Bender, Michael A.; Cole, Richard; Raman, Rajeev

Exponential structures for efficient cache-oblivious algorithms. (English) ⟨Zbl 1056.68511⟩

Widmayer, Peter (ed.) et al., Automata, languages and programming. 29th international colloquium, ICALP 2002, Málaga, Spain, July 8–13, 2002. Proceedings. Berlin: Springer (ISBN 3-540-43864-5). Lect. Notes Comput. Sci. 2380, 195-207 (2002).

Summary: We present cache-oblivious data structures based upon exponential structures. These data structures perform well on a hierarchical memory but do not depend on any parameters of the hierarchy, including the block sizes and number of blocks at each level. The problems we consider are searching, partial persistence and planar point location. On a hierarchical memory where data is transferred in blocks of size B, some of the results we achieve are:

We give a linear-space data structure for dynamic searching that supports searches and updates in optimal $O(\log_B N)$ worst-case I/Os, eliminating amortization from the result of Bender, Demaine, and Farach-Colton (FOCS '00). We also consider finger searches and updates and batched searches.

We support partially persistent operations on an ordered set, namely, we allow searches in any previous version of the set and updates to the latest version of the set (an update creates a new version of the set). All operations take an optimal $O(\log_B(m + N))$ amortized I/Os, where N is the size of the version being searched/updated, and m is the number of versions.

We solve the planar point location problem in linear space, taking optimal $O(\log_B N)$ I/Os for point location queries, where N is the number of line segments specifying the partition of the plane. The pre-processing requires $O((N/B)\log_{M/B} N)$ I/Os, where M is the size of the 'inner' memory.

For the entire collection see [Zbl 0993.00041].

MSC:

68P05	Data structures
68P10	Searching and sorting
68T20	Problem solving in the context of artificial intelligence (heuristics, search strategies, etc.)

Cited in **5** Documents

Full Text: Link