A Study of Reconfigurable Accelerators for Cloud Computing

Noor Mohammedali and Michael Opoku Agyeman
Department of Computing, University of Northampton, Northampton, UK.

Abstract— **Due to the exponential increase in network traffic in the data centers, thousands of servers interconnected with high bandwidth switches are required. Field Programmable Gate Arrays (FPGAs) with Cloud ecosystem offer high performance in efficiency and energy, making them active resources, easy to program and reconfigure. This paper looks at FPGAs as reconfigurable accelerators for the cloud computing presents the main hardware accelerators that have been presented in various widely used cloud computing applications such as: MapReduce, Spark, Memcached, Databases.**

Index Terms—reconfigurable computing, hardware accelerator, SPARK, reconfigurable architectures, cloud computing, FPGAs, data centers

I. INTRODUCTION

Cloud computing refers the virtualization and central management of data center resources over the internet [1]. Cloud contain an accelerator service used to transfer data to the user with a high performance and lower latency. Moreover, it is designed specifically for streaming or dynamic data. The benefit of using accelerators is to improve the performance of the system. Hence, there are many type of accelerators such as, Hardware accelerator, Graphics accelerator, Cryptographic accelerator, Web accelerator, PHP accelerator. The main challenge to implementing cloud acceleration are scalability, redundancy, consolidation of service and cost [2] [3]. Reconfigurable hardware on the cloud employs FPGA as reconfigurable device that has high performance, low power consumption, reduced size and energy. Using heterogenous system in cloud computing comes at a substantial. Consequently high level for programing language such as OpenCL, C, C++ are normally used to configure FPGA hardware accelerator files (bitstreams) in cloud computing. The customer should look at different aspect of the virtualization such as, performance, scalability, reliability and security technology before choosing it for his business [4]. The main problem is the performance of the acceleratory and the security in the cloud.

This paper presents a survey of the Reconfigurable Accelerators for Cloud Computing that have recently been presented in literature. The paper first gives an overview of the configuration framework in a cloud computing in Section II. Reconfigurable framework in cloud computing is presented in section III. The hardware accelerators for cloud computing are presented in section IV. Section V present the overview of the security in the cloud computing and the role of FPGA in the cloud computing. Section VI concludes the paper.

II. RECONFIGURABLE FREAMWORK IN A CLOUD COMPUTING

A reconfigurable hardware accelerators in a Cloud environment (RC3E) was presented in [5] as a hypervisor cloud which manages and monitors FPGA resources. Three service models were presented in this paper: RSaaS, RAaaS and BAaaS. In Reconfigurable Silicon as a Service – RSaaS provide full access to the user to reconfigure the FPGA that will cause some attack problem. The user can allocate a complete physical FPGA with own implemented hardware. In Reconfigurable Accelerators as a Service – RAaaS in this service the FPGA represented as a simple accelerator and only the vFPGAs are visible to the user with different sizes. In Background Acceleration as a Service – BAaaS in this service vFPGA will be work in the background to speed up the application that are visible to the user. The RAaaS- and BAaaS model allow multiple concurrent user designs on a single physical FPGA.

In [6], a virtualized FPGA accelerator where a conjunction of heterogeneous hardware resources in the cloud are used to to improve the performance and computational efficiency as well as the stalled CPU execution scalability is proposed. Moreover, a prototype framework for coordinating virtualised FPGA accelerators in the cloud using partial reconfiguration and virtualized correspondence interfaces were presented. Through partial reconfiguration, FPGAs shared multiple accelerators and dynamic loading of accelerators at run time that offer high communication bandwidth to improve a computational efficiency over software.

The advantages of implementing and integrating Network-on-Chip (NoC) based virtualized accelerators in in cloud computing are highlighted in [7]. Networks-on-Chip was used between the accelerators communication and the reconfigurable control to make the connection more efficient. NoC helps the hardware accelerators to have parallel communication between each other and the reconfiguration control manager and exchange of data through the routers connected to them. There are two service models in NoC based virtualized: Reconfigurable IPs as a Service (RIPaaS) and Reconfigurable Regions as a Service (RRaaS). In the RIPaaS service, a user will request accelerator and the cloud provider will check the request. If the accelerator is available, the connection between the user and accelerator will be established. If not, the accelerator will be reconfigured from the existing bitstream library. Whereas the RRaaS service, the user can access the top-level virtualization and choose the suitable RRv for his HDL implantation and I/O port, then, the cloud provider will send the template to the user to implant it and send it back. Finally, the service provider will check if there is no error will and put it in bitstream generation. So, in this service, the cloud provider offers a chance to add new accelerator to the cloud based on future expectation from users. Most of these papers cover details of reconfigurable hardware such as how FPGAs can be used for hardware accelerators in cloud which enables one physical FPGA to host multiple virtual FPGAs (vFPGAs) to increase the utilization and efficiency. In NoC based virtualized accelerators for cloud computing, the performance of the bus communication between the virtualized accelerators is improved with a NoC layer to support parallel communication to save resource and power.

