Ubiquitous Access to Distributed Data in Sensor Networks through Decentralized Erasure Codes

Alex Dimakis
Vinod Prabhakaran
Kannan Ramchandran

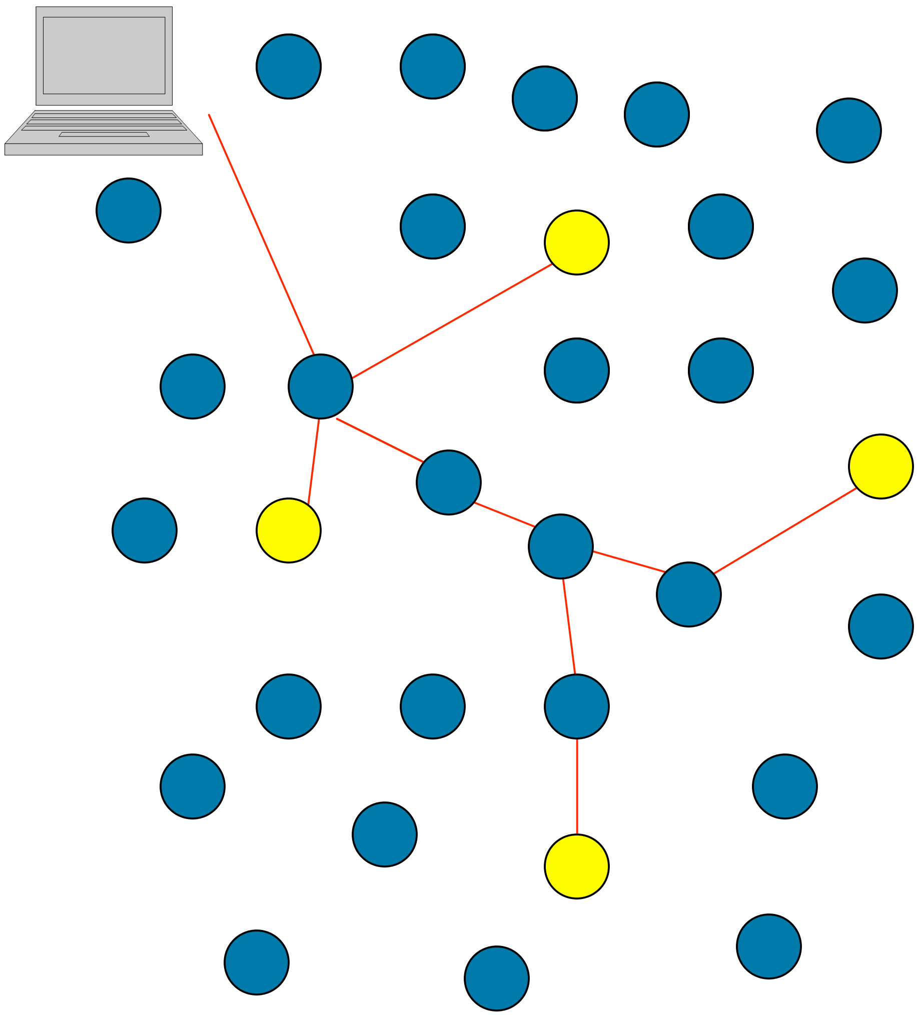
BASiCS group,
EECS department, UC Berkeley

Fourth International Symposium on Information Processing in Sensor Networks

Outline

- Problem description
- Decentralized erasure codes
- Extensions
- Sketch of Theory
- Conclusions – Future work

Problem description - Motivation



Data nodes:

