Coding for Online Adversaries



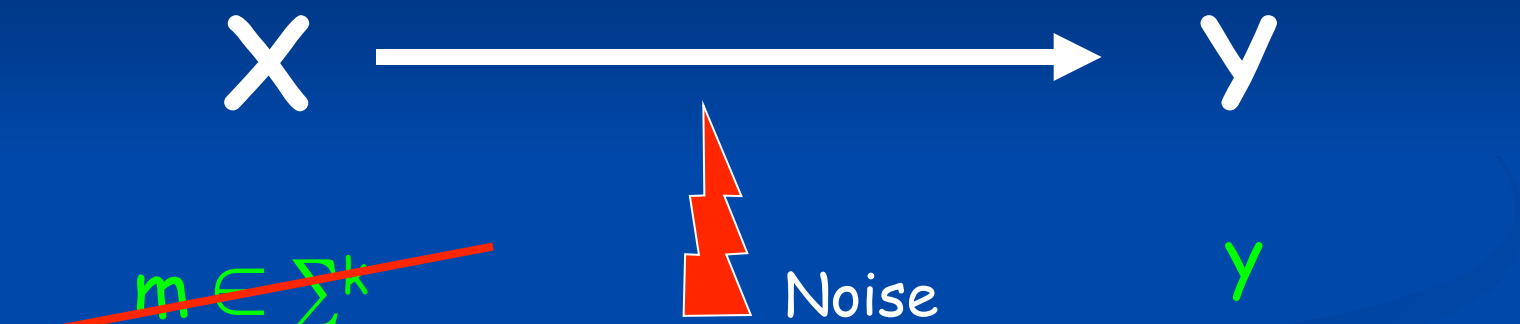


OPEN UNIVERSITY OF ISRAEL

UC San Diego

Bikash Dey Ishay Haviv Sidharth Jaggi Michael Langberg Anand Sarwate

Coding theory



~~$m \in \Sigma^k$~~

$x = C(m) \in \Sigma^n$

y

decode

m

Error correcting codes

$C: \Sigma^k \rightarrow \Sigma^n$

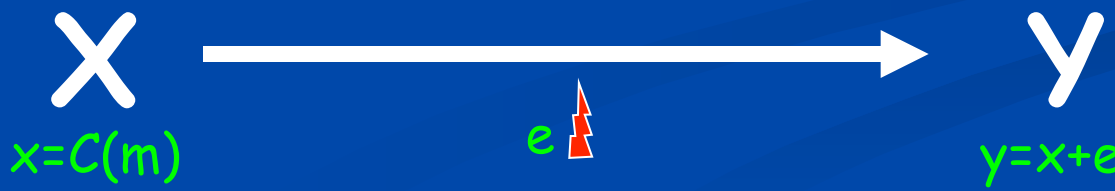
Communication channels

Design of C depends on properties of channel.

- This talk:
  - We view channel as a jammer that (may) be malicious.
  - Refer to malicious jammer also as adversary.
  - In general, y=x+e (this will change later …).
  - All jammers considered will have a power constrain p: at most pn characters of x can be changed: $|e| \leq pn$.

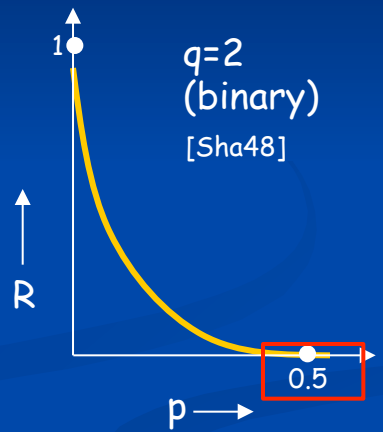
- Success criteria:
  - Average error ($Pr_{m,channel}[D(y)=m] \sim 1$).
  - Maximum error: stochastic codes (random encoding).
- Stress:
  - No shared information between X and Y.

- C,D is said to allow the communication of m over channel if $Pr_e[D(C(m)+e)=m] \sim 1$.
  - Probability over channel.
- C,D is said to allow the communication of $\sum^k$ over channel if $Pr_{m,e}[D(C(m)+e)=m] \sim 1$.
  - Probability uniform over $\sum^k$ and over channel.
- Rate of C is k/n; Capacity = maximal achievable rate.

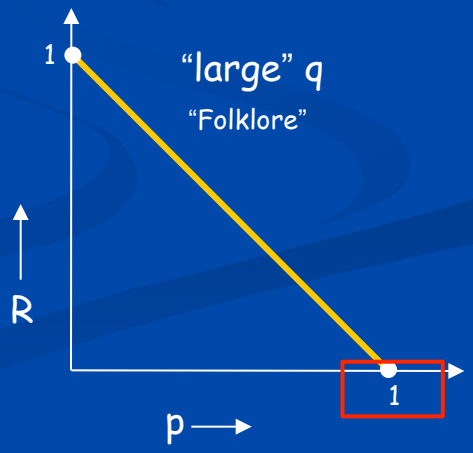


Random jammer

- What is known when error is random (change uniformly with probability p)?
- Will consider different alphabet sizes: $q=|\Sigma|$.



- For binary case (q=2):
  - Capacity = 1-H(p)

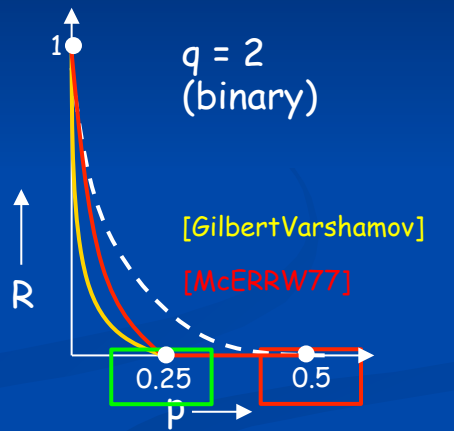


- For "large" alphabets:
  - Capacity = 1-p

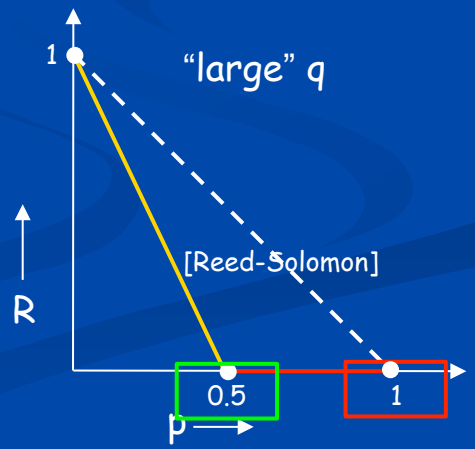
Adversarial jammer

- Malicious jammer controlling the error:
  - Knows code shared by X and Y.
  - Sees codeword x sent by X.
  - Plans an error e to disturb comm.
  - Error of weight at most pn.

