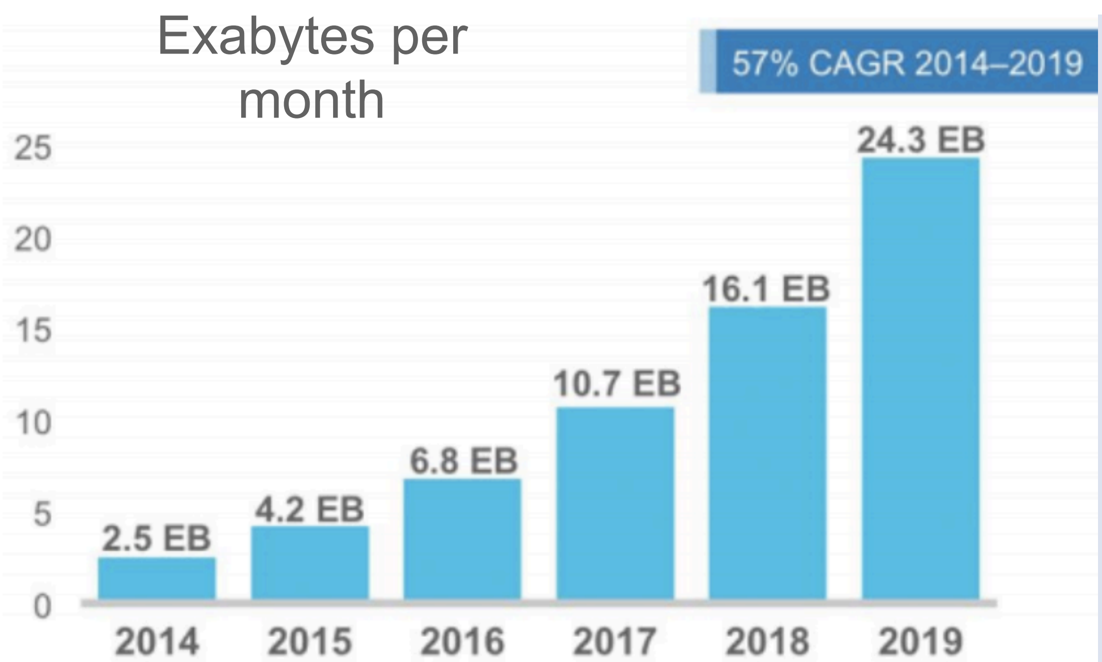
Index Coding & Caching in Wireless Networks

Salim El Rouayheb

ECE IIT, Chicago

Huawei University Days

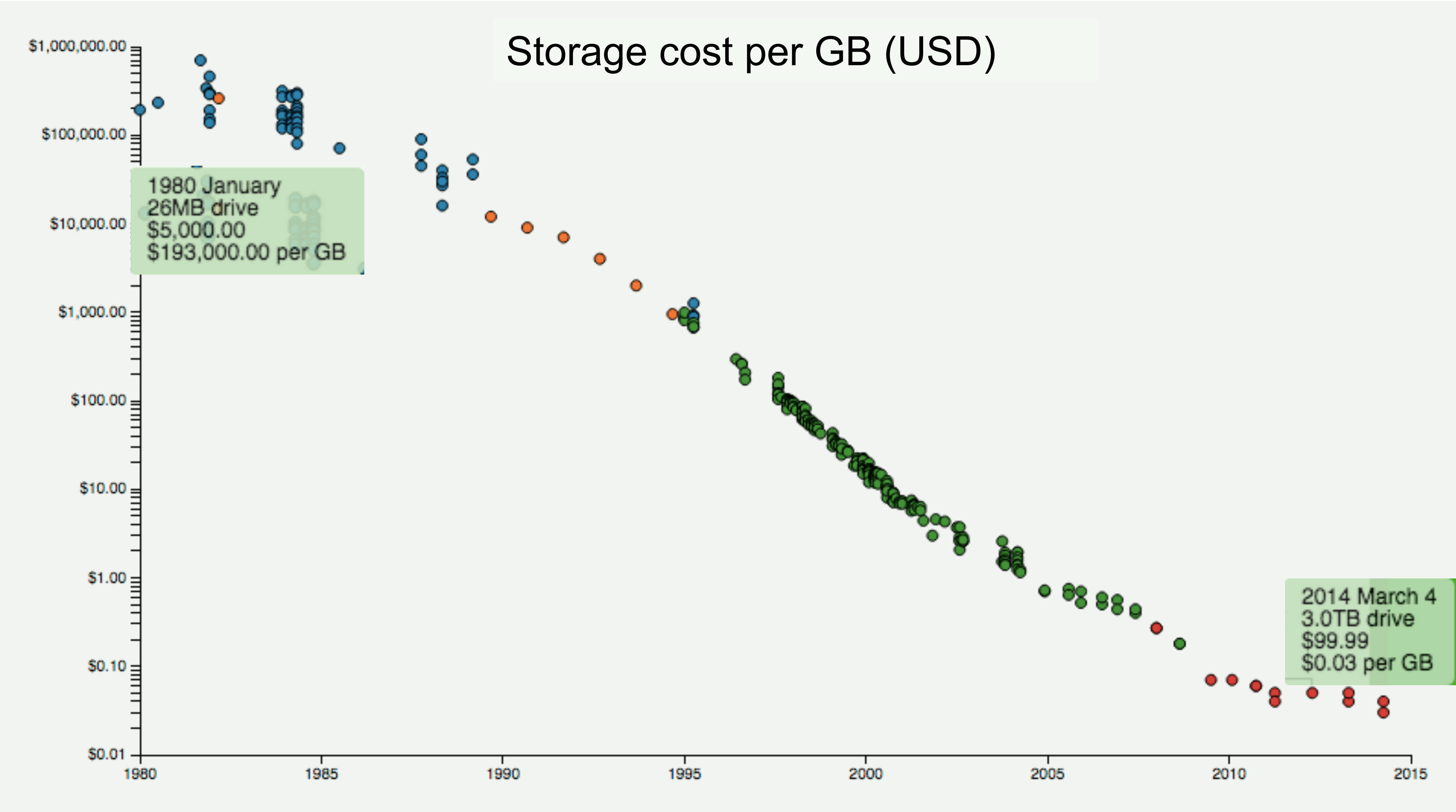
Big Data vs. Wireless



[Cisco]

$$\infty \equiv 5\ \text{GB} \pmod{\text{at\&t}}$$

Meanwhile, Storage is Getting Cheaper



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

Storage = Caching

- Index coding: [Birk & Kol ’98] + …
- Coded Caching: [Maddah-Ali & Niesen ’13] + …
- Femto-caching: [Golrezai et al. ’12]….
- ….

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

Index Coding Example

Trans-mission #	Index code 1
1	X_1
2	X_2
3	X_3
4	X_4

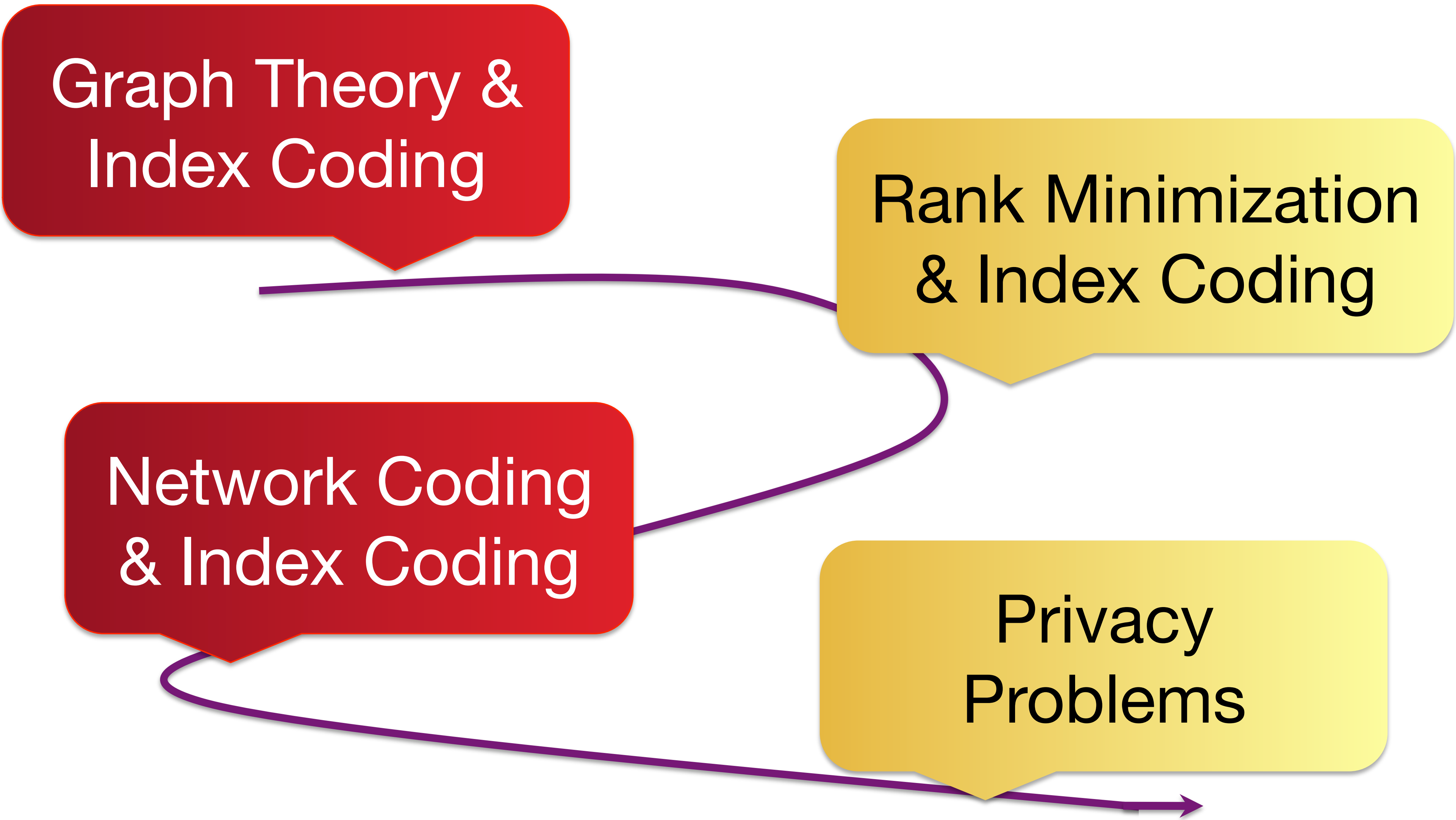
L=4 L=3

Min transmission rate?
Optimal schemes?

