Karlsruher Institut für Technologie
Fakultät für Informatik

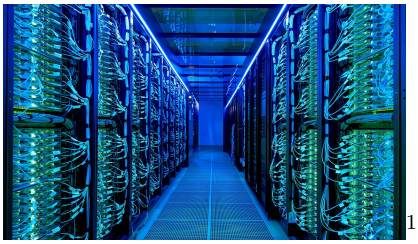
Lehrstuhl Algorithm Engineering
Prof. Dr. Peter Sanders

Master's Thesis

Distributed Suffix Sorting

Overview

Suffix array construction is one of the most fundamental full-text indices with many applications in text indexing, data compression and bioinformatics. The problem boils down of sorting all *suffixes* of a given text in lexicographic order. Although being closely related to the more general problem of *string* sorting, the suffix sorting problem can be solved with work only linear in the text length by exploiting the suffixes' internal structure.
The problem has received a lot of attention, especially in the sequential and shared-memory parallel setting. However, with the ever increasing amount of data, distributed-memory algorithms – which are able to efficiently use tens of thousands of processors – become more and more important.
While there exist linear work distributed-memory suffix array construction algorithm [4] (at least for specialized settings), they have not been implemented in a scaling fashion and the only practical implementations are quasilinear algorithms [3, 2, 1] The challenge of developing fast distributed suffix sorters lies in designing and engineering algorithms that efficiently distribute the workload and minimize communication overheads among the processors.



[1]

	1	2	3	4	5	6	7	8	9	10	11	12	13
T	a	b	a	b	c	a	b	c	a	b	b	a	$
SA	13	12	1	9	6	3	11	2	10	7	4	8	5
	$	a$	abacabba$	abba$	acabba$	abcabba$	ba$	babcabba$	ba$	bacabba$	bcabbacabba$	cabba$	cabbacabba$

Objective

The main objective of this master's thesis is to design, implement and engineer new scalable distributed suffix sorting algorithms building on our previous work on distributed suffix and string sorting [2, 5].

Requirements

- Good C++ and MPI programming skills
- Interest in string and distributed algorithms

[1] HoreKA Supercomputer, Foto: Amadeus Bramsiepe, KIT

References

[1] Timo Bingmann, Simon Gog, and Florian Kurpicz. Scalable construction of text indexes with thrill, 2018.

[2] Johannes Fischer and Florian Kurpicz. Lightweight distributed suffix array construction. In *ALENEX*, pages 27–38. SIAM, 2019.

[3] Patrick Flick and Srinivas Aluru. Parallel distributed memory construction of suffix and longest common prefix arrays. In *SC*, pages 16:1–16:10. ACM, 2015.

[4] Juha Kärkkäinen, Peter Sanders, and Stefan Burkhardt. Linear work suffix array construction. *J. ACM*, 53(6):918–936, 2006.

[5] Florian Kurpicz, Pascal Mehnert, Peter Sanders, and Matthias Schimek. Scalable distributed string sorting. *CoRR*, abs/2404.16517, 2024.

Florian Kurpicz, Matthias Schimek
Informatikgebäude am Fasanengarten, Raum 2{08,21}, {kurpicz, schimek}@kit.edu