SmartNICs: Current Trends in Research and Industry

Tristan Döring, Henning Stubbe*, Kilian Holzinger*
*Chair of Network Architectures and Services, Department of Informatics
Technical University of Munich, Germany
Email: tristan.doering@tum.de, stubbe@net.in.tum.de, holzinger@net.in.tum.de

***Abstract*—With ever rising demand, modern cloud environments had to evolve fast in the last years. One of these novel problems are the increasing speed requirements in combination with present Software-Defined Networks (SDNs). This paper gives an overview on a new hardware trend resulting from this. We illustrate the demand, development, implementation and use of the network accelerating SmartNICs. SmartNICs tackle existing problems of NIC-hardware such as the lack of flexibility, a requirement for virtualized networks. Furthermore the SmartNIC term will be analyzed to provide an universal definition.**

***Index Terms*—SmartNIC, network accelerator, data processing unit, fpga-based smartnic, asic-based smartnic, soc-based smartnic**

1. Introduction

Demand for network performance grows with the expanded use of the internet and the increasing popularity of cloud-based computing. The combination of standard network interface controllers, dedicated networking hardware and software based virtual packet routing can not keep up with the increasing speed requirements. One reason for the slowing development of CPU performance is the decline of Moore's law. E.g. network speeds in the Microsoft Azure platform improved by 50 times between 2009 and 2017, but CPU performance did not improve in the same speed [1]. Therefore cloud service providers, data center operators etc. had to think about new solutions which enable them to provide their services to the growing and performance demanding audience with faster and cost efficient performance. A problem known to exist is the lack of programmability/flexibility in regular NIC hardware which is built for a number of specific predefined use cases and at heart still serves the functionality as an interface from the host CPU to the network. This leaves much of the workload to the CPU. This static functionality does not comply with evolving Software Defined Network (SDN) policies and Virtual Network Functions (VNF) which are key parts of current virtualized network environments. There is an increasing demand for better performance compared to software solutions without compromising too much flexibility. This leads to the development of the SmartNIC. The novel NICs should be able to handle more complex tasks independent of the CPU to a certain degree. This is called an in-Network Interface Card (in-NIC) processing approach.

2. Trends and Technological Demands in Cloud/Hosting Industry

Before diving deeper into the topic of SmartNICs this section will give a short overview on the current state of the industry. The new guiding trend is the virtualization of networks, storage, GPUs etc. These techniques generate network related workloads not only on network devices as virtualization can not independently run on e.g. NICs. The network processing can be divided into two categories, i.e. the data plane and the control plane. The control plane is responsible for the control of the network structure, i.e. communications between network devices and assigning tasks to network devices. In other words it is the implementation of network policies. The data plane handles the actual movement and modification of application data packets. The legacy hardware layout in a server is a combination of standard NICs and CPU cores. Here the CPU handles the control and data plane. The NICs on the other hand at best accelerate certain data plane functions to ease up load on the CPU. That means in current server environments the network traffic can use up a significant amount of CPU resources [1, section 3] (see also section 5.3).

The following paragraphs will explain a few common network technologies present in typical up-to-date server environments that are demanding on the CPU.

Software-Defined Networking (SDN) decouples the network structure from its realization in hardware. That means the control plane is controlled by software control plane policies. These policies are mapped to the data plane and executed in hardware. That imposes additional packet steering and processing requirements and adds additional workload to the CPU. E.g. Open vSwitch (OVS), an open-source software implementation of a multilayer switch, is part of such a virtual network stack.

For example storage virtualization also called Software-Defined Storage (SDS) is part of the virtualization trend. The idea is to share storage between different servers over the network. Virtualization and abstraction is used to make the storage look local to applications. That means all the data traffic has to go through the network. This brings up some challenges to overcome with even more stress on the CPU which will further increase with higher bandwidths. The newest technology is Non-Volatile Memory express over Fabrics (NVMe-oF) which combines the low latency NVMe-protocol with virtualized storage addressing over the network.

