Accelerating Edge Metagenomic Analysis with Serverless-based Cloud Offloading

Piotr Grzesik[1] and Dariusz Mrozek[1]

Department of Applied Informatics, Silesian University of Technology
ul. Akademicka 16, 44-100 Gliwice, Poland
dariusz.mrozek@polsl.pl

Abstract. Third-generation nanopore sequencing technologies, along with portable devices such as MinION Nanopore and Jetson Xavier NX, allow performing cost-effective metagenomic analysis in a portable manner. At the same time, we observe the growth of the serverless computing paradigm that offers high scalability with limited maintenance overhead for the underlying infrastructure. Recent advancements in serverless offerings make it a viable choice for performing operations such as basecalling. This paper aims to evaluate if a combination of edge and serverless computing paradigms can be successfully used to perform the basecalling process, with the focus on acceleration of offline edge-based processing with serverless-based infrastructure. For the purposes of the experiments, we proposed a workflow in which DNA sequence reads are processed simultaneously at the edge with Jetson Xavier NX and in the cloud with AWS Lambda in different network conditions. The results of our experiments show that with such a hybrid approach, we can reduce the processing time and energy consumption of the basecalling process compared to fully offline or fully online processing. We also believe that while so far, the adoption of serverless computing for bioinformatic applications is not high, the recent improvements to platforms such as AWS Lambda make it a compelling choice for an increasing number of bioinformatics workflows.

Keywords: nanopore sequencing, edge computing, edge analytics, bioinformatics, Jetson Xavier NX, cloud computing, metagenomics, serverless computing, AWS Lambda

1 Introduction

In recent years, we have seen the fast growth of the popularity of third-generation sequencing technologies. These technologies allow performing metagenomics analysis in a cost-effective manner, thanks to devices such as MinION Nanopore. MinION Nanopore is a sequencing device released by Oxford Nanopore Technologies (ONT), which, due to its small dimensions, weight, and costs, enables portable analysis in mobile laboratories, helping out with monitoring Ebola virus outbreak in Kenya [19], Lassa virus outbreak in Nigeria [23], performing early detection of plant viruses in Africa [9], or monitoring sewage [8]. It even has

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_54

been used on International Space Station [11] or during an ice cap traverse expeditions [15].

The use cases as listed above are taking advantage of edge computing paradigm[30], which enables processing data closer to its source, reduces the amount of data that needs to be sent to the cloud, and provides resilience in situations where Internet connection is unreliable or even unavailable at times. However, the metagenomics analysis at the edge is problematic due to the limited access to computational power at edge devices. A lot of the popular bioinformatics tools are written with multi-node clusters in mind, and they require significant computing power to run processing successfully. Based on the official documentation [4], the MinION device can produce up to 15 GB of data per day, which makes it challenging to take advantage of cloud computing power in cases where network connectivity is unstable, and network throughput is limited. Another challenge in such field applications is access to a reliable power supply. Due to that, it is essential to preserve energy and take advantage of techniques that allow maintaining a sustainable ratio of computational power to energy consumption. In our previous paper [17], we have determined that devices such as Jetson Xavier NX can be successfully used for portable metagenomics. However, performing such analysis in real-time is challenging if the experiments require increased accuracy.

At the same time, we observe the growing popularity of the serverless computing paradigm, which commercially started with the release of AWS Lambda [22] in 2014. The serverless paradigm allows reducing infrastructure maintenance in comparison to cloud servers or virtual machines while at the same time providing a highly scalable execution environment that supports parallel processing very well. While mainly being adopted for use cases such as Web APIs, based on our previous research [16], we can see that serverless computing is also gaining popularity for bioinformatic workflows. In one of our recent works [18], we have validated that a serverless-based solution can be successfully used for performing basecalling of nanopore sequencing data.

This paper aims to evaluate if and how a combination of edge and serverless computing paradigms can be successfully used for performing metagenomics analysis and what benefits such a hybrid approach can offer compared to performing the analysis in a fully offline edge or fully online serverless manner. In particular, we focus on a use case where the fully offline edge basecalling process is accelerated with serverless-based basecalling processing if available network conditions allow for it. Throughout experiments, we aim to highlight the potential reductions in processing time and energy consumption. We believe that the low maintenance overhead and high scalability offered by the serverless paradigm can make it a good fit for this particular use case. The main motivation behind the study is to evaluate how edge-based basecalling process can be potentially speed up without sacrificing energy-efficiency and keeping in mind the constraints of edge-based deployments.

The rest of the paper is organized as follows. In section 2, we review the related works in the area. Section 3 describes the testing workflow and the environment used for experiments. Section 4 focuses on the testing methodology and the results of the performance experiments. Finally, section 5 provides a summary and concludes the results of the paper.

