The Small World Phenomenon:

An Algorithmic Perspective

Speaker: Bradford Greening, Jr.
Rutgers University – Camden

An Experiment by Milgram (1967)

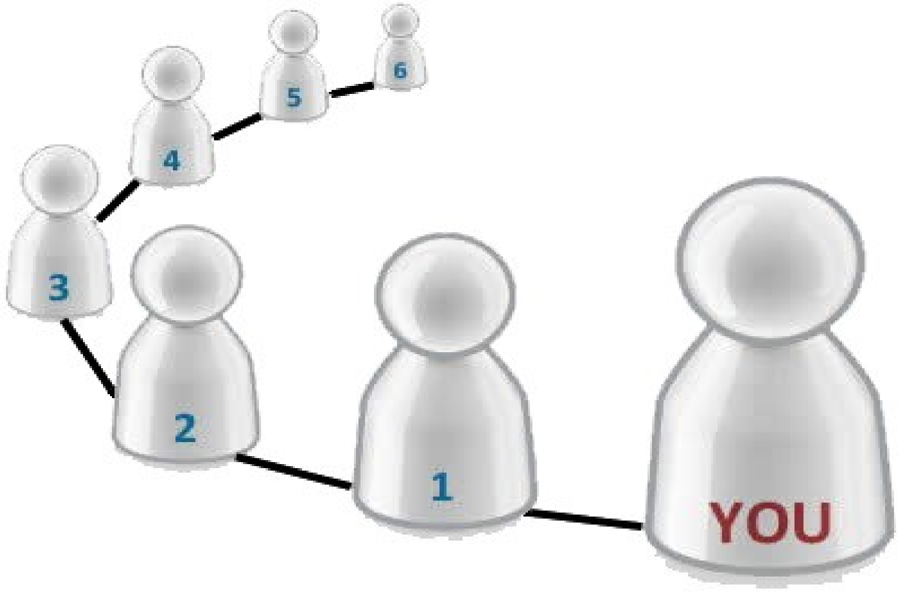
- Chose a target person
- Asked randomly chosen “starters” to forward a letter to the target
  - Name, address, and some personal information were provided for the target person
  - The participants could only forward a letter to a single person that he/she knew on a first name basis
  - Goal: To advance the letter to the target as quickly as possible

An Experiment by Milgram (1967)

- Outcome revealed two fundamental components of a social network:
  - Very short paths between arbitrary pairs of nodes
  - Individuals operating with purely local information are very adept at finding these paths

What is the “small world” phenomenon?

- Principle that most people in a society are linked by short chains of acquaintances
- Sometimes referred to as the “six degrees of separation” theory



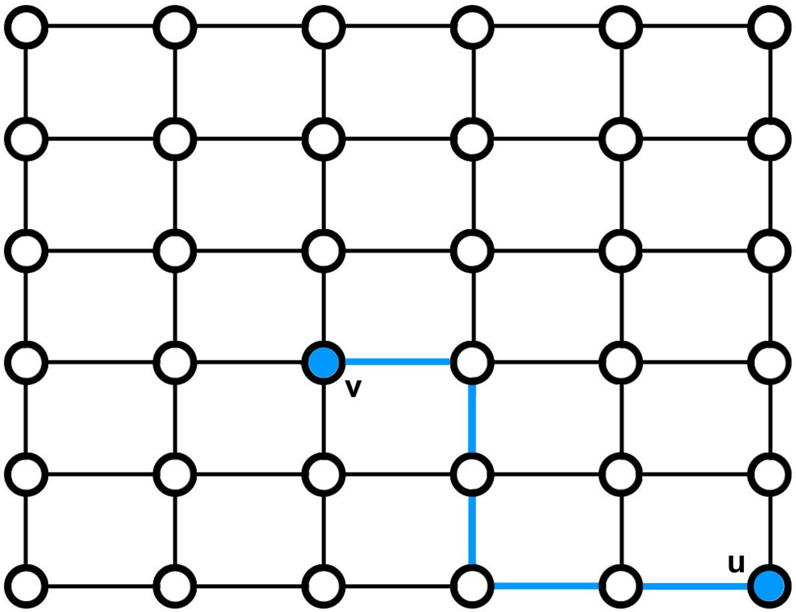
Modeling a social network

- Create a graph:
  - node for every person in the world
  - an edge between two people (nodes) if they know each other on a first name basis
- If almost every pair of nodes have “short” paths between them, we say this is a small world

