4) RasDaMan [6]: It is one of the earliest next-generation array DBMS for multi-dimensional discrete data, supporting an extended SQL query language. It stores its data as tiles, i.e., possibly non-aligned sub arrays, and resorts to blobs in an external DBMS. While their optimizer provides a rich set of heuristic-based rewrites, to the best of our knowledge, RasDaMan does not perform joint optimization over relational and array data.

5) SciDB [2]: It is an array database that, in contrast to RasDaMan, provides its own shared-nothing storage layer. This allows SciDB to store and query tiles more efficiently. It provides a variety of optimizations, like dealing with overlapping chunks and data compression. However, as it was shown in [8], it does not efficiently perform massive operations like mapping and joins.

6) TileDB [18]: It is a system that stores multi-dimensional array data in fixed size data tiles, which is optimized for both dense and sparse multi-dimensional arrays. GenomicsDB [10] is built on TileDB and it is used by the Broad Institute to store genomic variant data in 2D arrays, where columns and rows correspond to genome positions and samples, respectively.

Comment: All the aforementioned technologies provide a general solution for array databases; they build indexes for the dimensions to facilitate fast access which in turn speed up range queries and equi-joins on the dimensions. However, domain-specific join operations in Genomics are similar to theta-join which are not optimized by a dimensional organization. Coding genomic operations using SQL or SQL-like languages that are provided in RasDaMan and SciDB is rather difficult.

VIII. Conclusions

In this paper, we introduce a scalable algorithm for region-preserving genomic operations; we showed that our approach has a strong potential for performance improvement. Our approach is applicable to other interval-based domains, in our research work, we used the GDM model to represent twitter accounts, meteorological measures, and github commits. Region-preservation occurs when no new coordinate regions are created by the query; such situation is common in the interval-based domains.

Our future work will focus on evaluating the extension of the array-based model to arbitrary operations, by looking for classes of array-based optimization beyond region-preservation; we will also consider a stronger integration of the array-based approach with machine learning, by investigating the direct application of machine learning methods to the array-based model.

Acknowledgment

This research is funded by the ERC Advanced Grant project 693174 "GeCo" (Data-Driven Genomic Computing), 2016-2021, and has been supported through a grant by the German Ministry for Education and Research as Berlin Big Data Center BBDC (funding mark 01IS14013A).

