6.897
L14.2
Apr. 14, 2003

Density(node) = $\frac{\#\text{present elements in interval}}{\#\text{slots in (length of) interval}}$

Density constraint at each node: not too full/empty

- At depth d: density $\geq \frac{1}{2} - \frac{1}{4}\frac{d}{h} \in [1/4, 1/2]$
  density $\leq \frac{3}{4} + \frac{1}{4}\frac{d}{h} \in [3/4, 1]$
  } thresholds (at depth d)
- Allow temporary violation — but fix when seen.

Insert:
- Add element at appropriate location in leaf — $O(\lg n)$
- If leaf is full (outside threshold):
  - Walk up to first ancestor that's inside threshold
  - Rebalance all descendant elements, evenly redistributing among all descendant leaves/slots.

Delete similar ~ just first step different.

Analysis
- Thresholds get weaker as we go deeper
- So after rebalancing a node, densities of its children are far within threshold
  - at least by difference of adjacent thresholds $\sim \frac{1/4}{h} = \Theta\left(\frac{1}{\lg n}\right)$
    [here we need leaves of $\Theta(\lg n)$]
- Before this node is rebalanced again, a child must go outside threshold again, which requires
  $\Omega\left(\frac{\text{capacity}}{\lg n}\right)$ inserts or deletes in the subtree.
  note: capacity(parent) = Θ(capacity(child)).

∴ Amortized cost of inserting/deleting below a node is $\Theta(\lg n)$
- Each node is below $h = \Theta(\lg n)$ ancestors

∴ Amortized # moves = $\Theta(\lg^2 n)$.

TOPS FORM 33041
LITHO IN U. S. A.