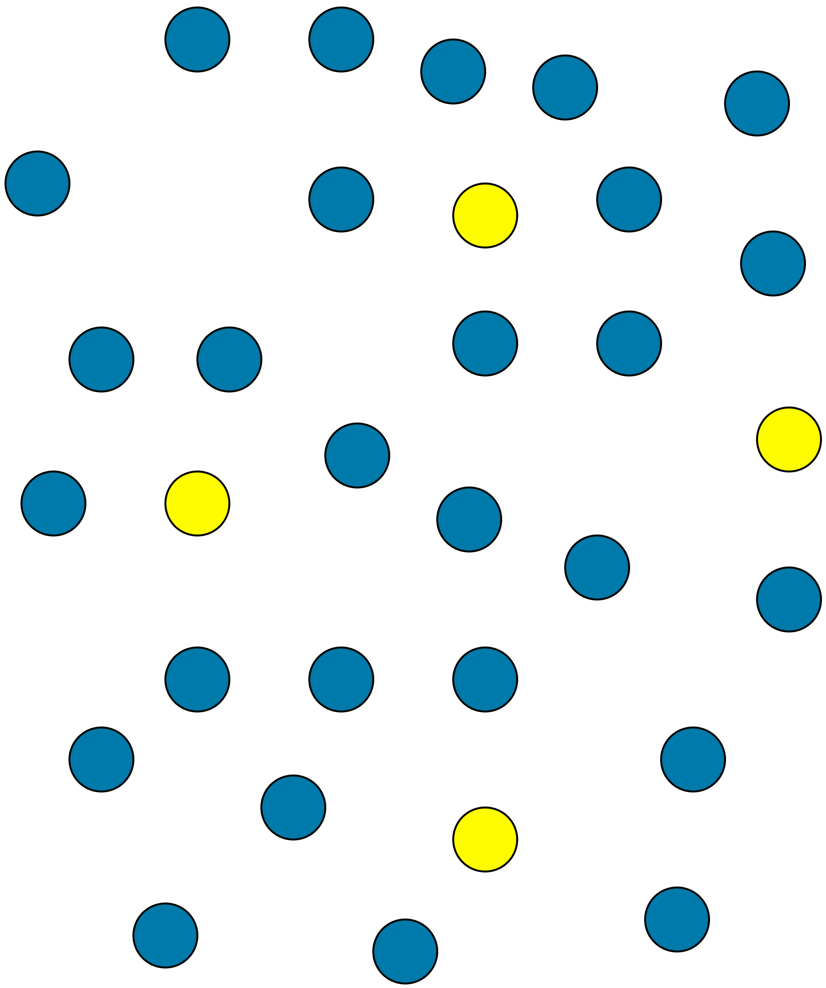
Storage nodes:

Traditional approach:

Pull data

- Find where data is
- Find routes to data
- Introduces latency, unreliability

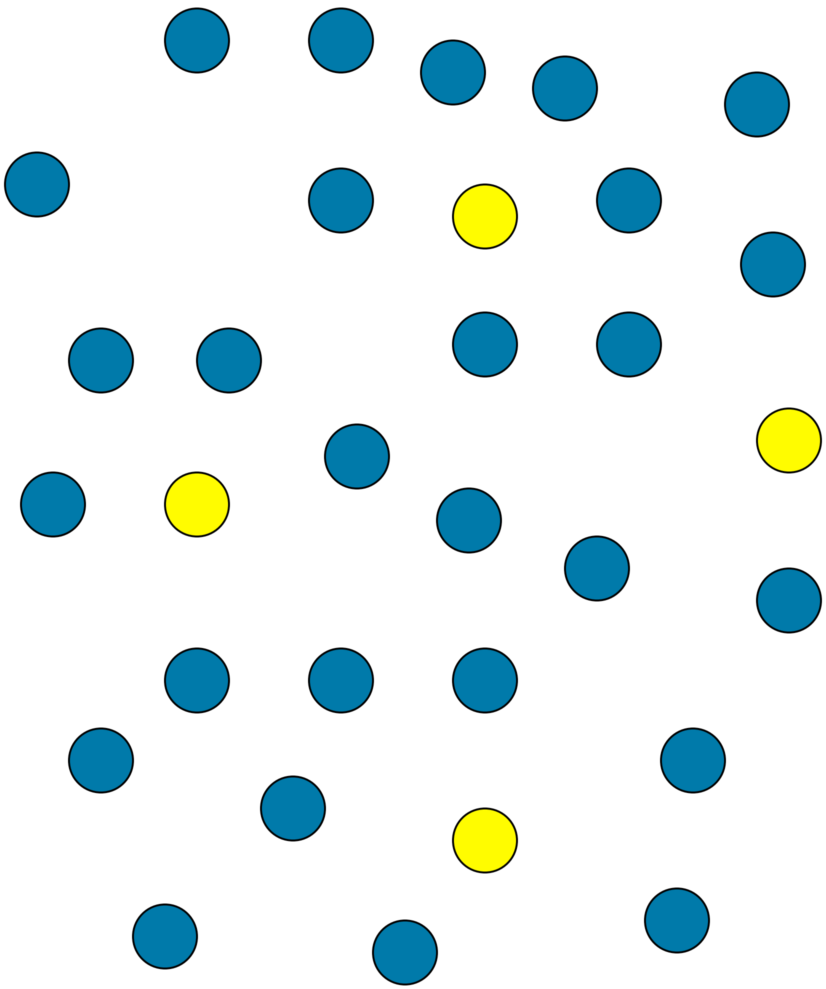
Ubiquitous access to data



Pre-route

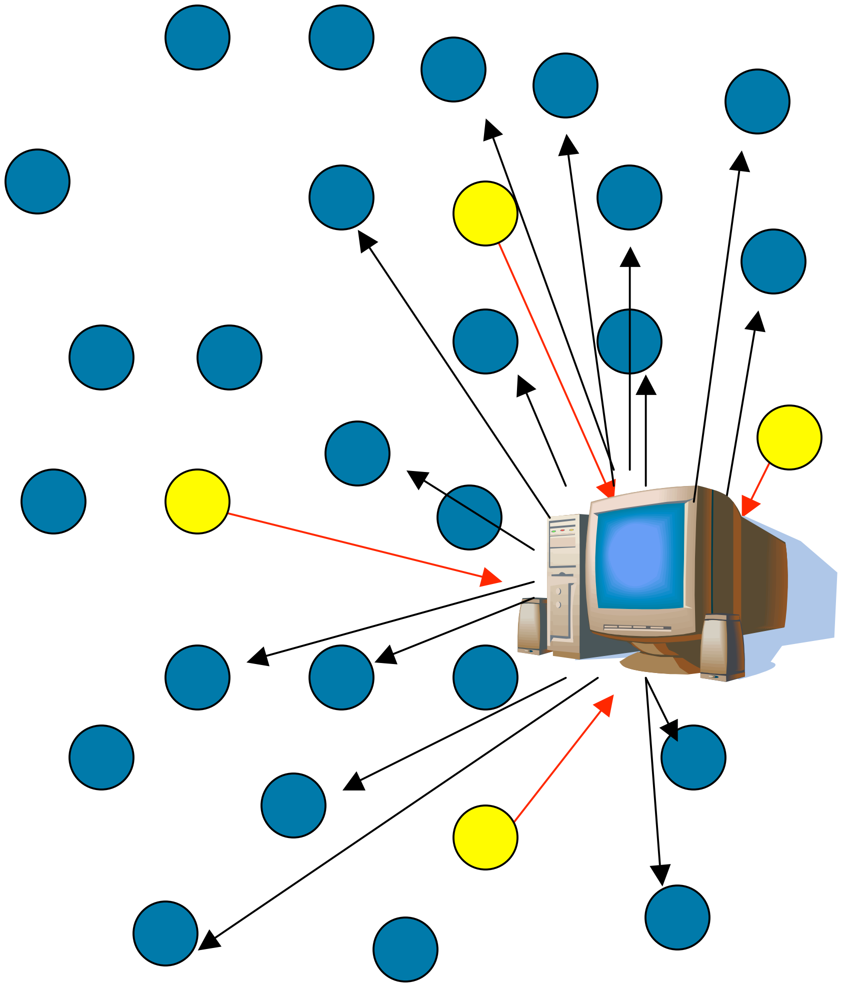
- Make data ubiquitous, network ready for any query, at any location
- Data available everywhere
- Robustness
- Zero latency

Problem statement



- **n** Storage nodes, k Data nodes,
- **k=α·n** data nodes (α<1, fixed)
- **Want to reconstruct original data packets from any k storage nodes**
- **Nodes can store only one packet.**

