Co-design Implications of Cost-effective On-demand Acceleration for Cloud Healthcare Analytics: The AEGLE approach

Dimosthenis Masouros*, Konstantina Koliogeorgi*, Georgios Zervakis*, Alexandra Kosvyra†, Achilleas Chytas†
Sotirios Xydis*, Ioanna Chouvarda†, Dimitrios Soudris*
*Microprocessors and Digital Systems Laboratory, ECE , National Technical University of Athens, Greece
†Lab of Computing, Medical Informatics and Biomedical Imaging Technologies, SM, Aristotle University of Thessaloniki, Greece
*{demo.masouros, konstantina, zervakis, sxydis, dsoudris}@microlab.ntua.gr
†{aekosvyra, achillec, ioannach}@auth.gr

***Abstract*—Nowadays, big data and machine learning are transforming the way we realize and manage our data. Even though the healthcare domain has recognized big data analytics as a prominent candidate, it has not yet fully grasped their promising benefits that allow medical information to be converted to useful knowledge. In this paper, we introduce AEGLE's big data infrastructure provided as a Platform as a Service. Utilizing the suite of genomic analytics from the Chronic Lymphocytic Leukaemia (CLL) use case, we show that on-demand acceleration is profitable w.r.t a pure software cloud-based solution. However, we further show that on-demand acceleration is not offered as a "free-lunch" and we provide an in-depth analysis and lessons learnt on the co-design implications to be carefully considered for enabling cost-effective acceleration at the cloud-level.**

***Index Terms*—Cloud, Platform as a Service (PaaS), Big Data Framework, Big Data Analytics, Co-design, On-demand Acceleration, Genomic.**

I. Introduction

At the centre of health debates there are open questions on how to manipulate, share and produce value out of data [1], [2]. Even though the term Big-Data has become a buzzword in the field of information technology, its applicability on biological data is still limited. Leveraging the increasing number of healthcare data, requires technologies capable of processing such amounts of data efficiently. To that end, business interest is growing, like in the case of Open Data initiative, where big data health providers, research institutes and industry aim to develop a vendor-neutral Big-Data platform [3].

The use of available medical data can allow clinicians to simulate potential outcomes and thus prevent patients from undergoing ineffective treatments or provide better treatment plans. In other words, accumulating data to develop a greater understanding of pathophysiological processes will result in significant healthcare improvements. However, the strategic advantage brought by Big-Data in healthcare still materializes at slow paces, as only some large-scale organizations have established few pilot or proof-of-concept projects.

Data-driven services are still needed to cater for the data versatility, volume, velocity and veracity within the whole data value chain of healthcare analytics. Currently, none of the existing Big-Data EU projects are completely dedicated to healthcare and the provision of corresponding services, or the management of diseases. The AEGLE project aspires to bridge this gap, by implementing a full data value chain to create new value out of rich, multi-diverse health data with the goal to revolutionize integrated and personalized healthcare services.

The project builds upon the synergy of cloud technologies together with heterogeneous high performance reconfigurable acceleration for delivering optimized analytic services on Big-Bio Data applications. At local level, the data are anonymized and uploaded to the cloud. At cloud level, the framework consists of the frontend and the backend part. On the frontend, an advanced visualization service and a friendly user interface simplify the data visualization and the execution of complex analytics workflows. On the other hand, the cloud-backend is the core of AEGLEs big data framework, as it is responsible for data storage and efficient execution of conventional and accelerated workflows. AEGLE's modular deployment envisions to support and enable a healthcare oriented analytic design framework whose functionalities and services are not provided solely by a unique user interface. Users of the framework can develop their own user interfaces at the local level and interconnect them with the services and tools provided at the Cloud level.

The rest of the paper is organized as follows. In section II, we present in brief AEGLE's targeted use-cases and we give detailed information regarding the overall architecture of AEGLE framework and its innovative on-demand acceleration mechanism. In section III we analyze the essential costs for the operation of the platform. Finally, section IV explores the acceleration capabilities of CLL workloads as well as the co-design implications for cost-effective acceleration at the cloud, while section V concludes the paper.

This work is partially funded by the EU Horizon 2020 research and innovation programme, under project AEGLE (http://www.aegle-uhealth.eu/en/), grant agreement No 644906.

