Big Data vs. Wireless

Exabytes per month

<table>
<thead>
<tr>
<th>Year</th>
<th>Exabytes</th>
</tr>
</thead>
<tbody>
<tr>
<td>2014</td>
<td>2.5 EB</td>
</tr>
<tr>
<td>2015</td>
<td>4.2 EB</td>
</tr>
<tr>
<td>2016</td>
<td>6.8 EB</td>
</tr>
<tr>
<td>2017</td>
<td>10.7 EB</td>
</tr>
<tr>
<td>2018</td>
<td>16.1 EB</td>
</tr>
<tr>
<td>2019</td>
<td>24.3 EB</td>
</tr>
</tbody>
</table>

57% CAGR 2014–2019

[Cisco]

\[ \infty \equiv 5 \text{ GB (mod at&t)} \]

ATT Free Msg: Your data usage on your 4G LTE smartphn is near 5GB this month. Exceeding 5GB during this or future billing cycles will result in reduced data speeds, though you will still be able to email & surf the web. Wi-Fi helps you avoid reduced speeds. Visit www.att.com/datainfo or call 866-344-7584 for more info.
Meanwhile, Storage is Getting Cheaper

Storage cost per GB (USD)

http://www.mkomo.com/cost-per-gigabyte-update
Storage = Caching

- Index coding: [Birk & Kol ’98] + …
- Femto-caching: [Golrezai et al. ’12]…
- ….

Content is cached (stored) on mobile devices during off-peak hours
Index Coding Example

- Content of the cache is given
- Cached data independent of a user’s preferences still help

Birk & Kol, “Informed-source coding-on-demand (ISCOD) over broadcast channels,” INFOCOM’98
Index Coding & Coloring

Side info graph $G_d$

Clique cover of $G$

$X_1 + X_2$

$X_3$

$X_4$

Chromatic nbr of $\overline{G}$

$X_1 + X_2 + X_3 + X_4$

$X_5$
Index Coding & Graph Coloring

- $L^*_\text{min}$ min length of linear index code
- Finding $L^*_\text{min}$ is NP hard by [R., Sprintson, Chaudhry ITW’07]

\[ \alpha(G_d) \leq c(G_d) \leq L^*_{\text{min}} \leq \chi_f(\bar{G}) \leq \chi(\bar{G}) \]

- Shannon capacity [Haemers ‘79]
- Fractional Chromatic nbr [Blasiak, Kleinberg, Lubetzky ‘11]

- More bounds [Dimakis et al.] [Arbobjalfoei & Kim], [Mazumdar et al.] etc…
Index Coding on Erdős-Rényi Graphs

Independence nbr \quad Chromatic nbr
\alpha(G) \leq L^*_\text{min} \leq \chi(\bar{G})

- When \( n \rightarrow \infty \), we have with prob 1
  \[
  \log n \leq L^*_\text{min} \leq \frac{n}{\log n}
  \]
- Can improve the lower bound [Haviv & Langberg “Index Coding on random graphs”, ISIT’12]
  \[
  c\sqrt{n} \leq L^*_\text{min} \leq \frac{n}{\log n}
  \]
- Recent results closes the gap \( L^*_\text{min} = \Theta(n/\log n) \)

Talk Roadmap

Graph Theory & Index Coding

Network Coding & Index Coding

Rank Minimization & Index Coding

Privacy Problems
Index Coding & Rank Minimization

- Linear case: \( L^*_{min} = \min rk(M) \) [Bar-Yossef et al. '06]
- Min rank introduced by Haemers in 79 to upper bound the Shannon graph capacity
- Min rank can be a tighter bound on Shannon capacity then Lovász Theta function.
Use Matrix Completion Methods to Construct Index Codes

- Minimizing nuclear norm [Recht & Candes ‘09] does not work here because the index coding matrices have a special structure.
- Try other rank minimization methods [Fazel et al. ‘04]

**Theorem:** [Alternating Projections (AP)]

If C and D are convex, then an alternating projection sequence between these 2 regions converges to a point in their intersection.