For efficient implementations of these protocols, SmartNIC acceleration is vastly beneficial.

doi: 10.2313/NET-2021-05-1_05

3. What is a SmartNIC?

Network environments develop to be more complex and the trend shifts to more and more virtualized network environments. This involves too much networking overhead on the CPU cores. With rising network speeds of currently up to 200Gbps per link the expensive CPU spends too much work for networking. Even though the true value of CPUs lies in their ability for general processing e.g. applications and data analysis. The virtualization trend makes this even worse with increasing local network traffic in servers due to SDS, SDN and big data. An obvious solution is to offload this kind of processing to specialized external devices. There are already some NIC devices that are not considered smart but assist the CPU by performing a huge diversity of network function acceleration including Network virtualization. These functions are hardcoded into these NICs. They are considered the first step that led to the development of SmartNICs. This combination of standard NICs and servers is not able to deliver enough performance to meet the rising demands for network speed because of the required application of SDN policies [1, section 2.3]. SmartNICs in comparison to CPU and NIC combinations target to offer a faster, more efficient and lower cost solution including the important reduction of CPU usage. Another important factor is the required flexibility that current software solutions offer. The programmable SmartNICs also try to meet this demand by offering various development kits or even the possibility to run existing software to engineers. We will see later how different manufacturers handle this problem.

3.1. The Term SmartNIC

"Also called a 'network interface card' (NIC), a network adapter is a plug-in card that enables a computer to transmit and receive data on a local network. [...]" [2]

On the basis of this definition of a classic NIC and also the knowledge of some SmartNIC functions and use cases a proper definition is worked out. A definition that is applicable to the new type of network accelerators called SmartNICs. The term SmartNIC implicates that it is an extension of standard NIC devices, but it is not as simple as just an increased functionality. The keyword “Smart” implies some kind of intelligence or the ability to act somewhat smart to solve complex tasks. The smartness in this new device class lies in the ability to perform numerous tasks independently and to be flexible enough to tackle future and current network tasks [1, section 3]. This was achieved by adding some kind of general processing unit to a NIC.

3.2. Other SmartNIC Definitions

There are numerous attempts to define what the term SmartNIC means. A widely spread definition is written by Alan Freedman, the author of a tech encyclopedia: “A network interface card (network adapter) that offloads processing tasks that the system CPU would normally handle. Using its own on-board processor, the smartNIC may be able to perform any combination of encryption/decryption, firewall, TCP/IP and HTTP processing. SmartNICs are ideally suited for high-traffic Web servers” [3]. But this definition lacks one of the most important properties of a SmartNIC, the programmability, the biggest difference to a regular NIC. In today’s rapidly changing software-defined networks updatability is key to have a future proof network environment. Microsoft also supports this opinion in their paper “Azure Accelerated Networking: SmartNICs in the Public Cloud” where they list "Maintain host SDN programmability of VFP” [1] as one of their design goals for their in-development Microsoft Azure SmartNIC. Also contrary to the definition by Alan Freedman use cases or application environments should not be mentioned in the definition as these are broad and ever changing.

Mellanox’s Vice President of marketing Kevin Deierling goes even further and defines three different kinds of NIC devices: Foundational NICs, Intelligent NICs and DPU based SmartNICs [4]. This variety exists because Mellanox has customers who ask for a SmartNIC because they need functionality they refer to as smart and which is superior to that of a Foundational NIC. But these functionalities were also supported by their base line of NIC products, the so-called Intelligent NICs which Mellanox does not refer to as smart. Kevin Deierling’s approach is to differentiate NICs by their offered functionality. The further differentiation does make sense for Mellanox, because they want customers to perceive their products superior to standard NICs. Mellanox defines an Intelligent NIC as a NIC with extended functionality that is not programmable. The further differentiation may be confusing because the border between NICs and Intelligent NICs can not be properly defined. In the future the comprehension of what extended NIC functionality is will probably change and with it the definition for Intelligent NICs.

3.3. Definiton of SmartNIC

Going off of multiple press and industry definitions we decided to work out a definition of our own that is universally applicable:

A SmartNIC is a device which connects to a network and a computer that specializes in hardware-acceleration of various standard and software-based network related tasks. The innovation drivers for its development are freeing up CPU resources, the requirement for formerly unreachable speeds and more efficiency. It is able to perform control and data plane functions by combining NICs with flexible processing units. That also allows SmartNICs to offer programmable and updatable functionalities. It can also be called programmable network function accelerator.

