Using Alignment Chain to Boost Genetic Sequence Alignment Process

Zhuochen Yang, Liwei Liu, Can Liu, Jin Zhang, and Shaolong Chen(✉)

Jiangxi Normal University, Jiangxi Provincial Engineering Research Center of Blockchain Data Security and Governance, Nanchang 330022, CN, China
schen@jxnu.edu.cn

Abstract. The rapid expansion of next-generation sequencing (NGS) technologies has led to a substantial increase in the amount of genetic sequence data. In the field of bioinformatics, variant analysis depends heavily on computationally demanding procedures, such as sequence alignment. Although the use of blockchain technology seems encouraging for executing read mapping in this domain, it poses specific challenges when it comes to examining genetic sequences. This paper presents a new blockchain-based system for aligning genetic sequences, with the goal of improving efficiency in read mapping in distributed networks that have multiple computing nodes. The alignment chain is structured into three tiers that incorporate blockchain features and hierarchical organization, including masters, verifiers, and aligners, which enable streamlined management and processing of sequence data. An alignment workflow has been established to improve the functionality of the alignment chain in use. In addition, we performed sequence alignment on a variety of processors to assess the chain's processing performance with regards to its potential implementation in future blockchain systems. This study contributes to the field of genetic sequence alignment by utilizing blockchain technology and a hierarchical structure, thereby facilitating better read mapping operations in distributed networks. The results emphasize the need for further exploration and refinement to enhance the overall performance and scalability of the alignment chain.

Keywords: Sequence alignment · alignment chain · blockchain · hierarchical structure

1 Introduction

The rapid evolution of NGS technologies since 2007 has revolutionized the field of biological research. However, the massive volume of data generated by NGS platforms remains a significant impediment to the optimal use of these technologies. In fact, the computational analysis required to handle such massive amounts of data has emerged as the most substantial bottleneck. This significant increase in data presents a challenge for bioinformatics and computational biology which requires the creation of strong and efficient algorithms for data storage, management, and analysis. As a result, while the

Y. Cao and X. Shao (Eds.): DIONE 2023, LNICST 515, pp. 76–86, 2025.
https://doi.org/10.1007/978-3-031-80713-8_6

advancements in NGS technologies have revolutionized the field of biological research, they also emphasize the necessity of effective computational strategies for handling the data produced.

The computational processing of data from NGS technologies involves several steps. However, the most significant obstacle is the process of read mapping or read alignment. This process requires identifying the potential loci of short DNA sequences within a larger reference genome. The computational burden of aligning reads increases steadily, not only because of the expanding amount of data that needs to be analyzed, but also because of updates and improvements in the reference genome assemblies. The apparently intuitive approach of creating huge computing clusters able to manage the computational needs associated with handling hundreds of petabytes of data is, regrettably, not feasible due to resource and infrastructure constraints, along with the intricate nature of maintaining systems on such a large scale.

The computational power of the bitcoin network is primarily devoted to maintaining the currency's integrity. This is accomplished by making sure that the creation of the blockchain, the foundation of Bitcoin's ledger, remains consistently difficult. The challenge arises from the proof-of-work mechanism that demands miners to find a specific value called the nonce. By implementing this system, it effectively regulates the rate at which the blockchain is generated by miners, protecting against the possibility of currency devaluation due to overmining. Although proof-of-work schemes in Bitcoin and similar cryptocurrencies offer strong security solutions for these digital currency systems, they are frequently censured for their high computational requirements, which have no practical value beyond the cryptocurrency domain. Therefore, their consumption of significant computational resources for the sole purpose of preserving the integrity of the cryptocurrency system raises apprehension regarding their impact on the environment and efficient use of resources.