References

[1] P. 4. Apache hadoop, oct 2017.
[2] P. 4. Scidb array database, oct 2017.
[3] F. N. Afrati and J. D. Ullman. Optimizing joins in a map-reduce environment. In *Proceedings of the 13th International Conference on Extending Database Technology*, pages 99–110. ACM, 2010.
[4] M.-C. Albutiu, A. Kemper, and T. Neumann. Massively parallel sort-merge joins in main memory multi-core database systems. *Proceedings of the VLDB Endowment*, 5(10):1064–1075, 2012.
[5] Apache. Flink, oct 2017.
[6] P. Baumann, A. Dehmel, P. Furtado, R. Ritsch, and N. Widmann. The multidimensional database system rasdaman. In *Acm Sigmod Record*, volume 27, pages 575–577. ACM, 1998.
[7] M. Bertoni, S. Ceri, A. Kaitoua, and P. Pinoli. Evaluating cloud frameworks on genomic applications. *IEEE Big Data Conference, Santa Clara*, Nov. 2015.
[8] S. Cattani, S. Ceri, A. Kaitoua, and P. Pinoli. Evaluating big data genomic applications on scidb and spark. *Proc. Web Engineering Conference*, June 2017; Rome.
[9] E. Consortium. An integrated encyclopedia of dna elements in the human genome. *Nature*, 2012.
[10] Intel. Intel. genomicsdb.
[11] A. Kaitoua, P. Pinoli, M. Bertoni, and S. Ceri. Framework for supporting genomic operations. *IEEE Transactions on Computers*, 66(3):443–457, 2017.
[12] W. J. Kent, C. W. Sugnet, T. S. Furey, K. M. Roskin, T. H. Pringle, A. M. Zahler, and D. Haussler. The human genome browser at UCSC. *Genome Res.*, 12(6):996–1006, Jun 2002.
[13] A. Kunft, A. Katsifodimos, S. Schelter, T. Rabl, and V. Markl. Blockjoin: efficient matrix partitioning through joins. *Proceedings of the VLDB Endowment*, 10(13):2061–2072, 2017.
[14] M. Masseroli, A. [illegible]akoglu, P. Pinoli, A. Kaitoua, A. Gulino, O. Horlova, L. Nanni, A. Bo[illegible]oni, S. Perna, E. Stamoulakatou, and S. Ceri. Processing of big heterogeneous genomic datasets for tertiary analysis of next generation sequencing data. *Bioinformatics*, 2018. DOI: http://dx.doi.org/10.1093/bioinformatics/bty688.
[15] M. Masseroli, P. Pinoli, F. Venco, A. Kaitoua, V. Jalili, F. Palluzzi, H. Muller, and S. Ceri. Genometric query language: a novel approach to large-scale genomic data management. *Bioinformatics*, 31(12):1881–1888, 2015.
[16] A. Okcan and M. Riedewald. Processing theta-joins using mapreduce. In *Proceedings of the 2011 ACM SIGMOD International Conference on Management of data*, pages 949–960. ACM, 2011.
[17] R. Palamuttam, R. M. Mogrovejo, C. Mattmann, B. Wilson, K. Whitehall, R. Verma, L. McGibbney, and P. Ramirez. Scispark: Applying in-memory distributed computing to weather event detection and tracking. *IEEE International Conference on Big Data (Big Data)*, pages 2020–2026, 2015.
[18] S. Papadopoulos, K. Datta, S. Madden, and T. Mattson. The tiledb array data storage manager. *Proceedings of the VLDB Endowment*, 10(4):349–360, 2016.
[19] J. M. Patel and D. J. DeWitt. Partition based spatial-merge join. *ACM SIGMOD Record*, 25(2):259–270, 1996.
[20] O. Polychroniou, R. Sen, and K. A. Ross. Track join: distributed joins with minimal network traffic. In *Proceedings of the 2014 ACM SIGMOD international conference on Management of data*, pages 1483–1494. ACM, 2014.
[21] W. Rödiger, S. Idicula, A. Kemper, and T. Neumann. Flow-join: Adaptive skew handling for distributed joins over high-speed networks. In *Data Engineering (ICDE), 2016 IEEE 32nd International Conference on*, pages 1194–1205. IEEE, 2016.
[22] J. W. Stamos and H. C. Young. A symmetric fragment and replicate algorithm for distributed joins. *IEEE Transactions on Parallel and Distributed Systems*, 4(12):1345–1354, 1993.
[23] Z. D. Stephens, S. Y. Lee, F. Faghri, R. H. Campbell, C. Zhai, M. J.Efron, R. Iyer, M. C. Schatz, S. Sinha, and G. E. Robinson. Big data: astronomical or genomical? *PLoS biology*, 13(7):e1002195, 2015.
[24] W. Wang, T. Liu, D. Tang, H. Liu, W. Li, and R. Lee. Sparkarray: An array-based scientic data management system built on apache spark. *IEEE International Conference on Networking, Architecture and Storage (NAS)*, August 2016.
[25] S. Wu, F. Li, S. Mehrotra, and B. C. Ooi. Query optimization for massively parallel data processing. In *Proceedings of the 2nd ACM Symposium on Cloud Computing*, page 12. ACM, 2011.
[26] M. Zaharia, M. Chowdhury, M. J. Franklin, S. Shenker, and I. Stoica. Spark: Cluster computing with working sets. *HotCloud*, 10(10-10):95, 2010.
[27] J. Zhou, P.-A. Larson, and R. Chaiken. Incorporating partitioning and parallel plans into the scope optimizer. In *Data Engineering (ICDE), 2010 IEEE 26th International Conference on*, pages 1060–1071. IEEE, 2010.