Centralized solution

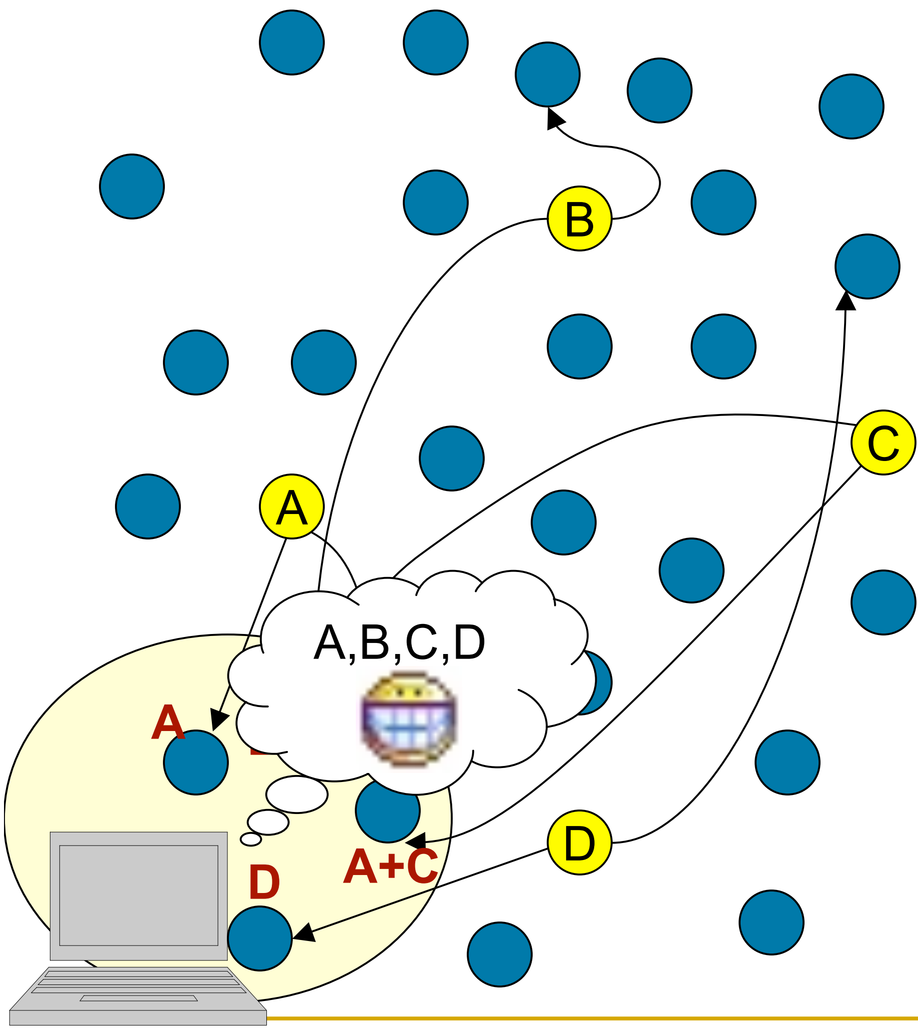


- Gather data in central node
- Central node creates an encoded packet for every node using erasure code (LT code , Reed Solomon)
- Central node routes encoded packets to all storage nodes
- Each storage node stores one erasure packet
- Any k storage nodes have all the data
- **Not scalable**

Want a truly distributed solution.

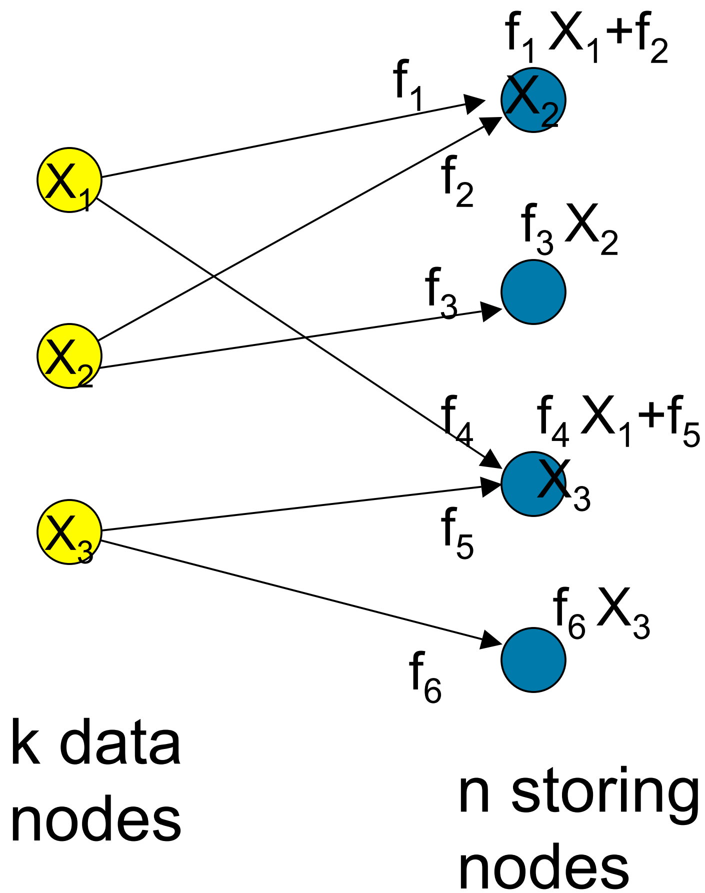
- Require a completely distributed and scalable solution. No centralized processing.
- Uncertain and dynamic environment. No global knowledge, no time coordination, nodes acting independently.
- Rely on a packet routing layer (e.g. Geographic routing) Not jointly optimize across layers.
- **How to build a code if your data is not in one place**