Min. distance: 2pn+1

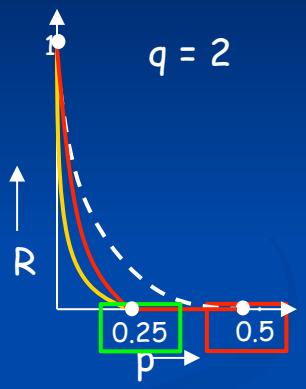


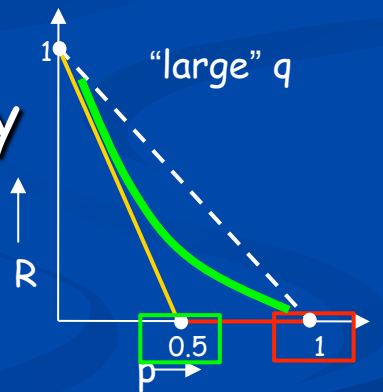
- For binary case (q=2):
  - 1-H(p) > Capacity ≥ 1-H(2p)
- For large alphabets:
  - Capacity = 1-2p

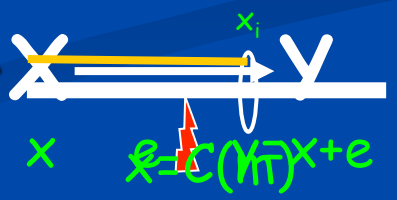


This talk: Online adversaries

- What happens if adversary behaves in an "online" or "causal" manner.
- Adversarial jammer is still malicious.
- Adversary still knows code shared by X and Y, but has partial information regarding the codeword x.
- Adversary sees codeword x "character by character", must base its decisions on what it has seen so far.
- Adv. is stronger than random jammer.
- Online adv. is weaker than "unlimited" adv.
- **What rate can be achieved?**

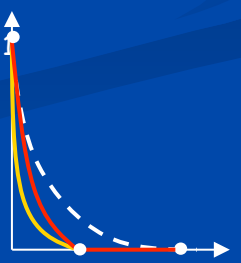






Theme: study natural channels that are somewhere between "unlimited adversarial jammer" and "random jammer".
- Combine perspective/tools from IT and TCS.

- Large alphabet setting:
  - Online setting fits applications in which X sends codewords consisting of n packets (characters).
  - Each packet is sent independently over time.
  - Decoding is done only after all packets arrive.
  - Adversary has limited jamming power: can corrupt only pn packets.
  - Corresponds to wireless comm. ("Jam or listen").
- Binary setting:
  - Understanding capacity of unlimited adversarial channel: major open question (codes of large min. dist.).
  - A better understanding of online adv. may advance our understanding.

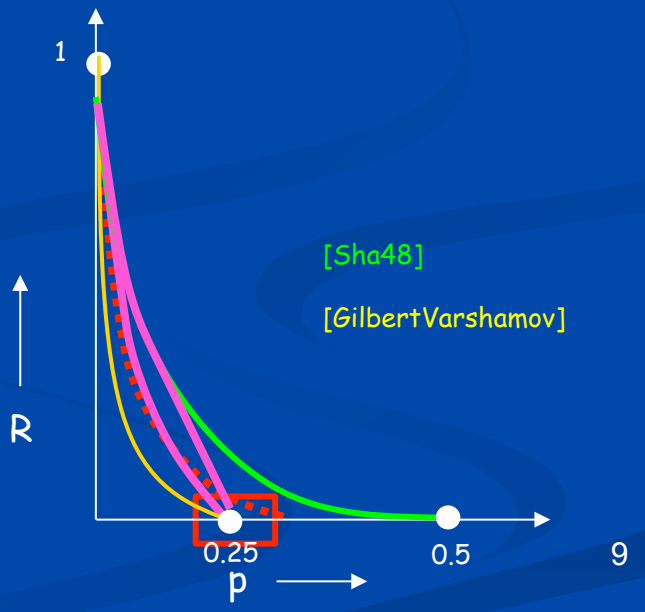


- Proof combinatorial in nature:
  - Turans theorem.
  - Plotkin bound.
  - Probabilistic arguments

$$C \leq \min_{\bar{p}\in[0,p]} \left[(1-4(p-\bar{p})) \left(1 - H\left(\frac{\bar{p}}{1-4(p-\bar{p})} \right) \right) \right].$$

- **Is the online capacity equal to that of unlimited adversary?**
- **Is the online capacity equal to that of random adversary?**

- Lets start with the binary alphabet:
- One may search for upper and lower bounds.
- Upper bound:
  - min(1-4p,1-H(p)).
  - Recently improved.
  - When p≥0.25 rate R=0.
  - Just like "unlimited adversary"!



- **Online capacity differs from that of random adversary.**
- **However, does it equal to that of unlimited adversary?**

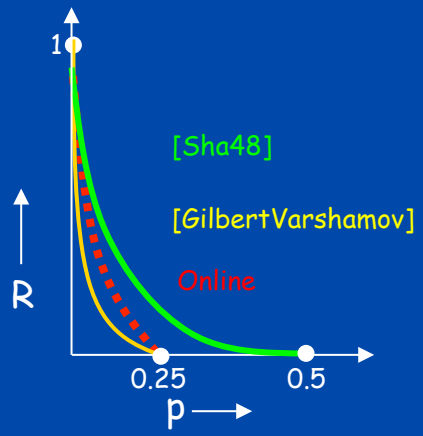
Results for binary case:

Q: Is the online capacity equal to that of unlimited adversary?

- GV bound of 1-H(2p) is best known lower bound for unlimited adversaries.
- We show: Online capacity is strictly greater than GV.

Theorem 1.2. *For any p such that $H(2p) \in (0, \frac{1}{2})$ there exists a $\delta_p > 0$ such that*

$$C_{online}(p) \geq 1 - H(2p) + \delta_p.$$



Hints on separation between online and unlimited adversarial channels.

