Random Graphs and All-to-All Communication

Linda Chen
Mentored by Jun Wan

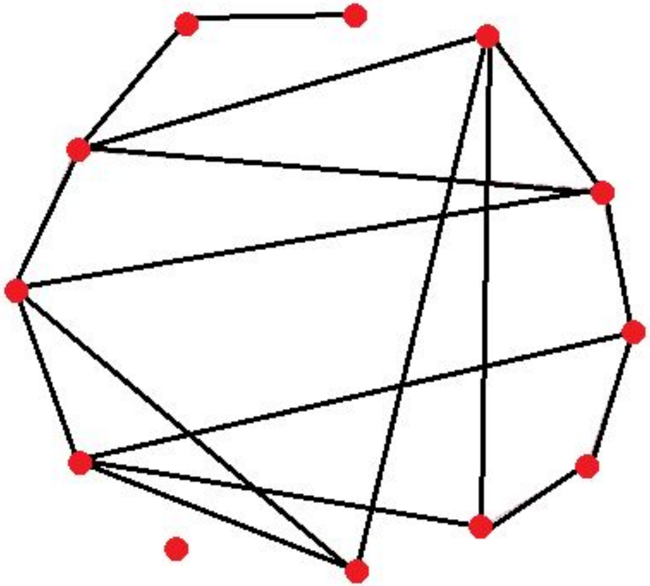
Graphs and Random Graphs

Graph G = (V, E)