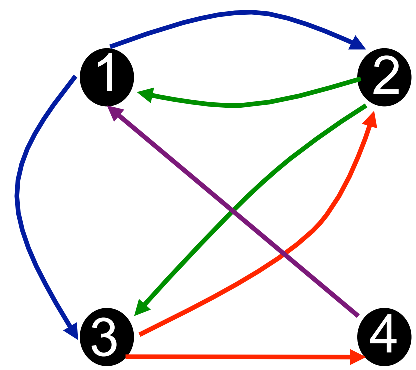
- Content of the cache is given
- Cached data independent of a user's preferences still help

Birk & Kol, "Informed-source coding-on-demand (ISCOD) over broadcast channels," INFOCOM'98

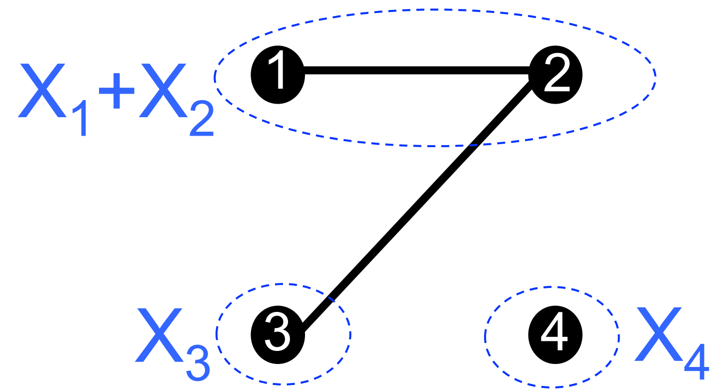
Talk Roadmap



Index Coding & Coloring



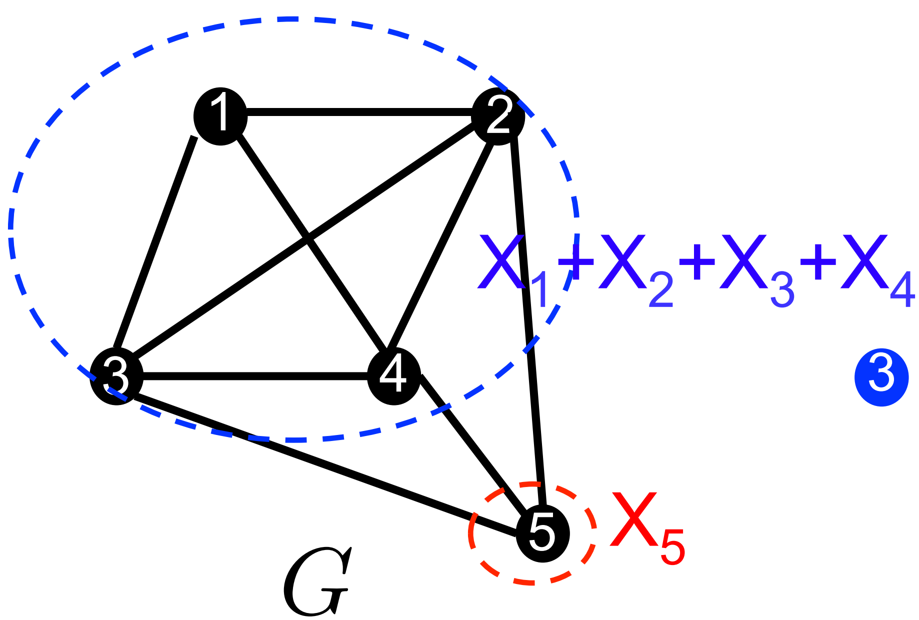
Side info graph G_d

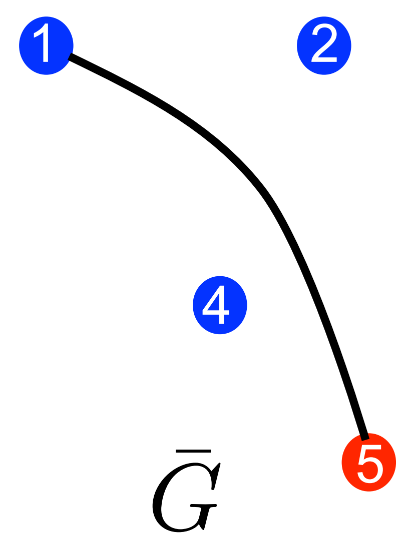


Clique cover of G

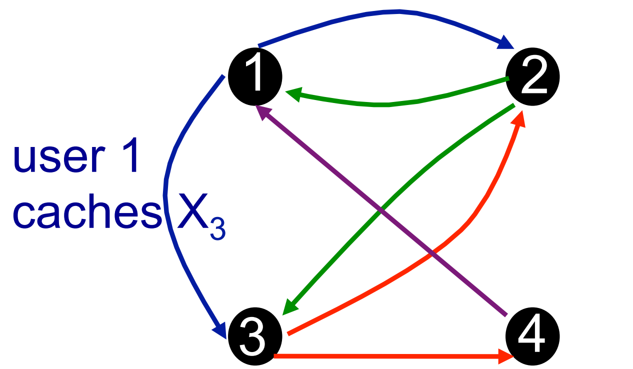
=

Chromatic nbr of $\bar{G}$

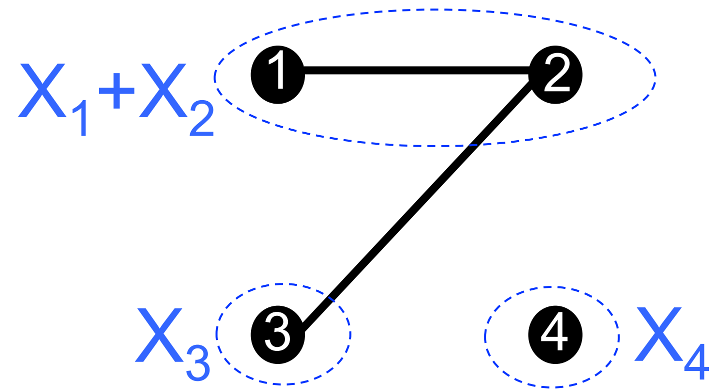




Index Coding & Graph Coloring



Side info graph G_d



Clique cover of G= Chromatic nbr of $\bar{G}$

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

Independ ence nbr $\alpha(G_d) \le c(G_d) \le L^*_{min} \le \chi_f(\bar{G}) \le \chi(\bar{G})$

Shannon capacity
[Haemers '79]

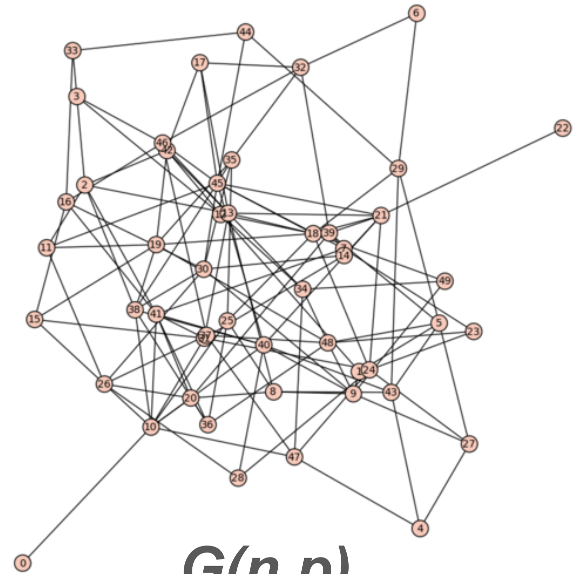
Fractional Chromatic nbr
[Blasiak, Kleinberg, Lubetzky '11]

- More bounds *[Dimakis et al.] [Arbobjalfoei & Kim], [Mazumdar et al.] etc…*

Index Coding on Erdős-Rényi Graphs

Independence nbr Chromatic nbr

$$\alpha(G) \leq L^*_{min} \leq \chi(\bar{G})$$



G(n,p)

- When $n \to \infty$, we have with prob 1

$$\log n \leq L^*_{min} \leq \frac{n}{\log n}$$

- Can improve the lower bound [Haviv & Langberg "Index Coding on random graphs", ISIT'12]

$$c\sqrt{n} \leq L^*_{min} \leq \frac{n}{\log n}$$

- Recent results closes the gap $L^*_{min} = \Theta(n/\log n)$

[Golovnev, Regev & Weinstein, "The Min Rank of Random graphs, Arxiv '16]

Talk Roadmap

Graph Theory & Index Coding

Rank Minimization & Index Coding

Network Coding & Index Coding

Privacy Problems

Index Coding & Rank Minimization

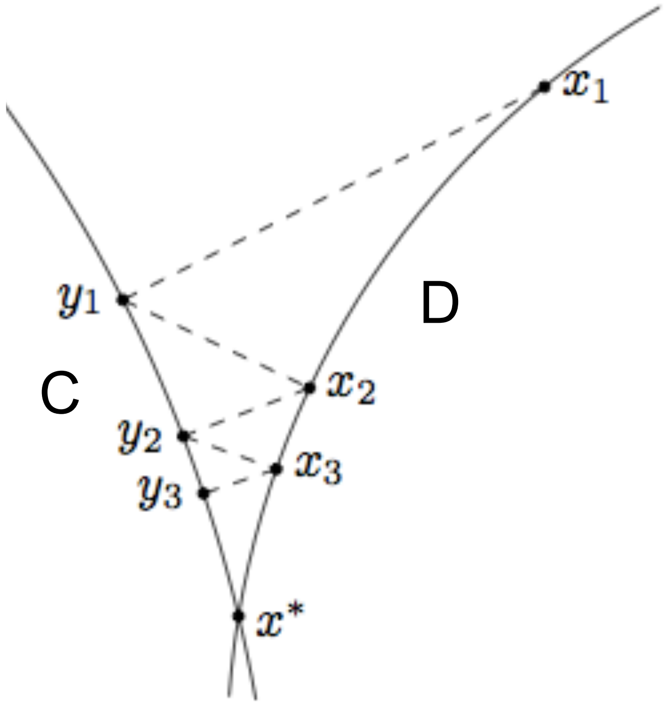
	X_1	X_2	X_3	X_4
X_1 t_1	1	1	1	0
X_1+X_2 t_2	1	1	1	0
$X_1+X_2+X_3$ t_3	0	1	1	1
X_1+X_4 X_4 t_4	1	0	0	1