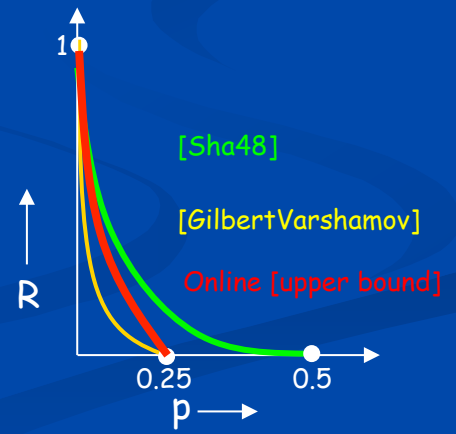
Summary for binary case

- Separation between the online channel and previously studied "strong" and "weak" channels.

Theorem 1.2. *For any p such that $H(2p) \in (0, \frac{1}{2})$ there exists a $\delta_p > 0$ such that*

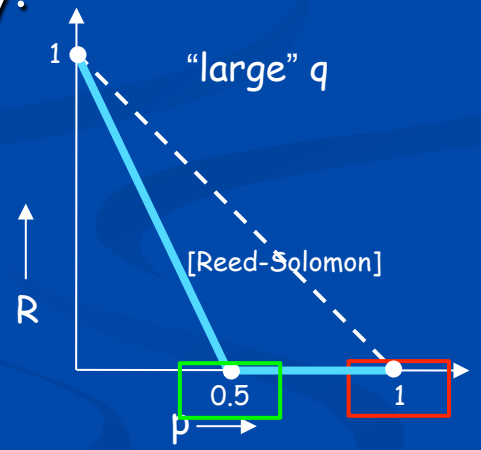
$$C_{online}(p) \geq 1 - H(2p) + \delta_p.$$

- What about large alphabets?
- We address same questions.
- Problem is significantly different as large alphabets allow "rich" encodings.



Our results: large $q=|\Sigma|$

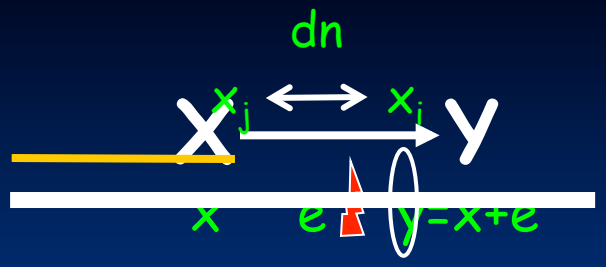
- Is the online capacity equal to one of the extremes?
  - Unlimited adversary/Random adversary?
- Yes!
  - We get a full characterization of the capacity.



- Turns out that an online adversary is just as strong as one which is not online.
- Namely, capacity equals that of unlimited adversary: 1-2p.

- Are we done?

What about delay?



- Up to now we considered restricting the adversary by forcing causality.
- Namely, after the jammer sees $x_1,x_2,\ldots,x_i$ it makes a decision on the value of e_i.
- What if due to computational or communication delays, the value of e_i must be decided on solely based on $x_1,x_2,\ldots,x_{i-dn}$ for some delay param $d>0$.
- New adv. is still stronger than random jammer.
- New adv. is still weaker than "unlimited" adv.
- What is the capacity in this case?
- Here also we have a full characterization for large q!

What about delay?

- What if due to computational or communicational delays, the value of ei must be decided on solely based on $x_1, x_2, \ldots, x_{i-d}$ for some delay param $d>0$.
- New adv. is still stronger than random noise.
- New adv. is still weaker than "unlimited" adv.
- What is the capacity in this case?
- Here also we have a full characterization!

- Wait! Once we have delay it is interesting to consider the error model:
  - Additive: $y=x+e$.
  - Overwrite: If sends e_i then $y_i=e_i$.
- When x_i is known to jammer there is no difference between the two. But in our setting ($d>0$) there is.

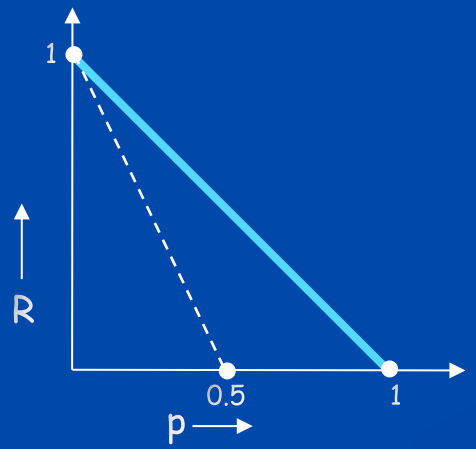
In both cases we prove upper and lower bounds:

- Achievability is efficient (encoding and decoding).

Large q.

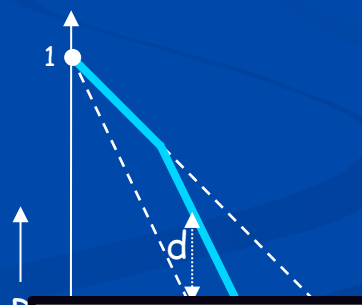
Additive error

- Turns out that for any delay $d>0$ the adversary is very weak.
- Namely, capacity equals that of random channel: $1-p$.



Overwrite error

- Differs substantially from additive case.
- Capacity depends on d.
- As expected, when d is larger the capacity is greater "=" $1-2p+d$.
- Cutoff at $p=0.5$.

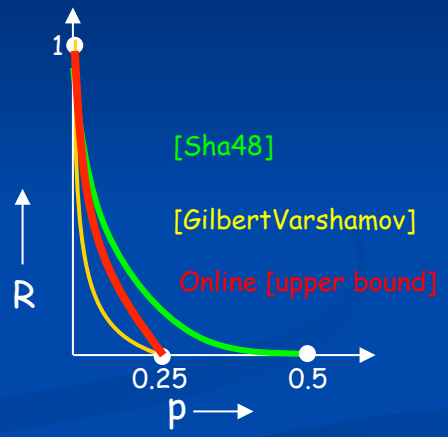


No delay:
Online = unlimited adv.

Summary: no delay

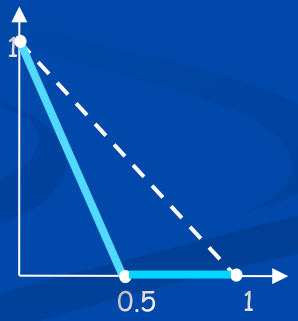
Binary:

- Capacity does not equal random channel (upper bound).
- Capacity seems to differ from adversarial channel (improved GV).



Large alphabets:

- Capacity equals to adversarial channel.