V = set of **vertices**, E = set of **edges**

Degree: number of edges coming out of vertex

Random graph: properties are randomly generated



The Problem

Graphs represent a **communication network**, vertices represent **users**

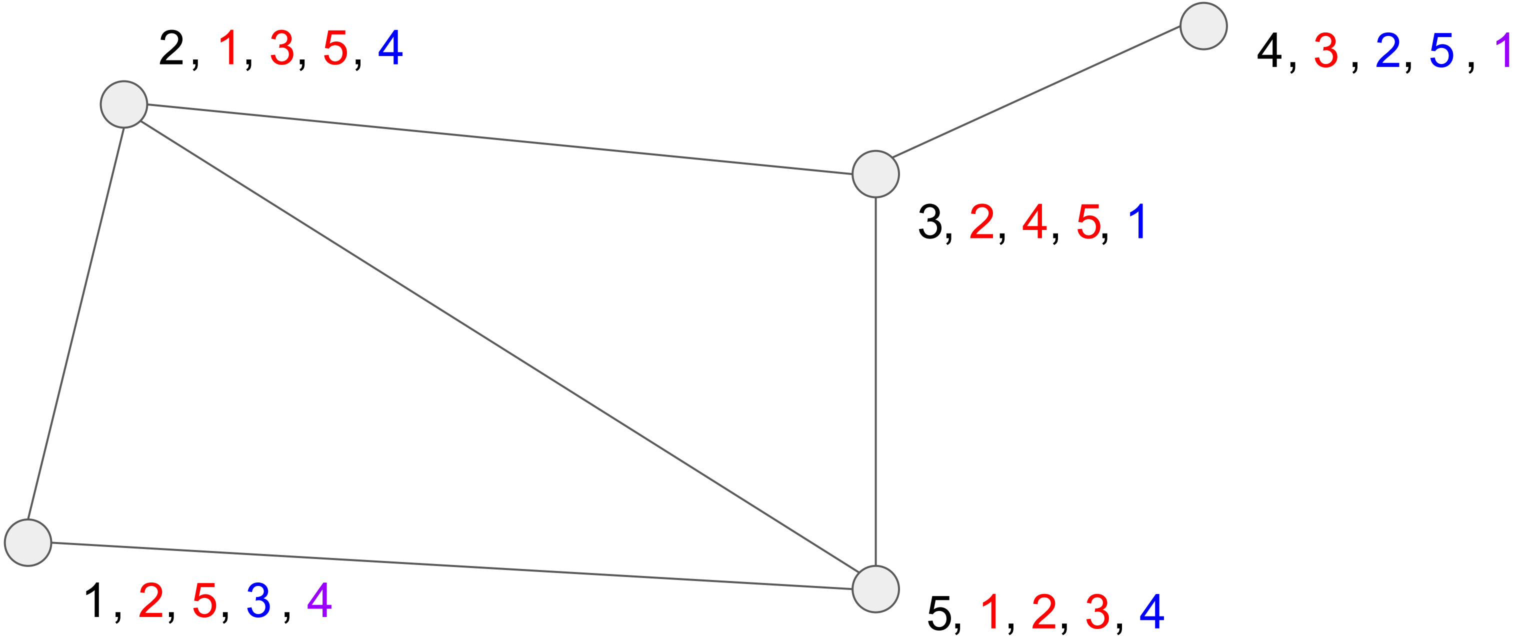
Users exchange **messages**

All-to-all communication: all users exchange with all other users

How to make communication more efficient and require less cost?

- Cryptocurrency
- Consensus protocols
- etc.

Example of Communication



Goals

Using random graphs: reduce number of exchanges from *n* to *d* * (round #)

Part I: compare different random graph models to reduce **round number**: # of rounds needed to receive all messages

Part II: reduce overall **communication cost**: # of bits received by a user

Part I: Comparison of Random Graph Models

Random graph models:

- Model 1: each edge exists with probability p
- Model 2: graph has total of m edges
- Model 3: each vertex has degree d undirected edges
- Model 4: each vertex has degree d directed edges

Giant Component

Giant component: largest connected component of a graph

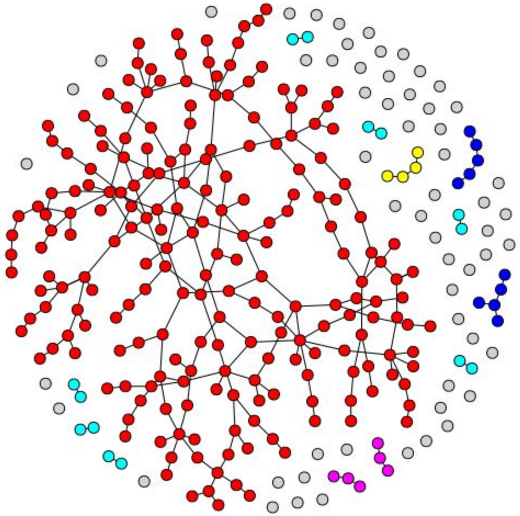
Average degree for the giant component to include more than $(1-\varepsilon)n$ vertices…

Previous results:

- Model 1 (probability *p*): $d > \dfrac{1 - \ln \varepsilon}{1 - \varepsilon}$
- Model 3 (*d* undirected): $d > 1$

Our results:

- Model 2 (*m* edges): $d > \dfrac{2 \ln\left((1-\varepsilon)^{1-\varepsilon} \cdot \varepsilon^{\varepsilon}\right)}{\ln(1 - 2\varepsilon(1-\varepsilon))}$
- Model 4 (*d* directed): $d > 1 + \dfrac{\varepsilon \ln \varepsilon}{(1-\varepsilon)\ln \varepsilon}$



Giant Component Proof Process

Split the set of V vertices into subsets V' and $V - V'$

$$\varepsilon n \leq |V'| \leq (1-\varepsilon)n$$

Find probability that the two subsets are disconnected

Apply a union bound for all subsets V'

