, IEEE 802.1Qca amendment of TSN uses a multipath route between nodes that want to communicate. There are several techniques using multipath routing like temporal redundancy [34, 35] and spatial redundancy [36]. However, using these techniques will increase the cost and size of the system. Therefor, we need to investigate a fault tolerance mechanism for TSN considering a time redundancy, cost and size of the system to improve the reliability.

Scalability and high throughput: Besides all the advantages of TSN, high-performance, aggregate throughput and scalable switches are notable features when designing switches for large-scale networks such as data center network environment. In our research, we can use either single-stage crossbar switches or multi-stage switches. However, single-stage switches can be implemented for small-sized switches, and they become too complex to design and unscalable for higher ports [37]. Multi-stage switches are a better approach than single-stage switches because they can be incrementally expanded by adding more modules to the existing design and also they are more scalable. They can use multiple routes between inputs and outputs to allow the traffic to be balanced across alternative paths [38].

Timing and synchronization: All TSN network nodes are synchronized by IEEE 802.1AS-Rev standard. This standard is based on IEEE 1588v2 for Layer 2 Ethernet to suggest Precise Time Protocol (PTP) to obtain low-jitter clocks and accurate synchronization of flows. If we have a small system with two nodes, it is easy to design a timing system. However, besides the collaboration of hardware and software parts, we have many nodes which need to be synchronized. Therefor, we need to investigate different techniques to achieve the best result in the synchronization from our research.

Memory management: On-chip memory limitation and relatively primitive memory abstraction model are the major bottlenecks in FPGAs. Therefore, memory optimization is a key issue especially for supporting more TSN interfaces since we need to keep buffered data in off-chip memory. For off-chip memories there are predefined IP Cores which need to be optimized to provide sufficiently large request queues. Moreover, up-coming memory architectures require cus-

tom IP modules for efficient packetizing request-response communication model, again with optimized request queues.

Once solutions and techniques to answer the challenges are proposed, we present a proof of concept by implementing the proposed solutions and we demonstrate their performance on a use-case.

3.1 Support for predictability

A key requirement for vehicular applications is predictability. It is not enough to obtain functional predictability and correctness, but also it is equally essential to achieve predictability in timing as the project targets real-time applications. The main intention of this part is to develop techniques to achieve timing predictability of TSN-based vehicular applications. In this part, we develop a Response-Time Analysis (RTA) for TSN. Then the next step is to integrate the newly developed analysis to the state-of-the-art end-to-end timing analysis considering computing nodes. The aim of this step is to achieve an extended end-to-end timing analysis framework for TSN. After this, we are going to integrate the developed framework to a software modeling framework, to support the timing analysis of the software architectures of TSN-based vehicular applications. The results from this activity will be used in the hardware realization.

3.2 Hardware realization

In this part of the research, we design and develop a parameterized and modular hardware IP core of the TSN switch fabric in VHDL, which will be deployable in any FPGA devices. The modularity and parameterized solutions help in updating the TSNs IP core to match the evolving TSN standards in a short time. The switch fabric IP core will be equipped with a set of optimized AXI-based memory modules supporting the present and up-coming types of memory. In this part at first step, we design a modular TSN switch architecture that can be easily modified/adapted by leveraging the FPGA reconfigurability. A portable memory controller is considered as data storing infrastructures, expected to greatly improve performance. We use an HW/SW co-design approach for designing the multi-stage TSN switch to increase the bandwidth utilization and diminish memory contention pressure based on the partitioning into communication and application tasks. Time-critical and computationally intensive parts of the communication task will be done in FPGA modules allowing the attached CPU to fulfill the timing requirements of the application without interference. The modularity is the next necessity of TSN switches to provide a portable solution that flexibly adapts to up-coming TSN standards. Finally, the scalability should be addressed in the design to support different use-cases with different levels of bandwidth requirements. The next step is to optimize memory interface in TSN switchs. As we know, performance of FPGA is not only defined by its raw computational power but also by the performance of the memory and I/O subsystems. The final step is to design and develop a specialized FPGA board that offers a scalable TSN switch fabric. The board supports TSN IP core,

and up-coming communication protocols. The prototype will be evaluated on a use-case provided by our business partners.

4 Summary

In this paper, we discussed a concrete research plan focusing on developing novel techniques to implement a high-performance TSN hardware. The hardware will be customized for vehicular domain as modular as possible such that it can be adapted based on the standards modifications. In addition, we presented the idea of using Zynq-based to optimize cost, performance and power of the hardware. We also considered multi-stage switches for scalability and high throughput. Finally, we mentioned the challenges in achieving features, such as timeliness, reliability, scalability and high throughput, timing and synchronization and memory management that need to be addressed for taking full advantage of TSN.