Decentralized Erasure Codes (Example)



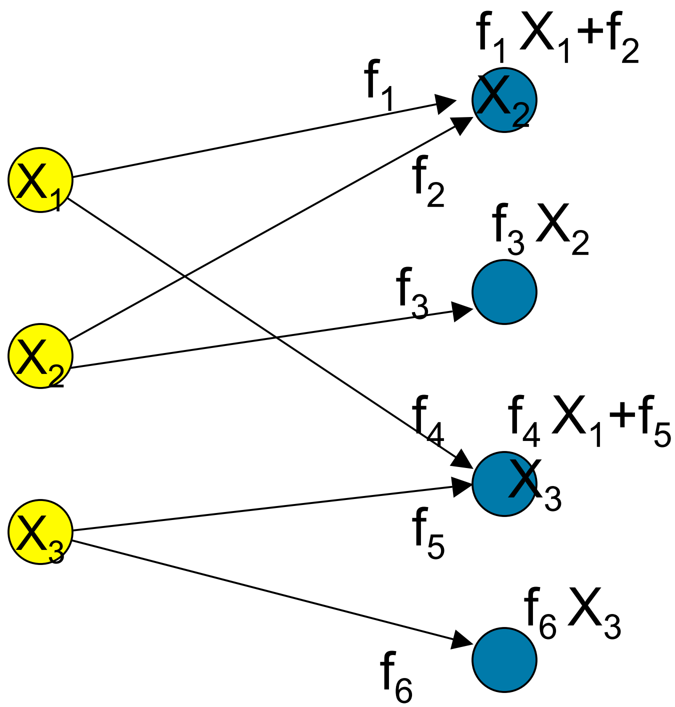
- Each node stores a linear combination in a finite field.
- k random storing nodes are queried.
- Want these k equations to have unique solution.
- **Result:** This is feasible with **random, independent pre-routing**.
- Communication cost:
  Number of pre-routed packets
  (**Data node degree**)

Decentralized Erasure Codes



- Data nodes **pre-route** packets to randomly selected storage nodes.
- Storage nodes select random coefficients f_i from a field F_q
- Store **linear combination** and **coefficients**

Decentralized Erasure Codes



k data nodes

n storing nodes

G'

$$[Y_1 \quad Y_2 \quad Y_3 \quad Y_4] = [X_1 \quad X_2 \quad X_3] \begin{bmatrix} f_1 & 0 & f_4 & 0 \\ f_2 & f_3 & 0 & 0 \\ 0 & 0 & f_5 & f_6 \end{bmatrix}$$