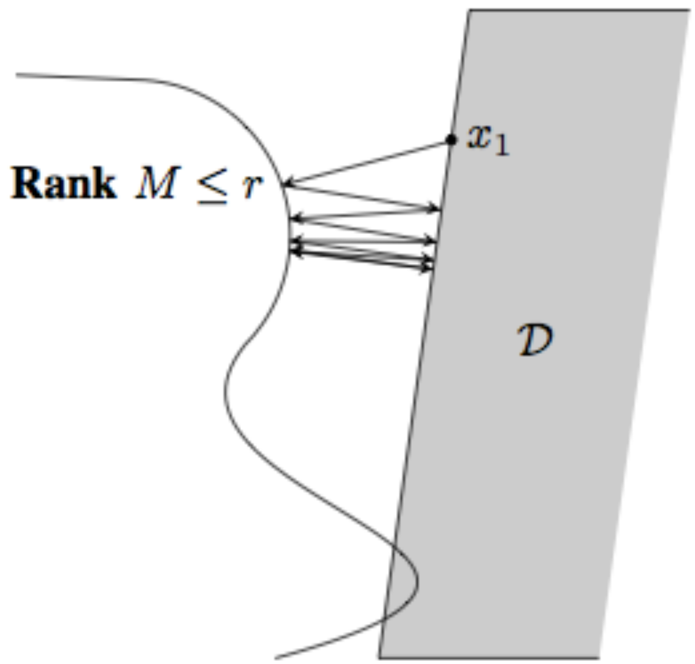
Matrix M

- 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]





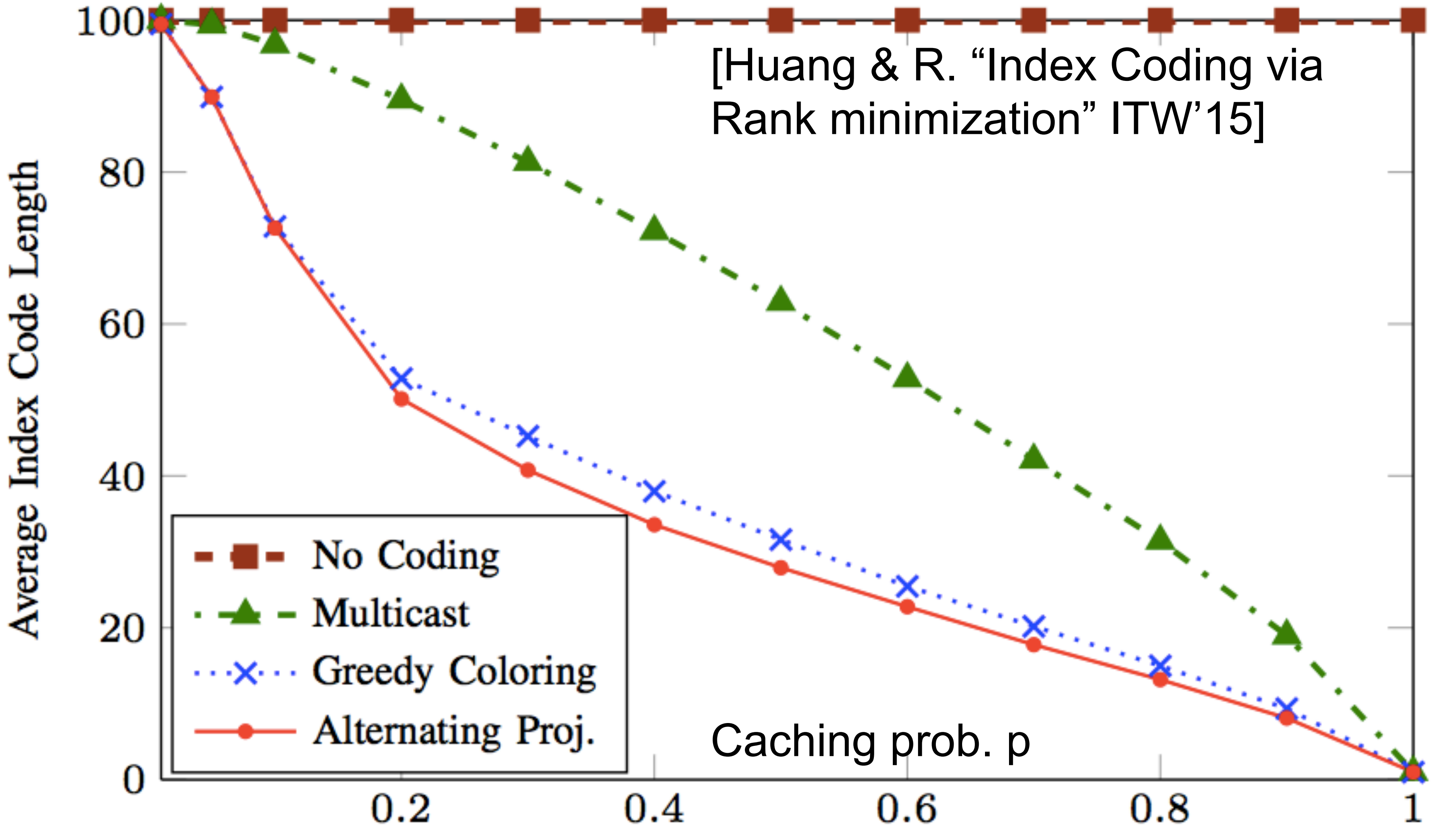
Index coding via AP

Two problems:
1) Regions not convex
2) Optimization over the reals

Theorem: [Alternating Projections (AP)]

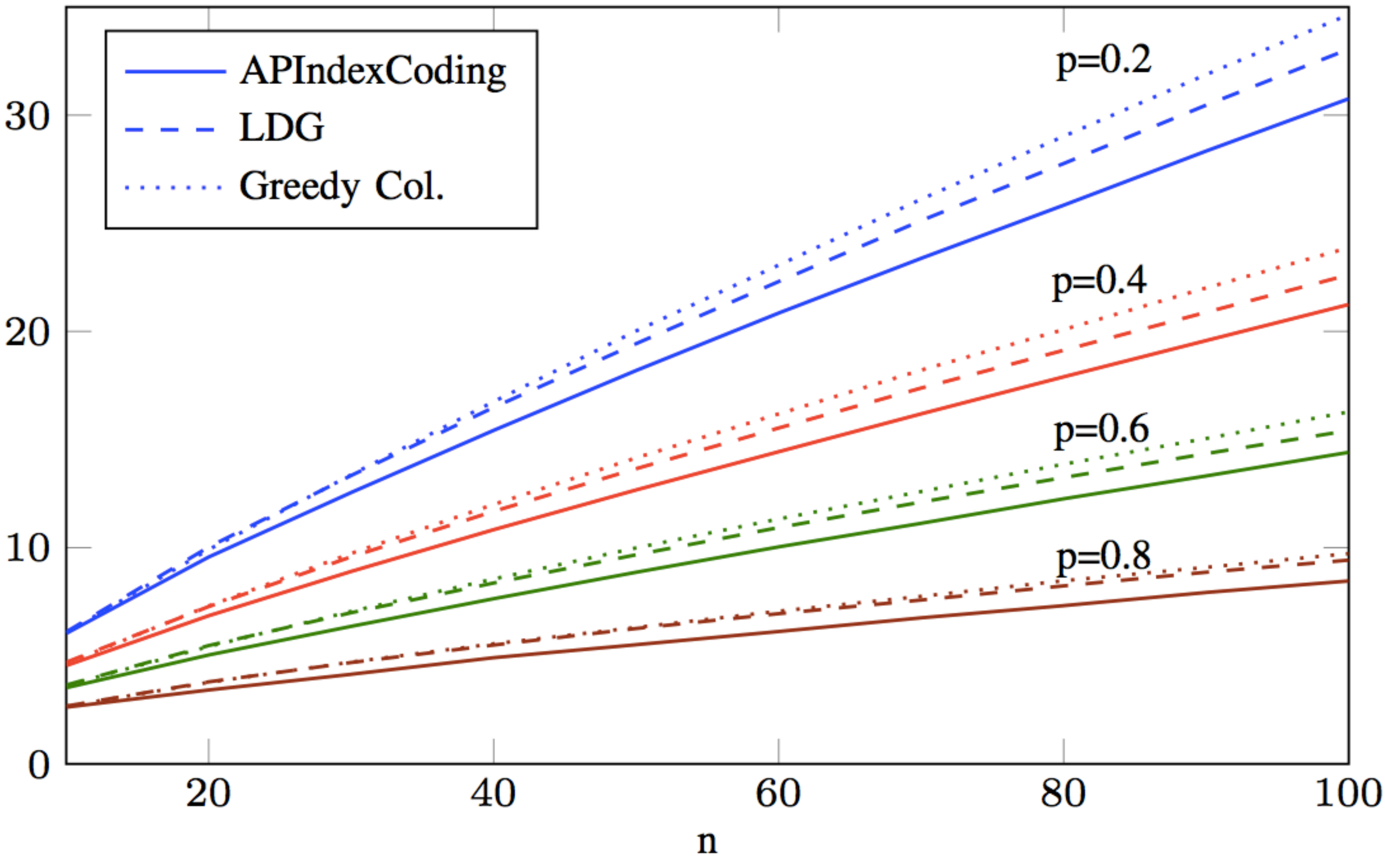
If C and D are convex, then an alternating projection sequence between these 2 regions converges to a point in their intersection.