4. SmartNIC Hardware Architectures

This section explains different hardware approaches to SmartNICs. There are three different processing units to build a SmartNIC upon. Figure 1 is a presentation slide of Microsofts paper "Azure accelerated networking: Smartnics in the public cloud" [1] and shows 5 different options to accelerate SDN speeds. Also the reciprocity of the flexibility versus the efficiency of the hardware solutions is illustrated. This paper will not explain the G/NPU hardware approach because there are no such devices released yet.

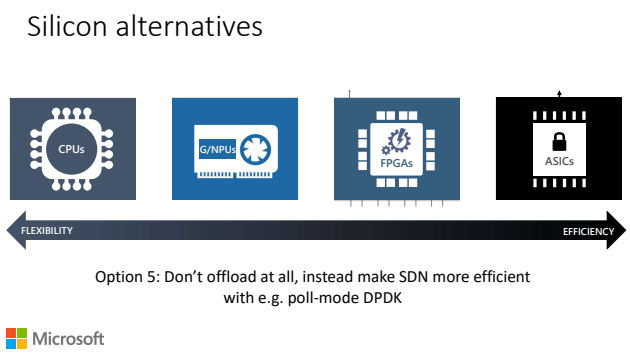

Figure 1: Overview on the different implementation options and their flexibility versus efficiency [1]

4.1. FPGA-based SmartNICs

The field-programmable gate array is an integrated circuit which is reconfigurable. This software-like programmability paired with faster performance and lower latency close to that of an Application-Specific Integrated Circuit (ASIC) makes it a perfect fit for SmartNIC hardware. The FPGA achieves faster performance with parallel processing of data flows in comparison to instruction by instruction temporal computing in CPUs. The downside is the comparatively difficult implementation of new functions as it requires expertise in Hardware Description Language (HDL) to make efficient use of the chip. Hiring dedicated development teams with sufficient skills is not a big problem for hyperscalers such as Microsoft, Amazon and Google FPGAs. Thus FPGAs are often used in these hyperscale environments. Though manufacturers try to facilitate development by offering compatibility with common developer environments such as the Data Plane Development Kit (DPDK). Microsoft estimates that the size of an FPGA compared to an ASIC with the same functionality is "around 2-3x larger" because of the flexibility they offer [1, section 4.1.3]. This generally makes them more expensive than ASICs. There are not many example products in the current market that only rely on FPGAs. One rare instance is the Intel FPGA Programmable Acceleration Card (PAC) N3000 that we will discuss later.

4.2. SoC-based SmartNICs

SmartNICs with a System-on-a-Chip work like embedded Linux servers [4, section 4.1.2]. They handle the data and control plane in a traditional manner as they basically run software-based VNFs on their SoC. Due to their compact size format and thus shorter distances between memory, processing unit and network interface efficiency and speed are usually better. Often some kind of other more specialized processing unit like FPGAs or ASICs are paired with SoCs, e.g. in the Silicom N5110A SmartNIC [5]. Many SoC-based SmartNICs offer easy C-programmability. Compared to FPGAs SoCs are easier to program, but slower/less efficient alternatives. As no special programming skills are required SoC based SmartNICs are well-suited for smaller companies e.g. the Broadcomm Stingray SmartNIC architecture which is specifically marketed as easy to deploy [6].

4.3. ASIC-based SmartNICs

Application-Specific Integrated Circuits (ASICs) achieve higher efficiency/performance by sacrificing the flexibility present in an FPGA. ASIC development traditionally means designing a specification and test methodology for everything that a system could possibly want to do over its lifetime upfront. As basic NICs usually rely on ASICs, it results in their development cycles being similar. FPGAs on the other hand allow hardware developers to be far more agile in their approach. The development phase of an ASIC chip can take up to 2 years from requirements engineering to the ready silicon parts [1, section 4.1.1]. But in that time the functional requirements of a NIC probably already changed because new network technologies are being developed all the time. Additionally ASICs are only able to perform data plane functions. To counter these issues manufacturers often add embedded CPU cores to handle new functionality and the control plane. These cores are comparatively slow and can consequently cause a bottleneck when more and more functionality is added to them. In every SmartNIC there is a foundational level of functionalities that is known during the development phases that benefits from the power and cost efficiency of ASICs. As a result ASICs in SmartNICs still play the role of a NIC but only in combination with other flexible compute units.

5. Two SmartNIC Products in Detail

At first sight the market of SmartNICs seems very confusing because every product is marketed for almost every application domain. Performance specifications are rare and differences must be found in the details. Thus the direct comparison of different SmartNIC products is hard. To illustrate the diversity of products two vastly different approaches were chosen to be displayed.