III. HARDWARE ACCELERATOR FOR A CLOUD COMPUTING

A reconfigurable Catapult fabric for accelerating datacenter services for large-scale production workloads is proposed in [8]. This pilot test on over 1,632 Microsoft servers run Intel Xeon to measure the accelerating efficacy in Bing web search engine. Using multiple FPGA to connect with multiple server. Moreover, processing Bing's custom algorithms used FPGAs because FPGAs are 40 times faster than CPUs. In a production search a performance evaluation for the FPGAs ranking increase by 95% at comparable latency and the power consumption increased by 10%. Google search engine used PageRank algorithm to measure the authority and the rank of the webpages [9]. In the same time PageRank details are proprietary. PageRank might have been the unique idea behind the creation of Google. This algorithm based on the paradigm that's means when the paper referenced by other paper will be considered as an important paper when it is had many citations [9]. In [10] presented the architecture for the FPGA accelerator board to improve the efficient of query processing in a Web search engine. The implantation includes ranker, matcher and list compression decompression. Moreover, this system boots up with real data from a search engine. They explore the design space for essential components in query processing on the new platform and implement the whole system. Furthermore, compared with an Intel Xeon server, the system could achieve up to 19.52X power efficiency and 7.17X price efficiency.

MapReduce uses in Hadoop as map and reduce (key / value). This takes the following general form

map: (K1, V1) → list(K2, V2)
reduce: (K2, list(V2)) → list(K3, V3)

The map has input key and value types like (K1 and V1). The input map is different from the output map types (K2 and V2). The reduced input must have the same types as the map output. Moreover, the reduced output types may be different again (K3 and V3) [11].

In [12], MapReduce accelerator that can be used to speed up the processing data based on FPGAs was proposed. Map tasks generation is different for each application and it has specialized hardware accelerator. The reduce tasks has shared configurable accelerator. The most challenging constraint in the data centers the power consumption in the data centers operator to sustain the increasing network traffic. The proposed platform allows the efficient mapping of MapReduce applications in FPGAs that allows the speedup of the applications and the significant reduction of the power consumption compared with typical server processors. Hence, the acceleration for both the Map and the Reduce tasks are used in their proposed platform. Depending on the application requirements a configuration of Reduce co-processor was developed to meet different processing requirement.
MapReduce used in different aspect: 1-Scalable MapReduce Accelerator. 2- FPMP: MapReduce. 3- Reconfigurable MapReduce. 4- Big Data Analysis acceleration. 5- MapReduce for K-means. [13] identified the limitation for the performance boost such as, Low parallelism exploitation, High memory conflicts and Low acceleration opportunity through the execution time. HLS tools used to examine the performance exploitation. MapReduce framework provides implementation abstraction, hardware architecture and an exemplary system-level framework for system designers. Those fundamental commitments considered in [13]:

1- A novel HLS-based MapReduce information stream architecture.
2- Improvement the hardware accelerators for MapReduce requisition to view of those HLS-enabled MapReduce architecture.
3- The performance of the MapReduce applications on a Virtex7 FPGA that shows up to 4.3x throughput gains and up to two orders of magnitude energy consumption savings.

A mechanism called JUMPRUN to accelerate key value operation and reduce a data traffic across the memory hierarchy is presented in [14]. Moreover, they presented the architecture and operating mechanism for item scanner. Moreover, they proposed item jump for software level acceleration schema and design NMP based on the acceleration engine. Their results show that, the performance of scanner operation improved up to 9 times and the total energy reduce up to 71%.

