Midterm out tuesday.
Collaborations.

**Shortest Paths**
classical shortest paths.

- dijkstra’s algorithm
- floyd’s algorithm. similarity to matrix multiplication

**Matrices**

- length 2 paths by squaring
- matrix multiplication. strassen.
- shortest paths by “funny multiplication.”
  - huge integer implementation
  - base-$(n + 1)$ integers

**Boolean matrix multiplication**

- easy.
- gives objects at distance 2.
- gives $nMM(n)$ algorithm for problem
- what about recursive?
- well can get to within 2: let $T_k$ be boolean “distance less than or equal to $2^k$. Squaring gives $T_{k+1}$.
- $O(\log n)$ squares for unit length
- what about exact?

**Seidel’s distance algorithm for unit lengths.**

- log-size integers:
  - parities suffice:
    - square $G$ to get adjacency $A'$, distance $D'$
      - if $D_{ij}$ even then $D_{ij} = 2D'_{ij}$
      - if $D_{ij}$ odd then $D_{ij} = 2D'_{ij} - 1$
    - For neighbors $i, k$,
      - $D_{ij} - 1 \leq D_{kj} \leq D_{ij} + 1$
exists \( k, D_{kj} = D_{ij} - 1 \)

- Parities
  - If \( D_{ij} \) even, then \( D'_{kj} \geq D'_{ij} \) for every neighbor \( k \)
  - If \( D_{ij} \) odd, then \( D'_{kj} \leq D'_{ij} \) for every neighbor \( k \), and strict for at least one
- Add
  - \( D_{ij} \) even iff \( S_{ij} = \sum_k D'_{kj} \geq D_{ij} d(i) \)
  - \( D_{ij} \) odd iff \( \sum_k D'_{kj} < D_{ij} d(i) \)
  - How determine? find \( S = AD' \)

To find paths: Witness product.

- easy case: unique witness
  - multiply column \( c \) by \( c \).
  - read off witness identity
- reduction to easy case:
  - Suppose \( r \) columns have witness
  - Suppose choose each with prob. \( p \)
  - Prob. exactly 1 witness: \( rp(1 - p)^{r-1} \approx 1/e \)
  - Try all values of \( r \)
  - Wait, too many.
- Approx
  - Suppose \( p = 2/r \)
  - Then prob. exactly 1 is \( \approx 2/e^2 \)
  - So anything in range \( 1/r \ldots 1/2r \) will do.
  - So try \( p \) all powers of 2.
  - suppose \( 2^k \leq r \leq 2^{k+1} \)
  - choose each column with probability \( 2^{-k} \).
  - prob. exactly one witness: \( r \cdot 2^{-k}(1 - 2^{-k})^{r-1} \geq (1/2)(1/e^2) \)
  - so try \( \log n \) distinct powers of 2, each \( O(\log n) \) times
- Mod 3:
  - Recall some neighbor distance down by one
  - so compute distances mod 3.
  - suppose \( D_{ij} = 1 \mod 3 \)
  - then look for \( k \) neighbor of \( i \) such that \( D_{kj} = 0 \mod 3 \)
  - let \( D_{ij}^{(s)} = 1 \iff D_{ij} = s \mod 3 \)
  - than \( AD^{(s)} \) has \( ij = 1 \) iff a neighbor \( k \) of \( i \) has \( D_{kj}^{(s)} \)
  - so, witness matrix mul!
Minimum Cut

- deterministic algorithms
- Min-cut implementation
- data structure for contractions
- alternative view—permutations.
- deterministic leaf algo
- recursion:

\[
\begin{align*}
p_{k+1} &= p_k - \frac{1}{4} p_k^2 \\
q_k &= \frac{4}{p_k} + 1 \\
q_{k+1} &= q_k + 1 + \frac{1}{q_k}
\end{align*}
\]

- cut counting
- Reliability
- Sampling