Modeling a social network

- Watts – Strogatz (1998)
  - Created a model for small-world networks
    - Local contacts
    - Long-range contacts
  - Effectively incorporated closed triads and short paths into the same model

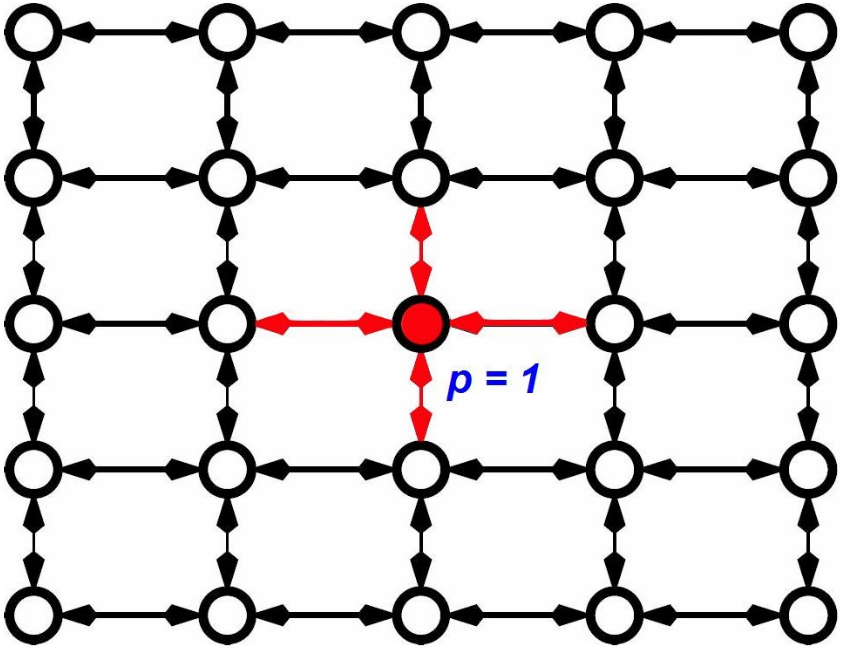
Modeling a social network



- Imagine everyone lives on an *n* x *n* grid
- “lattice distance” – number of lattice steps between two points
- Constants *p,q*

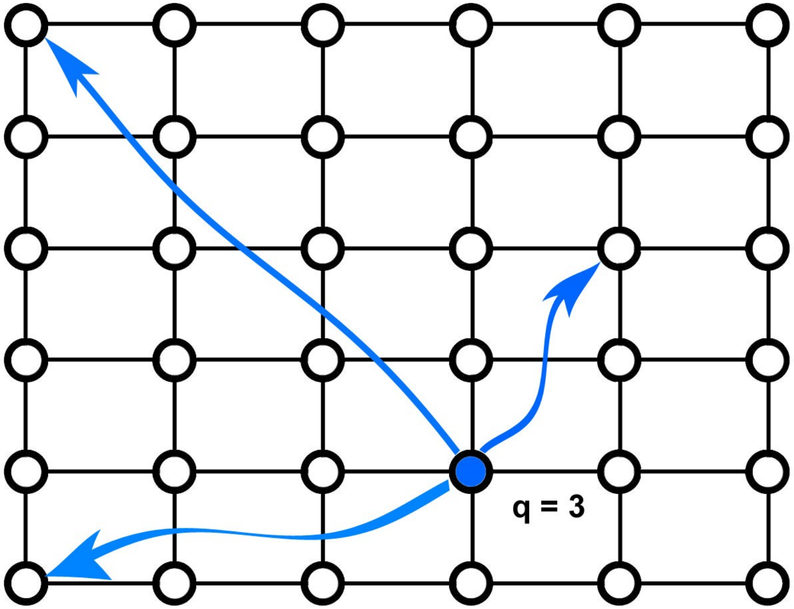
Modeling a social network

- *p: range of local contacts*
  - Nodes are connected to all other nodes within distance *p*.