Blockchain technology has received significant attention due to its possible applications in various fields, including genomics. Scalability is one of its main challenges and a significant drawback of the technology. Transaction validation and recording in blockchain rely on a distributed network of nodes, which can be computationally intensive and time-consuming. As a result, efficiently processing large-scale sequence alignment data, which requires fast and efficient algorithms, can pose challenges. One drawback of blockchain technology pertains to the privacy and confidentiality of the data put on the blockchain. Although blockchain solutions are celebrated for being immutable and transparent, they could prove problematic for secure and sensitive genetic or medical data. Guaranteeing the security and privacy of sequence alignment material is paramount, and blockchain methods may not offer sufficient security measures.

In this paper, we present a new blockchain-based system that employs genetic sequence alignment to improve the efficiency of read mapping across distributed networks consisting of multiple computing nodes. The system utilizes an alignment chain that follows a hierarchical structure and incorporates essential blockchain features to manage and process sequence data effectively. The framework includes masters, verifiers, and aligners for efficient management of genetic data. We have created an alignment workflow that enhances the alignment chain, leading to improved performance. To evaluate the processing potential of the chain, we carried out sequence alignment on various processors that could potentially be utilized in future blockchain systems. Our initial research has revealed challenges related to processing performance, scalability, and efficiency in the alignment chain. Therefore, it is crucial to conduct more extensive configurations and analyses in future endeavors to effectively address these concerns.

The following are our contributions in this research.

1. The paper proposes the alignment chain, a blockchain-based system designed to improve the efficiency of read mapping in distributed networks with multiple computing nodes. This hierarchical structure integrates blockchain features such as masters, verifiers, and aligners, enabling effective management and processing of sequence data.
2. The paper presents an alignment workflow that enhances the alignment chain and maximizes its output. The workflow includes several entities, namely the master, verifiers, and aligners. Initially, the master distributes certificates to the verifiers, who subsequently assign alignment assignments to numerous aligners. For instance, verifier 2 delegates jobs to aligner 2, aligner 3, and others. These aligners perform sequence alignment tasks assigned to them.
3. The paper examines the challenges relating to processing performance, scalability, and efficiency within the alignment chain, presenting initial findings identifying these concerns. It stresses the necessity for further research to tackle these issues and enhance the overall performance and scalability of the alignment chain.

The structure of the paper is as follows. In Sect. 2, we present the background information and knowledge. Afterwards, Sect. 3 demonstrates the related work and their pros and cons. Additionally, Sect. 4 delves into the architecture of the alignment chain, including its hierarchical structure and alignment workflow. Moreover, Sect. 5 outlines the experiments we conducted and the results we obtained. Lastly, in Sect. 6, we conclude our findings and suggest future research directions.

2 Background

In this section, we explain the most popular blockchain implementation, the bitcoin system. Cryptocurrencies using the Bitcoin protocol are digital or virtual currencies that rely on blockchain for security. Bitcoin, the pioneering cryptocurrency, introduced the world to blockchain technology, which serves as a public ledger for transaction data in the Bitcoin network. Transactions are added to a block or code link that makes up the chain, with each transaction recorded on a block. The following section provides a concise overview of the primary components and mechanisms.

- Decentralized Network: Bitcoin and similar cryptocurrencies are decentralized, unlike traditional currencies. This implies that there is no central authority or government regulating the currency's supply or management. Transactions are validated by a network of independent computers across the world, commonly known as miners.
- Blockchain Ledger: The blockchain records all transactions made with these cryptocurrencies. A blockchain is a decentralized public ledger, allowing anyone to view the complete transaction history while maintaining identity privacy through the use of public keys for anonymity.

- Consensus algorithm: The consensus algorithm utilizes powerful computers operated by miners to solve intricate mathematical problems that validate transactions via the proof-of-work process, ensuring transaction legitimacy and preventing double-spending. After solving a problem, the transactions are appended to the blockchain in a new block. The miner who solves the problem is then rewarded with a specific number of bitcoins.
- Wallets and Keys: To own and spend bitcoins, a digital wallet is required. Each wallet has two types of keys: a public key, resembling an address that can be shared for the purpose of receiving funds, and a private key utilized in signing transactions and accessing funds. Losing the private key implies forfeiting access to the bitcoins stored in that wallet.
- Security and Anonymity: Cryptocurrencies such as bitcoin provide pseudonymity as transactions are linked to blockchain addresses rather than names. In addition, the blockchain foundation of these currencies guarantees a high degree of security for transactions.