Two problems:
1) Regions not convex
2) Optimization over the reals
- Up to 13% savings over Greedy coloring. No theoretical guarantees.
- Recent work on min rank over finite field [Sauderson, Fazel, Hassibi ISIT’16]
- Index coding via LP [Blasiak et al. ‘10], via SDP [Chlamtac et al ’14]...
Performance with Increasing Number of Users

- APIIndexCoding
- LDG
- Greedy Col.

$p=0.2$
$p=0.4$
$p=0.6$
$p=0.8$

$n$
Talk Roadmap

Graph Theory & Index Coding

Network Coding & Index Coding

Rank Minimization & Index Coding

Privacy Problems
Equivalence to Network Coding

An index code of length L that satisfies all the users

A network code that satisfies all the terminals
Index coding is equivalent to the general network coding. If you can solve index coding efficiently you can solve any general network coding problem efficiently.

**Theorem:** [R, Sprintson, Georghiades’08] [Effros, R, Langberg ISIT’13]

For any network coding problem, one can construct an index coding problem and an integer \( L \) such that given any linear network code, one can efficiently construct a linear index code of length \( L \), and vice versa. (same block length, same error probability).
The linear case first

Butterfly network

All terminals in the index coding problem can decode

Any linear network code gives an index code of length $L=7$
Implications on Index Coding

Linear index codes are not optimal

Vector linear codes outperform scalar linear

No linear network code but a non-linear code over alphabet of size 4 [Zeger et al. '06]

Only vector linear codes exist when block length is even.
Connections to many problems

• Interference management: [Jafar et al. ‘12]
• Distributed storage & caching: [Mazumdar ’14], [Shanmugam et al. ’14]
• Matroid representations: [Rouayheb et al. ’09]
• Graph coloring: [Fragouli, Soljanin, Shokrollahi ‘04] [Alon et al. ‘08], [Shanmugam & Dimakis ’13]
• LP bounds: [Blasiak, Kleinberg, Lubetzky ‘11]
• Coded caching [Maddah-Ali & Niesen ’13] +…
1. Data Exchange Problem
[R., Sprintson, Sadeghi ITW’10]
[Milosavlevijc, Pawar, R., Ramchandran, ‘13]
[Courtade et al. ’13]…
• No Base station (D2D).
• Users wants missed parts

2. Pliable index coding [Fragouli et al ‘15]
• Like index coding but users want anything they don’t have

• Cached content is not fixed and can be designed
• Best paper, lots of follow up work…
Talk Roadmap

Graph Theory & Index Coding

Network Coding & Index Coding

Rank Minimization & Index Coding

Privacy Problems
Caching for Private Information Retrieval (PIR)

- PIR: user wants to hide which file it wants [chor et al’95]
- One server: User need to download all the data
- “Classical” PIR: data replicated on many servers
- Recent work: coded PIR [Jafar et al.], [Vardy et al.], [Rouayheb et al.], [Ulukus et al.], [Hollanti et al.]…
- Caching for PIR: user does not reveal cached data

**Theorem 1.** For the $W$-PIR-SI problem with $N = 1$ database, $K$ messages, and side information size $M$, when the demand $W$ and the side information set $S$ are jointly distributed according to (3), the capacity is

$$C_W = \left[ \frac{K}{M+1} \right]^{-1}.$$  \hspace{1cm} (9)

Kadhe, Garcia, Heidarzadeh, R., Sprintson, “PIR with Side Information”, Allerton ’17
**Secure Cooperative Computing in IoT**

- Local computations on untrusted workers
- Homomorphic Encryption very costly
- New codes for security

R. Bitar P. Parag, R., “Minimizing Latency for Secure Distributed Computing”, submitted to ISIT’17
Collaborators

- My students: Rawad Bitar, Razan Tajeddine, Peiwen Tian
- Camilla Hollanti (Aalto University, Finland)
- Olgica Milenkovic (UIUC)
- Hulya Seferoglu, (UIC)
- Parimal Parag (IISC, India)

Funding agencies:

- NSF: CCF-15-26875
- NSF: CCF-16-52867
- ARL: W911NF-17-1-0032
QUESTIONS?
All terminals in the index coding problem can decode

