Christopher Moh: Bloom Filters [Bloom 1970]
- approximate dictionaries (is it in set?)
- false positives with probability
  \(1 - e^{-nk/m})^k\) where \(k = \# \text{ hash functions},\)
  \(m = \text{size of filter},\) \(n = \# \text{elements}\)
- no false negatives
- set union & intersection = bitwise OR & AND
- two independent hash families suffice
  [Kirsch & Mitzenmacher]

Extensions:
- deletions, multisets

Bloomier filters, [Chazelle, Kilian, Rubinfeld, Tal 2004]
compact approximators [Boldi & Vigna 2002]

associated data with each key
returned data is only overestimate

NEW: conjecture Bloom filter analysis carries over
NEW: modification to upper bound on each bit
& lower bound as well

Applications:
- networks
- databases
- spam detection [Li & Zhong 2006]

[NOTE: SURVEY + THEORY]
Dawson Hwang: Nearest Neighbor for KL Divergence

- (ε,δ) Approximate Nearest Neighbor: find pt. within (1+ε)x of nearest with probability ≥ 1-δ
- KL Diverg: \( D(p∥q) = \sum_{i=1}^{k} p(i) \log \frac{p(i)}{q(i)} \) for two prob. distributions \( p, q \)
  (from info. theory)
- asymmetric, permutation invariance
- nonneg., zero only when \( p=q \)
- \( \infty \) if \( \exists i \) s.t. \( p(i)\neq0 \) & \( q(i)=0 \).
- typically \( p \) is experimental pdf,
  \( q \) is analytic pdf.
- \( \min D(p∥q) = \max \sum_{i=1}^{k} p(i) \log q(i) \)
- goal: upper or lower bound for NN/ANN
  with KL divergence: \( \nu \) find \( q \) to \( \min D(p∥q) \) given \( p \).
- offline & online versions
  \( \nu p \) known \( \nu p \) from experimental samples
- in progress

THEORY
Ivalyo Riskov: Implementing a new in-place sorting alg.
- $\Theta(n \log n)$ comparisons, $\Theta(n)$ moves, $O(1)$ aux. mem.
  - conjectured by Munro & Raman 1992
    where attained in average case
  - attained in Franceschini & Geffert 2003
- $2n \log n$ comparisons, $13n$ moves, but not simple!
- goal: implement, compare to other alg.s.
  on various metrics & data
- in progress, so far understand alg.

EXPERIMENTAL
Hooyeong Chung: Generalized Dynamic Connectivity
- two vertices \( k \)-vertex-connected if removal of \( \leq k \) edges leaves them connected

2-edge-connectivity: \( O(\sqrt{m})/\text{op.} \) [Frederickson 1997]
- \( O(\sqrt{n})/\text{op.} \) [Eppstein 1997] via general sparsification technique to make \( m \approx n \)
- \( O(\log^5 n)/\text{op. expected} \) [Henzinger & King]
- \( O(\log^4 n)/\text{op. det.} \) [Holm et al.]
- \( O(\log^3 n \log \log n) \) [Thorup]

based on TOT trees (like link-cut)
- similar kind of charging scheme as dynamic connectivity DS

(SURVEY)
Kuat Yessenov: Global k-edge-connectivity
- edge conn. = # edges need to remove to disconnect graph
- $\tilde{O}(n^2)$ w.c. update, list min-cut in $O(dg n/\text{edge})$
  [Thorup 2001]
- $\sqrt{2} + o(1)$ approximation factor in polylog/update
  [Thorup & Karger]
- maximal tree packing within factor of 2 of edge connectivity
- can find such a tree packing via greedy alg.
- can maintain using dynamic MST of Holm et al.
- can't hope to be exact
- use Eppstein's sparsification technique
  + TOP trees for MST
  + maintenance of cross edges

SURVEY
Eric Price: Functional Tries

- **motivation**: Subversion etc.
  - file system = rooted tree
  - functional ⇒ confluent persistence
- path copying: easy way (& what Subversion does)
  - copy nodes from root to modified nodes
  - expensive (esp. in space) for unbalanced tree

- **NEW**: use link-cut trees to represent unbal. tree
  - but uses parent ptvs. - not functional
  - use finger = stack of path to node
to point to a node (cdr = parent finger)
  - parent/preferred child in represented tree
  - predecessor/successor in auxiliary tree
- tricky to find those functionally ~
  - make finger a deque of deques! (catenable)

⇒ constant-degree tries in \(O(\text{depth})\) search & \(O(\log n)\) update time & space

- improvements:
  - arbitrary degree
  - \(\min \{\log n, \text{depth}\}\)
- want: "working set property"
  - faster searches than \(\log\) degree
  - multiple fingers

**THEORY**