Modeling a social network

- *q: number of long-range contacts*
  - add directed edges from node *u* to *q* other nodes using independent random trials



Modeling a social network

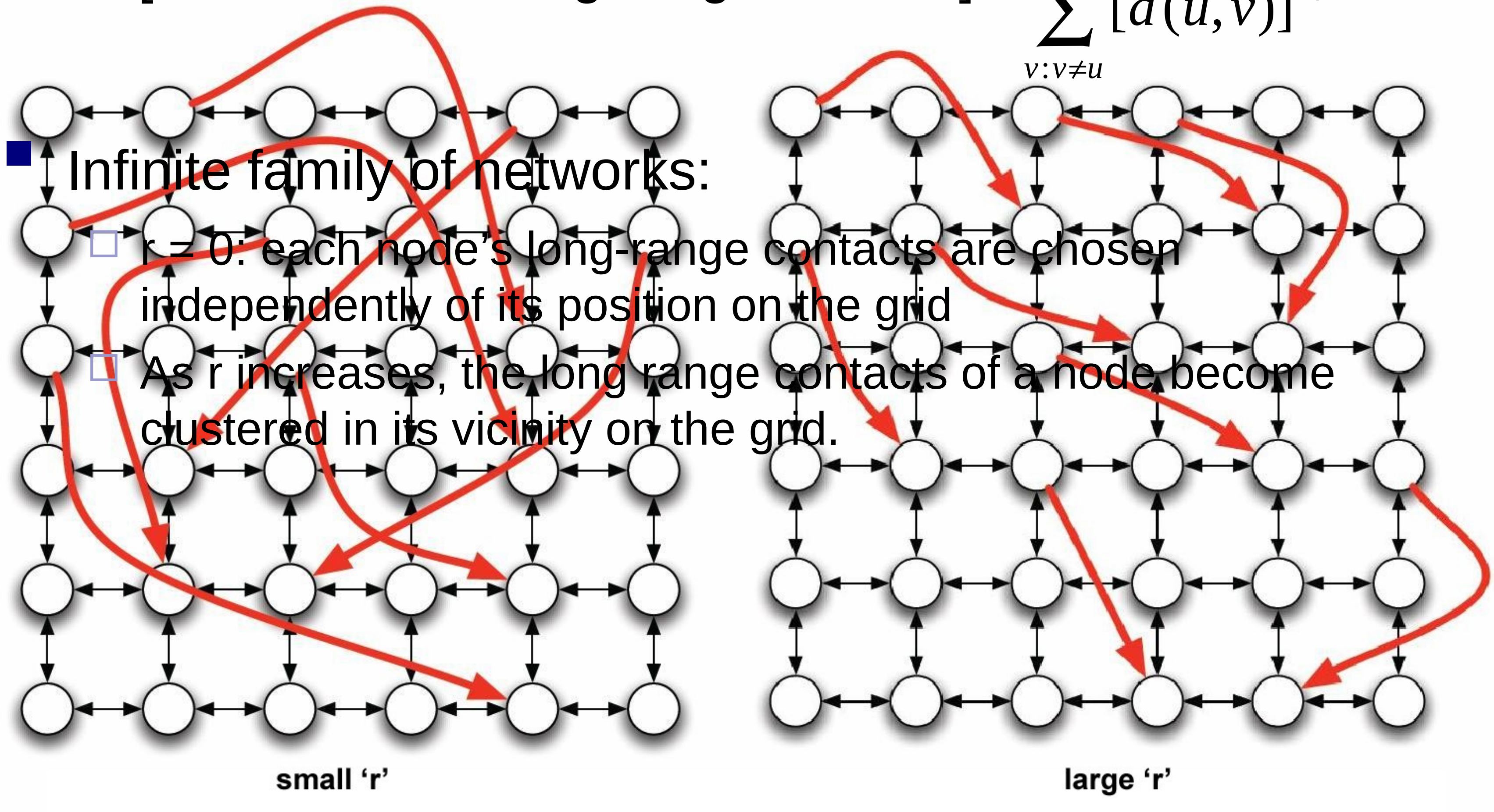
- Watts – Strogatz (1998)
  - Found that injecting a small amount of randomness (i.e. even $q = 1$) into the world is enough to make it a small world.

Modeling a social network

- Kleinberg (2000)
  - Why should arbitrary pairs of strangers, using only locally available information, be able to *find* short chains of acquaintances that link them together?
  - Does this occur in all small-world networks, or are there properties that must exist for this to happen?

Modeling a social network

- Pr [*u* has *v* as its long range contact] : $\frac{[d(u,v)]^{-r}}{\sum_{v:v\neq u}[d(u,v)]^{-r}}$
- Infinite family of networks:
  - r = 0: each node's long-range contacts are chosen independently of its position on the grid
  - As r increases, the long range contacts of a node become clustered in its vicinity on the grid.