5 Acknowledgement

This work is supported by the Swedish Governmental Agency for Innovation Systems (VINNOVA) through the DESTINE project.

References

1. D. Reinhardt and M. Kucera, "Domain controlled architecture - a new approach for large scale software integrated automotive systems," in *International Conference on Pervasive and Embedded Computing and Communication Systems (PECCS 2013)*, pp. 221–226, February 2013.
2. G. Gut, C. Allmann, M. Schurius, and K. Schmidt, *Reduction of Electronic Control Units in Electric Vehicles Using Multicore Technology*, pp. 90–93. 2012.
3. IEEE Std. 802.1Q, IEEE Standard for local and metropolitan area networks, bridges and bridged networks, 2014.
4. IEEE Std. 802.1Qbv, IEEE standard, amendment 25: Enhancement for scheduled traffic, 2015.
5. IEEE Std. 802.1Qbu, IEEE standard, amendment: frame pre-emption, 2015.
6. K. Sridhar, S. Ooghe, M. P. J. Vissers, and A. Suhail, "System and method for monitoring end nodes using ethernet connectivity fault management (cfm) in an access network," Mar. 30 2010. US Patent 7,688,742.
7. M. Felser, "Real-time ethernet–industry prospective," *Proceedings of the IEEE*, vol. 93, no. 6, pp. 1118–1129, 2005.
8. A. Lastovetsky, I.-H. Mkwawa, and M. O'Flynn, "An accurate communication model of a heterogeneous cluster based on a switch-enabled ethernet network," in *12th International Conference on Parallel and Distributed Systems-(ICPADS'06)*, vol. 2, pp. 6–pp, IEEE, 2006.
9. A. Xu, H. Wang, and Z. Yang, "Industrial automation network based on ethernet," *INFORMATION AND CONTROL-SHENYANG-*, vol. 29, no. 2, pp. 182–186, 2000.

10. M. D. J. Teener and G. M. Garner, "Overview and timing performance of ieee 802.1 as," in *2008 IEEE International Symposium on Precision Clock Synchronization for Measurement, Control and Communication*, pp. 49–53, IEEE, 2008.
11. IEEE Std. 802.1Qca, IEEE standard, amendment: Path Control and Reservation, 2015.
12. G. Alderisi, G. Patti, and L. L. Bello, "Introducing support for scheduled traffic over ieee audio video bridging networks," in *18th IEEE Conference on Emerging Technologies Factory Automation*, Sep. 2013.
13. S. Kehrer, O. Kleineberg, and D. Heffernan, "A comparison of fault-tolerance concepts for ieee 802.1 time sensitive networks (tsn)," in *Proceedings of the IEEE Conference on Emerging Technology and Factory Automation (ETFA)*, pp. 1–8, Sep. 2014.
14. F. A. R. Arif and T. S. Atia, "Load balancing routing in time-sensitive networks," in *3rd International Scientific-Practical Conference on Problems of Infocommunications Science and Technology*, Oct 2016.
15. K. S. Umadevi and R. K. Sridharan, "Multilevel ingress scheduling policy for time sensitive networks," in *International conference on Microelectronic Devices, Circuits and Systems (ICMDCS)*, Aug 2017.
16. M. H. Farzaneh and A. Knoll, "Time-sensitive networking (tsn): An experimental setup," in *IEEE Vehicular Networking Conference (VNC)*, pp. 23–26, Nov 2017.
17. F. Reimann, S. Graf, F. Streit, M. Gla, and J. Teich, "Timing analysis of ethernet avb-based automotive e/e architectures," in *18th IEEE Conference on Emerging Technologies Factory Automation*, Sep. 2013.
18. U. D. Bordoloi, A. Aminifar, P. Eles, and Z. Peng, "Schedulability analysis of ethernet AVB switches," in *The 20th IEEE International Conference on embedded and Real-Time Computing Systems and Applications*, August 2014.
19. J. Cao, P. J. Cuijpers, R. J. Bril, and J. J. Lukkien, "Independent yet tight wcrt analysis for individual priority classes in ethernet avb," in *Proceedings of the 24th International Conference on Real-Time Networks and Systems*, RTNS '16, pp. 55–64, ACM, 2016.
20. D. Maxim and Y.-Q. Song, "Delay analysis of avb traffic in time-sensitive networks (tsn)," in *Proceedings of the 25th International Conference on Real-Time Networks and Systems*, pp. 18–27, 2017.
21. L. Zhao, P. Pop, Z. Zheng, and Q. Li, "Timing analysis of avb traffic in tsn networks using network calculus," in *IEEE Real-Time and Embedded Technology and Applications Symposium (RTAS)*, April 2018.
22. I. Kuon and J. Rose, "Measuring the gap between fpgas and asics," *IEEE Transactions on computer-aided design of integrated circuits and systems*, vol. 26, no. 2, pp. 203–215, 2007.
23. S. M. Trimberger and J. J. Moore, "Fpga security: Motivations, features, and applications," *Proceedings of the IEEE*, vol. 102, no. 8, pp. 1248–1265, 2014.
24. B. Falsafi, B. Dally, D. Singh, D. Chiou, J. Y. Joshua, and R. Sendag, "Fpgas versus gpus in data centers," *IEEE Micro*, vol. 37, no. 1, pp. 60–72, 2017.
25. M. Loni, M. Daneshtalab, and M. Sjödin, "Adonn: Adaptive design of optimized deep neural networks for embedded systems," in *2018 21st Euromicro Conference on Digital System Design (DSD)*, pp. 397–404, IEEE, 2018.
26. P. Papaphilippou and W. Luk, "Accelerating database systems using fpgas: A survey," in *2018 28th International Conference on Field Programmable Logic and Applications (FPL)*, pp. 125–1255, IEEE, 2018.