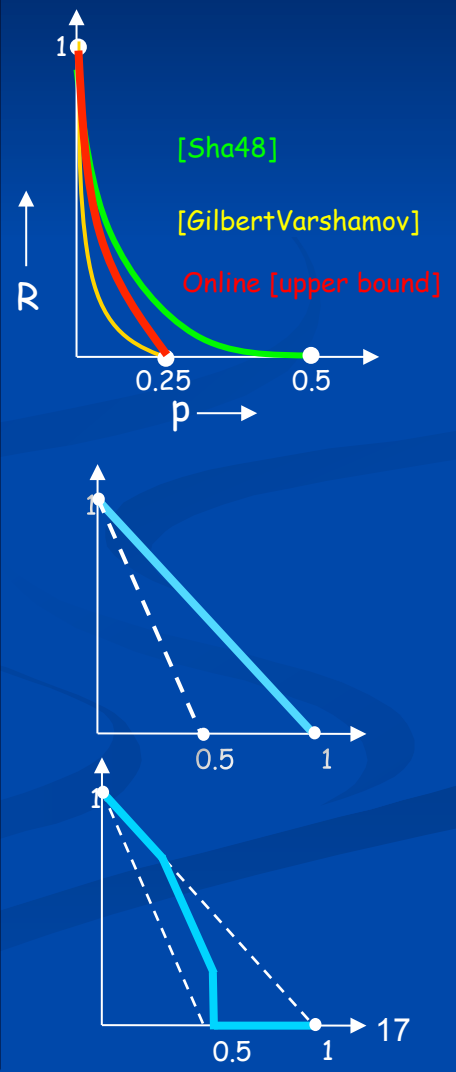
Summary: with delay

"Retrospect":

- Intriguing model of study with "unexpected" behavior.
- Being online does not seem to be that much of restriction to the adversary.
- Delay plays a significant role in capacity.
- "The present is more important than future".

Rest of talk:

- Will give a flavor of a few proof techniques.
- Present:
  - Upper bound for binary case (no delay).
  - Achievability for large alphabets with delay in additive model.
  - Achievability for large alphabets with delay in overwrite model.



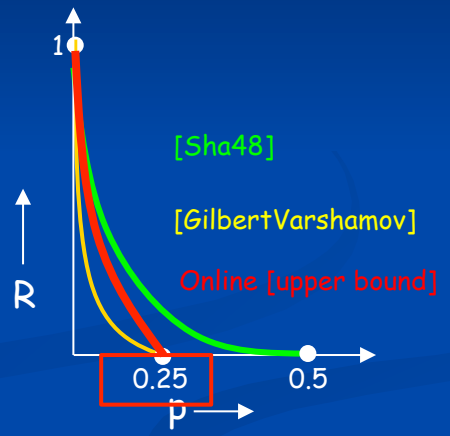
Related work

- To the best of our knowledge, communication in the presence of an online adversary (with or without delay) has not been explicitly addressed in the literature.
- Nevertheless, the model of online channels, being a natural one, has been "on the table" for several decades.
  - Appears as open question in book of Csisz´ar and Korner. In chapter of AVC's (arbitrarily varying channels).
- Variants of causal adversaries that have been defined/studied:
  - [BlackwellBreimanThomasian], [Csisz'arKorner], [JaggiL.HoEffros], [SahaiMitter], [Sarwate], [NutmanL.].
- Would be glad to hear of previous work.

Proof of binary upper bound

- Will show R = 0 for p=0.25.
- This matches the bound

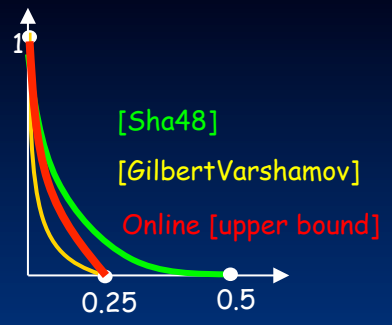
for "unlimited adversaries" and separates from random jammer.



Proof overview:

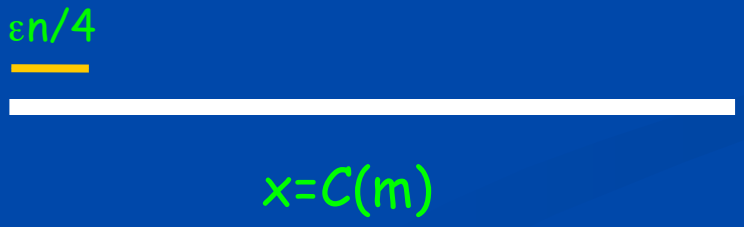
- Need to present adversarial strategy.
- Two step plan "wait and push":
  - "Wait" and listen to gather information.
  - Make a decision and "push" codeword in certain direction.

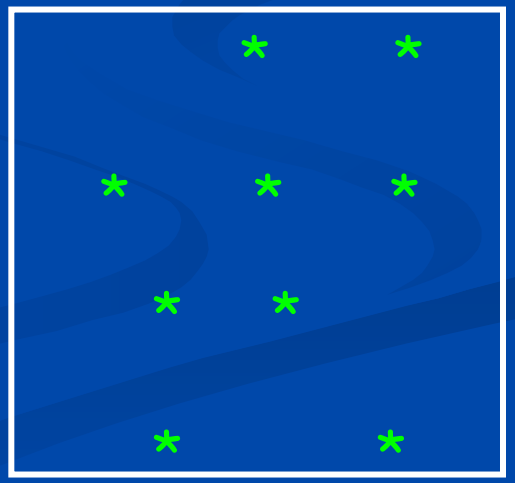
"Wait and push"



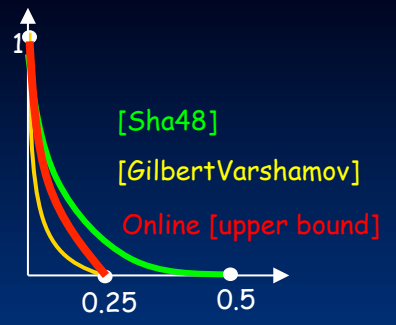
- Would like to prove that $R = 0$ for $p = 0.25$.
- Will show that using any code (encoder and decoder) of rate $\varepsilon > 0$ will imply a decoding error of at least $\approx \varepsilon$.
- Phase I: "wait"
  - Adversary just listens for a short while: say $\varepsilon n/4$ bits.
  - Constructs the set of codewords that are consistent with view.
  - * is the actual codeword transmitted.