2 Related Works

Portable metagenomics analysis has been gaining more interest in world research and scientific literature in recent years. Oliva et al. [27] presented an overview and benchmarks of bioinformatic tools that can be ported and used on an Android smartphone in order to evaluate if regular smartphones are powerful enough to support portable analytics. The authors considered 23 tools, but only 11 of them were successfully ported to work on an Android device. The only basecalling software that the authors managed to port was Nanocall [13], but the paper does not include a benchmark of basecalling with Nanocall. This research suggests that a new set of tools optimized for ARM architectures will be necessary to support portable analytics on regular smartphones reliably. Another example was presented by Grzesik et al. in [17], where the authors evaluated the feasibility of using a device such as Jetson Xavier NX for performing basecalling and classification operations in an edge computing manner. The authors developed a workflow based on Guppy basecaller and Kraken2 classification software, and throughout their experiments, they determined that Jetson Xavier NX can serve as an energy-effective and performant device that can be used for running metagenomic analysis in a portable manner. Yet another case of portable metagenomics was presented by D'Agostino et al. [14]. The authors proposed hybrid edge-cloud architecture for performing cost-effective metagenomic analysis. During the experiments, they evaluated a workflow that includes basecalling and classification steps using Deepnano and Kraken software. As the edge platform, they used Intel System-on-Chip boards. Based on the performed experiments, the authors suggested that while it is possible to run metagenomic analysis directly on selected devices, none supported the data processing in real-time. Similar research has been performed by Merelli et al. [26], where the authors described a fog computing architecture, based on low-powered portable devices, aimed at performing metagenomic analysis. However, Merelli et al. focused on energy consumption aspects and concluded that the system that would have to support real-time analysis could not be powered by batteries and would require multiple computing boards to process output from a single MinION device.

On the other hand, we also observe a growing interest in research related to the use of serverless computing for bioinformatics applications. Grzesik et al. [16] presented an overview of serverless techniques used for omics data analysis. The authors referred to multiple examples that consider using serverless computing for bioinformatic workflows in their work. One of the cases they mention is an API used for simulating the DNA sequencing data, proposed by Aboukhalil [7]. That solution is based on the ”wgsim” tool for simulating sequence reads based on a provided reference genome. By taking advantage of ”biowasm,” the author managed to compile ”wgsim” to WebAssembly to successfully run it on Cloudflare Workers Unbound, which is a serverless platform. Hung et al. [20] demonstrated how serverless computing could help with reducing computation time of RNA sequencing data analysis. In their application, the authors proposed a three-step architecture with split, merge, and align steps. They identified the merge step as the best one to be potentially parallelized and accelerated with the serverless approach. During the merge step, the human transcriptome reads are aligned by using the Burrows-Wheeler Aligner [25]. In the tested case, they sharded data into 60 MB files which resulted in a workflow that employed over 1,700 serverless functions in parallel. Taking advantage of such architecture reduced the total execution time of the workflow from 2.5 hours for a cloud server to 6 minutes when using serverless functions. Another case where serverless computing allowed for reduced computing time is sBeacon [21], a serverless implementation of the Beacon protocol, proposed and implemented by The Commonwealth Scientific and Industrial Research Organization (CSIRO). By taking advantage of AWS Lambda and AWS S3, CSIRO managed to reduce the time required to upload new genomes into the database from 33 hours when using a cloud server to only 22 seconds. The authors also mentioned that the selected architecture improves data privacy and allows reducing the costs of the infrastructure. CSIRO is one of the leading organizations involved in adopting serverless for bioinformatic workflows. In addition to sBeacon, it also proposed Serverless Variant Effect Predictor (sVEP) used for genomic variants prediction. Thanks to parallelization of the workflow being enabled by the use of AWS Lambda, the authors estimated that sVEP is 99% faster than traditional VEP implementations [29]. Yet another use case developed by CSIRO is GT-Scan [28], a web application that supports finding targets with minimal similar sequences in the genome. By taking advantage of the AWS Lambda and AWS DynamoDB, the authors managed to reduce the application costs from around $700 to $2.50 compared to a cloud server-based solution. The use of serverless for basecalling was validated by Grzesik et al. [17]. The authors implemented and evaluated the possibility of running a basecalling process of nanopore sequencing data. In their solution, they used AWS Lambda with Docker container support. During experiments, they determined that four Lambda functions running in parallel have enough computing power to support near real-time processing of data produced by a single MinION device. The authors also noted that in their experiments, they could scale up to 100 of such functions running simultaneously in less than a minute. In another case, Crespo-Cepeda et al., in their paper [12], analyse opportunities and challenges for using AWS Lambda for bioinformatic workflows. In their work, the authors propose an architecture for running Cloud-DmetMiner, based on AWS Lambda and AWS S3 services. Authors manage to run successful experiments and conclude the paper with suggestion that using serverless approach can reduce the time dedicated to managing and provisioning cloud infrastructure manually. Another successful instance of taking advantage of serverless computing for performing biomedical research has been presented by Kumanov et al., in their work [24]. The authors describe a proof of concept example of performing all-against-all pairwise comparison among 20,000 human