In [15], a novel GPU-accelerated MapReduce framework that amplifies Spark's in-memory is presented. Here, deep memory hierarchy is employed to reduce slow in I/O disk for iterative computing tasks. Hence, the main bottleneck in Spark system is data communication in a Java virtual machine. Moreover, they used a caching framework to minimize host-to-GPU communication and used a GPU over multiple mapper executions. In order to achieve a high performance, they used Widx because it contain strolling numerous hash buckets concurrently and input hashing keys depend on their use. They highlight many points of using Widx such as flexibility, minimizing area cost and improving the performance index by 3.1x on average and saving 83% of the energy. In short, in conventional Spark they achieved up to 50 times speed up and in GPU-accelerated Spark they achieved 10 times speed.

In [16] a radix tree-based index searching on the GPU was evaluated and implemented to supports a range queries on the GPU. Furthermore, it presented GPU based Radix Tree as an efficient index searching implementation that based on radix trees. GRT optimized for range queries, SIMD and different key lengths. They used ART because it is represented as is the most efficient radix-tree based index searching system in the CPU. The basic structure used by ART that adjust them with modern GPUs. Using benchmarks to get the overall study of GRT that contain sets of data that have millions of keys to support their results and findings. They also compared the performance of the runtime in GPU and CPU on a large dataset of over 64 million for 32-bit keys per second. for sparse keys GRT achieved 106 million lookups per second and130 million lookups per second was achieved for dense keys. For the excellent range, GRT achieved 1000 million lookups per second for sparse keys and over 600 million lookups per second for large range sizes for dense keys. They used different type of GPUs depend on their processor clock rate, peak GFLOPs and memory capacity. The GTX 580 model, The GTX 580 model and the Tesla K80. In addition, they used three key lengths: 4-byte, 8-byte, and 16-byte keys. At the end, their implementation was restricted for a single GPU to store and manage index structures.

In [17] Apache Spark was used as a big data analytics tools which resulted in less processing time than other tools. Moreover, it was 100 times faster than Hadoop and supports different file systems and many programing language as well as machine learning algorithms. FPGA-based accelerator for the distributed training of convolutional neural networks in presented in [18]. Here, Deep Convolutional Neural Networks were used, because of their high performance and complexity. They were considered as the most challenging aspect. Moreover they, Accelerated the training of deep convolutional neural networks by using the SPARK run time environment in a data center. In multi-layer involution operation, their accelerator achieves from 40 to 250 times speedup. FPGA design based on 2D involution filter that used in multi-layer involution operation and the distributed training of convolutional neural networks. Furthermore, the FPGA design has two circuit first one for a re-timing transformation and the second one for a feed-back loop. FPGA-based accelerator provided for machine learning and data analytic applications. In a data center FPGA run under the SPARK environment to supply a better performance per Joule. There are two types of approaches that were used to optimize the training, task distribution over clusters, and single node acceleration. These approaches were achieved by distributed task-loads in a data-center by formulating an FPGA-based accelerator design in amplifying SPARK's environment. Also, to achieve better results, a whole cluster was managed by SPARK. In short, experimentations shown that the computationally most expensive operations of the training process and the accelerators speedup over the implementations of a software while there is energy efficient. Using SPARK's extension and the accelerator's design to reduce the overhead of training helped distribute the training to the nodes, accelerating the nodes' computation.

A set of open source applications used to execute model-driven framework for individual Apache Spark setting and understand the performance influence models for each one was presented in [19]. They characterized two techniques: the workloads and applying statistical analysis techniques. For the workloads technique, they used KMeans, Word Count, PageRank, Matrix. In each case, their framework effectively focuses on the underlying components influencing the execution of individual settings. Their framework effectively focuses on the underlying Components influencing the execution of individual settings. However, this paper does not examine tuning offsetting automatically but they believe that from their investigation, they can get the motivations behind observed execution varieties because of the opposition on transforms in settings. Furthermore, Spark supports many applications such as graph computation, stream applications, machine learning, interactive queries, and stream applications and each setting effect on different applications.

IV. Heterogenous system in cloud

In [20], a dynamic fine-grained resource provisioning method supported by a novel Smart Controller (SC) is presented. This method used a non-equilibrium states algorithm in virtualized cloud data center (VCDC) to share multiple applications with different service classes. A hybrid meta-heuristic algorithm was used to determine the resource location in CPU and I/O, where the application services are maximized and machine-level energy is minimized. Furthermore, providing a head of time they assumed that admission control policy and dealing with several factors such as, the agreements between the customers and cloud providers, the amount of the requests that been done successfully, the amount of the failed and rejected requests. Finally, they solved the formulated optimization problem with particle swarm optimization and simulated annealing in hybrid algorithm and simulation the result demonstrated the accuracy and effectiveness of their proposed model and maximization method.