While bitcoin was the first and remains the most well-known cryptocurrency, there are now thousands of others based on similar principles, such as Ethereum, Litecoin, and Ripple, each with its own unique features and uses. The application of blockchain technology to sequence alignment in genomics is a groundbreaking concept that represents a new frontier in bioinformatics. Blockchain has the potential to offer solutions to some of the challenges posed by the massive datasets generated by NGS technologies.

However, to utilize blockchain for genomics, numerous challenges of significant magnitude need resolution. These include, but are not restricted to, scalability, computational efficiency, genomic data's exceptional complexity, and energy usage. Additionally, ethical and legal concerns about data privacy and sharing necessitate a thorough examination in any application of blockchain technology in genomics. The theoretical potential of utilizing blockchain technology in genomics necessitates significant research and development before it can be transformed into a practical reality.

3 The Related Work

Over the last ten years, there have been considerable advances in bioinformatics and computational biology driven by parallel computing and new hardware architectures. A number of researchers have introduced effective methods for parallelizing different algorithms, especially for DNA and protein sequence alignment, frequently capitalizing on the capabilities of multicore and multiprocessor systems. The constant advancement of methodologies and technologies, exemplified by the adoption of accelerators like manymap and the use of supercomputers such as Sunway TaihuLight, showcases the sustained pursuit of augmenting computational speed and efficiency within the field of bioinformatics.

In 2008, Helal et al. [1] presented a technique for parallelizing multi-dimensional dynamic programming and similar algorithms using multi-processor architectures. In the subsequent year, Nordin et al. [2] described a parallel methodology for executing the Smith-Waterman algorithm, a widely-used tool for bioinformatics analysis. In a further development, Almeida et al. [3] introduced a parallel programming framework specifically designed for multi-core DNA sequence alignment. This new approach is designed for homogeneous multi-core processor architectures, representing a significant advancement in the field. Later, in 2012, Sebastião [4] and his team introduced a flexible hardware architecture utilizing several innovative and highly efficient techniques that notably decrease the computational demands of the widely-used Smith-Waterman sequence alignment algorithm. This innovative approach utilizes data collected by a hardware accelerator to compute alignment scores, thereby restricting the post-processing required by a general-purpose processor during the backtracking phase to a smaller subsequence.

In 2014, Hudson et al. [5] introduced MEMRIS (Memory Random Instruction Sequence), an ARM internal tool for generating random instruction sequences. MEMRIS enables the verification and validation of memory consistency in multiprocessor and multi-cluster systems, supporting up to eight processors in AARCH32 and AARCH64 architectures. The tool is extensible, offering a valuable approach to testing memory consistency. The proposed architecture employs a structured approach to matrix padding, which enables time parallelism in generating replacement matrices for multiple sequences. Additionally, it features a backtracking handler for retrieving aligned sequences. Following this, Ghosh et al. [6] conducted a comparative analysis on the processing time of this framework, using three different scenarios based on temporal and spatial parallelism: dedicated, moderately-dedicated, and shared logic components. Turana et al. [7] analyzed a Hadoop cluster on a Raspberry Pi, demonstrating its potential as commercial hardware for DNA sequence comparison. Their objective analysis highlights the capabilities of this platform for such applications.

In 2019, Zhang et al. [8] introduced a highly efficient approach to search protein databases by utilizing the computational power of the Sunway TaihuLight supercomputer. Concurrently, Feng et al. [9] and their team developed a method named manymap, which enhances the performance of minimap2 - a prominent long read aligner - not only on CPUs but also on GPUs and Intel Xeon Phi processors. Schmidt et al. [10] have also contributed significantly to this field. In 2019, Turakhia et al. [11] and their colleagues developed “Darwin,” a co-processor, in a substantial step toward faster genomic sequence alignment. This innovative tool offers an impressive speedup of up to 15,000X over the leading software used for reference-guided assembly of third-generation reads, without compromising on sensitivity. In 2020, Xu’s team [12] introduced SLPal, a fast bit-parallel algorithm that efficiently compares long DNA sequences across Intel’s multicore and many-core architectures. This novel algorithm significantly enhances sequence alignment without any subjective evaluations. Technical term abbreviations are clearly explained upon initial use.