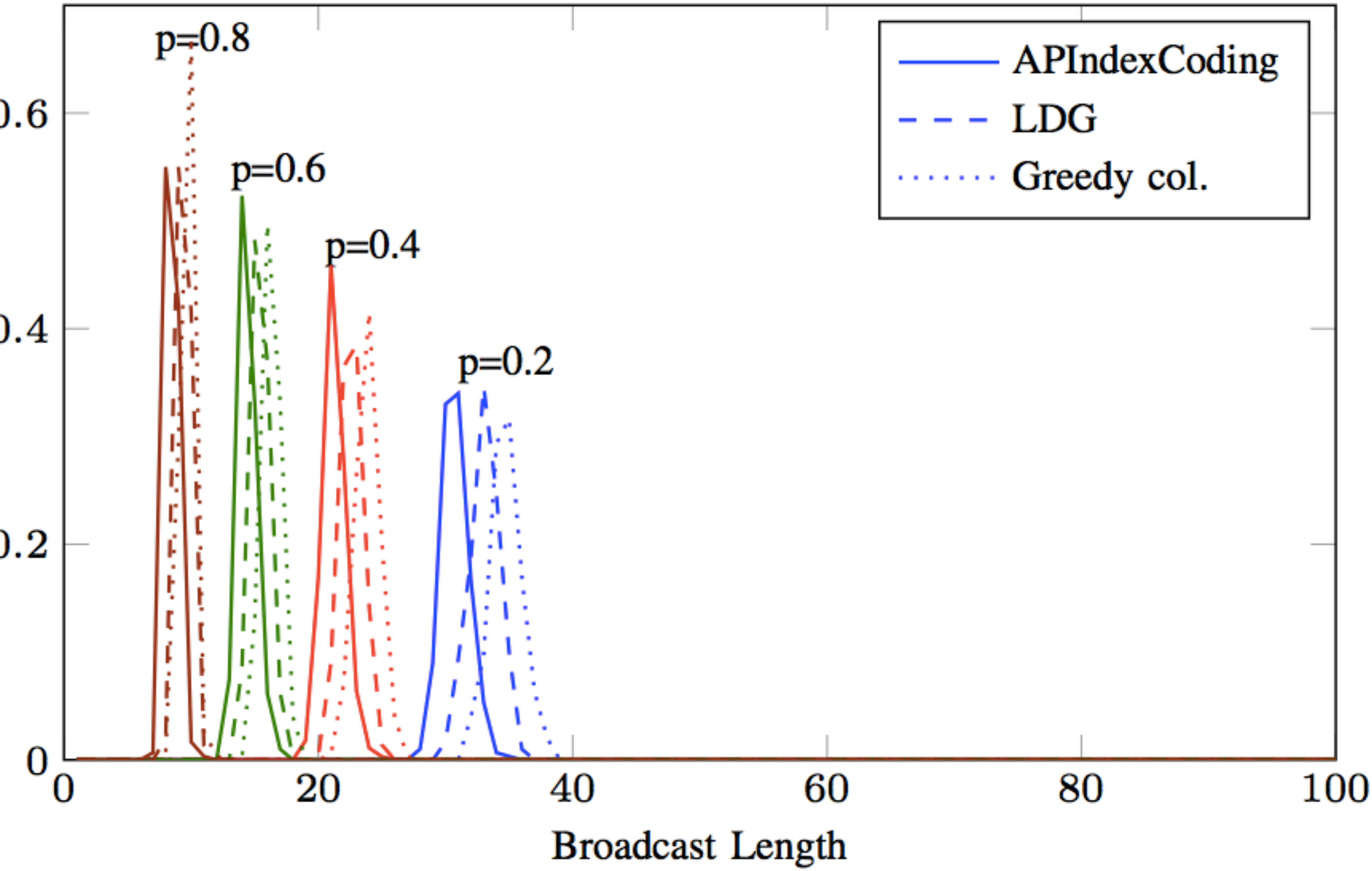
Alternating Projection on Random Undirected Graphs



- 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





p=0.8
p=0.6
p=0.4
p=0.2
APIndexCoding
LDG
Greedy col.
0.6
0.4
0.2
0
0
20
40
60
80
100
Broadcast Length

Talk Roadmap

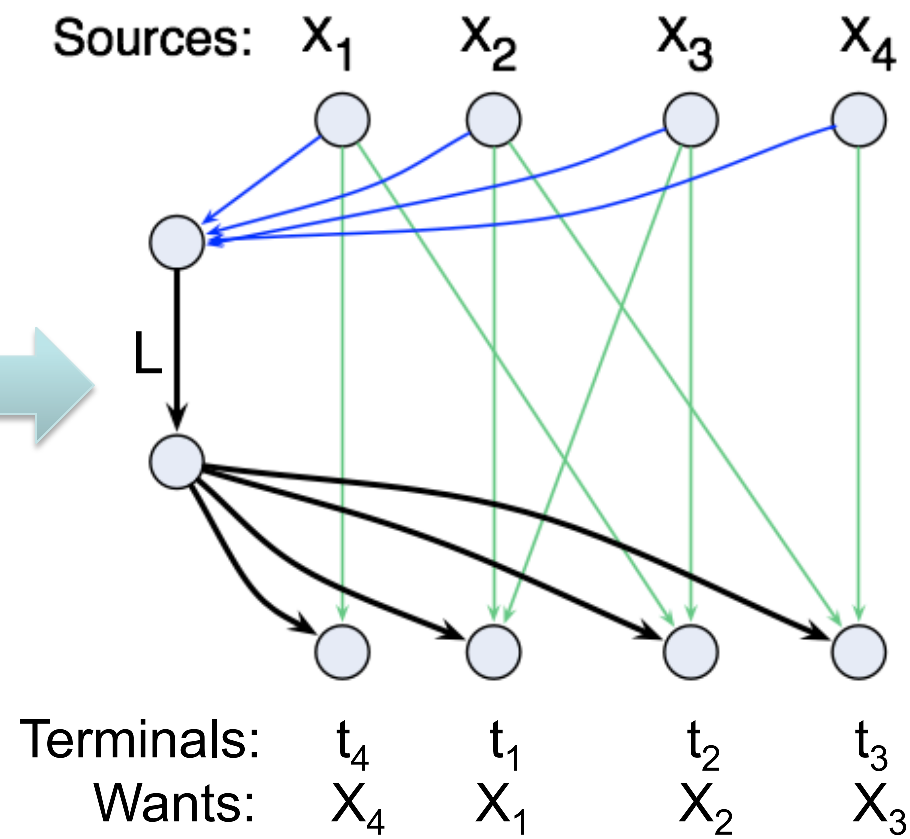
Graph Theory & Index Coding

Rank Minimization & Index Coding

Network Coding & Index Coding

Privacy Problems

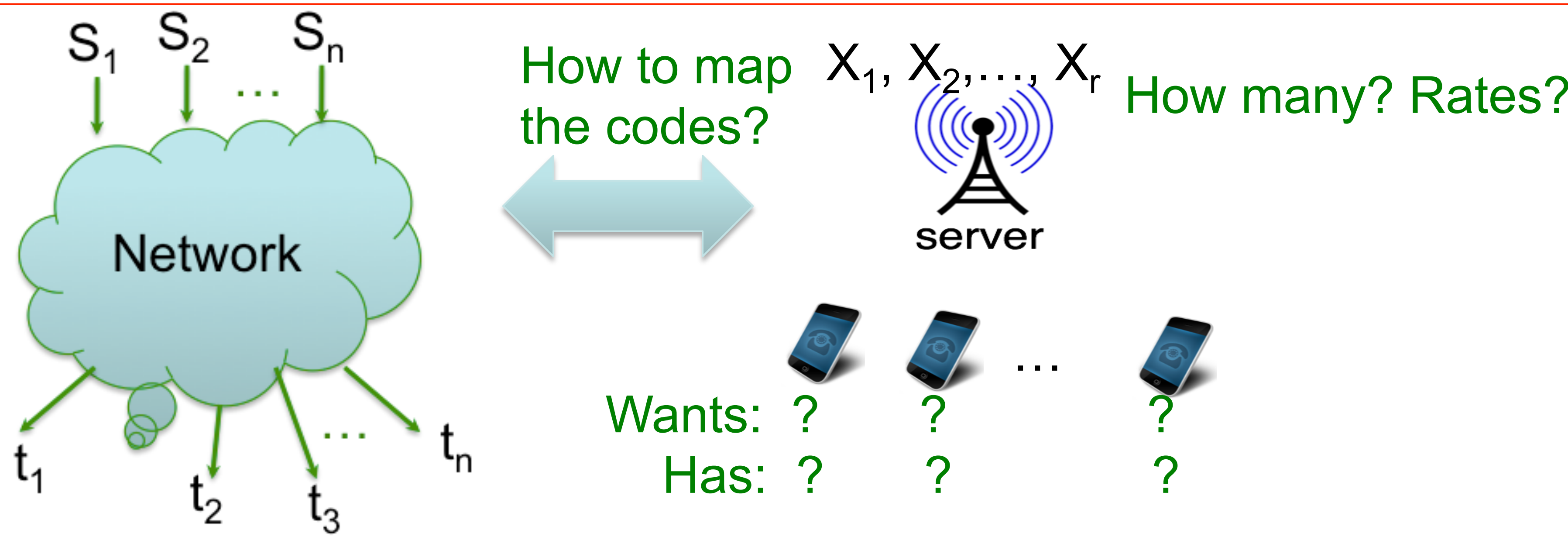
Equivalence to Network Coding



An index code of length L that satisfies all the users ⟺ **A network code that satisfies all the terminals**

From Network to Index Coding

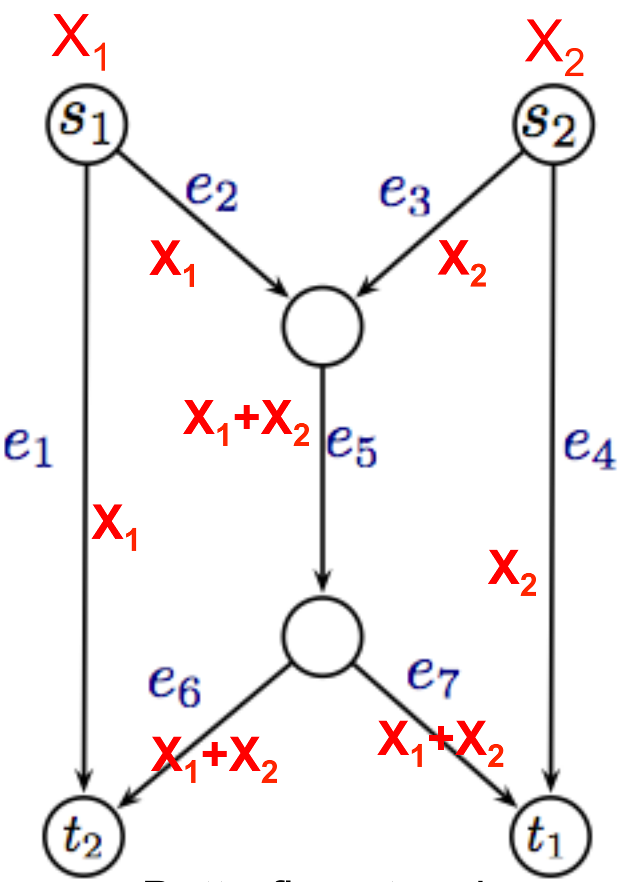
- Index coding is equivalent to the general network coding.
- If you can solve index coding efficiently you can solve any general network coding problem efficiently.