In [21], it was claimed that accelerating large-scale graph processing is very important for many data-intensive applications. Here a graphics processing unit in the cloud was presented. They develop a G2 based on vertex-oriented programming model. In additional, G2 represented as a GPU-accelerated in-memory graph processing engine. Also, G2 performance increase to by 50% on an Amazon EC2 in virtual cluster by using series of GPU-specific optimizations and it has three key feature such as, performance, scalability, programmability.

V. Security in cloud computing.

The FPGA usage of a cloud security instrument gives a secured zone inside the untrusted environment for safely performing touchy operations. Since information can be exchanged securely to the device, here it can be controlled without conceivable impedances from exterior components (other framework components or a chairman). Hence, the FPGA can play the part of a trusted computing device that unscrambles the input cipher-text, performs the computational operations and re-encrypts the results.

In [22], how FPGAs can used with cloud computing to build a secure and flexible trusted computing platform to save a sensitive data such as a medical record from attack is presented. Untrusted issue were solved by using Hardware-based systems that give as guarantees that are more powerful versus attack. FPGAs offer a unique practical alternative with in the cloud infrastructure to build a trusted third-party platform and emulating the effective behavior. Client may offload a sensitive data or may not trust these devices. Furthermore, protected bitstreams will be used to generate a root of trust for the clients of cloud computing services.

In [23], special benefits for the IT industry but brings along moreover particular challenges is presented. Given the complexity and fast development of the cloud computing framework, but there are many issues with respect to the security of the information, the data center is turning increasingly towards the basic equipment resources. Security solutions provided on a hardware with in a different area such as Data Security, User Enabled Collaboration Mechanism, CSU and CSV attestation. within FPGA using different type of solutions to ensure user authentication and data security, data collaboration and verifiable attestation to build a computational trust with cloud environment that can be served client or huge enterprises anywhere. These solutions implemented on implemented on cloud or individually depend on the requirement of the cloud client because it done by the Client.

VI. Conclusion

In this paper we have presented a survey of the Reconfigurable Accelerators for Cloud Computing. Various contributions have been reviewed with focus on hardware applications such as MapReduce, In-memory Databases, Spark, search engines and page ranking. It was observed from the considered literature that system speed increase while using FPGA in cloud computing rather than a kernel speed and the range of the speed is from 1x to 32x. configuration the hardware accelerator uses high level language such as OpenCL, C++. Moreover, SPARK environment supports high performance in cloud computing. GPU and FPGA accelerator developed faster from CG1 in 2011 to F1 in 2017. Moreover, GPU is 10X times faster than CPU in a performance and 5X time in energy efficiency. Also GPU has 1000's of cores to optimize the parallel task while CPU has few cores. Furthermore, a discussion of security in the cloud and how we can make the root secure between the client and the cloud service by using four security solutions is presented.

VII. References

[1] E. Knorr, "What is cloud computing? Everything you need to know now," InfoWord, 2017. [Online]. Available: https://www.infoworld.com/article/2683784/cloud-computing/what-is-cloud-computing.html. [Accessed 8 12 2017].

[2] Techopedia.com, "Techopedia," techopedia, 2017. [Online]. Available: https://www.techopedia.com/definition/26515/cloud-acceleration. [Accessed 8 12 2017].

[3] Techopedia, "Accelerator," Techopedia Inc, 2017. [Online]. Available: https://www.techopedia.com/definition/31677/accelerator. [Accessed 8 12 2017].

[4] A. Jain, "Arxiv," 8 may 2017. [Online]. Available: https://arxiv.org/pdf/1705.02730.pdf. [Accessed 11 12 2017].

[5] R. G. S. Oliver Knodel, "RC3E: Provision and Management of Reconfigurable Hardware Accelerators in a Cloud Environment," *semanticscholar,* pp. 48-53, 1 Sep 2015.

[6] K. V. a. S. S. S. A. Fahmy, "Virtualized FPGA Accelerators for Efficient Cloud Computing," *IEEE 7th International Conference on Cloud Computing Technology and Science (CloudCom), Vancouver, BC,,* pp. 430-435, 2015, pp.