978-3-9819263-2-3/DATE19/©2019 EDAA

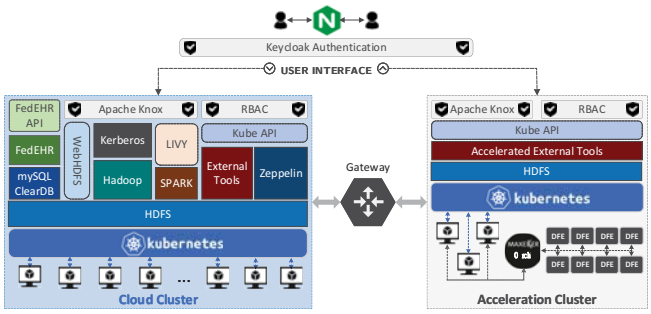

Fig. 1: AEGLE's Big Data Framework hierarchy.

II. The AEGLE cloud infrastructure

AEGLE cloud infrastructure hosts data and analytic tools from three distinct use cases, i) Chronic Lymphocytic Leukemia, ii) Type II diabetes and iii) Intensive Care Unit. In this paper, we utilize CLL as our driver case to highlight the technical contribution, due to the high workload divergence of CLL workflows, i.e. large scale descriptive statistics, machine learning and genomic pipelines, that make a strong claim on the utilization of the on-demand acceleration features.

Fig. 1 depicts an overview of the architecture of the AEGLE platform. On the bottom of the pyramid, lies the hardware stack. The hierarchical design of AEGLE allows the accommodation of any type of devices, i.e., bare-metal servers, virtual machines and accelerator devices. In addition, it allows the extension of its hardware stack, enabling new accelerators and/or entire acceleration clusters to be transparently federated in the existing infrastructure. Currently, AEGLE's hardware stack consists of two clusters, a *conventional* cluster comprised of ordinary virtual machines and an *accelerated* cluster that pairs virtual machines with Maxeler MPC-X servers providing dataflow acceleration of specific applications.

AEGLE platform exposes two kinds of services, *static* and *dynamic* ones. Static services are always up and running and constitute the backbone of the framework, as they are responsible for authentication of users, persistent storage of data, and mainly for the interaction of users with the framework through the user interface. On the other hand, dynamic services are operations that the users of AEGLE perform on the platform, such as workflows execution or dataset visualization.

Resource management: To efficiently manage the deployment of applications on this large and heterogeneous pool of hardware resources, Kubernetes [4] is adopted as the core resource manager, providing automated deployment of containerized applications. Kubernetes is responsible for scheduling workloads on the appropriate cluster (conventional or accelerated), as well as managing resources among VMs. The accelerated cluster supports multiple MPC-X nodes, which implicates a large pool of resources that could potentially be accessed by multiple independent applications at the same time. This requires some form of cluster level resource management for the acceleration engines, i.e. FPGAs, which is provided by the Maxeler Orchestrator that maintains the availability status of each FPGA. Applications are able to request Data Flow Engine (DFE) resources from a centralized management facility (i.e. the orchestrator) that can allocate, schedule and manage resources in the system.

Workloads: AEGLE provides a plethora of tools for analyzing and examining healthcare data. Within AEGLE, more than 100 analytics have been developed, including analytics for healthcare data and predictive models [5]. The *de-novo* analytics are developed over state-of-the-art Big Data platforms, e.g. Hadoop [6], Spark [7], allowing fast and general-purpose computations, interactive queries and stream processing. Additionally, AEGLE supports a variety of existing analytics, tools, and execution engines, that are widely used in the bio-medical research domain, e.g. SeqMule [8], TopHat [9], which provide genomics analyses, such as DNA or RNA sequencing. These tools are deployed and managed as containerized applications by Kubernetes. Last, to simplify the data representation produced by the execution of complex workflows, the platform provides advanced visualization techniques by utilizing the Apache Zeppelin framework [10].

III. Cost model of AEGLE Cloud Infrastructure

AEGLE deployment is amenable to the complex cloud cost models. In order to efficiently exploit the advanced on-demand acceleration features of AEGLE platform, a detailed and realistic assessment of cost w.r.t to performance is needed.