The Algorithmic Side

- Input:
  - Grid G = (V,E)
  - arbitrary nodes *s, t*

- Goal: Transmit a message from *s* to *t* in as few steps as possible using only locally available information

The Algorithmic Side

- Assumptions:
  - In any step, the message holder *u* knows
    - The range of <u>local</u> contacts of all nodes
    - The location on the lattice of the target *t*
    - The locations and long-range contacts of all nodes that have previously touched the message
  - *u* does not know
    - the long-range contacts of nodes that have not touched the message

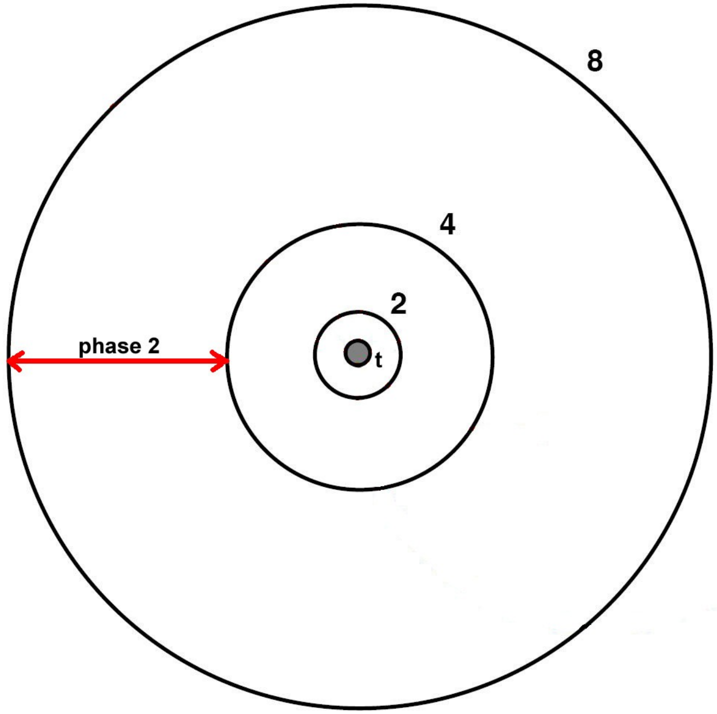
$$r = 2$$

The Algorithm

- In each step the current message holder passes the message to the contact that is as close to the target as possible.

Analysis

- Algorithm in phase *j*:
  - At a given step,
    $2^j < d(u,t) \leq 2^{j+1}$
  - Alg. is in phase 0 :
    - message is no more than 2 lattice steps away from the target t.
  - $j \leq \log_2 n$.



Analysis

Questions:

- How many steps will the algorithm take?
- How many steps will we spend in phase *j*?
- In a given step, with what probability will phase *j* end in this step?
- What is the probability that node *u* has a node *v* in the next phase as its long range contact?

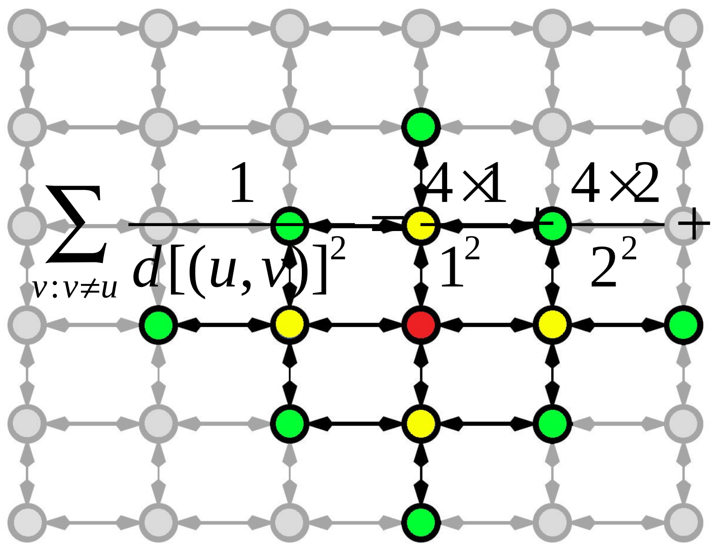
Analysis

Questions:

- How many steps will the algorithm take?
- How many steps will we spend in phase *j*?
- In a given step, with what probability will phase *j* end in this step?
- What is the probability that node *u* has a node *v* as its long range contact?