In 2021, Qureshi et al. [13] and colleagues argued that NGS applications often involve memory-bound computations and random memory access patterns, and could benefit significantly from high-bandwidth memory such as 3D-stacked HBM2, which is notably different from conventional DDR4 DRAMs. The authors further advocate the use of power-efficient computing cores to enhance energy efficiency and performance. In order to accelerate processing, Feng et al. [14] proposed using Heterogeneous Graph Aligner (HGA), a parallel sequence-to-graph alignment algorithm that works on both CPUs and GPUs. Additionally, Lalwani et al. [15] and their team introduced a new approach called cache-supported parallel multi-objective three-level particle swarm optimization algorithm (MO-3LPPSO) for solving the NP-hard problem of structural alignment of complex RNA sequences in bioinformatics. In 2017, Herruzo et al. [16] conducted an analysis of the performance and power consumption of two processor architectures running the FM-index precise matching algorithm, which serves as a benchmark for exact sequence comparison.

4 The Architecture of Alignment Chain

4.1 Hierarchical Structure

The alignment chain utilizes a three-tiered structure that blends blockchain and hierarchical organization to effectively manage and process sequence data.

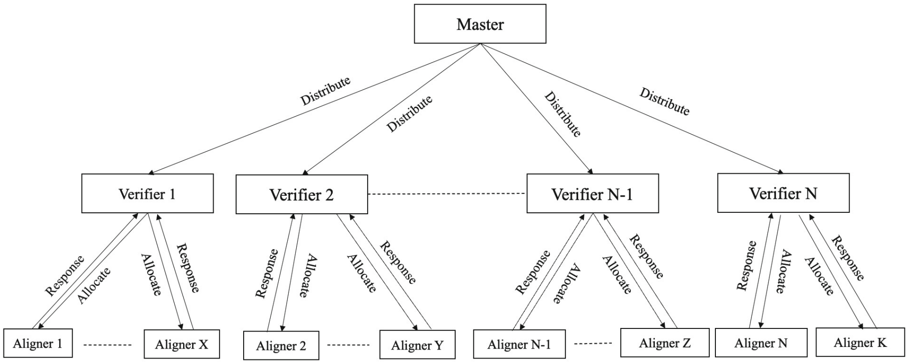

Fig. 1. Three-tiered structure of alignment chain

1. *Master*

At the top of this hierarchy is the root master, which tracks and validates authority servers in the middle-level that provide sequence data to the system. It functions as the central administrative body, ensuring data integrity and alignment authenticity. The master must be a reliable entity accepted by all system users responsible for validating only credible authority servers, such as major sequencing centers.

2. *Verifiers*

These entities are trusted and confirmed by the master. Verifiers serve as intermediaries between the master and the aligners, responsible for providing sequence data to the system. Additionally, they may initially validate or process sequence data and alignments before sending them to miners for further computational tasks.

3. *Aligners*

The third tier comprises aligners comparable to traditional blockchain miners. They perform alignment tasks on the sequence data supplied by authority servers, which could involve activities like sequence alignment, variant calling, or other forms of genomic data analysis. Aligners receive rewards for successfully completing these tasks, akin to bitcoin network miners receiving rewards for proof-of-work.

In Fig. 1, the master distributes certificates to the verifiers, who in turn allocate alignment tasks to multiple aligners. For example, verifier 2 assigns jobs to aligners 2, 3,…, and Y, and verifies the results. The validated alignments are then authenticated and returned to the aligners. Additionally, the aligners have the ability to handle tasks from various verifiers.