5.1. Intel FPGA PAC N3000

The FPGA-based Intel FPGA PAC N3000 is part of Intels newest line of FPGA-based accelerator cards. This Intel device combines the worlds of programmable accelerator cards (PACs) and SmartNICs into one product. It allows for optimization of data plane performance to achieve lower costs while maintaining a high degree of flexibility. The built-in Intel Arria 10 GT FPGA delivers up to 1.5 TFLOPS [7, section 2.2.1]. As a board management controller the Intel Max 10 FPGA is responsible for controlling, monitoring and giving low-level access to board features [7, section 2.4]. The Intel XL710-BM2 NIC is directly connected to the FPGA to provide basic ethernet connectivity, some virtualization and I/O Features [8]. Consequently the Intel Arria 10 GT can accelerate network traffic at up to 100 Gbps [7]. The Root-of-Trust technology prevents the loading or executing of unauthorized workloads and the unauthorized access to key interfaces and on-board flash storage. The scopes of application are Network Function Virtualizaton (NFV), Multi-Access Edge Computing (MEC), Video Transcoding, Cyber Security, High Performance Computing (HPC) and Finance [8]. Intel tries to make development as comfortable as possible

doi: 10.2313/NET-2021-05-1_05

with the implemented compatibility with various developing environments. The Intel Acceleration Stack for Intel Xeon CPU with FPGAs provides an allround development platform. The supported Open Programmable Acceleration Engine (OPAE) technology provides a consistent API across FPGA product generations and platforms. Also the popular Data Plane Development Kit (DPDK) is supported and provides control over both the FPGA and the Ethernet Controller. Due to its lack of a SoC the Intel FPGA PAC N3000 can not offload everything from the CPU as some functions like Internet Protocol Security (IPSec) still need additional CPU resources [9]. The Intel FPGA PAC N3000 compromises on flexibility to offer a network acceleration speed of up to 100 Gbps. It is mostly advertised to telecommunications service providers.

5.2. NVIDIA Bluefield-2 DPU

With the Bluefield-2 DPU, Nvidia fuses specialized network processing and general processing into one SoC. The SoC consist of a Nvidia Mellanox ConnectX-6 Dx intelligent NIC, two Very Long Instruction Word (VLIW) Acceleration Engines and a 64bit Arm processing unit [10]. It offers full programmability to the user. The Arm NEON SIMD execution unit is optimized for vector processing with an extended Instruction Set Architecture (ISA). Nvidia advertises the Bluefield-2 as a DPU made for data movement and security processing. In this case DPUs are defined as programmable Data Center Infrastructure-on-a-Chip. With the built-in Nvidia Mellanox ConnectX-6 Dx Nvidia integrated the fastest NIC in the ConnectX product line [11]. The general idea is to combine easy programmability of an Arm chip with the accelerator especially built for parallel data processing. Even though Nvidia rarely calls it a SmartNIC, it falls in this category when looking at the specifications and features. The DPUs features include some which are commonly seen in SmartNICs such as NFV, en-/ decryption and many other network, security or storage virtualizations such as IPSec, NVMeoF, OVS etc. GPUDirect, a family of technologies that enhances data movement and access for NVIDIA data center GPUs is also supported. Using GPUDirect, network adapters and storage drives can directly read and write to/from GPU memory, eliminating unnecessary memory copies, decreasing CPU overheads and reducing latency, resulting in significant performance improvements. This is an important functionality for AI computing. The Nvidia DOCA SDK integrates industry-standard open APIs for software-defined networking and storage, security services, and programmable P4 functionality. The DOCA SDK was just announced with the Bluefield-2 and is meant to become the standard SDK with intercompatibility to all other Bluefield devices [12]. Another product announcement based on the Bluefield-2 worth mentioning is the so-called AI-Powered DPU Bluefield-2X. With the addition of a Nvidia Ampere GPU it is the first instance of a GPU-based SmartNIC. The GPU enhances AI-based functions like real-time security analytics, identifying of abnormal traffic and dynamic security orchestration [13]. In conclusion the Nvidia Bluefield-2 DPU presents a novel approach to SmartNICs. It combines the hardware acceleration of the data plane with the addition of an easy-to-programm control plane which both run on the custom SoC.