27. N. Akbari, M. Modarressi, M. Daneshtalab, and M. Loni, "A customized processing-in-memory architecture for biological sequence alignment," in *2018 IEEE 29th International Conference on Application-specific Systems, Architectures and Processors (ASAP)*, pp. 1–8, IEEE, 2018.
28. T. Uchida, "Hardware-based tcp processor for gigabit ethernet," *IEEE Transactions on Nuclear Science*, vol. 55, no. 3, pp. 1631–1637, 2008.
29. N. Alachiotis, S. A. Berger, and A. Stamatakis, "Efficient pc-fpga communication over gigabit ethernet," in *2010 10th IEEE International Conference on Computer and Information Technology*, pp. 1727–1734, IEEE, 2010.
30. S. Shreejith, P. Mundhenk, A. Ettner, S. A. Fahmy, S. Steinhorst, M. Lukasiewycz, and S. Chakraborty, "Vega: A high performance vehicular ethernet gateway on hybrid fpga," *IEEE Transactions on Computers*, vol. 66, no. 10, pp. 1790–1803, 2017.
31. F. Groß, T. Steinbach, F. Korf, T. C. Schmidt, and B. Schwarz, "A hardware/software co-design approach for ethernet controllers to support time-triggered traffic in the upcoming ieee tsn standards," in *2014 IEEE Fourth International Conference on Consumer Electronics Berlin (ICCE-Berlin)*, pp. 9–13, IEEE, 2014.
32. C. Liß, M. Ulbricht, U. F. Zia, and H. Müller, "Architecture of a synchronized low-latency network node targeted to research and education," in *2017 IEEE 18th International Conference on High Performance Switching and Routing (HPSR)*, pp. 1–7, IEEE, 2017.
33. F. Balarin, P. Giusto, A. Jurecska, M. Chiodo, H. Hsieh, C. Passerone, E. Sentovich, L. Lavagno, B. Tabbara, A. Sangiovanni-Vincentelli, *et al.*, *Hardware-software co-design of embedded systems: the POLIS approach*. Springer Science & Business Media, 1997.
34. A. A. Atallah, G. B. Hamad, and O. A. Mohamed, "Reliability-aware routing of avb streams in tsn networks," in *International Conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems*, pp. 697–708, Springer, 2018.
35. F. Smirnov, M. Glaß, F. Reimann, and J. Teich, "Formal reliability analysis of switched ethernet automotive networks under transient transmission errors," in *2016 53nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, pp. 1–6, IEEE, 2016.
36. I. Álvarez, J. Proenza, and M. Barranco, "Mixing time and spatial redundancy over time sensitive networking," in *Proc. IEEE/IFIP Int. Conf. Depend. Syst. Netw. Workshops (DSN-W)*, pp. 63–64, 2018.
37. N. I. Chrysos, "Request-grant scheduling for congestion elimination in multi-stage networks," 2006.
38. F. Hassen and L. Mhamdi, "A scalable multi-stage packet-switch for data center networks," *Journal of Communications and Networks*, vol. 19, no. 1, pp. 65–79, 2017.