4.2 Alignment Workflow

The verifier is a central repository that hosts genomic sequences from N samples in the form of FASTQ files. It also includes a proprietary database of reference reads with predetermined mapping locations. The reference database is constructed for each unique combination of a reference genome and sequence aligner. The alignment process is presented in the following pseudocode.

When formulating assignments, the verifier engages in a process of multiplexing and randomizing reads that are extracted from various samples. The aligners then request these assignments from the verifier, which are regulated by a job scheduler. This system not only tracks tasks, but also enforces strict timelines or timeouts to prevent computational deadlocks.

Alignment process in alignment chain

```
Input: short-read sequences, genome reference.
Output: alignment results.
/*Initialize with N sample genomes and a reference database*/
Verifier.initialize(N_samples, ref_database)
Aligners.initialize()
While not Verifier.all_tasks_completed():
    /*Verifier creates assignments*/
    assignments = Verifier.create_assignments()
    For assignment in assignments:
        /*Aligners request assignments from Verifier*/
        requested_assignment = Aligners.request_assignment(assignment)
        /*Aligners map reads to the reference genome and generate BAM files*/
        BAM_file = Aligners.map_reads(requested_assignment)
        /*Aligners send BAM files back to Verifier*/
        Verifier.receive_BAM(BAM_file)
        /*Verifier reassigns alignments to N samples and checks location alignments*/
        Verifier.reassign_alignments_and_check_location()
        /*If locations are aligned correctly, Verifier signs block and sends to corresponding Aligner*/
        If Verifier.location_aligned_correctly():
        Verifier.sign_block_and_send_to_aligner(Aligners)
```

Upon receiving their assignments, the aligners proceed to download and execute the sequence alignment. During this process, reads are mapped to the specified reference genome following an established exchange protocol between the verifiers and the aligners. Once this mapping is completed, the aligners generate alignment result files and send them back to the authority.

In the final phase, the verifier reassigns the alignments to their respective N samples. Simultaneously, it validates the results by verifying the locations. If the locations are correctly aligned with their pre-determined locations, the resulting file is considered valid. The verifier then authenticates the block and returns it to the relevant aligner, thereby completing the process.

This model employs distributed computing and task delegation to efficiently manage the complicated task of aligning genomic sequences. This approach ensures computational efficiency and the accuracy of results.

5 Experiments and the Results

5.1 Experiment Configuration

We developed an initial model to simulate the proposed methodology using the object-oriented features of the C and C ++ programming languages. The prototype is currently designed to function smoothly on the Linux operating system, ensuring compatibility with various secure, open-source environments. To replicate the distributed network in the blockchain system, we executed multiple nodes utilizing general CPUs. Table 1 summarizes the operational environment used to develop and operate the alignment chain prototype.

In terms of the specific bioinformatics functionalities incorporated into the software, we utilized several key libraries and tools. BWA aligner was integrated into the system to perform the task of read mapping, which involves aligning short reads - DNA sequences to a reference genome. We also leveraged the capabilities of SAMtools to facilitate the conversion of sequence alignment map (SAM) files to the more compact binary alignment map (BAM) format, as well as to sort the resulting BAM files.

Table 1. Running environment configurations

Component	Details
Programming Language	C and C ++
Operating System	Linux
Processor	Intel Xeon E5 series CPUs
Bioinformatics Tools	BWA aligner (for read mapping), SAMtools (for SAM to BAM conversion and BAM sorting), OpenSSL and Crypto library (for BAM processing)
Cryptography Library	OpenSSL and Crypto library (for hashing, digital signature creation, and encryption/decryption of read names)

In terms of securing blockchain technology - which is integral to any blockchain-based platform - we employed the OpenSSL and Crypto library to undertake numerous BAM processing functions. These included pivotal measures such as detecting and eradicating false data and the retrieval of samples. This software library facilitated the implementation of critical blockchain functions, including hashing, digital signature generation, and read name encryption and decryption. This ensures the confidentiality and integrity of data.

