Google's PageRank, multilevel solvers, and ‘Gene's ranking’

Hans De Sterck
Department of Applied Mathematics
University of Waterloo

Waterloo Numerical Analysis Symposium
in commemoration of Gene Golub, 29 Feb 2008
ranking pages in web search...

- **Google search:**
  - keyword-based query results in list of web pages
  - pages are listed in order of ‘importance’: PageRank

- how does PageRank work?

- Markov chains
ranking pages in web search...

- how does PageRank work?

- “a page has high rank if the sum of the ranks of its backlinks is high”

(‘The PageRank Citation Ranking: Bringing Order to the Web’ (1998), Page, Brin, Motwani, Winograd)
ranking pages in web search...

- “a page has high rank if the sum of the ranks of its backlinks is high”
  (‘The PageRank Citation Ranking: Bringing Order to the Web’ (1998), Page, Brin, Motwani, Winograd)

- \[ B x = x \]

- PageRank = stationary vector of Markov chain

- ‘random surfer’, random walk on graph
stationary vector of Markov chain

\[ B \mathbf{x} = \mathbf{x} \quad \|\mathbf{x}\|_1 = 1 \]

- \( B \) is column-stochastic

\[ b_{i,j} \geq 0, \quad \sum_i b_{i,j} = 1 \quad \forall j \]

- if \( B \) is irreducible (every state can be reached from every other state in the directed graph)

\[ \exists! \mathbf{x} : B \mathbf{x} = \mathbf{x} \quad \text{and} \quad \|\mathbf{x}\|_1 = 1, \quad x_i > 0 \quad \forall i \]

(no probability sinks!)
Markov chains

\[ B \mathbf{x} = \mathbf{x} \quad ||\mathbf{x}||_1 = 1 \]

- largest eigenvalue of \( B \): \( \lambda_1 = 1 \)

- power method: \( x_{i+1} = B x_i \)
  
  - convergence factor: \( |\lambda_2| \)
  
  - convergence is very slow when \( |\lambda_2| \approx 1 \) (slow mixing)
Markov chains

\[ B \mathbf{x} = \mathbf{x} \quad \|\mathbf{x}\|_1 = 1 \]

• largest eigenvalue of \( B \): \( \lambda_1 = 1 \)

• power method: \( x_{i+1} = Bx_i \)
  
  – convergence factor: \( |\lambda_2| \)

  – convergence is very slow when \( |\lambda_2| \approx 1 \)

(slow mixing)
web matrix regularization

- PageRank (used by Google):

\[(\alpha = 0.15)\]
convergence of power method for PageRank

- convergence factor $|\lambda_2|$ independent of problem size
- convergence is fast, linear in the number of web pages (15+ billion!)
- model is accurate (hence Google’s $150B$ market cap...)

$|\lambda_2| = 1 - \alpha = 0.85$

$(0.85)^{50} \approx 0.0003$
Gene Golub and PageRank

• ‘Extrapolation Methods for Accelerating PageRank Computations’ (2003), Kamvar, Haveliwala, Manning, Golub

• quadratic extrapolation: subtract estimates of $\tilde{u}_2$ and $\tilde{u}_3$ from current estimate

$$\tilde{x}^{(k)} = \tilde{u}_1 + \alpha_2 \tilde{u}_2 + \alpha_3 \tilde{u}_3$$
Gene Golub and PageRank

- ‘Quadratic extrapolation’
- Google liked the idea!
- all authors received Google stock options
- >$500,000 after IPO
- Gene Golub donated his part to ‘Paul and Cindy Saylor Chair’ at UIUC, 2005 (his alma mater)
- Google’s PageRank is also a bit ‘Gene’s ranking’
but how about slowly mixing Markov chains?

- slow mixing: $|\lambda_2| \approx 1$

- one-level methods (Power, Jacobi, Gauss Seidel, ...)
  are way too slow

- need multi-level methods! (multigrid)

- applications: Google $\alpha \approx 0$, many other applications
but how about slowly mixing Markov chains?

• my own research with Tom Manteuffel, Steve McCormick, John Ruge, Quoc Nguyen, Jamie Pearson

• we want numerical methods that have computational complexity linear in the number of unknowns (O(n))
  (do not appear to exist yet for slowly mixing Markov chains!)

