Literature Review

Cheng Liu

2016-10-20

1 Ligra

Ligra [1] is a light-weight graph processing framework targeting shared-memory multi-core processing system. It simplifies graph traversal algorithm implementation on top of a few parallel libraries including cilkplus and openmp. Given a graph traversal algorithm, the user only needs to provide the implementation of a few half-done routines (i.e. Algorithm specific compute routine, F routines) and the framework will handle the rest. On top of the ease of the parallel graph algorithm development, this work contributes on the following aspects:

- It uses vertexSubset to support the frontier based graph traversal.
- It supports both edgeMap and vertexMap to meet requirements of different algorithms.
- It allows both graph traversal with a bottom-up and top-down approach for the sake of performance. Basically, iterations with larger frontier will prefer the bottom-up approach while the rest will adopt the top-down approach.
- Edge/vertex representations including sparse and dense are both supported.

Communication and synchronization relies on the atomic operation (compare-and-swap) over the shared memory.

2 Ligra+

This work [2] is an extension of the Ligra framework. It focuses on the graph compression on top of Ligra.

Inspired by the byte compression used in sparse matrix-vector multiplication, this work proposed a run-length byte encoding method. The basic idea is to encode the edges with variable length instead of the fixed byte block used in previous work. To support the run-length encoding, a head byte is used to specify the lengthy of each encoding block as well as other controlling information. The overall encoding roughly consists of three steps. 1) Edges associated with each vertex are sorted. 2) Difference encoding is applied on the sorted edges. 3) run-length encoding is further used. Due to the variation of edge numbers associated on each vertex, only

verteies with large associated edges are encoded. Then the encoding and decoding are further parallelized.

References

[1] Julian Shun and Guy E. Blelloch. Ligra: A lightweight graph processing framework for shared memory. *SIGPLAN Not.*, 48(8):135–146, February 2013.

[2] Julian Shun, Laxman Dhulipala, and Guy E. Blelloch. Smaller and faster: Parallel processing of compressed graphs with ligra+. In *Proceedings of the 2015 Data Compression Conference*, DCC '15, pages 403–412, Washington, DC, USA, 2015. IEEE Computer Society.