5.2 The Results and Its Evaluation

Table 2 demonstrates the processing performance of the alignment chain using various processors. Our objective is to provide a comprehensive evaluation of the computational abilities exhibited by the alignment chain network through a projected assessment. The table provides a performance analysis of the alignment chain using various processors, including E5–1620 v3, E5–2620 v3, and E5–2640 v3. The analysis considers the number of cores, the number of reads, time cost, GFLOP/s (gigaflops per second), GFLOP (total gigaflops), and Read/GFLOP (reads per gigaflop).

The E5–1620 v3 exhibits lower performance and efficiency in terms of read-to-GFLOP (gigaflop) ratio when compared to the E5–2620 v3 and E5–2640 v3. Nonetheless, all processors display scalability issues as the workload increases. The processors demonstrate inadequate scalability with the increase in time taken for higher read numbers serving as evidence. Besides, all processors have a high Read/GFLOP ratio, suggesting that achieving the same computational work requires a considerable number of reads.

Table 2. Performance analysis in alignment chain with various processors

Processor ID	Cores Num.	Reads Num.	Time-cost	GFLOP/s	GFLOP	Read/GFLOP
E5–1620 v3	8	1,000,000	112	4	3,584	279.02
	8	10,000,000	888	4	28,416	351.91
	8	100,000,000	8,122	4	259,904	384.76
E5–2620 v3	12	1,000,000	59	2.87	2,032	492.14
	12	10,000,000	666	2.87	22,937	435.98
	12	100,000,000	6,789	2.87	233,813	427.69
E5–2640 v3	16	1,000,000	41	3.25	2,132	469.04
	16	10,000,000	454	3.25	23,608	423.59
	16	100,000,000	4,304	3.25	223,808	446.81

Processing performance: The GFLOP/s metric measures the processor's computational ability by calculating the number of floating-point operations performed per second. Higher GFLOP/s values generally indicate better performance. When comparing processors within each model, the performance remains consistent for each configuration. However, when comparing between different processors, both the E5–2620 v3 and E5–2640 v3 underperform the E5–1620 v3.

Scalability analysis: The provided data displays the performance metrics for varying read quantities (1000000, 10000000, and 100000000). The preferred condition is for the processor to display excellent scalability, which implies enhanced performance or robust stability with increasing number of reads. As shown in the results in Table 2, the time taken to process increases as the number of reads increases for all processor models, which suggests sub-optimal scalability.

Efficiency analysis: This metric denotes the ratio of reads (Read/GFLOP ratio) performed versus the number of GFLOPs computed. It indicates the efficiency of the processor in terms of utilizing computational power for read mapping operations. Higher values are desirable, as they indicate that more reads were required to achieve the same computational work. Among the processors listed, the E5–1620 v3 has the lowest Read/GFLOP ratio, indicating that it requires more computational operations than the other processors to achieve the same workload - specifically, the read number.

In summary, the performance analysis highlights that as the number of reads increases, the time cost and total GFLOP increase for all processors. However, the Read/GFLOP ratio decreases, indicating a decrease in efficiency. These findings provide insights into the performance characteristics of the processors in the alignment chain and can inform decision-making for selecting an appropriate processor based on the desired read quantity and time constraints.

6 Conclusion and Future Research

The paper initially examines the challenges caused by the swift growth of genetic sequence data attributed to NGS technologies. The proposed solution, named the alignment chain, uses blockchain technology to improve genetic sequence alignment efficiency in distributed networks. The alignment chain operates on various tiers with a hierarchical structure, incorporating essential blockchain technology attributes. The paper describes a structure comprising of masters, verifiers, and aligners that handle and process sequence data. Furthermore, an alignment workflow is introduced, enhancing the performance of the alignment chain. In addition, the experimental analysis integrated into this paper identifies challenges associated with processing performance, scalability, and efficiency within the alignment chain. Further research is required to address concerns and enhance the overall performance and scalability of the alignment chain. This paper investigates the development and assessment of the alignment chain, utilizing blockchain technology to boost the efficiency of genetic sequence alignment in distributed networks. It emphasizes the necessity for more research to optimize the alignment chain's performance and scalability.