• use algebraic multigrid methods

• slow mixing: \[ |\lambda_2| \approx 1 \]
aggregation for Markov chains

- form three coarse, aggregated states

\[ B \mathbf{x} = \mathbf{x} \]

\[
B = \begin{bmatrix}
0 & 1/3 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 \\
1/2 & 1/3 & 0 & 0 & 1 \\
0 & 1/3 & 1 & 0 & 0 \\
1/2 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\]

\[
B_c \mathbf{x}_c = \mathbf{x}_c
\]

\[
B_c = \begin{bmatrix}
1/4 & 3/5 & 0 \\
5/8 & 2/5 & 1 \\
1/8 & 0 & 0 \\
\end{bmatrix}
\]

\[
x_{c,I} = \sum_{i \in I} x_i
\]

\[
b_{c,IJ} = \frac{\sum_{j \in J} x_j \left( \sum_{i \in I} b_{ij} \right)}{\sum_{j \in J} x_j}
\]

hdesterck@uwaterloo.ca
aggregation in matrix form

\[ B_c x_c = x_c \]

\[ b_{c,IJ} = \frac{\sum_{j \in J} x_j \left( \sum_{i \in I} b_{i,j} \right)}{\sum_{j \in J} x_j} \]

\[ B_c = Q^T B \text{ diag}(x) Q \text{ diag}(Q^T x)^{-1} \]

\[ x_{c,I} = \sum_{i \in I} x_i \]

\[ x_c = Q^T x \]

(Krieger, Horton, ... 1990s)
principle of multigrid (for PDEs)

\[-u_{xx} - u_{yy} = f(x, y) \quad Ax = b\]

- high-frequency error is removed by relaxation (weighted Jacobi, Gauss-Seidel, ...)
- low-frequency-error needs to be removed by coarse-grid correction
multigrid hierarchy: V-cycle

- multigrid V-cycle:
  - relax (=smooth) on successively coarser grids
  - transfer error using restriction ($P^T$) and interpolation ($P$)
- $W=O(n)$
choosing aggregates based on strength (SIAM J. Scientific Computing, 2008)

- error equation: \((I - B) \text{diag}(x_i) e_i = 0\)

- use strength of connection in \((I - B) \text{diag}(x_i)\)

- define row-based strength (determine all states that strongly influence a row’s state)

- state that has largest value in \(x_i\) is seed point for new aggregate, and all unassigned states influenced by it join its aggregate

- repeat
  (similar to AMG: Brandt, McCormick and Ruge, 1983)
we need ‘smoothed aggregation’...

(Vanek, Mandel, and Brezina, Computing, 1996)

after relaxation:

coarse grid correction with $Q$:

coarse grid correction with $Q_s$:
non-smoothed aggregation

- non-smoothed method:
  \[(I - B)x = 0 \quad Ax = 0\]

  \[R = Q^T \quad P = \text{diag}(x_i)Q\]

  \[A_c = RAP \text{diag}(Q^T x_i)^{-1} \quad A_c = I_c - B_c\]

\[A_c\] is an irreducible singular M-matrix
smoothed aggregation

non-smoothed method:

\[ A_c = RAP \text{diag}(Q^T x_i)^{-1} \quad A_c = I_c - B_c \]

smooth \( P \) and \( R \) with weighted Jacobi

\[
A = D - L - U \\
P_s = (1 - w) P + wD^{-1}(L + U)P \\
R_s = P_s^T(\text{diag}(1_c^T P_s^T))^{-1}
\]

\[
A_c = R_s A P_s (\text{diag}(P_s^T 1))^{-1} = (R_s D P_s \\
- R_s (L + U) P_s) (\text{diag}(P_s^T 1))^{-1} \neq I_c - B_c
\]

problem: \( A_c \) is not an irreducible singular M-matrix!...
lumped smoothed aggregation

- one solution: partially lump the ‘mass matrix’
  \[ A_c = \text{Lump}(R_s D P_s) - R_s (L + U) P_s \]
  
  \( A_c \) is an irreducible singular M-matrix!
  (off-diagonal elements remain positive)

- **theorem** \( A_c \) is a singular M-matrix on all coarse levels, and thus allows for a unique strictly positive solution \( x_c \) on all levels

- **theorem** Exact solution \( x \) is a fixed point of the multigrid cycle
  (SIAM J. Scientific Computing, submitted)
The performance of smoothed aggregation is illustrated through two tables.

**Table 5.1**

Uniform chain. G-AM with V-cycles and size-three aggregates. (No smoothing.)

<table>
<thead>
<tr>
<th>$n$</th>
<th>$\gamma_{res}$</th>
<th>iter</th>
<th>$C_{op}$</th>
<th>levels</th>
</tr>
</thead>
<tbody>
<tr>
<td>27</td>
<td>0.64</td>
<td>34</td>
<td>1.32</td>
<td>2</td>
</tr>
<tr>
<td>81</td>
<td>0.88</td>
<td>92</td>
<td>1.43</td>
<td>3</td>
</tr>
<tr>
<td>243</td>
<td>0.95</td>
<td>&gt; 100</td>
<td>1.47</td>
<td>4</td>
</tr>
<tr>
<td>729</td>
<td>0.97</td>
<td>&gt; 100</td>
<td>1.49</td>
<td>5</td>
</tr>
</tbody>
</table>

**Table 5.2**

Uniform chain. G-SAM with V-cycles and size-three aggregates. (Smoothing with lumping.)

