algorithms

Editorial: Special Issue on Efficient Data Structures

Jesper Jansson (ORCID)

Department of Computing, The Hong Kong Polytechnic University, Hung Hom, Kowloon 810005, Hong Kong, China; jesper.jansson@polyu.edu.hk

Abstract: This Special Issue of *Algorithms* is focused on the design, formal analysis, implementation, and experimental evaluation of efficient data structures for various computational problems.

1. Introduction

Data structures provide ways of compactly organizing and efficiently retrieving various kinds of information, and are consequently fundamental to computer science. Over the years, data structures have been used effectively in countless practical and conceptual applications. For example, a nineteenth-century data structure known as the *phylogenetic tree*, nowadays routinely used for representing the evolutionary history of a set of biological species, has helped scientists to understand the mechanisms of evolution. As another example, a twenty-first-century data structure known as the *blockchain*, which aims at achieving decentralized consensus, has many potentially important applications involving the creation of permanent ledgers for sharing information over the Internet and automated contracts.

2. Special Issue

To encourage further original research on efficient data structures, we set up a Special Issue of the MDPI journal *Algorithms* devoted to this topic. The call-for-papers invited articles dealing with the design, formal analysis, implementation, and experimental evaluation of efficient data structures for all kinds of computational problems. Of particular interest were algorithms for constructing data structures and extracting information from them efficiently. Articles focusing on complexity aspects of data structures related to time-space tradeoffs, information-theoretic entropy, and lower bounds in various models of computation were also welcomed.

All of the articles submitted to the Special Issue were evaluated by invited experts. In many cases, their detailed comments improved the technical strength and the quality of presentation. After several rounds of revisions and reviewing, six of the submitted articles were accepted for inclusion in the Special Issue. These six articles present new results for a wide variety of data structures with unexpectedly diverse applications: the *link-cut tree* (Russo et al. [1]), the *distributed combinatorial map* (Damiand et al. [2]), the *sliding suffix tree* (Brodnik and Jekovec [3]), the *DenseZDD* (Denzumi et al. [4]), the *dynamic DFS tree* (Nakamura and Sadakane [5]), and the *selectable sloppy heap* (Dumitrescu [6]). As a whole, they represent some of the current trends in the field. We hope that you will enjoy reading them and that you will find them useful in one way or another.

Conflicts of Interest: The author declares no conflict of interest.

References

1. Russo, L.M.S.; Teixeira, A.S.; Francisco, A.P. Linking and Cutting Spanning Trees. *Algorithms* **2018**, *11*, 53. [CrossRef]
2. Damiand, G.; Gonzalez-Lorenzo, A.; Zara, F.; Dupont, F. Distributed Combinatorial Maps for Parallel Mesh Processing. *Algorithms* **2018**, *11*, 105. [CrossRef]
3. Brodnik, A.; Jekovec, M. Sliding Suffix Tree. *Algorithms* **2018**, *11*, 118. [CrossRef]
4. Denzumi, S.; Kawahara, J.; Tsuda, K.; Arimura, H.; Minato, S.; Sadakane, K. DenseZDD: A Compact and Fast Index for Families of Sets. *Algorithms* **2018**, *11*, 128. [CrossRef]
5. Nakamura, K.; Sadakane, K. Space-Efficient Fully Dynamic DFS in Undirected Graphs. *Algorithms* **2019**, *12*, 52. [CrossRef]
6. Dumitrescu, A. A Selectable Sloppy Heap. *Algorithms* **2019**, *12*, 58. [CrossRef]