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_54

protein sequences, implemented with Striped Smith-Waterman algorithm. According to the authors, use of serverless cloud computing allowed for increasing speed of execution time at a low cost. In the cited case, the experiment can be accomplished in about 2 minutes for a cost of less than one dollar, which is a speed up of about 250 times in comparison to running the experiment on a laptop computer. The authors also suggest that the similar approach could be effective for tasks such as protein-folding, deep-learning or sequence alignment.

Based on the above findings, we can conclude that there is a growing interest in both taking advantage of serverless infrastructures for bioinformatic workflows and performing metagenomic analytics in a portable manner. This paper aims to expand knowledge in both areas by evaluating how portable analytic workflows can benefit, in terms of processing time and energy consumption reduction, from integration with serverless-based infrastructure. This makes our solution a unique one, since, to our best knowledge, there is no paper yet that proposes and evaluates such a hybrid approach to basecalling.

3 Testing workflow and environment

For the purposes of the evaluation, we propose the workflow in which the offline edge-based basecalling process is enhanced with optional serverless-based cloud acceleration. In this workflow, the FAST5 files containing MinION Nanopore sequencing reads are split into two batches - one to be processed directly on the edge device and the second to be processed in a serverless manner in the cloud environment. Splitting files is based on the estimated processing time for both approaches, taking into account the available network upload speed. After splitting the files into batches, the basecalling process for the first batch is started locally, where the files from the second batch are sent to the AWS S3 bucket. As soon as files appear in the S3 bucket, for each of them, an AWS Lambda function is created to perform the data processing in a parallel manner. The AWS Lambda functions are responsible for downloading the file from the S3 bucket, running the basecalling step, and uploading the results of the process to a separate S3 bucket. An additional process is running locally on the edge device that is responsible for monitoring the S3 bucket with results and downloading them to the edge device. After all processing is done and all files with results are sent back to the edge device, the data is ready for further processing, e.g., for the classification step. The described workflow is presented in Fig. 1.

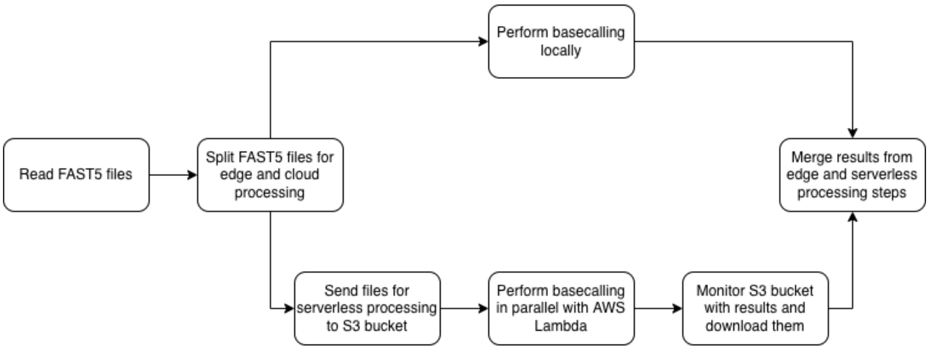
Read FAST5 files
Split FAST5 files for edge and cloud processing
Perform basecalling locally
Send files for serverless processing to S3 bucket
Perform basecalling in parallel with AWS Lambda
Monitor S3 bucket with results and download them
Merge results from edge and serverless processing steps

Fig. 1: Hybrid edge processing with serverless cloud offloading diagram.

As an edge device used during experiments, we selected Jetson Xavier NX. It has already been proven to be an effective and sufficient board for running basecalling experiments in our previous research, mainly thanks to its support for GPU acceleration. Another essential feature of Jetson Xavier NX is its energy efficiency and the capability to control power consumption with five distinct power consumption modes. The full technical specification of Jetson Xavier NX is presented below [3]:

- CPU - 6-core NVIDIA Carmel ARM®v8.2 64-bit CPU 6 MB L2 + 4 MB L3
- GPU - NVIDIA Volta™ architecture with 384 NVIDIA® CUDA® cores and 48 Tensor cores
- Memory - 8 GB 128-bit LPDDR4x 51.2GB/s
- OS Storage - SDHC card (32 GB, class 10)
- DB Storage - Solid State Drive, PNY 500GB M.2 PCIe NVMe XLR8 CS3030
- OS - Ubuntu 18.04.5 LTS with kernel version 4.19.140-tegra