Acknowledgments. This research is funded by Jiangxi Provincial Natural Science Foundation, with grant number 20212BAB212007.

References

1. Helal, M., El-Gindy, H., Mullin, L., Gaeta, B.: Parallelizing optimal multiple sequence alignment by dynamic programming. In: 2008 IEEE International Symposium on Parallel and Distributed Processing with Applications, pp. 669–674. IEEE Computer Society, Sydney (2008)
2. Nordin, M., Rahman, A.: Utilizing MPJ express software in parallel DNA sequence alignment. In: 2009 International Conference on Future Computer and Communication, pp. 567–571. IEEE Computer Society, Kuala Lumpar (2009)

3. de Almeida, T.J.B.M., Roma, N.F.V.: A parallel programming framework for multi-core DNA sequence alignment. In: 2010 International Conference on Complex, Intelligent and Software Intensive Systems, pp. 907–912. IEEE Computer Society, Krakow (2010)
4. Sebastião, N., Roma, N., Flores, P.: Integrated hardware architecture for efficient computation of the \$n\$-best bio-sequence local alignments in embedded platforms. IEEE Trans. Very Large Scale Integr. (VLSI) Syst. **20**(7), 1262–1275 (2012)
5. Hudson, J., Kurucheti, G.: A configurable Random Instruction Sequence (RIS) Tool for memory coherence in multi-processor systems. In: 2014 15th International Microprocessor Test and Verification Workshop, pp. 98–101. IEEE Computer Society, Austin (2014)
6. Ghosh, S., Mandal, S., Saha Ray, S.: A scalable high-throughput pipeline architecture for DNA sequence alignment. In: TENCON 2015 - 2015 IEEE Region 10 Conference, pp. 1–6. IEEE Computer Society, Macao (2015)
7. Turana, J.S., Sukoco, H., Kusuma, W.A.: Hadoop performance analysis on raspberry Pi for DNA sequence alignment. TELKOMNIKA (Telecommun. Comput. Elec. Control) **21**(3), 1059–1066 (2016)
8. Zhang, H., Fu, Y., Feng, L.-B., Zhang, Y., Hua, R.: Implementation of hybrid alignment algorithm for protein database search on the SW26010 many-core processor. IEEE Access **7**, 128054–128063 (2019)
9. Feng, Z., Qiu, S., Wang, L., Luo, Q.: Accelerating long read alignment on three processors. In: Proceedings of the 48th International Conference on Parallel Processing, pp. 1–6. Association for Computing Machinery, Kyoto (2019)
10. Schmidt, B.: Bioinformatics: High Performance Parallel Computer Architectures, 1st edn. CRC Press Inc, NW Boca Raton (2010)
11. Turakhia, Y., Bejerano, G., Dally, W.J.: Darwin: a genomics co-processor provides up to 15,000X acceleration on long read assembly. SIGPLAN Not. **25**(2), 199–213 (2019)
12. Xu, X., et al.: SLPal: accelerating long sequence alignment on many-core and multi-core architectures. In: 2020 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 2242–2249 (2020). IEEE International Conference on Bioinformatics and Biomedicine (BIBM), Seoul, Korea (South) (2020)
13. Qureshi, Y.M., et al.: Genome sequence alignment - design space exploration for optimal performance and energy architectures **70**(12), 2218–2233 (2021)
14. Feng, Z., Luo, Q.: Accelerating sequence-to-graph alignment on heterogeneous processors. In: Proceedings of the 50th International Conference on Parallel Processing, p. 26:10. Association for Computing Machinery, Lemont (2021)
15. Lalwani, S., Sharma, H.: Multi-objective three level parallel PSO algorithm for structural alignment of complex RNA sequences. Evol. Intel. **14**, 1251–1259 (2021)
16. Herruzo, J.M., Fernandez, I., González-Navarro, S., Plata, O.: Enabling fast and energy-efficient FM-index exact matching using processing-near-memory. J. Supercomput.Supercomput. **77**, 10226–10251 (2021)