Rate $\varepsilon = k/n$:
#codewords = $2^{\varepsilon n}$



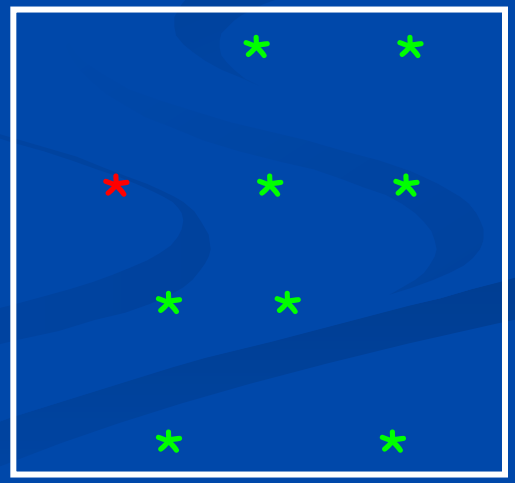


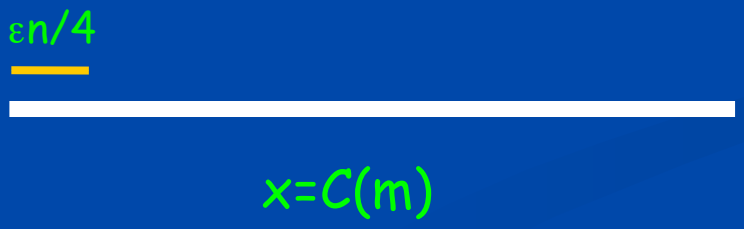
"Wait and push"



- Would like to prove that $R = 0$ for $p = 0.25$.
- Will show that using any code (encoder and decoder) of rate $\varepsilon > 0$ will imply a decoding error of at least $\approx \varepsilon$.
- Phase I: "wait"
  - Adversary just listens for a short while: s… /4 bits.
  - Constructs the set of codewords that c… consistent with view.
  - * is the actual codeword transmitted.
  - Claim 1: w.h.p. set is of size $2^{\Omega(\varepsilon n)}$.
  - Now pick a random codeword * from set.

Averaging argument.
Follows from rate = ε.





"Wait and push"

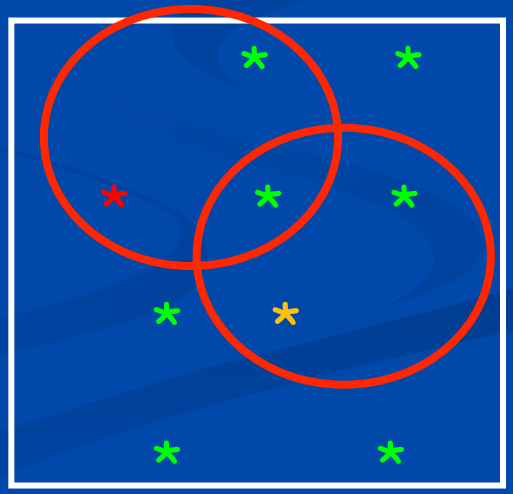
- Would like to prove that R = 0 for p=0.25.
- Will show that using any code (encoder and decoder) will imply a decoding error of at least $\approx \varepsilon$.
- Phase I: "wait"
  - Adversary just listens for a short while: s[illegible] bits.
  - Constructs the set of codewords that c[illegible]sistent with view.
  - * is the actual codeword transmitted.
  - Claim 1: w.h.p. set is of size $2^{\Omega(\varepsilon n)}$.
  - Now pick a random codeword * from set.
  - Claim 2: w.p. $\approx \varepsilon$: dist(*,*) < 2p = $\frac{1}{2}$.
  - Assume Claims 1, 2 hold.

$\varepsilon n/4$

x=C(m)

[Sha48]

- Plotkin: no large set of codewords that are mutually far apart.
- Turan's Theorem: must be many close pairs of codewords.

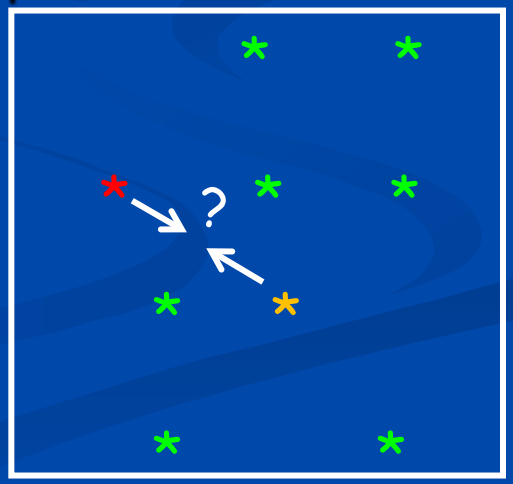


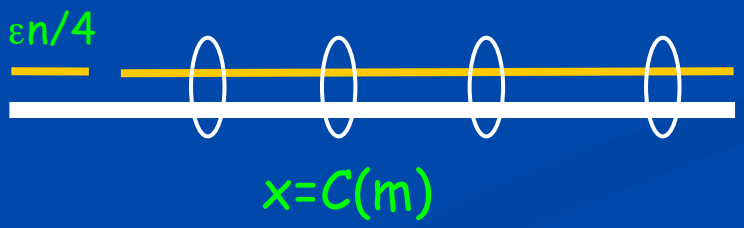
Theorem:
Any code (encoder and decoder) of rate $\varepsilon > 0$ will imply a decoding error of at least $\approx \varepsilon$.

- Would like to prove that $R = 0$ for $p=0.25$.
- Will show that using any code (encoder and
  imply a decoding error of at least $\approx \varepsilon$.
- Phase I: "wait" (* = actual codeword, * = rando picked by adv.)
  - Claim 1: w.h.p. set is of size $2^{O(\varepsilon n)}$.
  - Claim 2: w.p. $\approx \varepsilon$: dist(*,*) < $2p = 1/2$.
- Phase II: "push"
  - Each entry in * that differs from *: p w.p. $\frac{1}{2}$.
  - This pushed * towards *.
  - Claim 3: when Y receives corrupted word, cannot decide whether * or * were sent.

- In both cases, distribution Y views is exactly the same.
- Formally need Bayes' theorem (and few other ideas).

All in all, adv. forces error of $\approx \varepsilon$ (Claims 1,2 must hold).