As the serverless platform of choice, we selected AWS Lambda, which we determined in our previous research to be an effective and feasible solution for running basecalling in a serverless manner, thanks to its support for Docker [1] containers and the ability to use up to 10,240 MB of RAM and up to 6 vCPU cores. In our experiments, each Lambda function used a Docker container based on Ubuntu 16.04 operating system with Node.JS script that downloaded the file from AWS S3, invoked the basecalling process on it, and uploaded the results to the output bucket. The Lambda function was configured and deployed with the use of Serverless Framework. [6] As the basecalling software, we used Guppy, which is a closed-source basecaller developed by Oxford Nanopore Technologies. It supports GPU acceleration that can take advantage of GPU on Jetson Xavier NX and support multiple basecalling models (fast and high accuracy). We also considered alternative basecallers such as Deepnano-blitz[10], Bonito [2], or Causalcall [32], but after preliminary testing, they all proven to either not work on AWS Lambda or on Jetson Xavier NX, or offer much lower performance in comparison to the Guppy basecaller. The whole processing was coordinated by a custom Python program running directly on Jetson Xavier NX, that was responsible for splitting the read files and passing them for further processing. Upload to AWS S3 was handled by s3cmd [5], which also offers possibility to throttle upload speeds, that we used to achieve different upload speeds in our experiments.

4 Performance experiments

During experiments, we decided to evaluate the potential reduction in processing time and energy consumed during a basecalling process that is accelerated with serverless cloud offloading. Firstly, we ran a fully offline test where all processing was happening directly on the edge device. In the next step, we ran the fully online test for different upload speeds to measure the estimated processing time for processing all data only in the cloud. Throughout experiments, we ensured throttled upload speeds of 128kB/s, 256kB/s, and 512kB/s, which can be achieved by using, e.g., 3G/4G Internet connection from a smartphone or dedicated board module. During experiments, we used a subset of benchmarking dataset of *Klebsiella pneumoniae* reads [31]. The used dataset had the size of 178 MB and consisted of 2,240 separate sequence reads files. The Guppy basecaller was configured with a ”high accuracy” mode to ensure improved basecalling accuracy. The Jetson Xavier NX used the lowest power mode (2 Cores, 10 W) as well as the highest power mode (6 cores, 15W) to evaluate how the capabilities of edge device impact the results of the experiments.

In the preliminary testing, we determined that the fully offline edge processing of the prepared dataset took 320 seconds with an average power consumption of 11.96 W for lowest power mode and 240 seconds with an average power consumption of 15.3 W for highest power mode. The second workflow in which all files were uploaded to the cloud for online serverless processing took 700 seconds for 512kB/s upload speed, 980 seconds for 256 kB/s upload speed, and 1,520 seconds for 128 kB/s upload speed. For each of these scenarios, the average power consumption of edge device was measured to be around 4.25 W. Based on the outcomes of fully serverless processing for each of the average upload speeds, we determined the split ratio between files that should be processed at the edge and files that should be processed in the cloud that would result in processing all the data in the fastest manner, optimizing for execution speed. The ratio was obtained by comparing average processing speed of processed megabytes per second for edge and cloud processing. Based on that split ratio, we decided that for the upload speed of 128 kB/s, around 17.5% of the data was assigned for cloud processing for lowest power mode and 13.8% of of the data for the highest power mode, for the upload speed of 256 kB/s, 24.9% of data was sent for cloud processing for lowest and 19% for highest power mode, and for the upload speed of 512 kB/s, 31% of data was assigned to be processed on the cloud side for lowest power mode and around 25% for highest power mode.

During experiments in which we evaluated the hybrid edge-serverless workflow with the mentioned split ratio of the files to be processed, we observed a reduction in processing time to 264 seconds and 206.9 seconds for the upload speed of 128 kB/s, to 240 seconds and 193.5 seconds for 256 kB/s, and to 220 seconds and 179.1 seconds and for upload speed of 512 kB/s for lowest and highest power modes, respectively. The results and comparison of all three experimental scenarios for lowest power mode are presented in Fig 2 while results for highest power mode are presented in Fig 3. During the hybrid edge processing with cloud offloading, the average power consumption was equal to 12.4 W for lowest power mode in all tested cases and 15.6 W for highest power mode, also for all tested cases. Fig. 4 presents the energy consumed by the edge device during each considered processing scenario in lowest power mode, while Fig. 5 presents results for highest power mode.