- Pr [*u* has *v* as its long range contact] ?

$$= \frac{[d(u,v)]^{-2}}{\sum [d(u,v)]^{-2}}$$



$$\sum_{v:v\neq u} \frac{1}{d[(u,v)]^2} = \frac{4\times 1}{1^2} + \frac{4\times 2}{2^2} + \frac{4\times 3}{3^2} + \dots = \sum_{j=1}^{2n-2} \frac{4j}{j^2}$$

Analysis

Questions:

- How many steps will the algorithm take?
- How many steps will we spend in phase *j*?
- In a given step, with what probability will phase *j* end in this step?
- What is the probability that node *u* has a node *v* as its long range contact?

- Pr[*u* has *v* as its long range contact]?

$$\sum_{v:v\neq u}[d(u,v)]^{-2} \leq \sum_{j=1}^{2n-2}\frac{4j}{j^2} = 4\sum_{j=1}^{2n-2} 1/j \leq 4[1+\ln(2n-2)] \leq 4\ln(6n)$$

$$\geq \frac{[d(u,v)]^{-2}}{4\ln(6n)}$$

- Thus *u* has *v* as its long-range contact with probability

$$\geq \frac{1}{4\ln(6n)\times[d(u,v)]^2}$$

Analysis

Questions:

- How many steps will the algorithm take?
- How many steps will we spend in phase *j*?
- In a given step, with what probability will phase *j* end in this step?
- What is the probability that node *u* has a node *v* as its long range contact?

$$\geq \frac{1}{4\ln(6n)\times[d(u,v)]^2}$$

- In any given step, Pr[phase j ends in this step]?
  - Phase j ends in this step if the message enters the set B_j of nodes within distance 2^j of t. Let v_f be the node in B_j that is farthest from *u*.

$$\Pr[\text{phase } j \text{ ends in this step}] = \sum_{v \in B_j} \Pr\left[u \text{ is friends with } v \in B_j\right]$$

$$\geq |B_j| \times \left(\frac{1}{4\ln(6n)\times[d(u,v_f)]^2} \right)$$

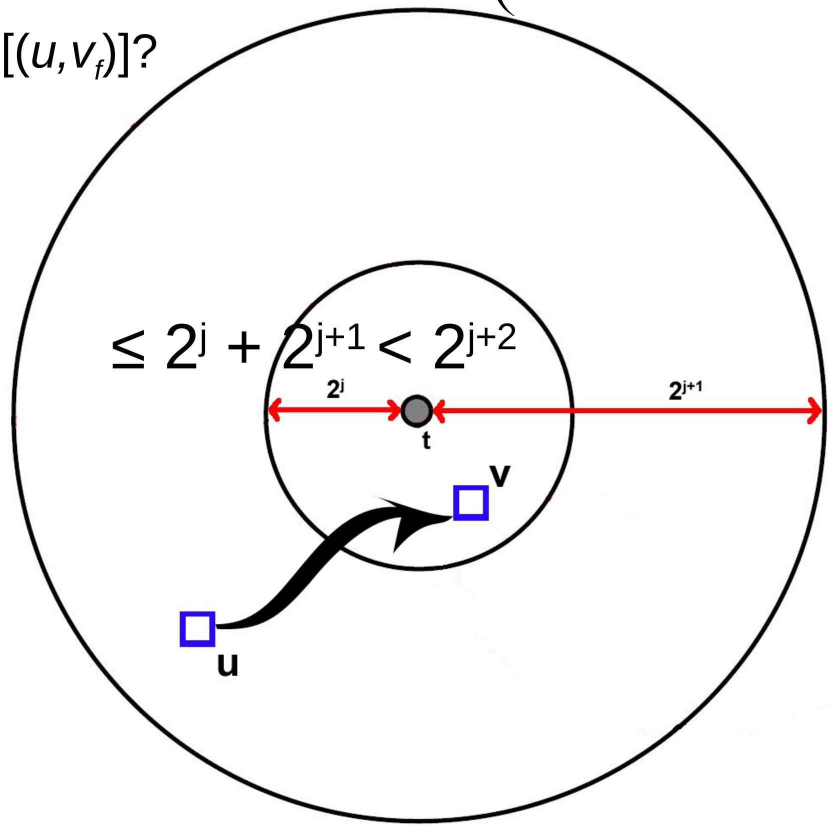
Analysis

Questions:

- How many steps will the algorithm take?
- How many steps will we spend in phase *j*?
- In a given step, with what probability will phase *j* end in this step?
- What is the probability that node *u* has a node *v* as its long range contact?

$$\geq \frac{1}{4\ln(6n)\times[d(u,v)]^2}$$

- $\Pr[\text{phase } j \text{ ends in this step}] \geq |B_j| \times \left(\frac{1}{4\ln(6n)\times[d(u,v_f)]^2}\right)$
  - What is $d[(u,v_f)]$?



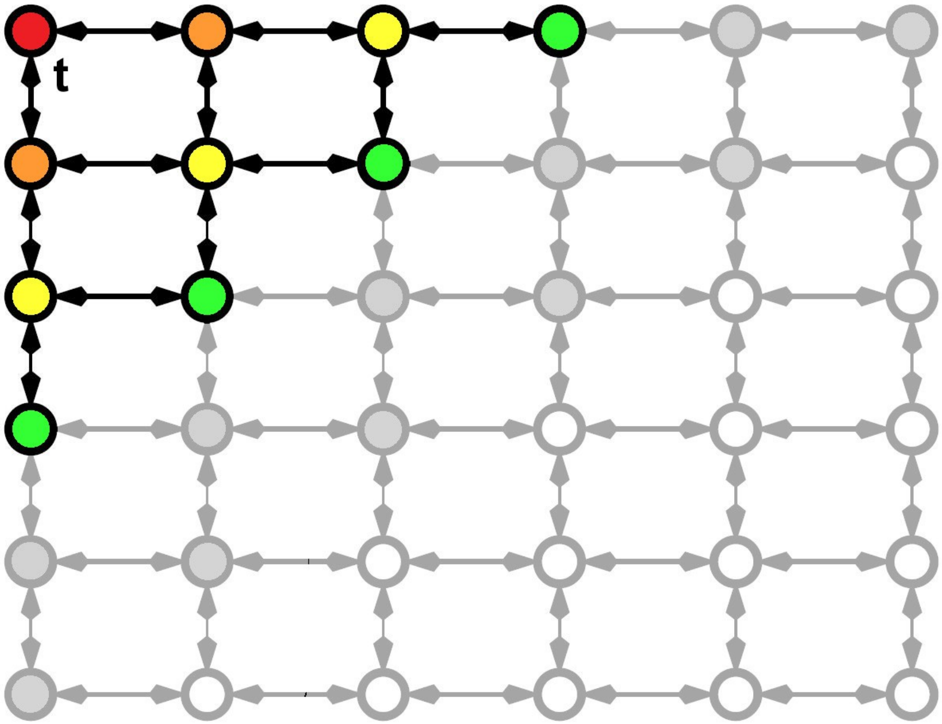
Analysis

Questions:

- How many steps will the algorithm take?
- How many steps will we spend in phase *j*?
- In a given step, with what probability will phase *j* end in this step?
- What is the probability that node *u* has a node *v* as its long range contact?

$$\geq \frac{1}{4\ln(6n)\cdot[d(u,v)]^2}$$

- $$\Pr[\text{phase } j \text{ ends in this step}] \geq |B_j| \cdot \left(\frac{1}{4\ln(6n)\cdot 2^{2j+4}}\right)$$
- How many nodes are in B_j?



Analysis

Questions:

- How many steps will the algorithm take?
- How many steps will we spend in phase *j*?
- In a given step, with what probability will phase *j* end in this step?
- What is the probability that node *u* has a node *v* as its long range contact?

$$\geq \frac{1}{4\ln(6n)\cdot[d(u,v)]^2}$$

- In any given step, Pr[phase j ends in this step]?
  - Pr[*u* has a long-range contact in B_j]?

$$\geq \#\ of\ nodes\ in\ B_j \cdot (probability\ u\ is\ friends\ with\ farthest\ v \in B_j)$$

$$\geq 2^{2j-1}\left(\frac{1}{4\ln(6n)\cdot 2^{2j+4}}\right) = \frac{2^{2j-1}}{4\ln(6n)\cdot 2^{2j+4}} = \frac{1}{128\ln(6n)}$$