The encoded vector (what is stored) is

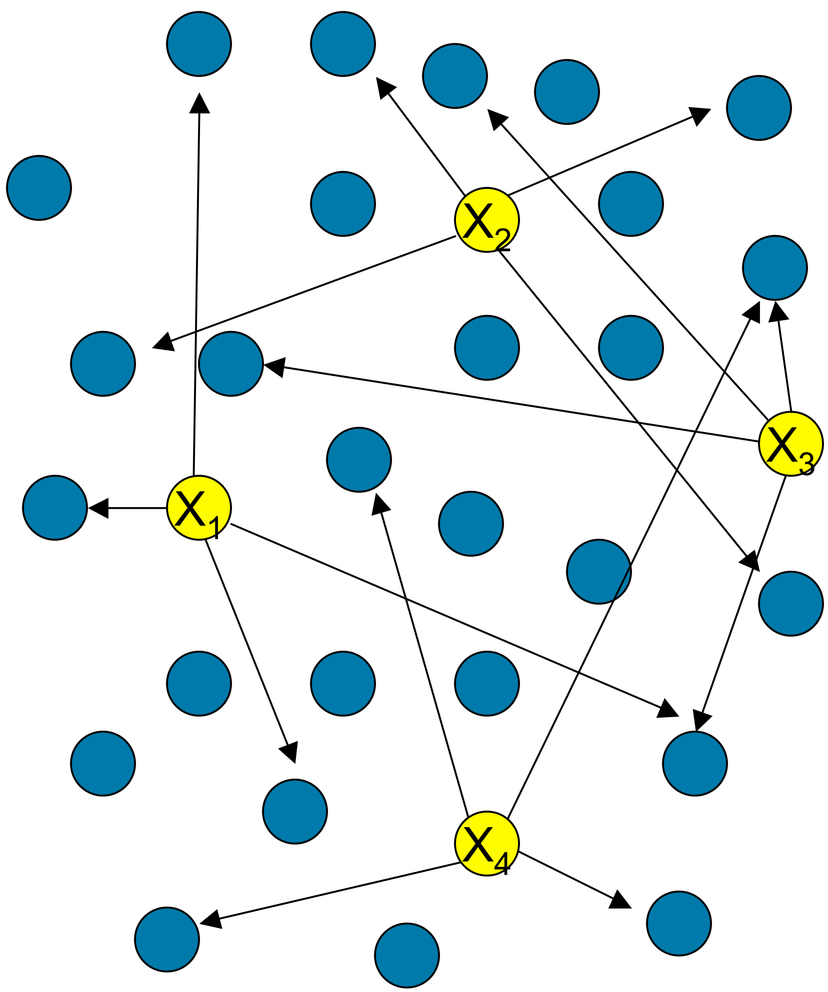
Y=X G.

Each **row of G is generated independently =Decentralized**

Want **G** as sparse as possible.

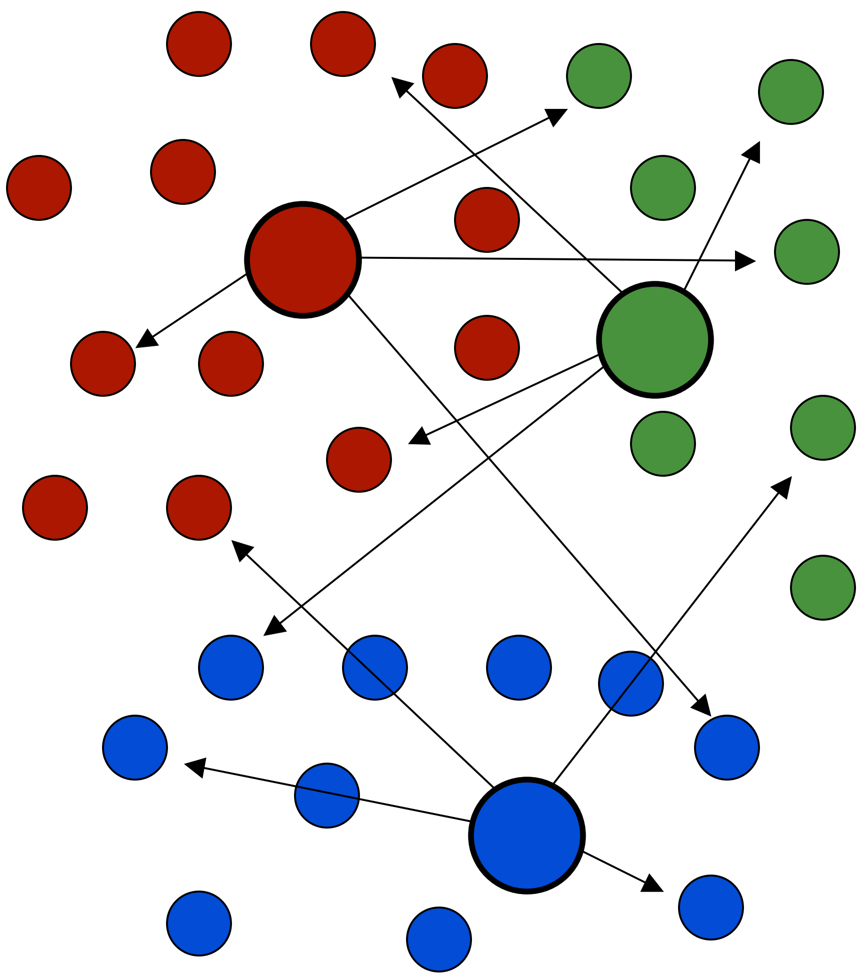
Now assume only storing nodes 1-3 are queried. To reconstruct it suffices to have **G'** to be full rank