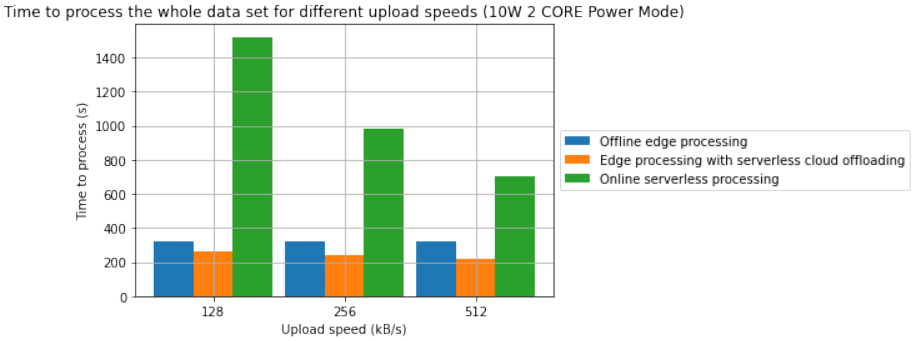

Fig. 2: Dependency between the basecalling execution time and upload speed for various strategies of data processing for lowest power mode of Jetson Xavier NX (offline edge processing vs. edge with cloud offloading vs. online serverless processing).

Based on the obtained results, we can observe that the proposed hybrid approach (edge processing with serverless cloud offloading) allows reducing consumed energy by 14% (lowest power mode) and 12% (highest power mode) as well as processing time by 17.3% (lowest power mode) and by 14% (highest power mode) for the upload speed of 128 kB/s. For 256 kB/s, we can achieve reductions by 22% (lowest power mode) and 20% (highest power mode) as well as 25% (lowest power mode) and 18% (highest power mode) for energy consumed and processing time, respectively, in comparison to the offline edge processing scheme. As we initially expected, we can observe the best results for upload speed of 512 kB/s, where we achieved a 28.7% (lowest power mode) and 24% (highest power mode) reduction of energy consumption as well as 31.2% (lowest power mode) and 25.5% (highest power mode) reduction of processing time compared to offline edge processing. In all considered cases, we see that fully on-

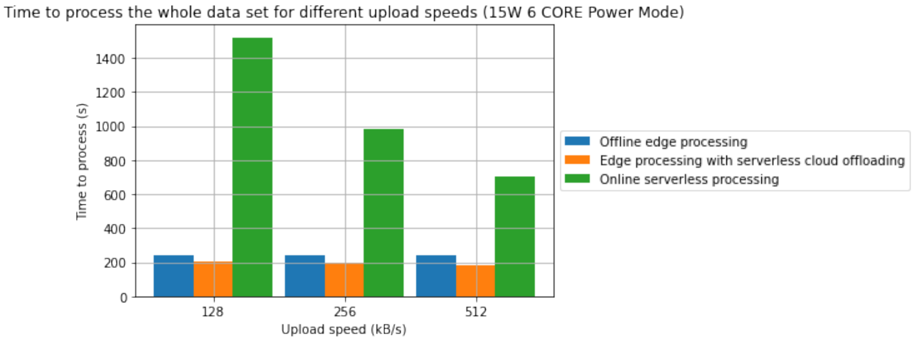

Fig. 3: Dependency between the basecalling execution time and upload speed for various strategies of data processing for highest power mode of Jetson Xavier NX (offline edge processing vs. edge with cloud offloading vs. online serverless processing).

line serverless processing was the slowest of all compared approaches. However, what is interesting, we observed that for the upload speed of 512 kB/s, the fully online serverless basecalling process is more energy-effective than the fully offline edge processing approach, offering a reduction of energy consumption by 22.2% (lowest power mode) and 19.1% (highest power mode), at the cost of processing time being increased to 700 seconds. This suggests that when energy preserving is more critical, a fully online basecalling process might potentially be more effective. We also observe that using highest power mode of Jetson Xavier NX does not necessarily means reduced energy-effectiveness, as in tested cases the energy consumption was similar in comparison to lowest power mode, while the processing time was significantly lower for highest power mode.

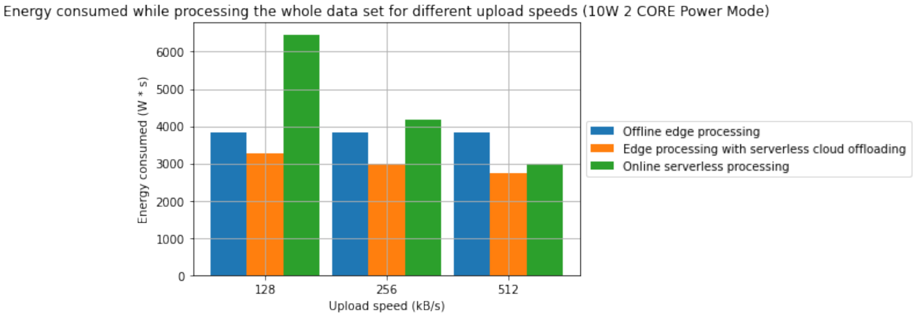

Fig. 4: Dependency between the energy consumed during basecalling for various strategies of data processing for lowest power mode of Jetson Xavier NX (offline edge processing vs. edge with cloud offloading vs. online serverless processing).