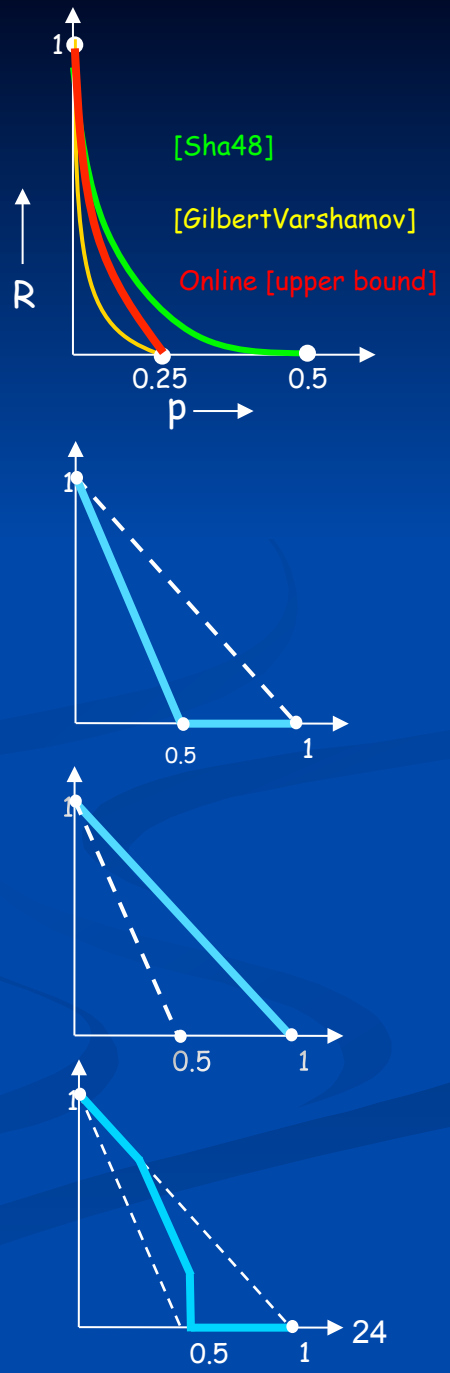
Breather …

Just seen:

- Upper bound for binary alphabets (no delay).
- Same technique gives upper bound for large alphabets.

Large alphabets with delay:

- Additive: single character of delay = random jammer.
- Overwrite: capacity somewhere between random and unlimited adv.

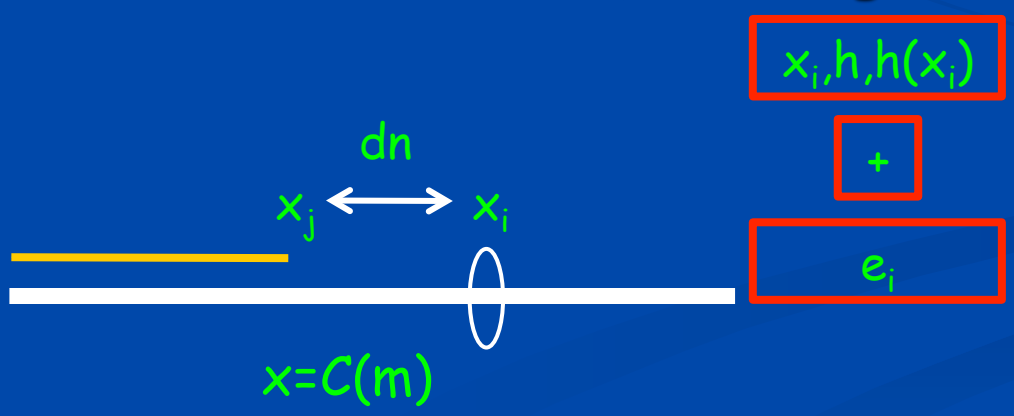


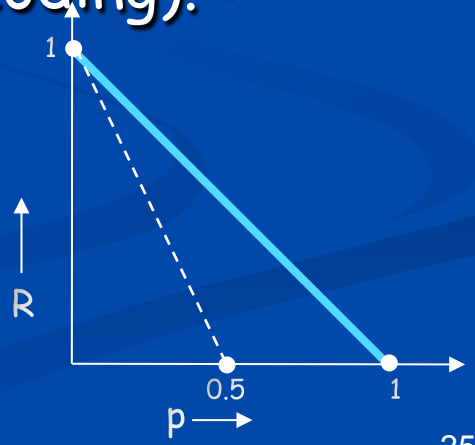
Main idea: use codes for an erasure channel:

- Let $m=m_1 \ldots m_k$ be X's message.
- Encode m using an erasure code (RS for example).
- To each symbol x_i of codeword add an authentication mechanism.
- Namely, to each symbol x_i will add a pair $(h,h(x_i))$.
- h is will be drawn independently by X from a family H of hashes.
- Design H in such a way that:
  - Easy for Y to authenticate.
  - "Cannot" add an error e_i that will pass authentication.
- So after authentication, Y uses erasure decoding.

Model: delay > 0, errors additive.

- $1-p$ for any $d > 0$.
- As this is the capacity of random channel - we only need a lower bound (encoding + decoding).



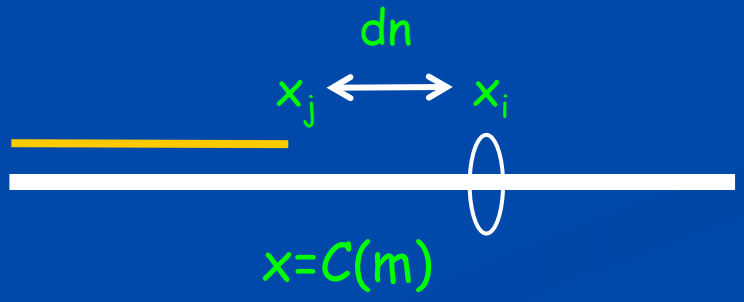


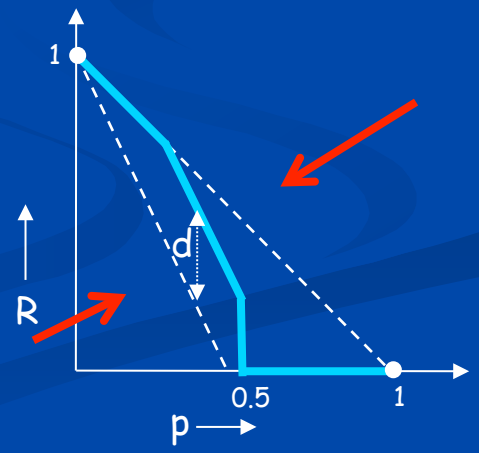
Why won't previous scheme work?