A. Cost model of AEGLE's PaaS

AEGLE's software components are based on open source solutions. Apache licensing 2.0 conditions enable the free commercial use of this open source software without further costs. AEGLE software platform is only amenable to licensing conditions for the GNUBILAs FedEHR Capsule software that provides data anonymization, upload and repository services

The big data platform of AEGLE, has been deployed on Microsoft's Azure Cloud platform [11], utilizing the Azure Kubernetes Service (AKS). The total platform cost (TC) per month can be calculated as follows:

$$TC = C_M + C_{SS} + C_S + C_{LIC} + C_{DS} + C_{HW} + C_{BW} \quad (1)$$

where C_M is the cost for the Virtual Machines used for the deployment of the master nodes of Kubernetes cluster, C_{SS} is the cost for the Virtual Machines hosting the required static services, C_{DS} is the cost for the Virtual Machines hosting dynamic services, C_S is the cost for storage purposes, C_{BW} is the cost for bandwidth transfer inside and outside the cloud cluster, C_{HW} is the cost for hardware accelerators and C_{LIC} is the cost for the software licenses (FedEHR capsule).

Kubernetes master nodes: The master nodes of the cluster should operate indefinitely (24/7), as they are the backbone of Kubernetes ensuring the fluid operation of the cluster. For reliability and fault tolerance reasons, 3 Master nodes were deployed, however the number of master nodes may vary. As a managed Kubernetes service, AKS is free, so the master nodes of the cluster are free of charge ($C_M = 0$).

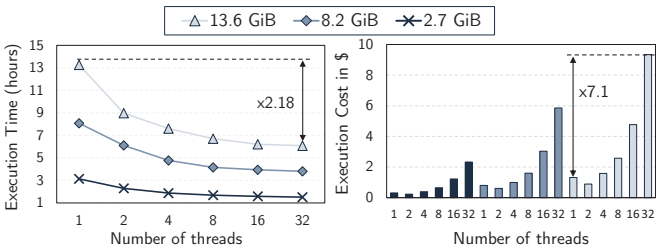

Fig. 2: Execution time and total cost for different number of threads and input datasets for the SeqMule workload.

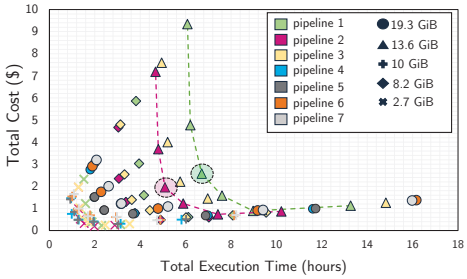

Fig. 3: Total cost and time for software execution of Seqmule pipelines for different input datasets and number of threads.

Static services: The static services of AEGLE comprise of the user interface, the security services, the FedEHR anonymizer and the HDFS cluster, which should be always up and running. In order to provide a high availability and fault tolerant cluster, Kubernetes is configured to schedule HDFS nodes to different VMs. Under normal traffic conditions on the platform, 3 datanodes and 1 namenode are utilized. For the static services, we have chosen the L4 machines with a cost of \$0.372/hour ($C_{SS} = \$1071.36$/month).

Storage and Traffic: Storage costs depend on the amount of data hosted on the platform. Azure charges \$60 per 1TB of File Disk per month. Within AEGLE we host approximately 1TB of anonymized medical data and provide a replication factor of 3 on the HDFS side. Therefore, we utilize 1TB of persistent disk for each HDFS node ($C_S = \$240$/month). Moreover, the outbound traffic is charged for \$0.07 per GB.

In total, the expenses for the basic operation of the platform are estimated to $TC = \$2900.34$. This cost might seem high at a first glance, but the platform can provide services to tens of patients, lowering the individual costs to lower levels.

B. Cost-aware acceleration services

Although acceleration services are available on demand, they should be used cautiously. The tradeoff between speedup gains and cost increment should always be taken into account, even when deciding whether to scale on pure software (i.e., number of threads) or not. For example, Fig. 2 shows the scaling of cost and execution time, w.r.t. the number of threads and input dataset size for the SeqMule pipeline implementing full exome sequencing. As depicted, cost growth escalates quickly compared to savings in execution time. Even for the largest dataset sizes, where we observe better scaling in terms of threads (x2.18 for 32 threads), the increase in cost is greater than time gains by a factor of three (x7.1). This shows that utilizing resources recklessly should be avoided, especially when aiming to keep costs at low levels.