Main results



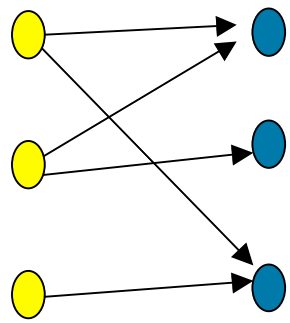
- Communication Cost:
  Data node degree
- Theorem: Degree O(**log n**) is sufficient. (whp)
- Theorem: Ω(**log n**) is necessary for independent data nodes.
- **G'** full rank with probability at least (1-k/q), k : number of data nodes, q: size of finite field=2^u .
- Error probability decreasing exponentially in the number of codeword bits u. (storage overhead)

Multiresolution storage



- Create averages (**headline news**) at each neighborhood
- Nodes that have averages act as data nodes.
- Headline news **are everywhere**, details are localized.
- Can be made in many layers

Proof ideas



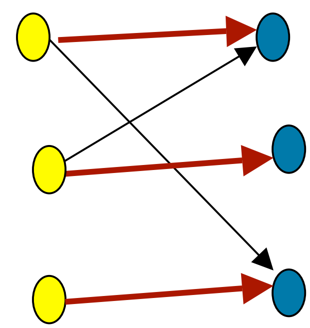
$$[Y_1 \quad Y_2 \quad Y_3 \quad Y_4] = [X_1 \quad X_2 \quad X_3] \begin{bmatrix} f_1 & 0 & f_4 & 0 \\ f_2 & f_3 & 0 & 0 \\ 0 & 0 & f_5 & f_6 \end{bmatrix}$$

G' (the first three columns of the matrix)

- We want G' invertible (det(G') not zero).
- det(G') is a multivariate polynomial of random coefficients f_1, f_2, .. f_L.
- Two cases:
- Case I det(G') could be identically zero (for any f_i) e.g. if one column is all zeros (e.g. when one storing node receives nothing)
- Case II It could just be that f_1, f_2, .. f_L is a root of the determinant polynomial

Proof ideas (Case I: hard case)

- Use Edmonds' Theorem: The determinant is not identically zero iff the random bipartite graph has a **perfect matching**



- We show that random bipartite graphs we introduced have a perfect matching (whp).

- Proof uses modification of technique introduced by Erdös for similar problem (*Each bipartite graph with 2k nodes and **k log(k)** edges assigned randomly has a perfect matching whp*)

- Result might be of independent interest.

Proof ideas (Case II: easy case)

- Case 2) Maybe f_1, f_2, .. f_L is a root of the determinant polynomial

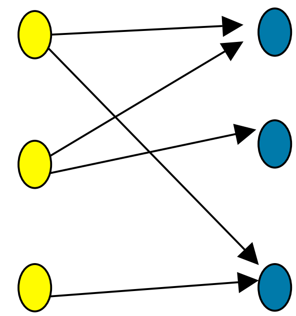
- Intuition: very unlikely to hit a root with a random assignment.

- Formally: Schwartz Zippel theorem:

If $Q(f_1, f_2, \ldots, f_L)$ multivariate polynomial of degree d, and if f_i chosen iid from a finite field F(q) , then
$$\mathbf{P[Q(f_1,f_2, \ldots, f_L)=0 \mid Q \not\equiv 0] \leq d/q}$$

- Easy to see d=k in the determinant polynomial

Related work



- Fountain codes *(Luby 02)*
  Rateless property ~ Decentralized property
- Network Coding *(Ho et al. 03, Li et al. 03, Chou et al. '03, Effros et al. 02, Deb et al.)*
- Distributed storage in Sensor Networks *(Ganesan et al. 03)*
- Information Dispersal Algorithm *(Rabin '89)*

Conclusions – Future Work

- Decentralized Erasure Codes efficiently solve the Distributed Networked Storage problem.
- Solution requires no coordination, centralized processing or global knowledge of locations. Matches the minimal communication required under these constraints.
- Future work involves investigating multiresolution storage, aging of data and jointly optimizing across layers.
- Can be applied in any scenario where we want to create an erasure code from distributed data. (P2P networks, RAID systems)