- Adding authentication info. in each packet to get an erasure channel.
- Overwrite adv.: can put in fake packet that will pass authentication.
- Need new ideas …

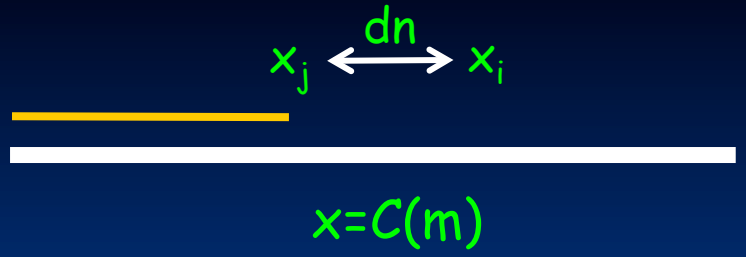
- Model:
  - Errors: overwrite.
  - Delay: decide on e_i based on x_j, $j \le i-dn$.
- Capacity:
  - 0 if $p > \frac{1}{2}$.
  - 1-p if $p < d$.
  - 1-2p+d if $p \ge d$.

Differs from capacity of 1-p in additive case when d>0.





Lower Bound

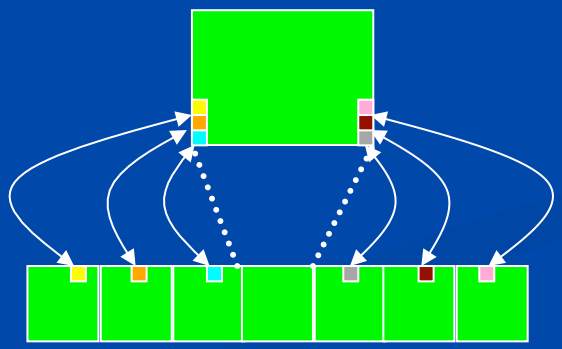


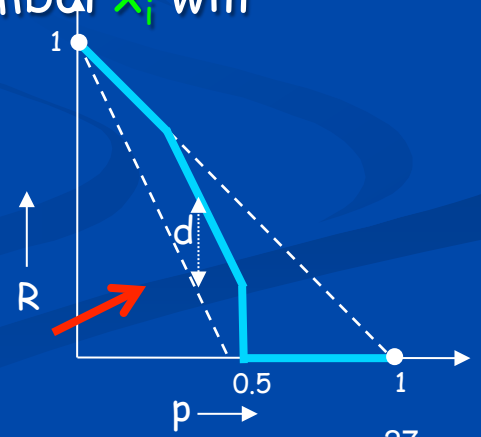
Model: errors = overwrite, delay = e_i based on x_j, $j \le i-dn$.

Lower bound: $1-p$ if $p < d$, $1-2p+d$ if $p \ge d$

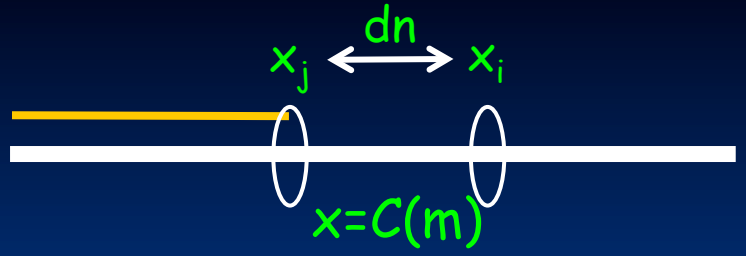
Variant on reduction to erasure codes. More involved.

- Let $m=m_1 \ldots m_k$ be X's message.
- Encode m using an erasure code (RS for example).
- To each symbol x_i of codeword add authentication mechanism.
- This time authentication information added to symbol x_i will include information from all other symbols x_j!



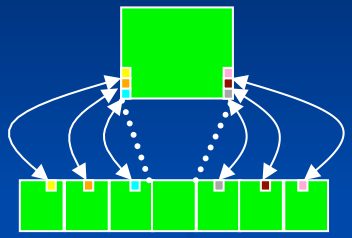


Lower Bound



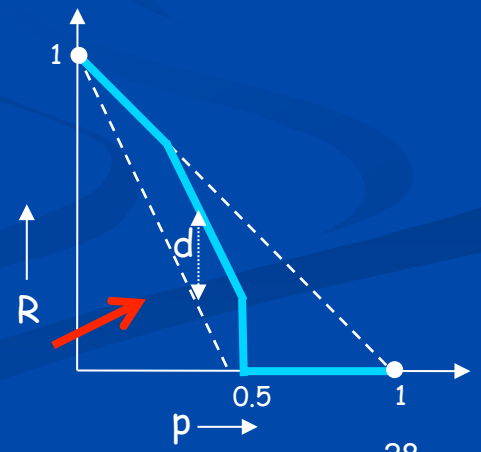
Model: errors = overwrite, delay = e_i based on x_j, $j \le i-dn$.

Lower bound: $1-p$ if $p < d$, $1-2p+d$ if $p \ge d$

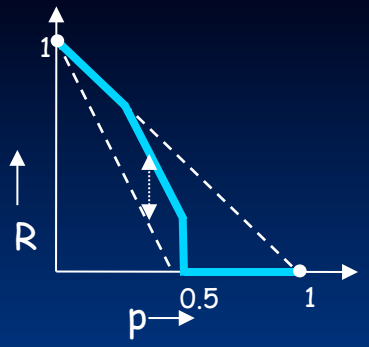


Variant on previous idea that reduces to erasure codes.

- Let $m=m_1 \ldots m_k$ be X's message.
- Encode X using an erasure code (RS for example).
- This time authentication information in symbol x_i will include information from all other symbols x_j!
- Enables pairwise authentication (x_i,x_j).
- Corrupted info will pass pairwise test only if:
  - Both x_i and x_j are corrupted.
  - x_i corrupted after adv. knows value of x_j.
- Need to use pairwise independent hash family.
- How to decode?

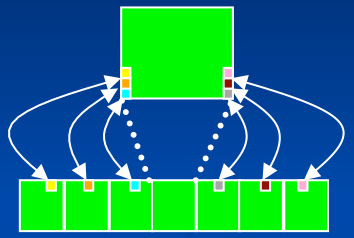


Lower Bound



Lower bound: 1-p if p < d, 1-2p+d if p ≥ d

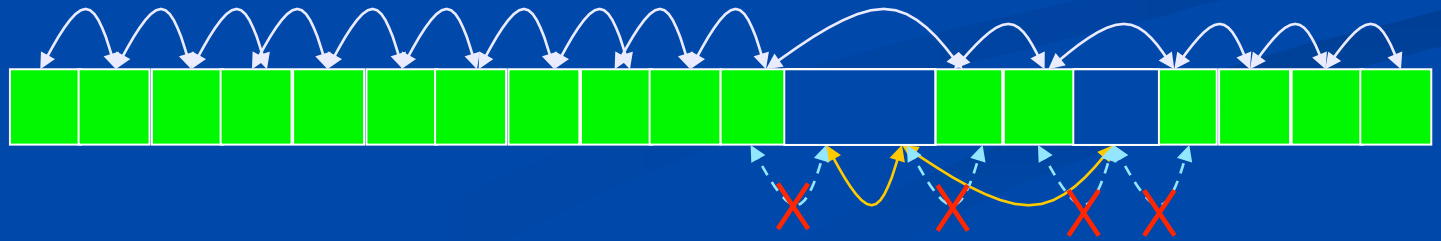
Variant on previous idea that reduces to erasure codes.