<table>
<thead>
<tr>
<th>$n$</th>
<th>$\gamma_{res}$</th>
<th>iter</th>
<th>$C_{op}$</th>
<th>levels</th>
</tr>
</thead>
<tbody>
<tr>
<td>27</td>
<td>0.24</td>
<td>12</td>
<td>1.32</td>
<td>2</td>
</tr>
<tr>
<td>81</td>
<td>0.26</td>
<td>12</td>
<td>1.43</td>
<td>3</td>
</tr>
<tr>
<td>243</td>
<td>0.26</td>
<td>12</td>
<td>1.47</td>
<td>4</td>
</tr>
<tr>
<td>729</td>
<td>0.26</td>
<td>12</td>
<td>1.49</td>
<td>5</td>
</tr>
<tr>
<td>2187</td>
<td>0.26</td>
<td>12</td>
<td>1.50</td>
<td>6</td>
</tr>
<tr>
<td>6561</td>
<td>0.26</td>
<td>12</td>
<td>1.50</td>
<td>7</td>
</tr>
</tbody>
</table>
performance of smoothed aggregation

**Fig. 5.6.** Tandem queueing network.

**Fig. 5.7.** Graph for tandem queueing network.
performance of smoothed aggregation

<table>
<thead>
<tr>
<th>n</th>
<th>$\gamma_{res}$</th>
<th>iter</th>
<th>$C_{op}$</th>
<th>levels</th>
</tr>
</thead>
<tbody>
<tr>
<td>121</td>
<td>0.90</td>
<td>&gt;100</td>
<td>1.20</td>
<td>3</td>
</tr>
<tr>
<td>256</td>
<td>0.91</td>
<td>&gt;100</td>
<td>1.19</td>
<td>3</td>
</tr>
<tr>
<td>676</td>
<td>0.90</td>
<td>&gt;100</td>
<td>1.18</td>
<td>3</td>
</tr>
<tr>
<td>1681</td>
<td>0.95</td>
<td>&gt;100</td>
<td>1.19</td>
<td>4</td>
</tr>
</tbody>
</table>

**Table 5.13**
Tandem queueing network. G-AM with V-cycles and three-by-three aggregates. (No smoothing.)

<table>
<thead>
<tr>
<th>n</th>
<th>$\gamma_{res}$</th>
<th>iter</th>
<th>$C_{op}$</th>
<th>levels</th>
<th>$\gamma_{res}$</th>
<th>iter</th>
<th>$C_{op}$</th>
<th>levels</th>
</tr>
</thead>
<tbody>
<tr>
<td>121</td>
<td>0.51</td>
<td>22</td>
<td>1.26</td>
<td>3</td>
<td>0.32</td>
<td>15</td>
<td>1.60</td>
<td>3</td>
</tr>
<tr>
<td>256</td>
<td>0.50</td>
<td>22</td>
<td>1.28</td>
<td>3</td>
<td>0.34</td>
<td>15</td>
<td>1.60</td>
<td>3</td>
</tr>
<tr>
<td>676</td>
<td>0.51</td>
<td>23</td>
<td>1.26</td>
<td>3</td>
<td>0.33</td>
<td>15</td>
<td>1.56</td>
<td>3</td>
</tr>
<tr>
<td>1681</td>
<td>0.55</td>
<td>28</td>
<td>1.27</td>
<td>4</td>
<td>0.33</td>
<td>15</td>
<td>1.61</td>
<td>4</td>
</tr>
<tr>
<td>3721</td>
<td>0.60</td>
<td>30</td>
<td>1.28</td>
<td>4</td>
<td>0.33</td>
<td>15</td>
<td>1.63</td>
<td>4</td>
</tr>
</tbody>
</table>

**Table 5.14**
Tandem queueing network. G-SAM with V-cycles (left) and W-cycles (right), using three-by-three aggregates. (Smoothing with lumping.)
conclusions

• multilevel method applied to Google’s PageRank with $\alpha=0.15$ is not faster than power method (SIAM J. Scientific Computing, 2008)

• smoothed multilevel method applied to slowly mixing Markov chains is much faster than power method (in fact, close to optimal $O(n)$ complexity!) (SIAM J. Scientific Computing, submitted)
conclusions

• multilevel method applied to Google’s PageRank with $\alpha=0.15$ is not faster than power method (SIAM J. Scientific Computing, 2008)

• smoothed multilevel method applied to slowly mixing Markov chains is much faster than power method (in fact, close to optimal $O(n)$ complexity!) (SIAM J. Scientific Computing, submitted)

• ‘Gene’s ranking’ (PageRank and power method with quadratic extrapolation, $\alpha=0.15$) is hard to beat

• Gene is ranking pretty high up there ;-)

hdesterck@uwaterloo.ca
PageRank web matrix regularization as a function of coupling factor $\alpha$