Determine what d must be in order for this probability to be negligible

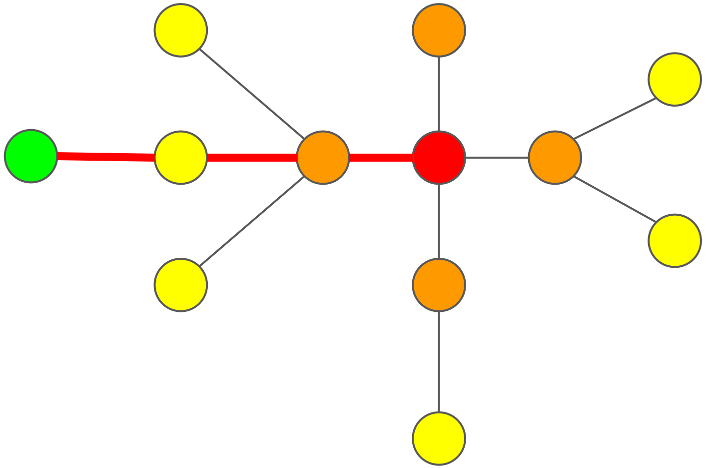
Diameter and Round Number

Diameter longest shortest path between any two vertices of the graph

$\delta=6$

Diameter = round number

Round *i*: users receive messages from users that are a distance *i* from them



Diameter

For each user to receive…

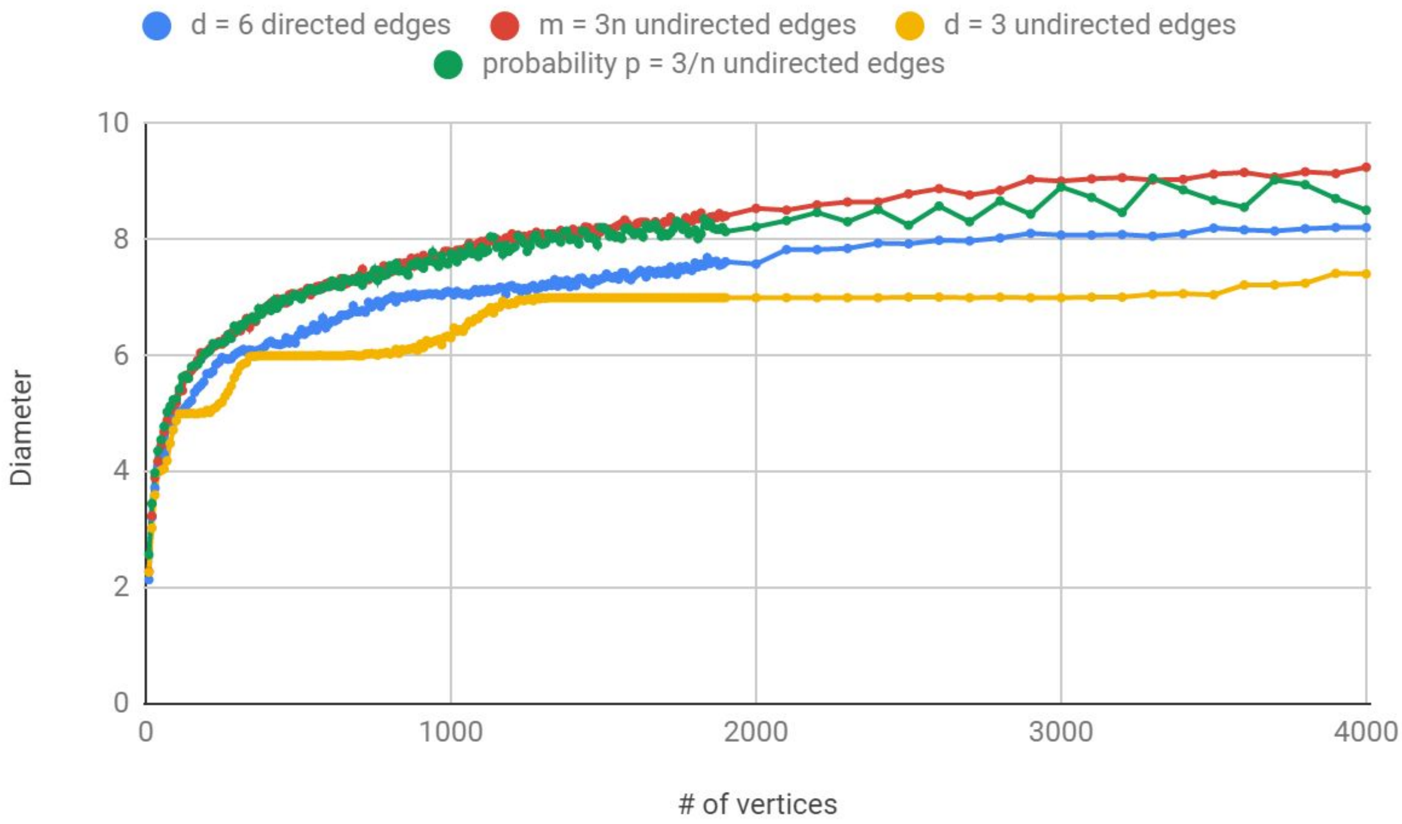
From 1 to log(n) messages: log(n) rounds

