# Bounds for randomized rumour spreading protocols

Abbas Mehrabian

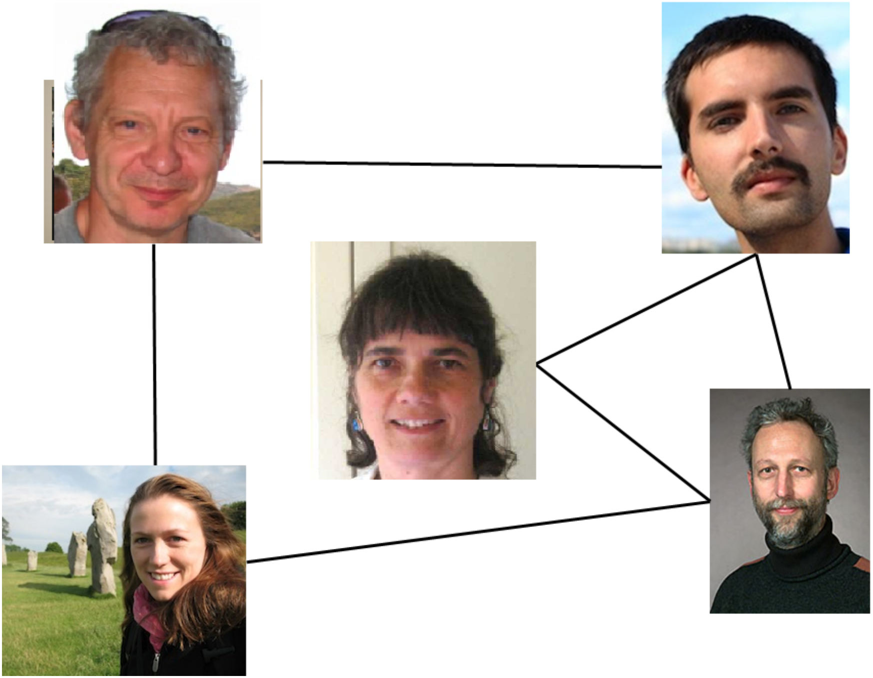
amehrabi@uwaterloo.ca

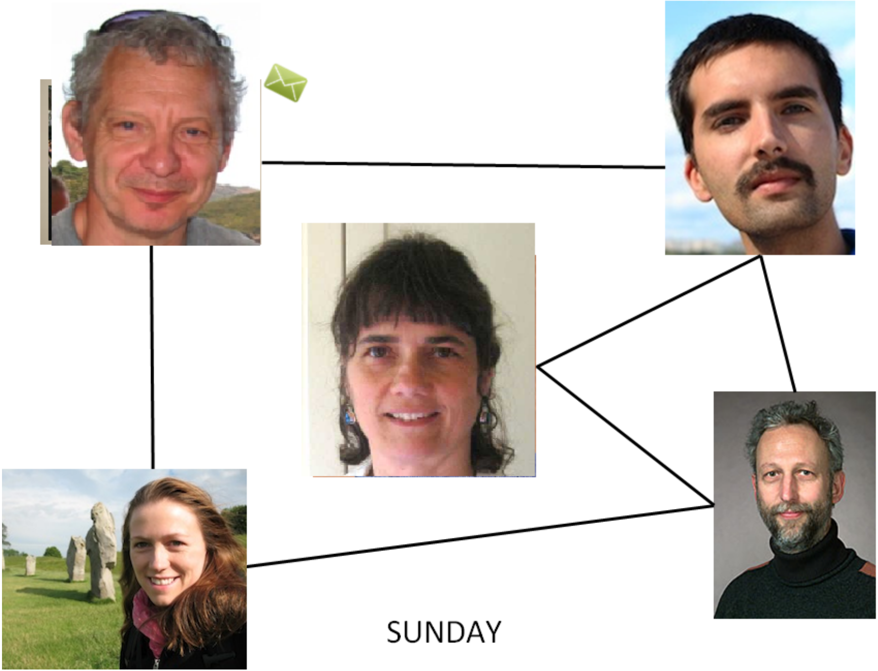
University of Waterloo

20 November 2014

