RECAP (KARGER-STEIN ALGORITHM)

OBSERVATION: FULL-CONTRACTION: early contractions much more valuable than later contractions in preserving min-cut C^*

IDEA: "share" valuable contractions

RECURSIVE STRATEGY

```
R_KS(G=(V,E)) ; n ← |V|
if (n ≤ 8) then * solve directly *
K ← n - ⌈n/√2 + 1⌉  {K ≃ (1 - 1/√2)n ≃ n/3.4}
G1=(V1,E1) ← PC(G, K); t1 ← R_KS(G1) {T1}
G2=(V2,E2) ← PC(G, K); t2 ← R_KS(G2) {T2}
  {two independent trials, T1 and T2}
  {|V1|, |V2| ≃ n/√2 ≃ n/1.41  (> n/2)}
return MIN(t1, t2)
```

$T(n) = O(n^2 \log n)$ $\quad T(n) = 2T\left(\frac{n}{\sqrt{2}}\right) + O(n^2)$

OBSERVATION: $h \in \lceil 2 \log_2 n \rceil + 1$ levels : $2^h = \Theta(n^2)$ leaves! We perform $\Theta(n^2)$ "full" contractions at the price of $\log n$! BUT "FULL" CONTRACTIONS ARE NOT INDEPENDENT ANYMORE (THEY SHARE COMMON SEGMENTS OF PARTIAL CONTRACTIONS)

COLLECTIVE ANALYSIS

Let: $p'(h) = Pr$ (R_KS(G) with h recursion levels is correct)

For $|V|=n$: $h \le 2\lceil \log_2 n \rceil + 1$

We prove: $p'(h) \ge \begin{cases} 1 & h=1 \\ p'(h-1) - \frac{1}{4}(p'(h-1))^2 & h>1 \end{cases}$

LEMMA $p'(h) \ge \frac{1}{h}$

B: $h=1$ ok $1 \ge 1$ ($p(1)=1$!)

HP: $p'(h-1) \ge \frac{1}{h-1}$ $h \ge 2$

$p'(h) \ge p'(h-1) - \frac{1}{4}(p'(h-1))^2 = f(p'(h-1))$:

$f(x) = x - \frac{1}{4}x^2$: $f'(x) = 1 - \frac{x}{2} > 0$ in $[0,1)$

$f(x)$ increasing in $[0,1]$

Since $p'(h-1) \ge \frac{1}{h-1}$ from HP

then

$p'(h) \ge f(p'(h-1)) \overset{HP}{\ge} f\left(\frac{1}{h-1}\right)$

$= \frac{1}{h-1} - \frac{1}{4(h-1)^2} > \frac{1}{h-1} - \frac{1}{2(h-1)^2} =$

$= \frac{1}{h-1} - \frac{1}{2(h-1)(h-1)} \ge \frac{1}{h-1} - \frac{1}{h(h-1)}$

(since $2(n-1) \geq n$ for $n \geq 2$)

$$= \frac{n-1}{n(n-1)} = \frac{1}{n}$$ Q.E.D.

We know that the number of levels of the recursion tree is $h \leq \lceil 2\log_2 n + 1 \rceil$

Therefore

$$p(n) = Pr(\text{R\_KS}(G=(V,E)) \text{ returns } |C^*|, |V|=n) \cdot p'(h) \geq$$
$$\geq \frac{1}{h} \geq \frac{1}{\lceil 2\log_2 n + 1 \rceil} \geq \frac{1}{4\log_2 n}$$

This is *not* high probability!

However we can AMPLIFY this probability via multiple runs of R_KS:

```
KARGER_STEIN(G, s)
  min ← +∞
  repeat s times
    t ← R_KS(G)
    if (t < min) then min ← t
  return min
```

We have that

$$Pr(KS(G,s) \text{ fails returning } |C^*|) \leq$$
$$\leq (1 - p(n))^s \leq \left(1 - \frac{1}{4\log_2 n}\right)^s$$

setting $s = 4 \cdot d \cdot \log_2 n \ln n = \Theta(\log^2 n)$ we obtain:

$$\Pr[\text{KS}(G,s) \text{ fails returning } |e^*|) \leq$$

$$\leq \left[\left(1 - \frac{\Delta}{4\log_2 n}\right)^{4\log_2 n} \right]^{d \ln n} < e^{-1}$$

$$\leq e^{-d \ln n} = \frac{1}{n^d}$$

$\Rightarrow$ $\Theta(\log^2 n)$ executions of RKS(G) guarantee high probability. Therefore:

$$T_{KS}(n) = O(n^2 \log n \cdot \log^2 n) = O(n^2 \log^3 n)$$

$T_{RKS}(n)$ # executions

This is much faster than any MAX-FLOW based algorithm

STATE OF THE ART:

- Deterministic direct algorithms (existing max-flow): $O(m \log^2 n \log\log n)$ (TOTALLY UNPRACTICAL, only theoretical)
- Karger's improvement over KS $O(m \log^3 n)$ w.h.p.
- Many heuristic and approximation algorithms

FURTHER RANDOMIZED TECHNIQUES:

COUPON COLLECTING/RANDOMIZED ENUMERATION

Problem: Assume that an algorithm makes multiple calls to RANDOM(1..n). How many calls are necessary to generate all numbers in $\{1, \ldots, n\}$?

Important primitive:
- How many card packets to complete collection of player cards?
- Collaborative work (e.g. SETI@HOME) volunteers donate CPU time. Computational task divided into units. Each volunteer picks random unit. How many volunteer slots to execute all units?
  (Pros: decentralized, fault-tolerant
  Cons: redundancy)

ANALYSIS IN EXPECTATION

Define z_i = "# calls to RANDOM(1..n) needed to obtain i-th distinct number"

ATTENTION: i-th distinct number means that we have already extracted i-1 distinct ones!

Then, if Z = "# of total calls":

$$Z = \sum_{i=1}^{n} Z_i$$

OBSERVATION: We expect that, on average, the values of the Z_i's increase

e.g., $\Pr(Z_1 = 1) = 1$ (the first call always return a new value)

$\Pr(Z_2 = 1) = 1 - \frac{1}{n}$ (very high)

$\Rightarrow E[Z_2] = 1/(1-1/n) > 1$, etc...

In general: if we have already generated $i-1$ distinct values, then each successive call will return a new value with probability $p_i = \frac{n-(i-1)}{n} = \frac{n-i+1}{n}$

Z_i must then count the number of trials until the first head of a coin-flip with $\Pr(\text{head}) = p_i$

$\Rightarrow$ $Z_i \sim \text{Geom}(p_i)$:

$$\Pr(Z_i = K) = p_i (1-p_i)^{K-1} \qquad K \geq 1$$

We have $E[Z_i] = \frac{1}{p_i}$ (exercise)

$$= \frac{n}{n-i+1}$$

Therefore

$$E[Z] = E\left[\sum_{i=1}^{n} Z_i\right] = \sum_{i=1}^{n} \frac{n}{n-i+1}$$

$$= n \sum_{i=1}^{n} \frac{1}{n-i+1} = \underset{j=n-i+1}{=} n \sum_{j=1}^{n} \frac{1}{j}$$

$$= n H(n) = n\left(\ln n + O(1)\right)$$

$$= n \ln n + O(n)$$

ANALYSIS IN HIGH PROBABILITY

We follow a different approach. Let us fix a given $i \in \{1, \dots, n\}$, and evaluate

$$P_i = Pr\left(\text{"the first } r \text{ calls of RANDOM}(\{1,\dots,n\}) \text{ have not returned } i\text{"}\right) = Pr(E_i)$$

OBSERVATION By symmetry: $P_i = P_j = P(r,n)$ $\forall\, 1 \le i \ne j \le n$

We have

$$P(r,n) = \left(1 - \frac{1}{n}\right)^r = \left[\left(1 - \frac{1}{n}\right)^n\right]^{\frac{r}{n}} < e^{-r/n}$$

($\left(1 - \frac{1}{n}\right)^n < e^{-1}$)

($1 - \frac{1}{n}$: probability of returning a value $\ne i$)

Observe that by setting $r = (d+1)\, n \ln n$

we obtain $P(r,n) < \frac{1}{n^{d+1}}$

To finish the analysis:

$$\Pr(\text{"all numbers obtained after } (d+1)n \text{ calls"})$$
$$= 1 - \Pr(\text{"at least one value is not obtained after } (d+1)n \text{ calls"})$$
$$= 1 - \Pr\left(\bigcup_{i=1}^{n} E_i\right) \geq_{\text{UNION BOUND}} 1 - \sum_{i=1}^{n} P_i = 1 - n \cdot p(r,n)$$
$$> 1 - \frac{1}{n^d}$$

HIGH PROBABILITY!

FURTHER RANDOMIZED TECHNIQUES:

OCCUPANCY PROBLEMS / BALLS INTO BINS

EXPERIMENT: $m \geq n$ identical balls are thrown randomly and independently into n bins (boxes)

AIM: Study events related to the final distribution of the balls into the bins.