The aforementioned scenario is becoming even more complex, when making use of the acceleration cluster. Acceleration value depends on whether the gained speed-up makes up for traffic charges and the increased cost of occupying an acceleration machine rather than a conventional one. Equation 2 provides a lower bound on the acceleration speedup,x, required in order to have cost-effective acceleration services:

$$\frac{t_{CPU}}{x} * C_{HW} + C_{BW} \leq t_{CPU} * C_{VM} \quad (2)$$

where t_{CPU} is the execution time (hours) of a workload on a conventional VM, i.e. VM without having access to acceleration cluster, and C_{VM} is the cost of the VM. The above equation implies a fundamental principle for cost effective on-demand acceleration, i.e. the gained acceleration speedup should encompass the increased costs of dedicated HW resources, e.g. FPGAs, as well as the data transfer costs from/to the cloud storage. As shown in the next section, this fundamental principle directly questions the currently dominating model of kernel-specific cloud acceleration libraries/stores and suggests for more holistic HW/SW co-designed solutions.

IV. Cost-effective acceleration of AEGLE's CLL Use Case

Predictive and descriptive CLL analytics are exhibiting low execution runtimes, violating Eq. 2 bounds, e.g. transferring the input data to the accelerated cluster would take up as much time as pure software execution. On the contrary, SeqMule forms a good candidate for acceleration due to its intense time requirements (8-12 hours for realistic inputs). SeqMule performs automated human exome/genome variants detection, where short fragments of DNA reads are first aligned to a genome reference and then used for variant calling. AEGLE includes two major types of genome analysis in its workflows. The first one includes a single aligner (BWAMEM/Bowtie2) operating on one input dataset, followed by multiple variant callers (GATKLite, SamTools, Freebayes, Varscan). The second type performs alignment on two input datasets (e.g pre- and after-treatment read sequences) and a single variant caller (Varscan) is utilized. Fig. 4 illustrates the distribution of time per stage for each of the examined Seqmule analyses. Aligners and variant callers take up most of the computation time and thus are usually the target for acceleration. Still, the overall time of minor stages is not negligible either.

A. Co-design implications of genomic workflows acceleration

Careful consideration is required to decide if it is cost-efficient for users to execute a SeqMule pipeline on the accelerated cluster. For that purpose, a small exploration of the available options is presented. Each pipeline is executed

(a) Normal Analysis (b) Somatic Analysis

Fig. 4: Execution's time distribution for SeqMule pipelines.

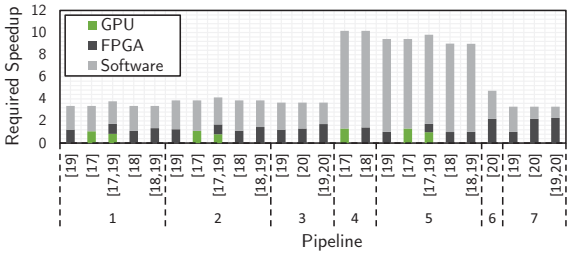

Fig. 5: Total cost and time for software execution of Seqmule pipelines for different input datasets and number of threads.

for different input sizes and different number of threads (1-2-4-8-16-32) and the resulting set of configurations is depicted in Fig. 3. As illustrated with dotted lines, configurations of the same pipeline and input size lie on the same frontier, where lower execution times correspond to higher number of threads. Executing the specific configuration on the accelerated cluster should be more efficient than the point of the frontier with the best trade-off between cost and latency (highlighted points).

After identifying the best cost-latency trade-off points, we apply Eq.2 to acquire the lower acceleration factor x for each pipeline. Given this value and the profiling results of Fig. 4, we investigate how this speedup can be actually acquired. We form our acceleration library adopting state-of-the-art hardware accelerators for SeqMule aligners and variant callers utilized in these pipelines. To broaden the analysis, we considered accelerators targetting both GPUs and Xilinx FPGAs, as well as Maxeler DFEs [12], [13], [14], [15]. For each pipeline, we consider all combinations of hardware accelerators from our library and measure the resulting speedup and cost.

Fig.5 presents the results for all pipelines for a single dataset input size (8.2GB). Each bar stands for a threshold speedup, from which value and onwards hardware acceleration is more cost efficient. Fig.5 shows that the need for an holistic co-design solution is evident in the case of cloud level acceleration, since none of the examined accelerated pipelines achieves the cost-effectiveness threshold. Configurations for pipelines 1,2,3,6,7 partly achieve the target speedup value by relying on FPGAs, GPUs or both. The remaining speedup can be acquired by allocating more software resources in order to boost the performance of the pipeline stages that are still executed on software. That is not the case for pipelines 4 to 5, whose hardware-acquired speedup is far below the threshold, making it potentially hard for software resources to compensate for the difference.