From log(n) to 0.1n messages: $\log(\frac{0.1n}{\log n})$ rounds

From 0.1n to (1−ε)n messages: O(1) rounds

Upper bound of diameter = $\log n + \log(\frac{0.1n}{\log n}) + O(1)$

Diameter



d = 6 directed edges
m = 3n undirected edges
d = 3 undirected edges
probability p = 3/n undirected edges
10
8
6
4
2
0
Diameter
0
1000
2000
3000
4000
# of vertices

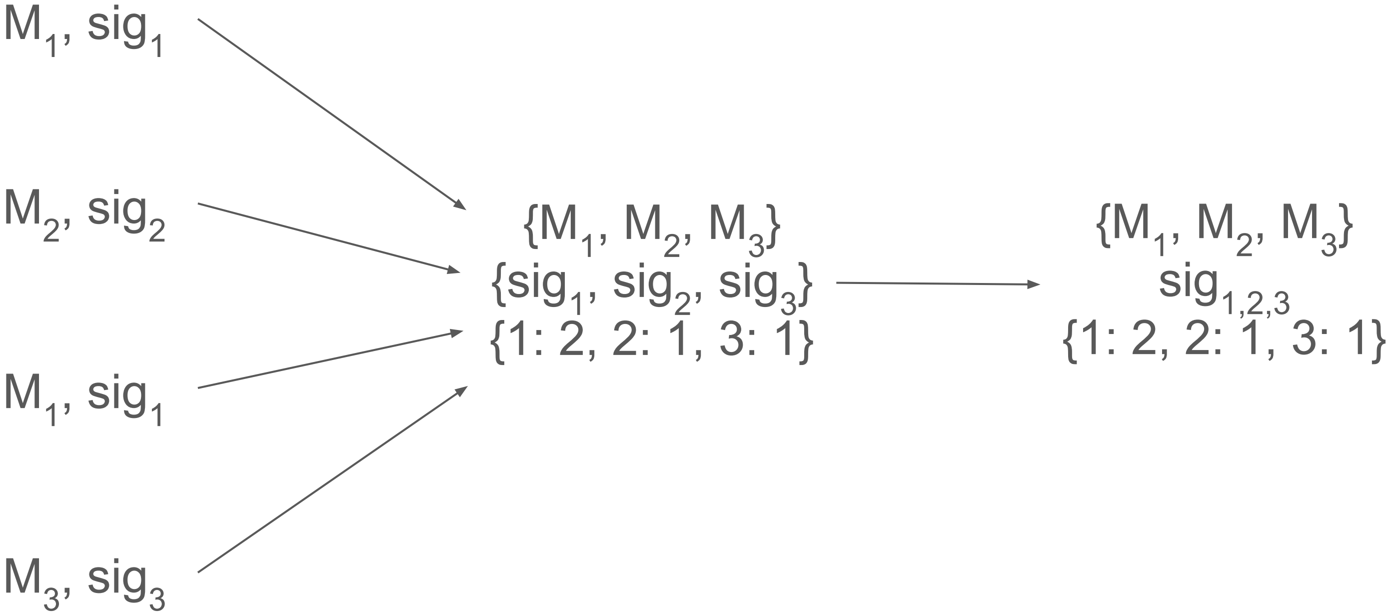
Part II: Communication Cost

Communication cost: total number of bits received by a user

Each round, users send to each other an **aggregate signature**

- Consists of **message set**, **signature**, and **multiset** storing components
- Aggregates signatures from multiple distinct users into one signature

Aggregate Signatures



Protocol

Randomly generate graph G = (V, E)

n users each start with their own message and signature on that message

For 1 to *k* (round number) rounds, each user...