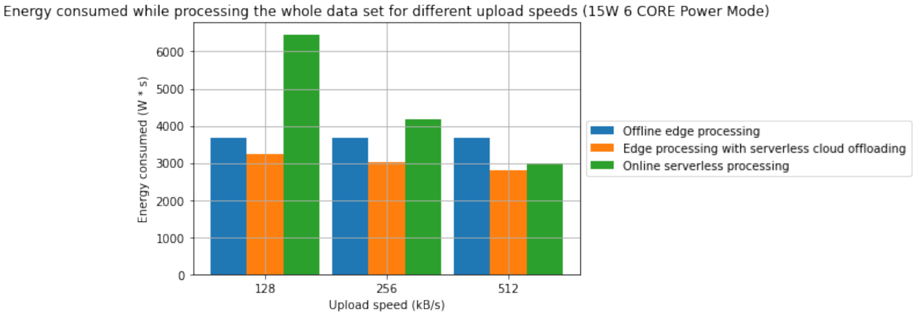

Fig. 5: Dependency between the energy consumed during basecalling for various strategies of data processing for highest power mode of Jetson Xavier NX (offline edge processing vs. edge with cloud offloading vs. online serverless processing).

5 Results Summary and Concluding Remarks

The works presented in the paper expand our previous research where we consider edge and serverless workflows totally separately. Considering the results of the experiments presented in the previous section, we can notice that the combination of offline edge processing with online serverless processing, given access to sufficient network connection, can be an effective strategy for reducing energy consumption and processing time when performing basecalling operations. In one of the cases, we managed to achieve a 31.2% reduction of processing time and 28.7% reduction of energy consumption in comparison to a fully offline process. Our approach may thus optimize the utilization of available computing resources. Throughout the experiments, we have also observed that in the tested scenarios, fully online serverless processing can be more effective than offline processing from an energy consumption standpoint. The use of the serverless AWS Lambda offering ensures scalability of the underlying infrastructure on demand while at the same time keeping the maintenance overhead low, which makes it a very compelling solution as an optional acceleration engine for bioinformatics computations. Thanks to its scale-up capabilities, it can be especially effective when the workloads are rarely executed during the day. While so far we did not see massive adoption of serverless computing for accelerating bioinformatics workflows, we believe that with the rapid growth of the technology in the future, it can enable even more use cases within the bioinformatics domain. This observation causes that there is still a lot of room for potential improvements and developments in this area.

Acknowledgments

The research was supported by the Polish Ministry of Science and Higher Education as a part of the CyPhiS program at the Silesian University of Technology, Gliwice, Poland (Contract No. POWR.03.02.00-00-I007/17-00) and by Statutory Research funds of Department of Applied Informatics, Silesian University of Technology, Gliwice, Poland (grant No BK/RAu7/2022).

