PRESENTATION OUTLINE: Parallel Buffer Trees and Searching

Cory Fraser
School of Computer Science
Carleton University
Ottawa, Canada K1S 5B6
cfraser3@connect.carleton.ca

November 12, 2012

1 Title Page

- Introduce Topic

2 Outline

- Computational Model
- Parallel Buffer Tree
- Implementation
- Results

3 Computational Model

- Sequential Version operated in the External Memory Model
- Parallel version operates in the Parallel External Memory Model
- Goal of these models.

4 Search Data Structures

- Binary Search Trees - Analyzed in the RAM/PRAM model.
- B-trees - Analyzed in the External Memory Model / PEM model
- How does a Parallel Buffer differ from these?

5 What is a Parallel Buffer Tree?

- Offline Data Structure
- Batched Nature reduces I/Os = good for high volume applications
- Motivating examples - Databases, sorting

6 Parallel Buffer Tree Complexity

- For sequences of N insert/delete/find(/range) operations:
- $O(sort_P(N))$ Parallel I/Os without range search
- $O(sort_P(N) + K/PB)$ Parallel I/Os with range search
- Comparison vs Parallel B-tree Complexity.

7 Required Parallel Algorithms

- Parallel sorting for batched operation buckets.
- Parallel merge sort used.
- Parallel Prefix sums for the range query operation and computing buckets.

8 Required Parallel Algorithms

- Merging two sorted lists in $O(\lceil \frac{S_1+S_2}{PB} \rceil) + logP)$ I/Os.
- Distributing a sorted sequence into k buckets in $O(\lceil \frac{N+k}{PB} \rceil + \lceil k/P \rceil + logP)$ I/Os.

9 Implementation Overview

- Free CILK++ extension for the GCC compiler used.
- Parallel merge sort described in class used.
- Did not implement support for range queries.

10 Implementation Details

- Buffer tree is an (a,b)-tree, a= f/4, b=f, $f >= PB$
- Each leaf node stores B elements.
- Each non-leaf has a buffer of size $g = fB$.
- Internal nodes have k-1 routing elements to direct values, k = num of children.

11 Implementation Details - Operations

- Tree builds up batches of PB operations.
- An operation is its type, value and timestamp.
- The PB batched operations are split into P blocks and sent to the root node.

12 Implementation Details - Emptying Buffers (Possibly 2 slides)

- Describe non-fringe buffer emptying process.
- Describe fringe buffer emptying process.

13 Results

- Test system specifications
- Branching factor variation summary.
- Results showing speedup for different input sequences / block sizes
- Results compared to online search structures

14 Conclusion

- Conclusions
- Was the expected performance met?
- Practical for industry use?