Analysis

Questions:

- How many steps will the algorithm take?
- How many steps will we spend in phase *j*?
- In a given step, with what probability will phase *j* end in this step?

$$\geq \frac{1}{128 \ln(6n)}$$

- What is the probability that node *u* has a node *v* as its long range contact?

$$\geq \frac{1}{4 \ln(6n) \cdot [d(u,v)]^2}$$

- How many steps will we spend in phase *j*?
  - Let X_j be a random variable denoting the number of steps spent in phase *j*.
    - X_j is a geometric random variable with a probability of success at least

Analysis

Questions:

- How many steps will the algorithm take?
- How many steps will we spend in phase *j*?
- In a given step, with what probability will phase *j* end in this step?

  $\geq \frac{1}{128\ln(6n)}$

- What is the probability that node *u* has a node *v* as its long range contact?

  $\geq \frac{1}{4\ln(6n)\cdot[d(u,v)]^2}$

- How many steps will we spend in phase *j*?
  - Since X_j is a geometric random variable, we know that

$$E[X_j] = \frac{1}{p} \leq \frac{1}{1/128\ln(6n)} = 128\ln(6n)$$

Analysis

Questions:

- How many steps will the algorithm take?
- How many steps will we spend in phase *j*?
- In a given step, with what probability will phase *j* end in this step?

$$\geq \frac{1}{128\ln(6n)}$$

- What is the probability that node *u* has a node *v* as its long range contact?

$$\geq \frac{1}{4\ln(6n)\times[d(u,v)]^2}$$

- How many steps will we spend in phase *j*?
  - Let X_j be a random variable denoting the number of steps spent in phase *j*.

$$E[X_j] = \sum_{i=1}^{\infty} \Pr[X_j \geq i]$$

$$\leq \sum_{i=1}^{\infty} \left(1 - \frac{1}{128\ln(6n)}\right)^{i-1}$$

$$= 128\ln(6n)$$

Analysis

Questions:

- How many steps will the algorithm take?
- How many steps will we spend in phase *j*?

  $\leq 128\ln(6n)$

- In a given step, with what probability will phase *j* end in this step?

  $\geq \frac{1}{128\ln(6n)}$

- What is the probability that node *u* has a node *v* as its long range contact?

  $\geq \frac{1}{4\ln(6n)\times[d(u,v)]^2}$

- How many steps does the algorithm take?
  - Let X be a random variable denoting the number of steps taken by the algorithm.
  - By Linearity of Expectation we have

$$E[X] \leq (1+\log n)(128\ln(6n)) = O(\log n)^2$$

Analysis

Questions:

- How many steps will the algorithm take?
- How many steps will we spend in phase *j*?

$$\leq 128 \ln(6n)$$

- In a given step, with what probability will phase *j* end in this step?

$$\geq \frac{1}{128 \ln(6n)}$$

- What is the probability that node *u* has a node *v* as its long range contact?

$$\geq \frac{1}{4 \ln(6n) \times [d(u,v)]^2}$$

- When r = 2, expected delivery time is

$$O(\log n)^2$$

$$r \neq 2$$

Summary of results

- $0 \leq r < 2$: The expected delivery time of <u>any</u> decentralized algorithm is $\Omega(n^{(2-r)/3})$.
- $r > 2$: The expected delivery time of <u>any</u> decentralized algorithm is $\Omega(n^{(r-2)/(r-1)})$.

Revisiting Assumptions

- Recall that in each step the message holder *u* knew
  - the locations and long-range contacts of all nodes that have previously touched the message
- Is knowledge of message's history too much info?
- Upper-bound on delivery time in the good case is proven without using this.
- Lower-bound on delivery times for the bad cases still hold even when this knowledge is used.

The Intuition

- For a changing value of *r*
  - *r* = 0 provides no “geographical” clues that will assist in speeding up the delivery of the message.
  - 0 < *r* < 2: provides some clues, but not enough to sufficiently assist the message senders
  - r > 2: as *r* grows, the network becomes more localized. This becomes a prohibitive factor.
  - r = 2: provides a good mix of having relevant “geographical” information without too much localization.

References

- Kleinberg, J. *The Small-World Phenomenon: An Algorithmic Perspective*. Proc. 32nd ACM Symposium on Theory of Computing, 2000

- Kleinberg, J. *Navigation in a Small World.* Nature 406(2000), 845.