- Exchanges messages with *d* neighbors
- Verifies messages using aggregate signature
- Updates their current messages and aggregate signature with the new messages received

Communication Cost

Using aggregate signatures, **signature cost** is reduced to (# of rounds) * (degree) * (sig size)

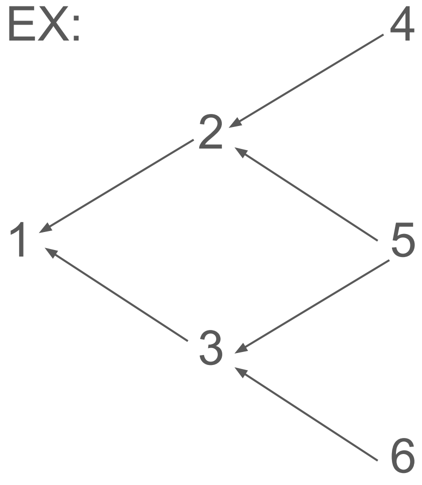
Less than the **message cost**, so we can just focus on the messages when considering communication cost

Communication Cost - Messages

A user's set of messages can be expressed as multisets

> **Multiset**: a modification of a set that can have multiple instances of the same element

EX:



User 1's multisets:

Start: {1}

Round 1: {2, 3}

Round 2: {4, 5, 5, 6}

Communication Cost - Messages

Multisets assigned numbers in order of probability of appearing

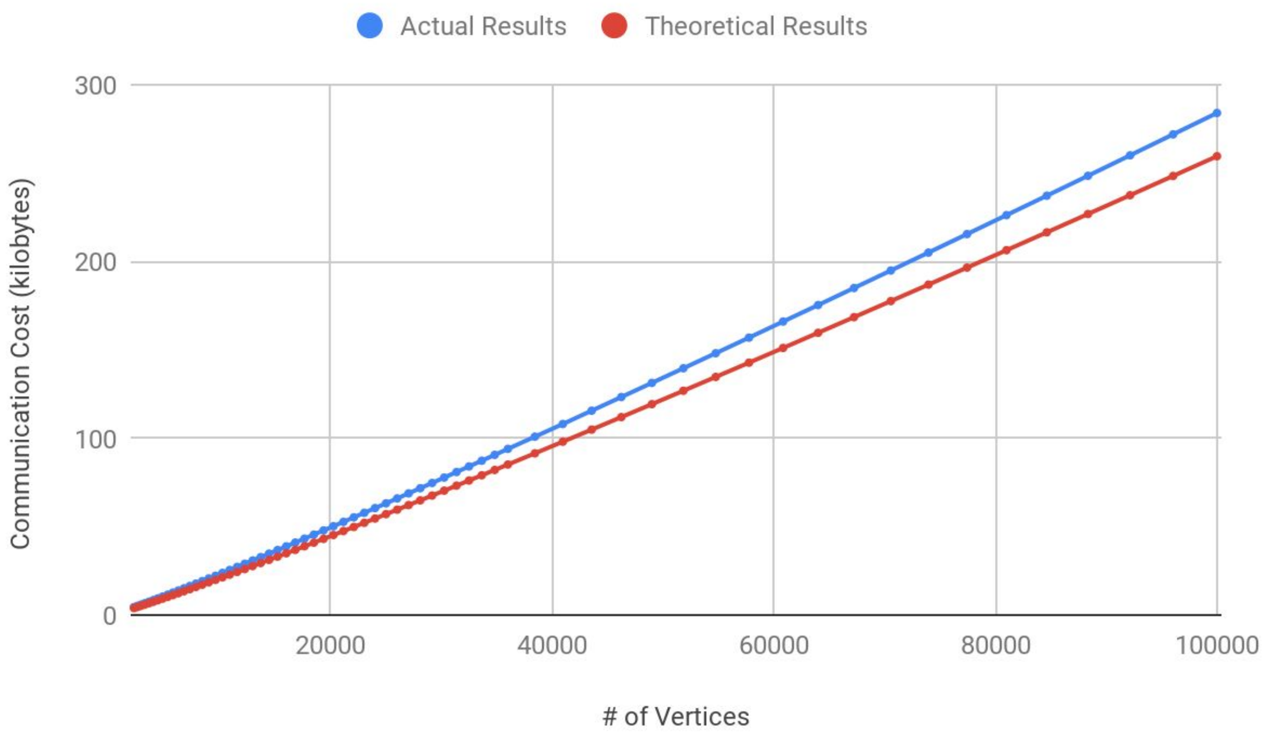
EX: {1, 2, 3, 4} is assigned a smaller number than {2, 2, 2, 2}