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).

Network Code → Index Code
The linear case first



Butterfly network

$Y_{e1}+X_1$

X_1

X_2

$Y_{e4}+X_2$

$\boxed{Y_{e5}}+X_1+X_2$

$Y_{e6}+X_1+X_2$

$Y_{e7}+X_1+X_2$

X_1, X_2

$Y_{e1}, Y_{e2}, \ldots, Y_{e7}$

$H(Y_{ei})=c(e_i)=1$

Terminal	Wants	Has
U_{e_1}	Y_{e_1}	X_1
U_{e_2}	Y_{e_2}	X_1
U_{e_3}	Y_{e_3}	X_2
U_{e_4}	Y_{e_4}	X_2
U_{e_5}	Y_{e_5}	Y_{e_2} Y_{e_3}

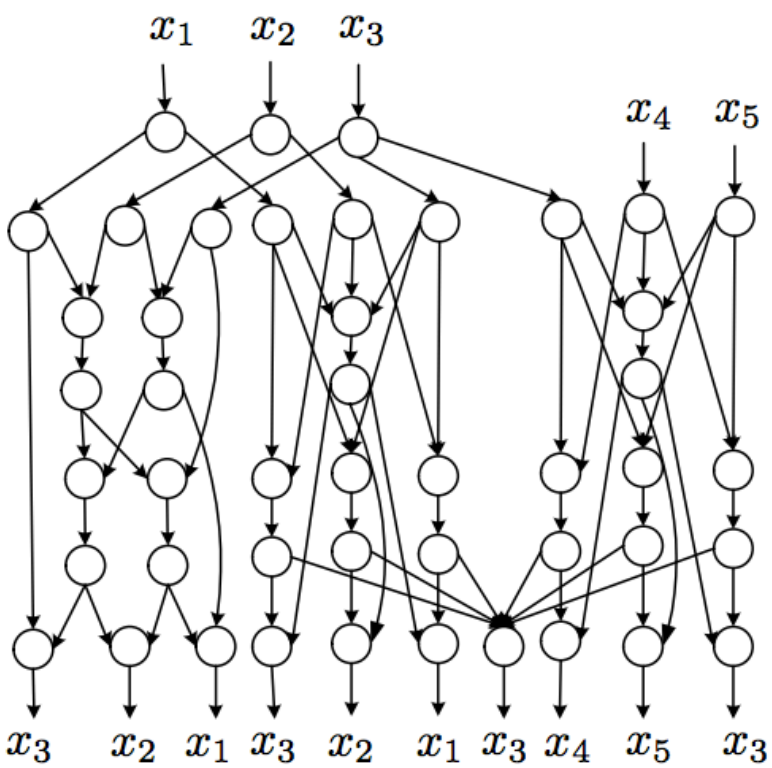
Terminal	Wants	Has
U_{e_6}	Y_{e_6}	Y_{e_5}
U_{e_7}	Y_{e_7}	Y_{e_5}

Equivalent index code

- All terminals in the index coding problem can decode
- Any linear network code gives an index code of length L=7

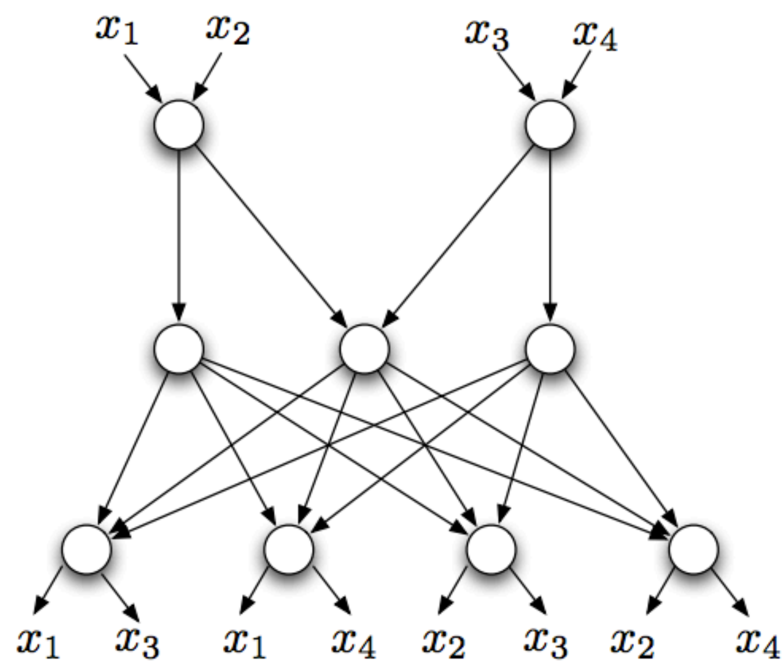
Implications on Index Coding

Linear index codes are not optimal



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

Vector linear codes outperform scalar linear



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] +…

Variations on Index Coding

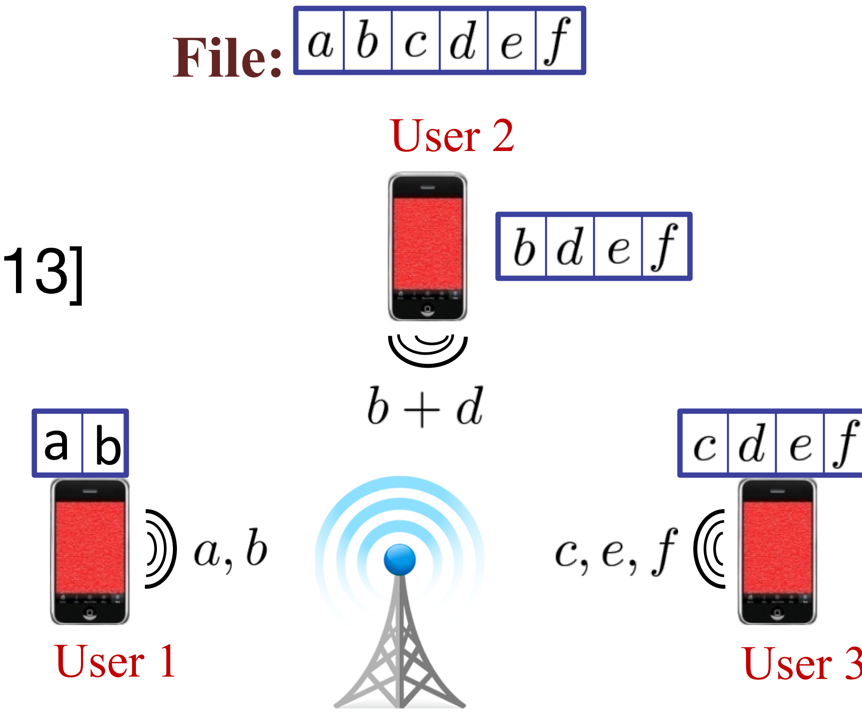
1. Data Exchange Problem

[R., Sprintson, Sadeghi ITW'10]

[Milosavlevijc, Pawar, R., Ramchandran, '13]

[Courtade et al. '13]…

- No Base station (D2D).
- Users wants missed parts



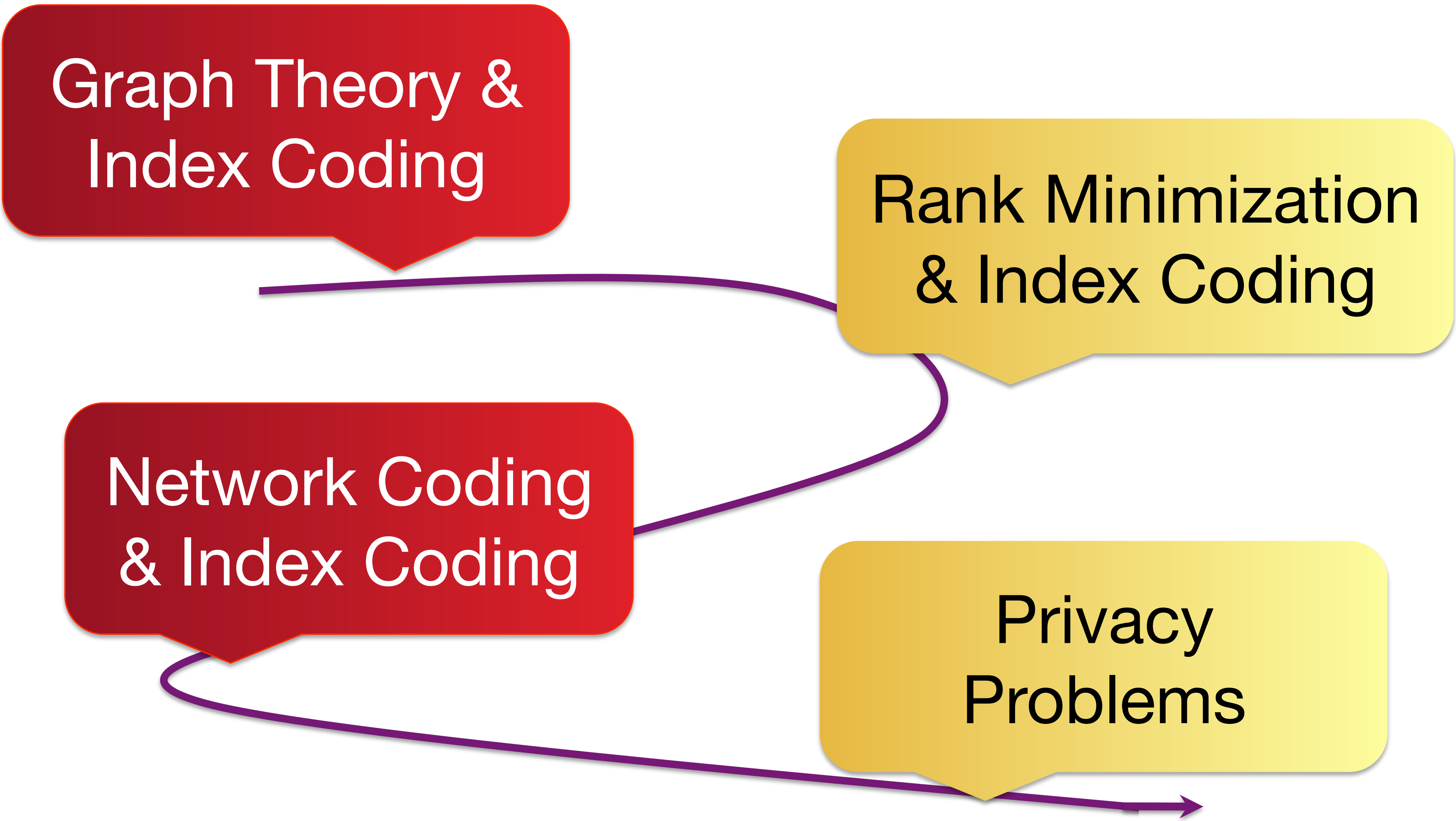
2. Pliable index coding [Fragouli et al '15]

