

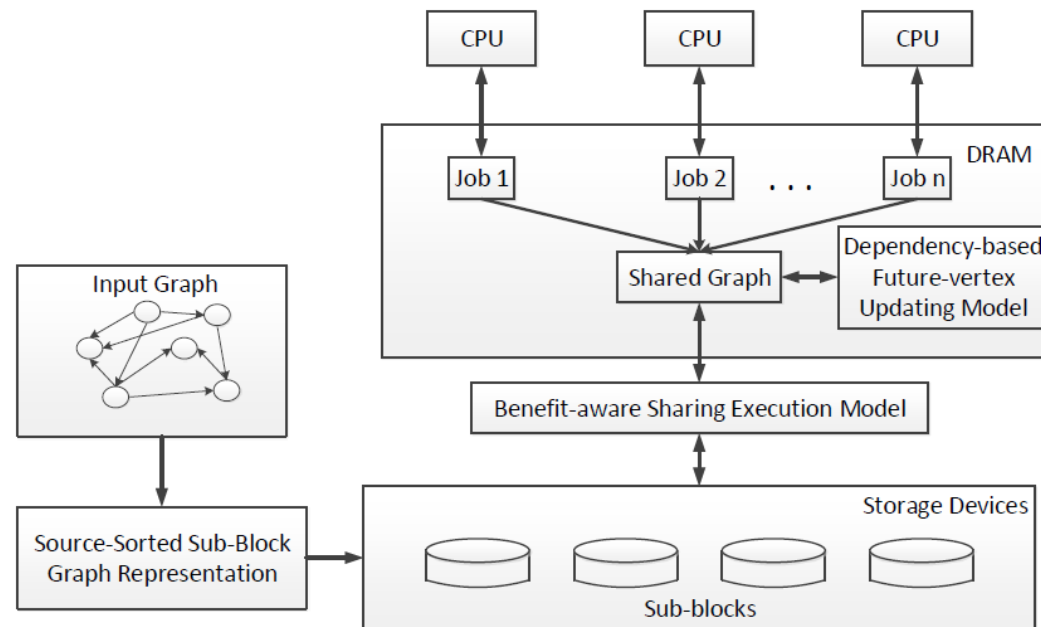
A Disk I/O Optimized System for Concurrent Graph Processing Jobs

**Xianghao XU, Fang WANG, Hong JIANG, Yongli CHENG,
Dan FENG, Peng FANG**

Frontiers of Computer Science, DOI: [10.1007/s11704-023-2361-0](https://doi.org/10.1007/s11704-023-2361-0)

Problems & Ideas

- Problems of existing concurrent graph processing (CGP) systems:
 - Inactive data access.
 - No support for future-value updating.
 - Limited processing capacity and scalability
- Ideas: A disk I/O optimized out-of-core graph processing system that efficiently supports the processing of CGP jobs.



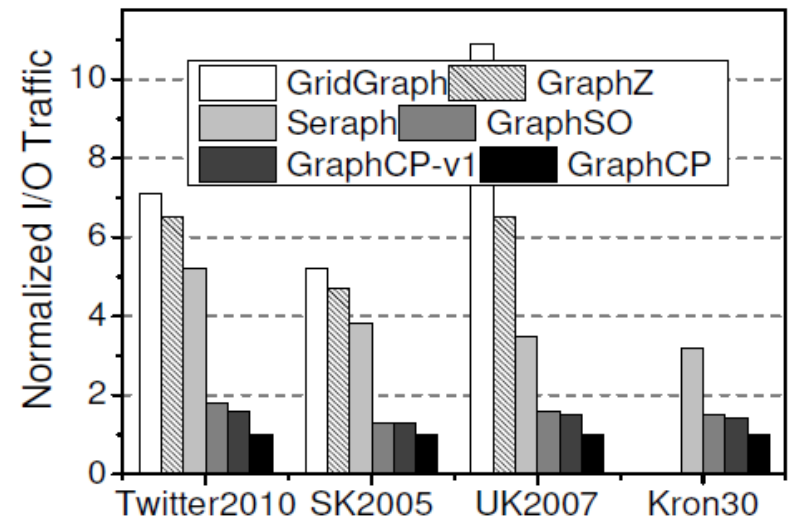
The architecture of our proposed out-of-core graph processing system GraphCP.

Main Contributions

- Contributions:
 - A benefit-aware sharing execution model to share the I/O access and processing of graph data among the CGP jobs and adaptively schedule the graph data loading based on the states of vertices;
 - A dependency-based futurevertex updating model so as to reduce disk I/Os in the future iterations;
 - A Source-Sorted Sub-Block graph representation for better processing capacity and I/O access locality.

	LJ	TT	SK	UK	KR
GridGraph	16.9	3115.4	3807.1	6911.2	-
GraphZ	9.4	1639.6	1586.3	2303.7	-
Seraph	9.6	328.2	768.4	1573.3	130944.5
GraphSO	8.5	197.5	326.1	702.5	62354.5
GraphCP-v1	6.3	205.1	349.3	561.9	56932.4
GraphCP	6.4	128.2	232.9	330.5	40666.2

“-” indicates that the system failed to finish execution in 48 hours.



Performance of GraphCP compared with other systems. Left: the comparisons of graph algorithms' execution time; Right: the comparisons of I/O traffic.