Any linear network code gives an index code of length $L=7$
Given a linear index code

- $Y_{e1} + Y_{e2}$
- $Y_{e3}$
- $Y_{e4}$
- $Y_{e5} + Y_{e4}$
- $Y_{e6} + Y_{e7}$

Can always diagonalize

- $Y_{e1} + X_1$
- $Y_{e2} + X_1$
- $Y_{e3} + X_2$
- $Y_{e4} + X_2$
- $Y_{e5} + X_1 + X_2$
- $Y_{e6} + X_1 + X_2$
- $Y_{e7} + X_1 + X_2$

Butterfly network

- Any linear index code of length $L=7$ can be mapped to a linear network code
- Works for scalar linear and vector linear
Non-Linear Network Code ➔ Index Code

Butterfly network

\[ f_{e_i}(X_1, X_2) : \text{message on edge } e_i \]
"Diagonalization" May Not Work for Non-Linear

Given a non-linear index code

If we can we diagonalize?
Non-linear Index Code $\Rightarrow$ Network Code

- Destinations can decode with no errors:
- Recall that $B = f(X_1, X_2, Y_{e1}, \ldots, Y_{e7})$
- For a fixed $B$ and given values of $X_1$ and $X_2$, there is a unique possible vector $(Y_{e1}, \ldots, Y_{e7})$
- Otherwise, $U^*$ cannot decode correctly

\[
X_1 = D_{U_{t1}} (B, Y_{e4}, Y_{e7})
\]
\[
Y_{e4} = D_{U_{e4}} (B, X_2)
\]
\[
Y_{e7} = D_{U_{e7}} (B, Y_{e5})
\]

Fix a value for $B$, say $B=0$
Dealing with Errors

- Consider an index code where decoding errors only happen when the broadcast message $B=0$
- $\varepsilon$: Probability of error in the index code $=1/2^c=1/2^7=0.0078$
- Probability of error in the network code $=1$ (bad).

**Claim:** There exists $\sigma$, such that for $B=\sigma$, in the previous construction, the network code will have a probability of error at most $\varepsilon$ ($\varepsilon=$ error prob of the index code).

- Intuition: if for every value of $B$, the resulting network code will have a probability of error $>\varepsilon$, this implies that the probability of error in the index code $>\varepsilon$. A contradiction.

Each $\checkmark$ corresponds to a different “good” value of $(X, Y_e)$

- $X=(X_1, X_2)$
- $Y_e=D_{U^*}(B, X_1, X_2)$

Total # of $\checkmark$ $< (1-\varepsilon)|\Sigma_B|\cdot|\Sigma_X|$

But $|\Sigma_B|=|\Sigma_{Y_e}|$

$\Rightarrow$ Total # of “good” values $< (1-\varepsilon)|\Sigma_{Y_e}|\cdot|\Sigma_X|$

contradiction
Capacity Regions

If there is a code that achieves $P$ “exactly”, then $P'$ is in $\mathcal{R}_I \cap \mathcal{H}$, and vice versa.

What if a sequence of points (not necessarily in $\mathcal{H}$) converges to $P$. Does this mean that $P$ is in $\mathcal{R}_N$?

If true this will solve a long-standing open problem: Is zero-error capacity= $\epsilon$-error capacity of networks?

True for index coding problems [Langberg, Effros ‘11]
The Case of Co-located Sources

\[ R_{X_1}, R_{X_2} \]: Capacity region of a network

\[ R_N \]: Capacity region of the equivalent index code

\[ R_B = 7 \]

\[ R_B \]: Capacity region of the equivalent index code

\[ \mathcal{H} \]: Capacity region of the equivalent index code

**Theorem**: For any network \( \mathcal{N} \) with co-located sources one can efficiently construct an index coding problem \( \mathcal{I} \) and an integer \( L \) such that \( \mathbf{R} \) is in the capacity region of \( \mathcal{N} \) iff \( \mathbf{R}' \) is in the capacity region of \( \mathcal{I} \) with broadcast length \( L \).