- Like index coding but users want anything they don't have

3. Coded Caching [Maddah-Ali & Niesen '14]

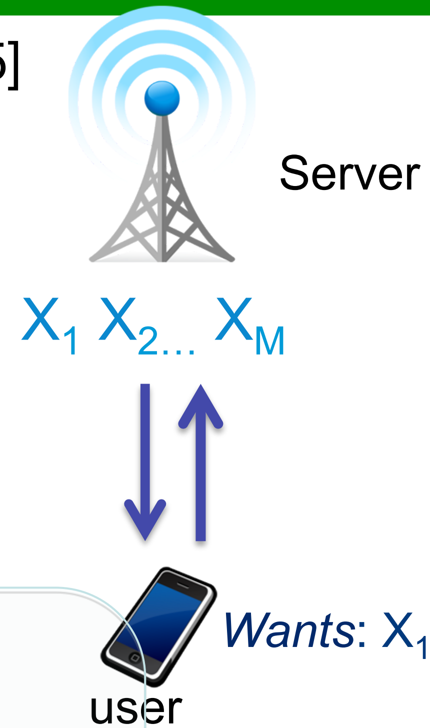
- Cached content is not fixed and can be designed
- Best paper, lots of follow up work…

Talk Roadmap



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

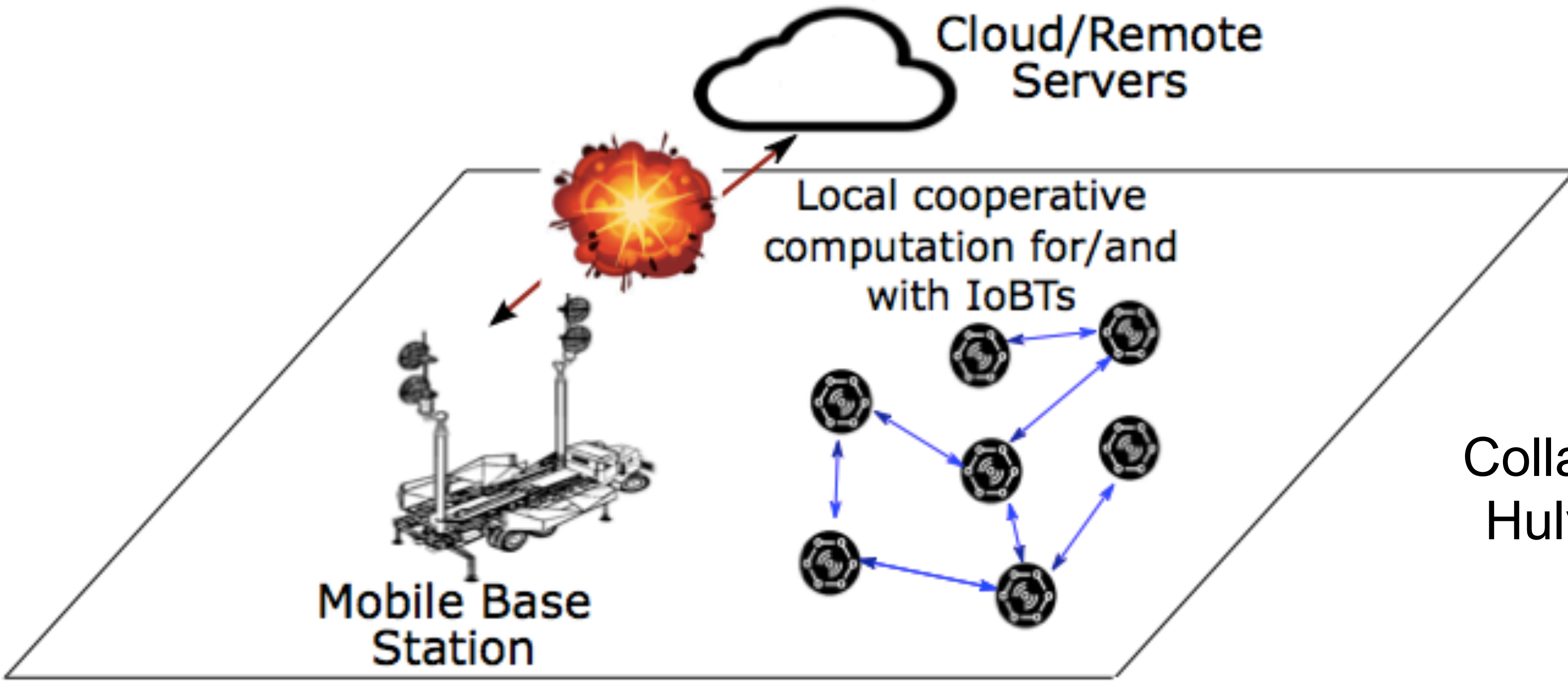


Theorem 1. *For the W-PIR-SI problem with $N = 1$ database, K messages, and side information size M, when the demand* $\mathbf{W}$ *and the side information set* $\mathbf{S}$ *are jointly distributed according to* (3)*, the capacity is*

$$C_W = \left\lceil \frac{K}{M+1} \right\rceil^{-1}. \tag{9}$$

Kadhe, Garcia, Heidarzadeh, R., Sprintson, "PIR with Side Information", Allerton '17

Secure Cooperative Computing in IoT



Collaboration with Hulya Seferoglu UIC

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

Bitar, R., "Staircase Codes for Secret Sharing with Optimal Communication Overhead," Trans. on info th., 2017.
R. Bitar P. Parag, R., "Minimizing Latency for Secure Distributed Computing", submitted to ISIT'17

Acknowledgment

Collaborators

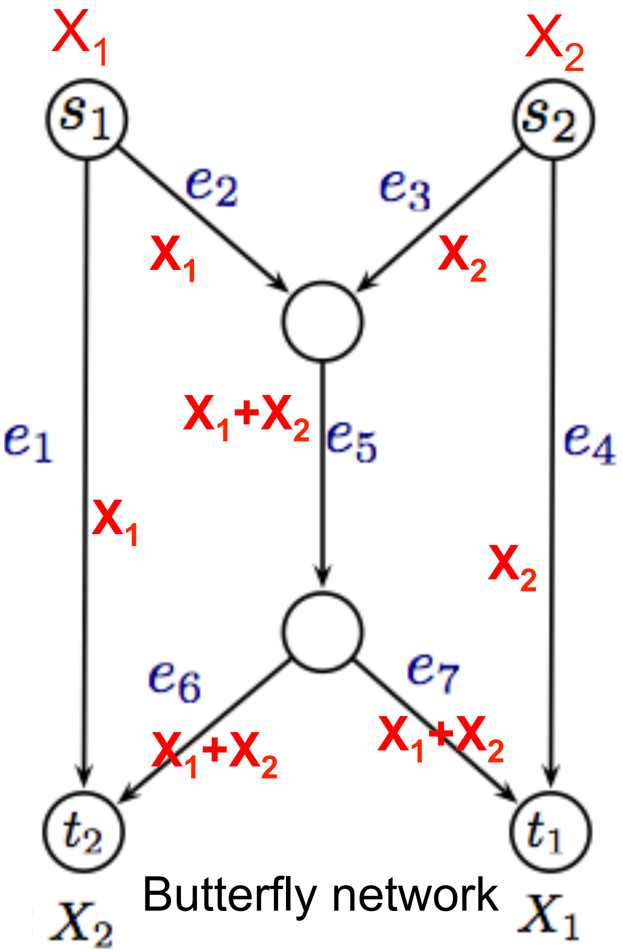
- My students: Rawad Bitar, Razan Tajeddine, Peiwen Tian
- Camilla Hollanti (Aalto University, Finland)
- Olgica Milenkovic (UIUC)
- Hulya Seferoglu, (UIC)
- Parimal Parag (IISC, India)

Funding agencies:

- NSF: CCF-15-26875
- NSF: CCF-16-52867
- ARL: W911NF-17-1-0032

QUESTIONS?