INTERESTING PHENOMENA

1) Maximum number of balls into any bin
2) Number of empty bins
3) How large must m be so that each bin contains at least a ball?

SOME APPLICATIONS:

1. Analysis of hash functions under chaining: max number of collisions = max length of the list (max search time)
2. Idle workers under random distribution of load

3. Coupon collecting symmetry:
Balls = random extractions with replacement
Bins = values in the urn

3: We already know that $m = \Theta(n \log n)$ balls suffice to fill all bins w.h.p.
Also, on average, $nH(n) = n\ln n + O(n)$ balls are necessary

1. MAXIMUM NUMBER OF BALLS INTO ANY BIN (after throwing m balls into n bins)

Let X_b = "# balls into bin b", $1 \le b \le n$

Clearly $X_b \sim \text{Binomial}\left(m, \frac{1}{n}\right)$

(sum of m Bernoulli trials)

Y_j^b : $1 \le j \le m$

$\Pr(Y_j^b = 1) = \Pr(j\text{-th ball into bin } b) = \frac{1}{n}$

Thus: $\Pr(X_b = i) = \binom{m}{i} \frac{1}{n^i} \left(1 - \frac{1}{n}\right)^{m-i}$

FACT $\binom{m}{i} \le \left(\frac{em}{i}\right)^i$ $\forall m \ge 1$ $\forall 0 < i \le m$

STIRLING'S APPROXIMATION TO BIN. COEFF.

Let $C_b(k)$ = "bin b has $\ge k$ balls"

Then:

$$\Pr(C_b(k)) = \sum_{i=k}^{m} \binom{m}{i} \left(\frac{1}{m}\right)^i \left(1 - \frac{1}{m}\right)^{m-i}$$

$$\le \sum_{i=k}^{m} \left(\frac{em}{i}\right)^i \left(\frac{1}{m}\right)^i = \sum_{i=k}^{m} \left(\frac{em}{im}\right)^i$$

For $i \ge k$:

$$\left(\frac{em}{im}\right)^i \le \left(\frac{em}{km}\right)^i$$

Thus

$$\Pr(C_b(k)) \le \sum_{i=k}^{m} \left(\frac{em}{km}\right)^i \le \sum_{i=k}^{\infty} \left(\frac{em}{km}\right)^i$$

HP: $\left(k \ge 2e \frac{m}{n}\right)$ $\le \frac{1}{2}$

$$= \left(\frac{em}{km}\right)^k \sum_{j=0}^{\infty} \left(\frac{em}{km}\right)^j =$$

$$= \left(\frac{em}{km}\right)^k \frac{1}{1 - \frac{em}{km}} \le 2\left(\frac{em}{km}\right)^k$$

$\le \frac{1}{2}$

Consider the case $m = n$

Letting $k^* = \lceil (3 \ln n) / \ln \ln n \rceil$ we obtain: ($\geq 2e!$)

$$\Pr(C_b(k^*)) \leq 2\left(\frac{e}{k^*}\right)^{k^*} = 2\left(e^{1-\ln k^*}\right)^{k^*}$$

ATTENTION:

$$\leq 2\left(e^{(\overbrace{1 - \ln 3}^{<0} - \ln\ln n + \ln\ln\ln n)}\right)^{k^*}$$

($\ln k^*$)

HOMEWORK:

$$< 2 \cdot e^{(-\ln\ln n + \ln\ln\ln n)k^*}$$

$$\leq 2 \cdot e^{-3\ln n + 4\ln n \frac{\ln\ln\ln n}{\ln\ln n}} \quad \left(\frac{\ln\ln\ln n}{\ln\ln n} \to 0\right)$$

$$< \frac{1}{8} \quad \forall n$$

MEMO:

$$\leq 2 \cdot e^{-3\ln n + \frac{\ln n}{2}} =$$

$$= 2 \cdot e^{-2.5\ln n} = 2 \cdot \frac{1}{n^{2.5}} \leq \frac{1}{n^2} \quad (n \geq 4)$$

Thus

$$\Pr(\exists \text{ bin with} \geq k^* \text{ balls}) = \Pr\left(\bigcup_{b=1}^{n} C_b(k^*)\right)$$

UNION BOUND:

$$\leq \sum_{b=1}^{n} \Pr(C_b(k^*)) \leq \frac{n}{n^2} = \frac{1}{n}$$

Therefore the maximum number of balls into a bin when $m = n$, is $O\left(\frac{\log n}{\log \log n}\right)$ w.h.p.

COROLLARY

Under a fully random hash function with chaining, the search time for a key is $O\left(\frac{\log n}{\log \log n}\right)$ w.h.p.

(better than any deterministic dictionary data structure)
