End of SL vs. L and Amplification of Randomness

Jonathan Kelner
4/26/2011

Today

• Finish SL vs. L
• Talk about expander walks for error reduction

Last Time

• Wanted to show SL=L by giving deterministic algorithm for undirected s-t connectivity
• Defined derandomized squaring operation, which improves connectivity without increasing degree too much
• Logspace Algorithm for UPATH
  ◦ Preprocessing: Show WLOG graph is 4-regular, consistently labeled, and has self-loop at each vertex (so nonbipartite)
  ◦ Perform derandomized squaring $O(\log N)$ times using constant degree expanders
  ◦ Will increase spectral gap by a constant each step until get second eigenval to be constant
  ◦ Want eigenval to be $1/poly(n)$, to guarantee diameter 1
    ◦ If continue as above, can’t get past constant
    ◦ Will use really nonconstant degree graphs at end, but not too many times, so total deg stays poly
  ◦ See paper
• Have to show can simulate all of this in log space
• You can. See paper.

Derandomized Squaring

• Labeling: assign $K$ distinct #s in $[K]$ to outedges of each vertex
• Say consistent if all edges into each vertex have distinct labels
• Let $X$ be labeled $K$-regular graph with vertex set $[N]$
• Let $G$ be labeled $D$-regular graph with vertex set $[K]$
• Define $KD$-outregular graph $X\circ G$ with:
  ◦ Vertex set $[N]$
  ◦ Edges are subset of paths in $X$ of length 2
  ◦ Have edges $v[x][y]$ if $y$ is neighbor of $x$ in $G$
  ◦ I.e., Let $x\in[K]$ be an edge label in $X$,
    ◦ $a\in[D]$ be an edge label in $G$
    ◦ Make edges from $v$ to $v[x][x[a]]$ for all $x,a$
• If $X$ is consistently labeled, will be regular
• If $X,G$ consistently labeled, will be consistently labeled
• Will show that if $G$ is an expander, will improve connectivity almost as much as squaring, but only mult degree by a constant
• If $G$ constant degree, can do $O(\log(N))$ times and still have $poly(N)$ degree
Main Technical Result
- If \( X \) is a consistently labeled \((N,K,\lambda)\)-graph, \( G \) is a \((K,D,\mu)\) graph, then
  \[ X \ominus G \text{ is an } (N,KD,f(\lambda,\mu)) \text{ graph, with} \]
  \[ f(\lambda, \mu) = 1 - (1 - \lambda^2) \cdot (1 - \mu) \]
- Note that:
  - \( f(\lambda, \mu) \leq \lambda + \mu \)
  - \( 1 - (1 - \gamma) \cdot \frac{3}{2} \) when \( \gamma < 1/4 \)
  - When \( \mu \to 0 \) (\( G \) really good expander), \( f(\lambda, \mu) \to \lambda^2 \)
- Proof will use:
  - Lemma: Let \( A = \text{walk matrix of } (N,D,\lambda) \) graph, \( J_N = N \times N \) all-ones matrix. Then
    \[ A = (1 - \lambda)J_N + \mu C, \quad ||C|| = \max_{v \in \mathbb{R}^n} \frac{||Cv||}{||v||} \]
  - Intuitively, means random step like going to unif. distr. with prob \((1 - \lambda)\), not getting further from unif with prob \(\lambda\)
  - Not exactly, b/c \( C \) not necessarily stochastic, but right idea
  - Proved last time
- See board for proof of main result from lemma

Error Reduction with Expander Walks
- Have previously seen a few ways to improve error probability of randomized algorithms
  - Repetition, pairwise independent repetition, deterministic error reduction
  - Say initial alg needs \( r \) random bits, want to reduce error from \( 1/3 \) to \( 2^{-t} \). Our techniques gave:

<table>
<thead>
<tr>
<th></th>
<th># Repetitions</th>
<th># Random bits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Repetition</td>
<td>( O(t) )</td>
<td>( O(tr) )</td>
</tr>
<tr>
<td>Pairwise independent repetition</td>
<td>( O(2^t) )</td>
<td>( O(t+r) )</td>
</tr>
<tr>
<td>Deterministic error reduction</td>
<td>( O(2^t) )</td>
<td>( r )</td>
</tr>
<tr>
<td>Today</td>
<td>( O(t) )</td>
<td>( r+O(t) )</td>
</tr>
</tbody>
</table>

Error Reduction with Expander Walks
- Will do for RP. BPP a little more work, same idea
- Implicitly construct constant degree expander \( G \) with \( N = 2^t \) vertices, spectral gap \( \epsilon \)
  - Verts. correspond to random bits for alg, in \( \{0,1\}^t \)
  - Choose a random vertex using \( r \) random bits
  - Take random walk of length \( k \) using \( O(k) \) random bits
    - Let verts be \( u_1, \ldots, u_t \)
    - Run alg with randomness corresponding to \( u_1, \ldots, u_t \)
      accept if ever accept
  - The fact that this gives error red follows from:
    - **Claim:** For \( G \) and \( u_i \) as above, let \( B \subseteq V \) be any set with \( |B| \leq (1 - \delta)n \). Then
      \[ \Pr [u_i \in B \forall i \in \{1, \ldots, t\}] \leq \sqrt{1 - \delta} (1 - \epsilon \delta)^{\frac{t}{2}} \leq (1 - \epsilon \delta)^{\frac{t}{2}} \]
  - See board for proof