mapalgo ----CENTER FOR MASSIVE DATA ALGORITHMICS

External Memory Breadth First Search

Problem: Given a massive sparse undirected graph, compute its BFS level decomposition

Problem

Internal memory solution: Simple linear time algorithm for computing BFS

RAM model BFS algorithm perform very poor on real architecture



Cause

- RAM model does not capture the cost of I/Os, which becomes the bottleneck on real architecture
- Remembering visited nodes may cause $\Omega(m)$ I/Os
- Unstructured accesses to adjacency lists may cause $\Omega(n)$ I/Os

Solution

Need to design and analyze the BFS algorithm in the I/O model



Mehlhorn and Meyer BFS algorithm - Randomized clustering

Randomly select master nodes (marked *) with uniform probability and expand around them "in parallel"



Key features

- Open source software freely available
- Uses pipelining to reduce constant factors in I/Os
- Can exploit disk parallelism
- Maximizes overlap between I/O and computation
- Based on external memory library STXXL which has the same interface as the standard template library, STL



Deepak Ajwani Max-Planck-Institute for Informatics

- External memory: *D* disks
- Data transfer in blocks of size B (~1M)
- Up to $D \cdot B$ elements transfer in one I/O
- Goal: Minimize total number of I/O steps

BFS algorithm by Munagala and Ranade

Observation

In an undirected graph, the edges from BFS level *t*-1 go only to levels *t*-2, *t*-1 and *t*



Mehlhorn and Meyer BFS algorithm - Deterministic clustering

Key Idea

Cluster the graph and re-arrange the graph layout on the disk to make accesses to adjacency lists structured





Steps involved in deterministic clustering:

- Compute an Euler tour around the spanning tree
- Sort nodes in the order in which they appear in the Euler tour
- Chop the ordered list into blocks of size \sqrt{B} and remove duplicates

Implementation

Applications		Graph class	п	т	MR_BFS	MM_BFS_R	MM_BFS_D
er layer	Streaming layer	Random	228	2 ³⁰	1.4	7x	бх
vector, stack, set lority_queue, map rt, for_each, merge	Pipelined sorting,	Web Graph	$\sim 1.4 \cdot 2^{28}$	$\sim 1.2 \cdot 10^{9}$	2.6	3.5x	2x
		Grid (2 ¹⁴ x 2 ¹⁴)	2 ²⁸	229	2.5x	1.25x	21
ock management layer k, block manager, buffered streams, prefetcher, buffered block writer		Grid (22 ¹ x 2 ⁷)	228	$\sim 2^{29}$	> 100x	> 10x	4.0
		Grid (2 ²⁷ x 2)	2^{28}	$\sim 2^{28} + 2^{27}$	> 500x	>25x	3.8
ronous I/O primitives layer		Simple Line	2 ²⁸	2 ²⁸ - 1	0.4	7x	7x
s, I/O requests, disk queues, completion handlers		Random Line	2 ²⁸	$2^{28} - 1$	>1300x	>75x	3.6
Operating System		Max			~ ½ year	~ 1 week	< 1 day

MADALGO – Center for Massive Data Algorithmics, a Center of the Danish National Research Foundation







$$N(b)\begin{bmatrix} a \\ c \\ d \\ d \\ a \\ b \\ d \end{bmatrix} \begin{bmatrix} a \\ c \\ d \\ d \\ b \\ e \end{bmatrix} \begin{bmatrix} a \\ c \\ d \\ d \\ d \\ d \end{bmatrix} \begin{bmatrix} a \\ d \\ d \\ d \\ d \\ d \end{bmatrix} \begin{bmatrix} a \\ d \\ d \\ d \\ d \\ d \\ d \end{bmatrix}$$

Compute a minimum spanning tree of the input graph