The difficulty to acquire the target speedup can be attributed to the fact that the accelerated solutions are applied only to some stages of the pipeline, directly originated from the current cloud model of kernel-specific acceleration libraries. To add to that, although there are many hardware accelerators for the bottlenecks of these tools, there are only a few integrated co-designed implementations that manage to deliver a maximum of only $\times 2$ speedup, i.e. below the cost-effectiveness acceleration threshold.

V. Conclusion

The scope of this work is to present AEGLE's Platform as a Service, highlighting the on-demand acceleration services that are offered through a framework that seamlessly combines cloud and big data technologies. A detailed description of the cost model of AEGLE platform is provided, along with an exploration that aims to export preliminary guidelines for enabling cost-effective acceleration on demand. Utilizing CLL use case as a driver application on this exploration, we present the challenges of effective co-design and highlight the importance of providing an elegant and effective solution for Big Data Health analytics.

References

[1] *Transforming health care through big data*. [Online]. Available: http://www.ihealthtran.com/
[2] *Big data: What is it and why is it important?* [Online]. Available: http://ec.europa.eu/digital-agenda/en/news/big-data-what-it-and-why-it-important
[3] *Top Big Data opportunities for health startups*. [Online]. Available: http://healthstartup.eu/2012/05/top-big-data-opportunities-for-health-startups/
[4] B. Burns, B. Grant, D. Oppenheimer, E. Brewer, and J. Wilkes, "Borg, omega, and kubernetes," *Queue*, vol. 14, no. 1, pp. 10:70–10:93, Jan. 2016. [Online]. Available: http://doi.acm.org/10.1145/2898442.2898444
[5] C. Maramis, A. Gkoufas, A. Vardi, E. Stalika, K. Stamatopoulos, A. Hatzidimitriou, N. Maglaveras, and I. Chouvarda, "Irprofiler–a software toolbox for high throughput immune receptor profiling," *BMC bioinformatics*, vol. 19, no. 1, p. 144, 2018.
[6] K. Shvachko, H. Kuang, S. Radia, and R. Chansler, "The hadoop distributed file system," in *Mass storage systems and technologies (MSST), 2010 IEEE 26th symposium on*. Ieee, 2010, pp. 1–10.
[7] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica, "Spark: Cluster computing with working sets." *HotCloud*, vol. 10, no. 10-10, p. 95, 2010.
[8] Y. Guo, X. Ding, Y. Shen, G. J. Lyon, and K. Wang, "Seqmule: automated pipeline for analysis of human exome/genome sequencing data," *Scientific reports*, vol. 5, p. 14283, 2015.
[9] C. Trapnell, L. Pachter, and S. L. Salzberg, "Tophat: discovering splice junctions with rna-seq," *Bioinformatics*, vol. 25, no. 9, pp. 1105–1111, 2009.
[10] *Apache Zeppelin*. [Online]. Available: https://zeppelin.apache.org/
[11] *Microsoft Azure Cloud Computing Platform and Services*. [Online]. Available: https://azure.microsoft.com/
[12] E. J. Houtgast, V. Sima, K. Bertels, and Z. AlArs, "An efficient gpuaccelerated implementation of genomic short read mapping with bwamem," *ACM SIGARCH Computer Architecture News*, vol. 44, no. 4, pp. 38–43, 2017.
[13] N. Ahmed, V.-M. Sima, E. Houtgast, K. Bertels, and Z. Al-Ars, "Heterogeneous hardware/software acceleration of the bwa-mem dna alignment algorithm," in *Proceedings of the IEEE/ACM International Conference on Computer-Aided Design*. IEEE Press, 2015, pp. 240–246.
[14] B. Dutro, "Hardware acceleration of the samtools variant caller," 2015.
[15] J. Arram, T. Kaplan, W. Luk, and P. Jiang, "Leveraging fpgas for accelerating short read alignment," *IEEE/ACM Transactions on Computational Biology and Bioinformatics (TCBB)*, vol. 14, no. 3, pp. 668–677, 2017.