Network Code → Index Code
The linear case first



Butterfly network

$Y_{e1}+X_1$

X_1

X_2

$Y_{e4}+X_2$

$\boxed{Y_{e5}}+X_1+X_2$

$Y_{e6}+X_1+X_2$

$Y_{e7}+X_1+X_2$

X_1, X_2

$Y_{e1}, Y_{e2}, \ldots, Y_{e7}$

$H(Y_{ei})=c(e_i)=1$

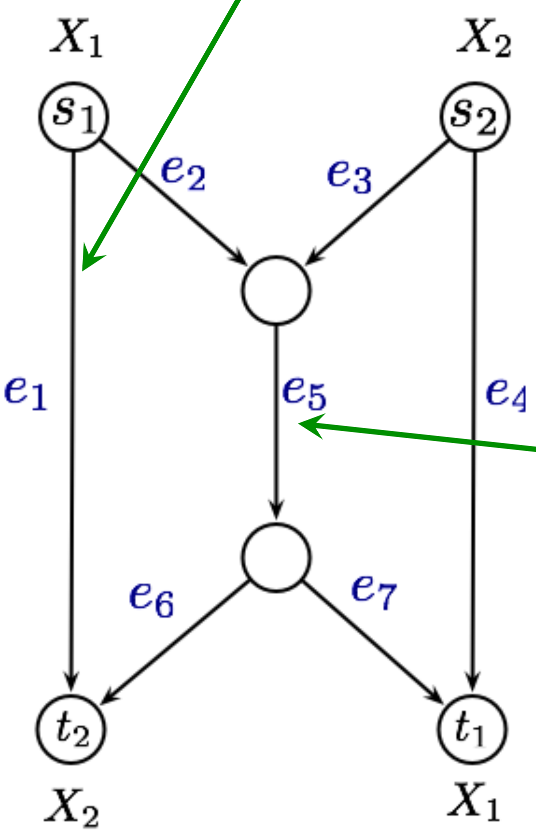
Terminal	Wants	Has
U_{e_1}	Y_{e_1}	X_1
U_{e_2}	Y_{e_2}	X_1
U_{e_3}	Y_{e_3}	X_2
U_{e_4}	Y_{e_4}	X_2
U_{e_5}	Y_{e_5}	$Y_{e_2}\ Y_{e_3}$

Terminal	Wants	Has
U_{e_6}	Y_{e_6}	Y_{e_5}
U_{e_7}	Y_{e_7}	Y_{e_5}

Equivalent index code

- All terminals in the index coding problem can decode
- Any linear network code gives an index code of length L=7

Index Code → Network Code



Butterfly network

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

Terminal	Wants	Has
U_{e_6}	Y_{e_6}	Y_{e_5}
U_{e_7}	Y_{e_7}	Y_{e_5}
U_{t_1}	X_1	$Y_{e_4}\ Y_{e_7}$
U_{t_2}	X_2	$Y_{e_1}\ Y_{e_6}$
U^*	$Y_{e_1} \ldots Y_{e_7}$	$X_1\ X_2$

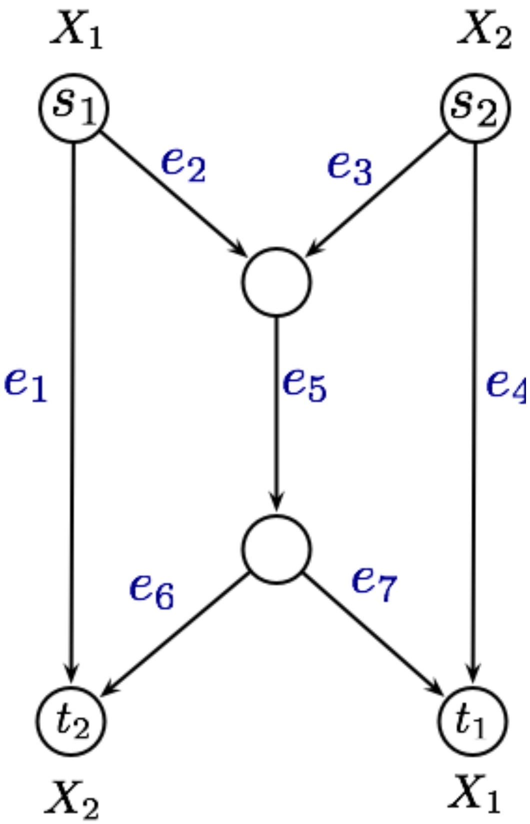
Given a linear index code

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

Can always diagonalize

- 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

$$Y_{e_1} + f_{e_1}(X_1, X_2)$$
$$f_{e_2}(X_1, X_2)$$
$$f_{e_3}(X_1, X_2)$$
$$Y_{e_4} + f_{e_4}(X_1, X_2)$$
$$\boxed{Y_{e_5}} + f_{e_5}(X_1, X_2)$$
$$Y_{e_6} + f_{e_6}(X_1, X_2)$$
$$Y_{e_7} + f_{e_7}(X_1, X_2)$$

X_1, X_2

$Y_{e_1}, \ldots, Y_{e_7}$

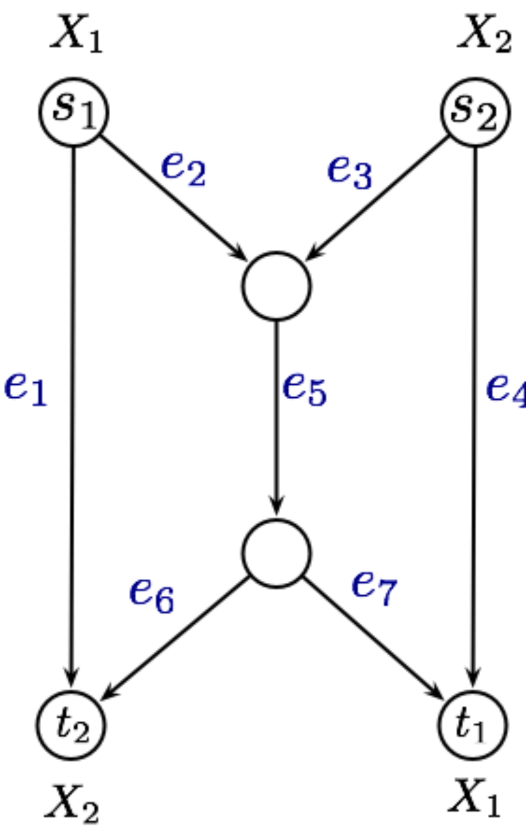
Terminal	Wants	Has
U_{e_1}	Y_{e_1}	X_1
U_{e_2}	Y_{e_2}	X_1
U_{e_3}	Y_{e_3}	X_2
U_{e_4}	Y_{e_4}	X_2
U_{e_5}	Y_{e_5}	$Y_{e_2}\ Y_{e_3}$

Terminal	Wants	Has
U_{e_6}	Y_{e_6}	Y_{e_5}
U_{e_7}	Y_{e_7}	Y_{e_5}
U_{t_1}	X_1	$Y_{e_4}\ Y_{e_7}$
U_{t_2}	X_2	$Y_{e_1}\ Y_{e_6}$
U^*	$Y_{e_1} \ldots Y_{e_7}$	$X_1\ X_2$

Equivalent index code

$f_{e_i}(X_1, X_2)$: message on edge e_i

"Diagonalization" May Not Work for Non-Linear



Butterfly network

$$B'_1 = g'_1(Y_{e_1}, \bar{X})$$
$$B'_2 = g'_2(Y_{e_2}, \bar{X})$$
$$B'_3 = g'_3(Y_{e_3}, \bar{X})$$
$$B'_4 = g'_4(Y_{e_4}, \bar{X})$$
$$B'_5 = g'_5(Y_{e_5}, \bar{X})$$
$$B'_6 = g'_6(Y_{e_6}, \bar{X})$$
$$B'_7 = g'_7(Y_{e_7}, \bar{X})$$

Terminal	Wants	Has
U_{e_6}	Y_{e_6}	Y_{e_5}
U_{e_7}	Y_{e_7}	Y_{e_5}
U_{t_1}	X_1	$Y_{e_4}\ Y_{e_7}$
U_{t_2}	X_2	$Y_{e_1}\ Y_{e_6}$
U^*	$Y_{e_1} \ldots Y_{e_7}$	$X_1\ X_2$

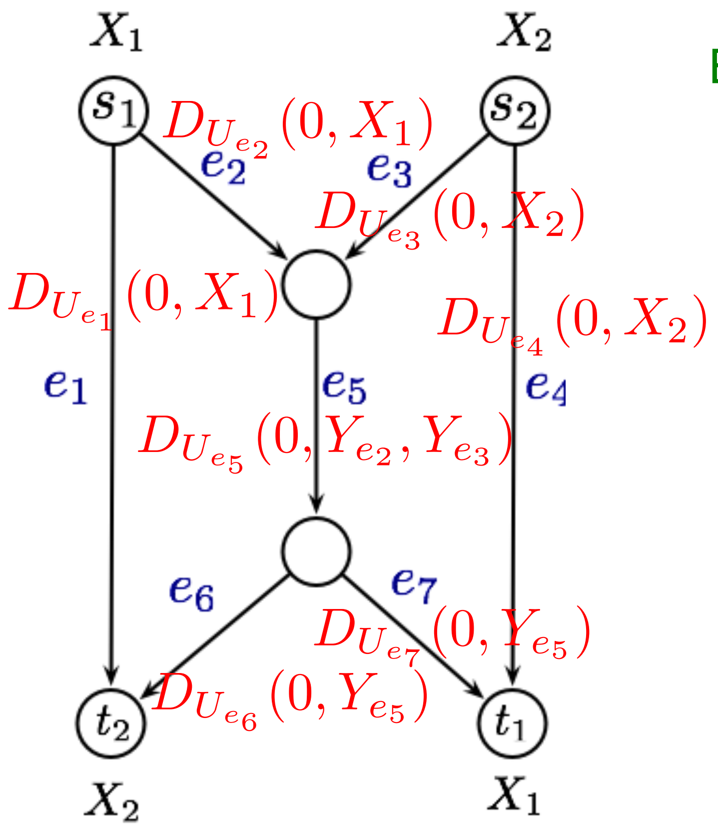
If we can we diagonalize

?