References

1. AWS Lambda container image support (accessed on February 5th, 2022), `https://aws.amazon.com/blogs/aws/new-for-aws-lambda-container-image-support/`
2. Bonito basecaller repository on github (accessed on February 5th, 2022, `https://github.com/nanoporetech/bonito`
3. Jetson Xavier NX specification (accessed on February 5th, 2022), `https://developer.nvidia.com/embedded/jetson-xavier-nx-devkit`
4. Nanopore product comparison (accessed on February 5th, 2022), `https://nanoporetech.com/products/comparison`
5. s3cmd (accessed on April 5th, 2022), `https://s3tools.org/s3cmd`
6. Serverless framework (accessed on April 5th, 2022), `https://github.com/serverless/serverless`
7. Aboukhalil, R.: Serverless genomics - using WebAssembly and Cloudflare Workers to power genomics analysis (accessed on February 5th, 2022), `https://robaboukhalil.medium.com/serverless-genomics-c412f4bed726`
8. Acharya, K., Blackburn, A., Mohammed, J., Haile, A.T., Hiruy, A.M., Werner, D.: Metagenomic water quality monitoring with a portable laboratory. Water Research 184, 116112 (2020), `https://www.sciencedirect.com/science/article/pii/S0043135420306497`
9. Boykin, L.M., Sseruwagi, P., Alicai, T., Ateka, E., Mohammed, I.U., Stanton, J.A.L., Kayuki, C., Mark, D., Fute, T., Erasto, J., Bachwenkizi, H., Muga, B., Mumo, N., Mwangi, J., Abidrabo, P., Okao-Okuja, G., Omuut, G., Akol, J., Apio, H.B., Osingada, F., Kehoe, M.A., Eccles, D., Savill, A., Lamb, S., Kinene, T., Rawle, C.B., Muralidhar, A., Mayall, K., Tairo, F., Ndunguru, J.: Tree lab: Portable genomics for early detection of plant viruses and pests in sub-saharan africa. Genes 10(9) (2019), `https://www.mdpi.com/2073-4425/10/9/632`
10. Boža, V., Perešíni, P., Brejová, B., Vinař, T.: Deepnano-blitz: A fast base caller for minion nanopore sequencers. Bioinformatics (Oxford, England) 36, 4191–4192 (05 2020)
11. Castro-Wallace, S.L., Chiu, C.Y., John, K.K., Stahl, S.E., Rubins, K.H., McIntyre, A.B.R., Dworkin, J.P., Lupisella, M.L., Smith, D.J., Botkin, D.J., Stephenson, T.A., Juul, S., Turner, D.J., Izquierdo, F., Federman, S., Stryke, D., Somasekar, S., Alexander, N., Yu, G., Mason, C.E., Burton, A.S.: Nanopore dna sequencing and genome assembly on the international space station. Scientific Reports 7(1), 18022 (Dec 2017), `https://doi.org/10.1038/s41598-017-18364-0`
12. Crespo-Cepeda, R., Agapito, G., Vazquez-Poletti, J.L., Cannataro, M.: Challenges and opportunities of amazon serverless lambda services in bioinformatics. In: Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics. p. 663–668. BCB '19, Association for Computing Machinery, New York, NY, USA (2019), `https://doi.org/10.1145/3307339.3343462`
13. David, M., Dursi, L.J., Yao, D., Boutros, P.C., Simpson, J.T.: Nanocall: an open source basecaller for Oxford Nanopore sequencing data. Bioinformatics 33(1), 49–55 (09 2016), `https://doi.org/10.1093/bioinformatics/btw569`

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_54

14. D'Agostino, D., Morganti, L., Corni, E., Cesini, D., Merelli, I.: Combining edge and cloud computing for low-power, cost-effective metagenomics analysis. Future Generation Computer Systems 90, 79–85 (2019), https://www.sciencedirect.com/science/article/pii/S0167739X18300293
15. Gowers, G.O.F., Vince, O., Charles, J.H., Klarenberg, I., Ellis, T., Edwards, A.: Entirely off-grid and solar-powered dna sequencing of microbial communities during an ice cap traverse expedition. Genes 10(11) (2019), https://www.mdpi.com/2073-4425/10/11/902
16. Grzesik, P., Augustyn, D.R., Wyciślik, L., Mrozek, D.: Serverless computing in omics data analysis and integration. Briefings in Bioinformatics 23(1) (09 2021), https://doi.org/10.1093/bib/bbab349, bbab349
17. Grzesik, P., Mrozek, D.: Metagenomic analysis at the edge with jetson xavier nx. In: Paszynski, M., Kranzlmüller, D., Krzhizhanovskaya, V.V., Dongarra, J.J., Sloot, P.M. (eds.) Computational Science – ICCS 2021. pp. 500–511. Springer International Publishing, Cham (2021)
18. Grzesik, P., Mrozek, D.: Serverless nanopore basecalling with aws lambda. In: Paszynski, M., Kranzlmüller, D., Krzhizhanovskaya, V.V., Dongarra, J.J., Sloot, P.M.A. (eds.) Computational Science – ICCS 2021. pp. 578–586. Springer International Publishing, Cham (2021)
19. Hoenen, T., Groseth, A., Rosenke, K., Fischer, R.J., Hoenen, A., Judson, S.D., Martellaro, C., Falzarano, D., Marzi, A., Squires, R.B., Wollenberg, K.R., de Wit, E., Prescott, J., Safronetz, D., van Doremalen, N., Bushmaker, T., Feldmann, F., McNally, K., Bolay, F.K., Fields, B., Sealy, T., Rayfield, M., Nichol, S.T., Zoon, K.C., Massaquoi, M., Munster, V.J., Feldmann, H.: Nanopore sequencing as a rapidly deployable ebola outbreak tool. Emerging infectious diseases 22(2), 331–334 (Feb 2016), https://pubmed.ncbi.nlm.nih.gov/26812583, 26812583[pmid]
20. Hung, L.H., Niu, X., Lloyd, W., Yeung, K.Y.: Accessible and interactive rna sequencing analysis using serverless computing. bioRxiv (2020), https://www.biorxiv.org/content/early/2020/10/03/576199
21. Jain, Y., Hosking, B., Twine, N., Wilson, L., Zarnegar, A., Rophina, M., Scaria, V., Bauer, D.: sBeacon: cloud-native genomic data exchange. ABACBS-2020 2020, 1 (11 2020)
22. Jonas, E., Schleier-Smith, J., Sreekanti, V., Tsai, C., Khandelwal, A., Pu, Q., Shankar, V., Carreira, J., Krauth, K., Yadwadkar, N.J., Gonzalez, J.E., Popa, R.A., Stoica, I., Patterson, D.A.: Cloud programming simplified: A berkeley view on serverless computing. CoRR abs/1902.03383 (2019), http://arxiv.org/abs/1902.03383
23. Kafetzopoulou, L.E., Pullan, S.T., Lemey, P., Suchard, M.A., Ehichioya, D.U., Pahlmann, M., Thielebein, A., Hinzmann, J., Oestereich, L., Wozniak, D.M., Efthymiadis, K., Schachten, D., Koenig, F., Matjeschk, J., Lorenzen, S., Lumley, S., Ighodalo, Y., Adomeh, D.I., Olokor, T., Omomoh, E., Omiunu, R., Agbukor, J., Ebo, B., Aiyepada, J., Ebhodaghe, P., Osiemi, B., Ehikhametalor, S., Akhilomen, P., Airende, M., Esumeh, R., Muoebonam, E., Giwa, R., Ekanem, A., Igenegbale, G., Odigie, G., Okonofua, G., Enigbe, R., Oyakhilome, J., Yerumoh, E.O., Odia, I., Aire, C., Okonofua, M., Atafo, R., Tobin, E., Asogun, D., Akpede, N., Okokhere, P.O., Rafiu, M.O., Iraoyah, K.O., Iruolagbe, C.O., Akhideno, P., Erameh, C., Akpede, G., Isibor, E., Naidoo, D., Hewson, R., Hiscox, J.A., Vipond, R., Carroll, M.W., Ihekweazu, C., Formenty, P., Okogbenin, S., Ogbaini-Emovon, E., Günther, S., Duraffour, S.: Metagenomic sequencing at the

