August 18-21, 2008
MADALGO Summer School on Cache-Oblivious Algorithms

June 8-10, 2009
25th Annual ACM Symposium on Computational Geometry

www.madalgo.au.dk
"You have to provide reliability on a software level. If you're running 10,000 machines, something is going to die every day."

— Google™ fellow Jeff Dean
A bit in memory changed value because of e.g. background radiation, system heating,...
Binary Search for 16

$O(\log N)$ comparisons
**Requirement**: If the search key occurs in the array as an uncorrupted value, then we should report a match!
Where is Kurt?
Where is Kurt?
Where is Kurt?

If at most 4 faulty answers then Kurt is somewhere here.
Faulty-Memory RAM Model

- Content of memory cells can get corrupted
- Corrupted and uncorrupted content cannot be distinguished
- $O(1)$ safe registers
- **Assumption**: At most $\delta$ corruptions
Faulty-Memory RAM: Searching

Problem?

Low confidence  High confidence

4  7  10  13  14  15  16  18  19  23  8  26  27  29  30  31  32  33  34  36  38

16?
Faulty-Memory RAM: Searching

When are we done ($\delta=3$)?

Contradiction, i.e. at least one fault

If range contains at least $\delta+1$ and $\delta+1$ then there is at least one uncorrupted and, i.e. $x$ must be contained in the range
Faulty-Memory RAM: $\Theta(\log N + \delta)$ Searching

If verification fails

→ contradiction, i.e. $\geq 1$ memory-fault
→ ignore 4 last comparisons
→ backtrack one level of search

Brodal, Fagerberg, Finocchi, Grandoni, Italiano, Jørgensen, Moruz, Mølhave, ESA’07
Faulty-Memory RAM: \( \Theta(\log N + \delta) \) Searching

- Standard binary search + verification steps
- At most \( \delta \) verification steps can fail/backtrack
- **Detail**: Avoid repeated comparison with the same (wrong) element by grouping elements into blocks of size \( O(\delta) \)

Brodal, Fagerberg, Finocchi, Grandoni, Italiano, Jørgensen, Moruz, Mølhave, ESA’07
Faulty-Memory RAM: Reliable Values

- Store $2\delta+1$ copies of value $x$ - at most $\delta$ copies uncorrupted
- $x =$ majority
- Time $O(\delta)$ using two safe registers (candidate and count)

Boyer and Moore ‘91

$\delta=5$

<table>
<thead>
<tr>
<th>Candidate</th>
<th>Count</th>
</tr>
</thead>
<tbody>
<tr>
<td>y y y y x x y x x x y x</td>
<td>1 2 3 2 1 2 1 0 1 0 1</td>
</tr>
</tbody>
</table>
Faulty-Memory RAM: Dynamic Dictionaries

- Packed array
- Reliable pointers and keys
- Updates $O(\delta \cdot \log^2 N)$
- Searches = fault tolerant $O(\log N + \delta)$

- 2-level buckets of size $O(\delta \cdot \log N)$
- Root: Reliable pointers and keys
- Bucket search/update amortized $O(\log N + \delta)$

- Search and update amortized $O(\log N + \delta)$

Itai, Konheim, Rodeh, 1981

Brodal, Fagerberg, F Rocchi, Grandi, Italiano, Jørgensen, Moruz, Mølhave, ESA’07
Fault-Tolerant Results

- Merging, time $\Theta(N+\delta^2)$  
  Finocchi, Grandoni, Italiano, ICALP’06

- Priority queue, time $\Theta(\log N+\delta)$  
  Jørgensen, Moruz, Mølhave, WADS’07

- Sorting, time $\Theta(N\cdot\log N+\delta^2)$  
  Finocchi, Grandoni, Italiano, ICALP’06

- Static and dynamic dictionary, time $\Theta(\log N+\delta)$  
  Brodal, Fagerberg, Finocchi, Grandoni, Italiano, Jørgensen, Moruz, Mølhave, ESA’07
  Finocchi, Grandoni, Italiano, ICALP’06

- External-memory fault tolerant searching, $\Theta\left(\frac{1}{\varepsilon}\log_B N + \frac{\delta}{B^{1-\varepsilon}}\right)$ I/Os  
  Brodal, Jørgensen, Mølhave, Submitted
Allan G. Jørgensen and Gerth S. Brodal at MFCS’07