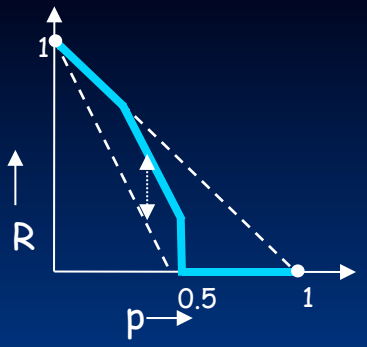
- Corrupted info will pass pairwise test only if:
  - Both x_i and x_j are corrupted.
  - x_j corrupted after adv. knows value of x_i.
- Use pairwise independent hash family.

Close = distance at most dn.
- Will not enable authentication of corrupted and uncorrupted pair.

- How to decode?
- Case 1: p < d.
- Main idea: construct "chains" of consistent "close" pairs.
- Largest chain will be of length (1-p)n and thus allow decoding.

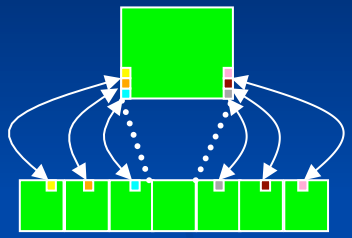


Lower Bound



Lower bound: 1-p if p < d, 1-2p+d if p ≥ d

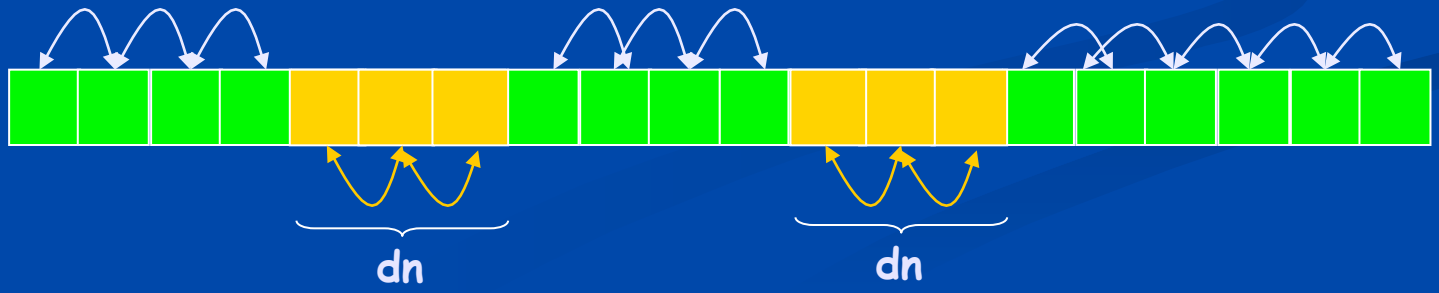
Variant on previous idea that reduces to erasure codes.



- Corrupted info will pass pairwise test only if:
  - Both x_i and x_j are corrupted.
  - x_j corrupted after adv. knows value of x_i.
- Use pairwise independent hash family.

Close = distance at most dn.
- Will not enable authentication of corrupted and uncorrupted pair.

- How to decode?
- Case 2: p ≥ d.
- As before: construct chain of mutually consistent "close" pairs.
- Chain may be disconnected!



Which connected components are good ones?

- Need to construct at least (1-2p+d)n uncorrupted entries to decode.
- Check consistency among all possible chain combinations ⇒ expensive.

- Turns out that one can prove:
  - Not too many correct (green) chains ≤ p/d.

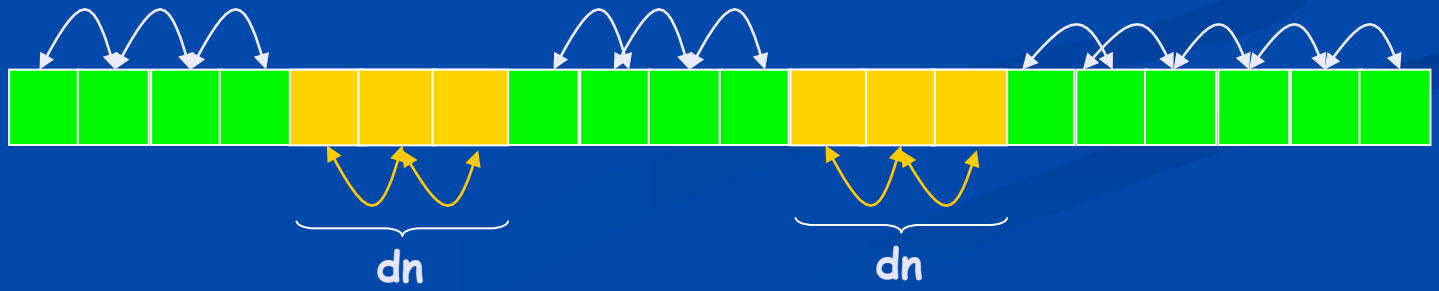
Rate: 1-2p+d

- Put it all together ⇒ limited exhaustive search + erasure decoding!

  - Both x_i and x_j are corrupted.
  - x_j corrupted after adv. knows value of x_i.
- Use pairwise independent hash family.

Close = distance at most dn.
- Will not enable authentication of corrupted and uncorrupted pair.

- How to decode?
- Case 2: p ≥ d.
- As before: construct chain of mutually consistent "close" pairs.
- Chain may be disconnected!



Summary/thoughts

- Theme: study channels that are somewhere between "unlimited adversarial jammer" and "random noise".
- This talk: online (causal) adversaries.
  - Large alphabets (now) understood, small not fully …
  - Causal adversary still strong…… but delays can weaken him.
  - Types of error matter (add./overwrite/…?).
  - "Present is more important than future".

Thanks!

- May consider other channel models based on theme:
  - Jammer has other limited views of codeword [L].
  - Jammer does not have full knowledge of codebook [Ahlswede][Lipton][MicaliPeikertSudanWilson] [SarwateGastpar].
  - Gaussian additive channel.
  - Causality in network error correction (time vs. topology [Nutman L]).