Reduces communication cost: more likely to send smaller numbers (less bits)

Final cost: $\frac{|\ln \varepsilon|\, n \log n}{k}$

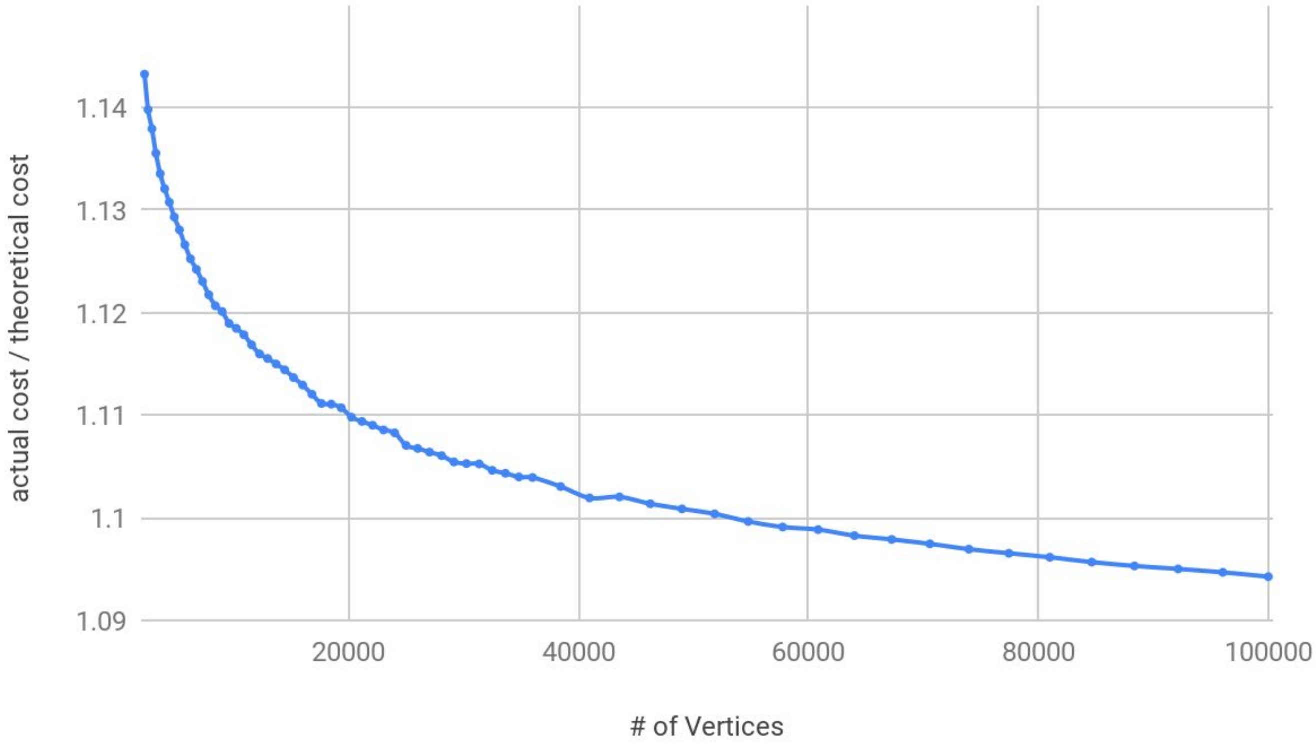
k = round number

Communication Cost



Communication Cost

As *n* gets bigger, the ratio between actual cost to theoretical cost gets smaller



Adversaries

Crash model: each user fails with probability p

Is similar to original model, but with reduced degree

When generating graph, increase degree by a factor of $\frac{1}{1-p}$

Can still follow original method of assigning numbers to multisets

Open questions - What else can the adversary do?

Conclusion

Found "good" model of random graph: minimizes diameter and maximizes giant component size

We show an all-to-all communication protocol with:

$\log n + \log(\frac{0.1n}{\log n}) + O(1)$ # of rounds

$\frac{|\ln \varepsilon| \, n \log n}{k}$ communication complexity

In contrast, previous work does:

$|\ln \varepsilon| \, n \log n$ communication complexity

Thank you!

Questions?