5.3. Nvidia Bluefield-2 performance benchmark

To illustrate the independence of the Bluefield-2 its performance in a DPDK 20.08 benchmark is compared to that of a Nvidia Mellanox ConnectX-6Dx so-called intelligent NIC. Nvidia benchmarked the throughput at zero packet loss at a linerate of 2x25 Gbps. The ConnectX-6Dx uses 4 additional CPU cores, while the BlueField-2 SmartNIC is on its own. The ConnectX-6Dx CPU combination achieves to throughput the maximum amount of packages per second possible at the specified line rate. The BlueField-2 is only slightly slower (maximum 2.06% slower) but uses none of the CPU cores [14].

6. DDoS Mitigation Use Case

The reference paper [15] for this section was published by IEEE, it explores the use of SmartNICs for DDoS attack mitigation. The idea is to offload a portion of the DDoS Mitigation rules from the server to the SmartNIC. For comparison they try three different approaches to the problem.

6.1. Host-based Mitigation

All traffic is processed by the host CPU. The packets matching a given blacklist are dropped. The Linux-Kernel-based solution with iptables and its derivatives is too slow for modern DDoS attacks. The opposite solution using specialized NIC and network drivers in combination with an userspace applications proves to be faster, but requires allocating a fixed number of CPU cores. A mix of the previous solutions is XDP, in essence a kernel framework. The early kernelspace program is injected by userspace before the netfilter framework, hence performing an order of magnitude faster. Also its event-driven execution allows it to use CPU resources only when necessary.

6.2. SmartNIC-based Mitigation

The idea is to spare the CPU cores by prefiltering all packets in the SmartNIC. Depending on its hardware features there are different options to do so. If available the built-in hardware filters should be used, which can only hold a certain number of mitigation policies. That means the SmartNICs CPU has to apply the remaining ones by using custom programs. The surviving traffic is then directed to the server applications in the host system. This option works well until the number of mitigation policies is too big to store for the hardware tables. When additional use of the SmartNICs CPU becomes unavoidable, it presents a bottleneck resulting in declining performance.

6.3. Hybrid (SmartNIC + XDP Host)

A mix of the above combines the SmartNICs hardware tables which run at line rate and the enormous processing power of the host CPU. If the hardware tables prove to be too small to hold all mitigation rules the remaining ones will be implemented by XDP on the CPU.

6.4. Conclusion

In the experiments performed by IEEE, the hardware offloading approaches prove to be the most effective [15]. Precious CPU resources are protected as long as possible and even when additional CPU cores are required due to an increasing number of attack sources the hardware tables provide a proper prefiltering for CPU-based mitigation applications. SmartNICs can help mitigate the network load on congested servers, but only to a certain extent. In a real server environment a DDoS-aware load balancer would come in handy to distribute the load among multiple hosts and thus limit the number of mitigation policies.

7. SmartNICs: Conclusion

In this paper we discussed why the development of SmartNICs is inevitable, the different hardware approaches, two product instances and finally an interesting real world application of them. Beside all good about them it is clearly that their performance is not yet strong enough to solely handle high bandwidth throughput of a server. Often there is still no other choice but to use additional CPU cores as they still exceed the SmartNICs performance as seen in the DDoS Mitigation use case.

The future seems bright for SmartNICs. In the last years most companies in the sector announced their first line of products, e.g. Nvidia, Intel, Broadcom, Silicom, Inventec etc. Also AMD will probably try to enter the market with its acquisition of Xilinx, one of the major FPGA developers. In the future the development will be driven by even faster network speeds. E.g. Nvidia plans to increase the throughput of their devices to 400 Gbps including a 100-times performance improvement until 2023 [13]. The market is expected to grow substantially, with it the hardware-acceleration trend and the utilization of SmartNICs.