Given a non-linear index code

$$B_1 = g_1(\bar{Y}_e, \bar{X})$$
$$B_2 = g_2(\bar{Y}_e, \bar{X})$$
$$B_3 = g_3(\bar{Y}_e, \bar{X})$$
$$B_4 = g_4(\bar{Y}_e, \bar{X})$$
$$B_5 = g_5(\bar{Y}_e, \bar{X})$$
$$B_6 = g_6(\bar{Y}_e, \bar{X})$$
$$B_7 = g_7(\bar{Y}_e, \bar{X})$$

Non-linear Index Code → Network Code



Broadcast message

Decoding function

$$X1 = D_{U_{t1}}(B, Y_{e_4}, Y_{e_7})$$

$$Y_{e_4} = D_{U_{e_4}}(B, X_2)$$

$$Y_{e_7} = D_{U_{e_7}}(B, Y_{e_5})$$

Fix a value for B, say B=0

X_1, X_2

$Y_{e_1}, \ldots, Y_{e_7}$

Terminal	Wants	Has
U_{e_1}	Y_{e_1}	X_1
U_{e_2}	Y_{e_2}	X_1
U_{e_3}	Y_{e_3}	X_2
U_{e_4}	Y_{e_4}	X_2
U_{e_5}	Y_{e_5}	$Y_{e_2}\, Y_{e_3}$
U_{e_6}	Y_{e_6}	Y_{e_5}
U_{e_7}	Y_{e_7}	Y_{e_5}
U_{t_1}	X_1	$Y_{e_4}\, Y_{e_7}$
U_{t_2}	X_2	$Y_{e_1}\, Y_{e_6}$
U^*	$Y_{e_1} \ldots Y_{e_7}$	$X_1\, X_2$

- 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 <u>unique</u> possible vector $(Y_{e1},\ldots,Y_{e7})$
- Otherwise, U* cannot decode correctly

Dealing with Errors

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

X_1, X_2

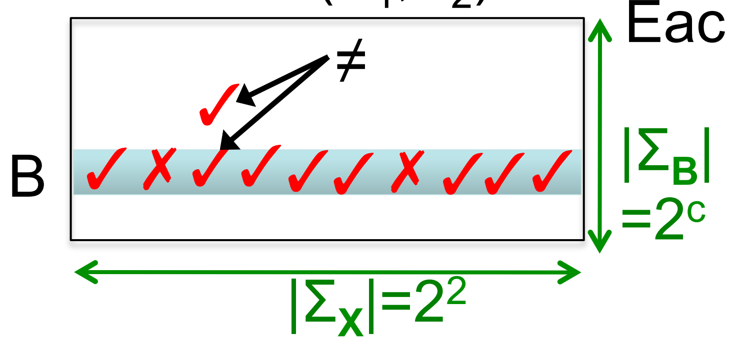
$Y_{e_1}, \ldots, Y_{e_7}$

Terminal	Wants	Has
U_{e_1}	Y_{e_1}	X_1
U_{e_2}	Y_{e_2}	X_1
U_{e_3}	Y_{e_3}	X_2
U_{e_4}	Y_{e_4}	X_2
U_{e_5}	Y_{e_5}	$Y_{e_2}\,Y_{e_3}$
U_{e_6}	Y_{e_6}	Y_{e_5}
U_{e_7}	Y_{e_7}	Y_{e_5}
U_{t_1}	X_1	$Y_{e_4}\,Y_{e_7}$
U_{t_2}	X_2	$Y_{e_1}\,Y_{e_6}$
U^*	$Y_{e_1}\ldots Y_{e_7}$	$X_1\,X_2$

Claim: There exists σ, such that for B=σ, in the previous construction, the network code will have a prob of error at most ε (ε=error prob of the index code).

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

X=(X_1,X_2)



Ye=$D_{U*}(B, X_1, X_2)$

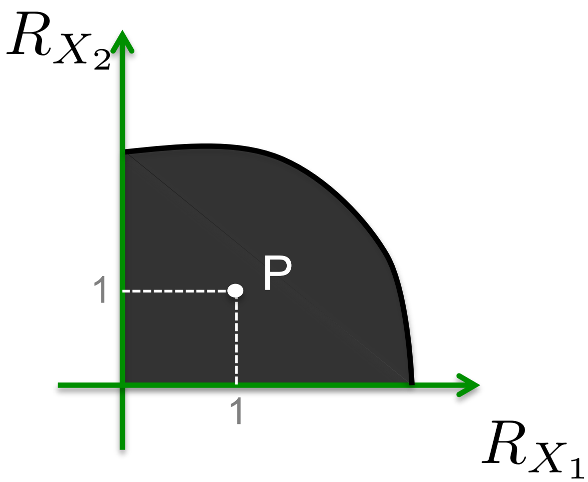
✗: decoding error

Each ✓ corresponds to a different "good" value of (**X**,**Ye**)

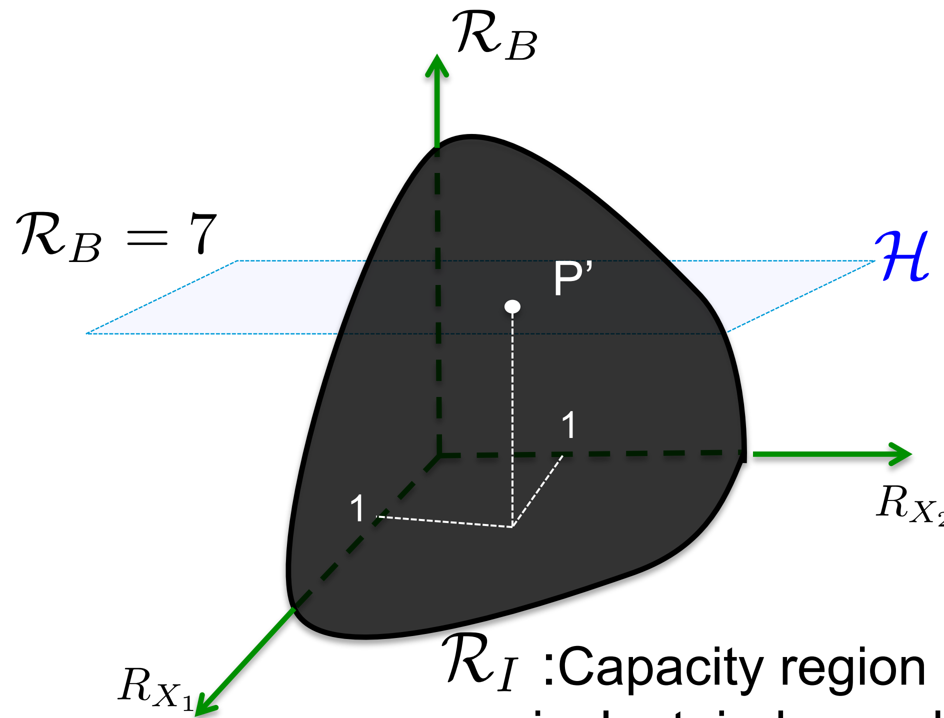
Total # of ✓<$(1-\varepsilon)|\Sigma_B|.|\Sigma_X|$
But $|\Sigma_B|=|\Sigma_{Ye}|$
➔Total # of "good" values<$(1-\varepsilon)|\Sigma_{Ye}|.|\Sigma_X|$

contradiction

Capacity Regions



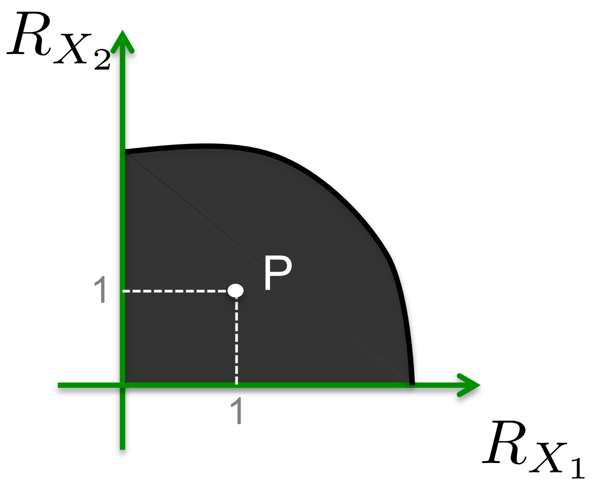
$\mathcal{R}_N$:Capacity region of a network



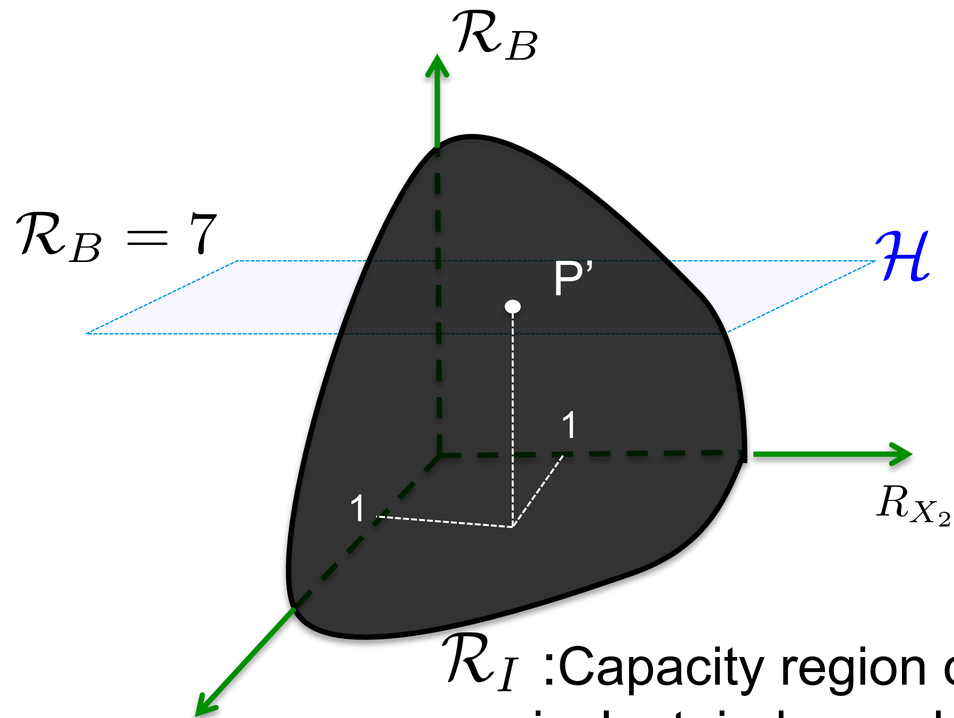
$\mathcal{R}_I$:Capacity region of the equivalent index code

- If there is a code that achieves P "exactly", then P' is in $\mathcal{R}_{\mathcal{I}} \cap \mathcal{H}$, 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= ε-error capacity of networks?
- True for index coding problems [Langberg, Effros '11]

The Case of Co-located Sources



$\mathcal{R}_N$:Capacity region of a network



$\mathcal{R}_I$: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 **R** is in the capacity region of $\mathcal{N}$ iff **R**' is in the capacity region of $\mathcal{I}$ with broadcast length L.