epicenter of the nigeria 2018 lassa fever outbreak. Science 363(6422), 74–77 (2019), https://science.sciencemag.org/content/363/6422/74

24. Kumanov, D., Hung, L.H., Lloyd, W., Yeung, K.Y.: Serverless computing provides on-demand high performance computing for biomedical research (2018), https://arxiv.org/abs/1807.11659
25. Li, H., Durbin, R.: Fast and accurate short read alignment with Burrows–Wheeler transform. Bioinformatics 25(14), 1754–1760 (05 2009), https://doi.org/10.1093/bioinformatics/btp324
26. Merelli, I., Morganti, L., Corni, E., Pellegrino, C., Cesini, D., Roverelli, L., Zereik, G., D'Agostino, D.: Low-power portable devices for metagenomics analysis: Fog computing makes bioinformatics ready for the internet of things. Future Generation Computer Systems 88, 467–478 (2018), https://www.sciencedirect.com/science/article/pii/S0167739X17324123
27. Oliva, M., Milicchio, F., King, K., Benson, G., Boucher, C., Prosperi, M.: Portable nanopore analytics: are we there yet? Bioinformatics 36(16), 4399–4405 (04 2020), https://doi.org/10.1093/bioinformatics/btaa237
28. What is ”serverless” and ”cloud-native” and when to use it? (accessed on February 5th, 2022), https://bioinformatics.csiro.au/blog/converting-traditional-architecture-to-cloud-native-applications/
29. Serverless VEP (accessed on February 5th, 2022), https://bioinformatics.csiro.au/serverless-vep/
30. Singh, S.: Optimize cloud computations using edge computing. In: 2017 International Conference on Big Data, IoT and Data Science (BID). pp. 49–53 (Dec 2017)
31. Wick, R.R., Judd, L.M., Holt, K.E.: Performance of neural network basecalling tools for oxford nanopore sequencing. Genome Biology 20(1), 129 (Jun 2019), https://doi.org/10.1186/s13059-019-1727-y
32. Zeng, J., Cai, H., Peng, H., Wang, H., Zhang, Y., Akutsu, T.: Causalcall: Nanopore basecalling using a temporal convolutional network. Frontiers in Genetics 10, 1332 (2020), https://www.frontiersin.org/article/10.3389/fgene.2019.01332

ICCS Camera Ready Version 2022
To cite this paper please use the final published version:
DOI: 10.1007/978-3-031-08754-7_54