[7] H. L. Kidane, E.-B. Bourennane and G. Ochoa-Ruiz, "NoC Based Virtualized Accelerators for Cloud Computing," *IEEE 10th International Symposium on Embedded Multicore/Many-core Systems-on-Chip (MCSOC), Lyon,* pp. 133-137, 2016.

[8] A. Putnam, "A Reconfigurable Fabric for Accelerating Large-Scale Datacenter Services," *IEEE Micro,* vol. 35, no. 3, pp. 10-22, 2015.

[9] a. J. Singh, "Effective Model and Implementation of Dynamic Ranking in Web Pages," *Fifth International Conference on Communication Systems and Network Technologies, Gwalior*, 2015, pp. 1010-1014.

[10] J. Yan, Z.-X. Zhao, N.-Y. Xu, X. Jin, L.-T. Zhang and F.-

H. Hsu, "Efficient Query Processing for Web Search Engine with FPGAs," *IEEE 20th International Symposium on Field-Programmable Custom Computing Machines,*
Toronto, 2012*,* pp. 97-100.

[11] Safari, "Chapter 7. MapReduce Types and Formats, " Safari Books Online, 2017. [Online]. Available: https://www.safaribooksonline.com/library/view/hadoop-the-definitive/9781449328917/ch07.html. [Accessed 20 12 2017].

[12] D. D. a. C. Kachris, "High-level synthesizable dataflow MapReduce accelerator for FPGA-coupled data centers," *nternational Conference on Embedded Computer Systems: Architectures, Modeling, and Simulation (SAMOS), Samos*, 2015*,* pp. 26-33.

[13] D. D. C. S. S. Christoforos Kachris, "An FPGA-based Integrated MapReduce Accelerator," *J Sign Process Syst*, 2017*,* p. 357–369.

[14] H. L. a. G. Park, "JUMPRUN: A hybrid mechanism to accelerate item scanning for in-memory databases, " *IEEE International Conference on Big Data and Smart Computing (BigComp), Jeju*, 2017*,* pp. 231-238.

[15] O. Kocberber, B. Grot, J. Picorel, B. Falsafi, K. Lim and P. Ranganathan, "Meet the walkers accelerating index traversals for in-memory databases,"46th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO), Davis, CA, 2013, pp. 468-479.

[16] S. B. Y. a. K. S. P. M. Alam, "Performance of Point and Range Queries for In-memory Databases Using Radix Trees on GPUs," *IEEE 18th International Conference on High Performance Computing and Communications;* pp. 1493-1500.

[17] K. V. a. N. Radhika, "A big data framework for intrusion detection in smart grids using apache spark," *International Conference on Advances in Computing, Communications*
and Informatics (ICACCI), 2017,pp. 198-204.

[18] M. E. a. H. A. R. Morcel, "FPGA-Based Accelerator for Deep Convolutional Neural Networks for the SPARKEnvironment," *EEE International Conference on Smart Cloud (SmartCloud), New York, NY,* , 2016, pp. 126-133.

[19] N. Nguyen, M. M. H. Khan, Y. Albayram and K. Wang, "Understanding the Influence of Configuration Settings: An Execution Model-Driven Framework for Apache Spark Platform," *IEEE 10th International Conference on Cloud Computing (CLOUD)*, Honolulu, CA, 2017, pp. 802-807.

[20] J. Bi, H. Yuan, Y. Fan, W. Tan and J. Zhang, "Dynamic Fine-Grained Resource Provisioning for Heterogeneous Applications in Virtualized Cloud Data Center," *2015 IEEE 8th International Conference on Cloud Computing*, New York City, NY, 2015, pp. 429-436.

[21] J. Zhong and B. He, "Towards GPU-Accelerated Large-Scale Graph Processing in the Cloud," *2013 IEEE 5th International Conference on Cloud Computing Technology and Science*, Bristol, 2013, pp. 9-16.

[22] K. Eguro and R. Venkatesan, "FPGAs for trusted cloud computing," *22nd International Conference on Field Programmable Logic and Applications (FPL)*, Oslo, 2012, pp. 63-70.

[23] J. A. M. Mondol, "Cloud security solutions using FPGA," *Proceedings of 2011 IEEE Pacific Rim Conference on Communications, Computers and Signal Processing*, Victoria, BC, 2011, pp. 747-752.