References

[1] D. Firestone, A. Putnam, S. Mundkur, D. Chiou, A. Dabagh, M. Andrewartha, H. Angepat, V. Bhanu, A. Caulfield, E. Chung, H. K. Chandrappa, S. Chaturmohta, M. Humphrey, J. Lavier, N. Lam, F. Liu, K. Ovtcharov, J. Padhye, G. Popuri, S. Raindel, T. Sapre, M. Shaw, G. Silva, M. Sivakumar, N. Srivastava, A. Verma, Q. Zuhair, D. Bansal, D. Burger, K. Vaid, D. A. Maltz, and A. Greenberg, "Azure accelerated networking: Smartnics in the public cloud," in *15th USENIX Symposium on Networked Systems Design and Implementation (NSDI 18)*. Renton, WA: USENIX Association, Apr. 2018, pp. 51–66. [Online]. Available: https://www.usenix.org/conference/nsdi18/presentation/firestone

[2] A. Freedman, "Definition: network adapter," 2020, accessed: 2021-01-05. [Online]. Available: https://www.computerlanguage.com/results.php?definition=network%20adapter

[3] A. Freedman, "Definition: SmartNIC," 2020, accessed: 2020-12-29. [Online]. Available: https://www.computerlanguage.com/results.php?definition=Smart+NIC

[4] K. Deierling, "Defining the DPU (Data Processing Unit) based SmartNIC : What is a SmartNIC and How to Choose the Best One," 2018, accessed: 2020-12-29. [Online]. Available: https://blog.mellanox.com/2018/08/defining-smartnic/

[5] Silicom Limited, "Silicom N5110A SmartNIC Intel® based - Server SmartNIC with Advanced Computing and 100Gbps Network Switching," 2021, accessed: 2021-02-28. [Online]. Available: https://www.silicom-usa.com/pr/fpga-based-cards/100-gigabit-fpga-cards/n5110a-pcie-smartnic-intel-based/

[6] Fazil Osman, Broadcom Limited, "Using SmartNICs as New Platform for Storage Services," 2019, accessed: 2020-12-30. [Online]. Available: https://www.snia.org/educational-library/using-smartnics-new-platform-storage-services-2019

[7] Intel Corporation, *Intel FPGA Programmable Acceleration Card N3000 Data Sheet*, Intel Corporation, 2020, accessed: 2021-01-01. [Online]. Available: https://www.intel.com/content/www/us/en/programmable/documentation/dfy1538000574521.html

[8] Intel Corporation, "Product Brief Intel® Ethernet 700 Series Network Adapters Intel® Ethernet Converged Network Adapter XL710," 2020, accessed: 2021-01-03. [Online]. Available: https://www.intel.de/content/www/de/de/products/docs/network-io/ethernet/network-adapters/ethernet-xl710-brief.html

[9] Intel Corporation, "Accelerating IPSec with Arrive Technologies on the Intel® FPGA Programmable Acceleration Card N3000," Intel Corporation, Tech. Rep., 2019, accessed: 2021-01-05. [Online]. Available: https://www.intel.de/content/dam/www/programmable/us/en/pdfs/literature/solution-sheets/sb-accelerating-ipsec-arrive-technology-intel-fgpa-pac3000.pdf

[10] Nvidia Corporation, "NVIDIA BLUEFIELD-2 DPU Data Center Infrastructure on a chip," 2020, accessed: 2021-01-05. [Online]. Available: https://www.nvidia.com/content/dam/en-zz/Solutions/Data-Center/documents/datasheet-nvidia-bluefield-2-dpu.pdf

[11] Tolly Group, "NVIDIA Mellanox ConnectX-5 25GbE Ethernet Adapter Adapter Performance vs Broadcom NetXtreme E," no. 219130, Oct. 2019.

[12] Nvidia Corporation, "Nvidia doca sdk data center infrastructure on a chip architecture," 2021, accessed: 2021-01-08. [Online]. Available: https://developer.nvidia.com/networking/doca

[13] Nvidia Corporation, "Nvidia Bluefield Data Processing Units Software-Defined, Hardware-Accelerated Data Center Infrastructure on a Chip," 2021, accessed: 2021-01-05. [Online]. Available: https://www.nvidia.com/de-de/networking/products/data-processing-unit/

[14] NVIDIA Corporation, "Mellanox NIC's Performance Report with DPDK 20.08 Rev 1.1," 2020, accessed: 2020-12-30. [Online]. Available: http://fast.dpdk.org/doc/perf/DPDK_20_08_Mellanox_NIC_performance_report.pdf

[15] S. Miano, R. Doriguzzi-Corin, F. Risso, D. Siracusa, and R. Sommese, "Introducing smartnics in server-based data plane processing: The ddos mitigation use case," *IEEE Access*, vol. 7, pp. 107 161–107 170, 2019.