Abstract

Caching (also known as *paging*) is a classical problem concerning page replacement policies in two-level memory systems. *General caching* is its variant with pages of different sizes and fault costs. We aim at a better characterization of the computational complexity of general caching in the offline version.

General caching in the offline version was recently shown to be strongly NP-hard, but the proof needed instances of caching with pages larger than half of the cache size. The primary result of this work addresses this problem as we prove: General caching is strongly NP-hard even when page sizes are limited to $\{1, 2, 3\}$. In the structural part of this work, a new simpler proof for the full characterization of work functions by layers for classical caching is given and then extended to caching with variable cache size. We invent two algorithms for restricted instances of general caching building on results around caching with variable cache size.