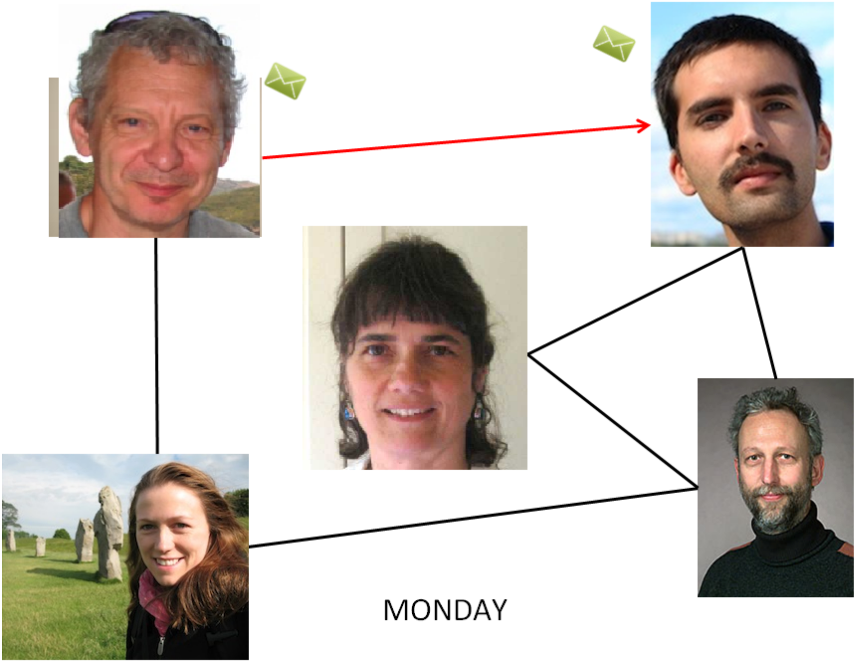
Dalhousie University

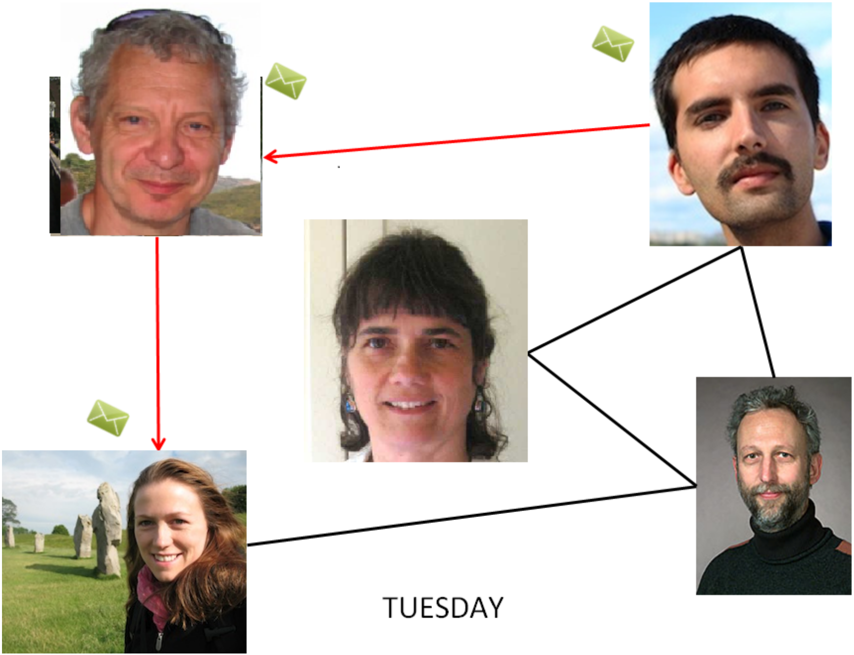
joint work with H. Acan, A. Collevecchio, A. Pourmiri, N. Wormald



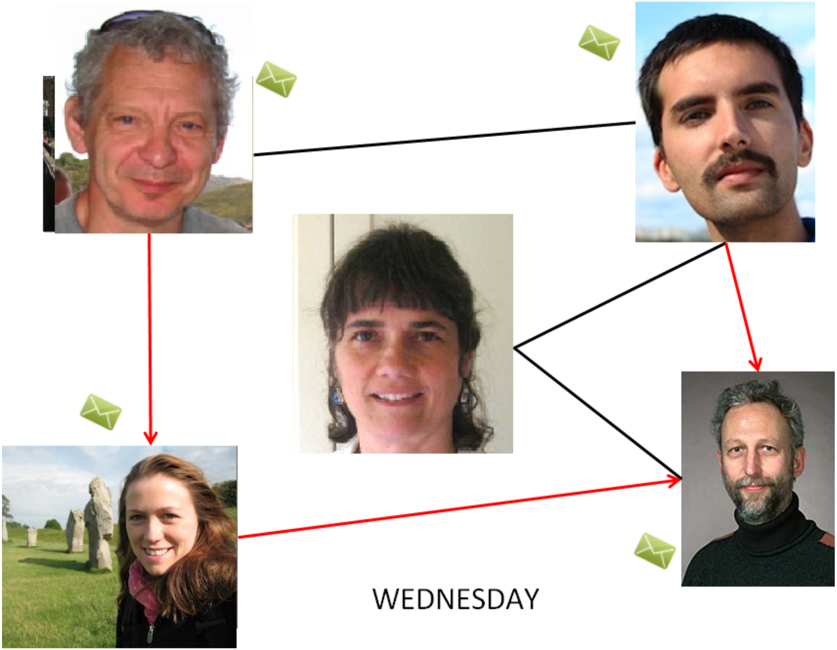


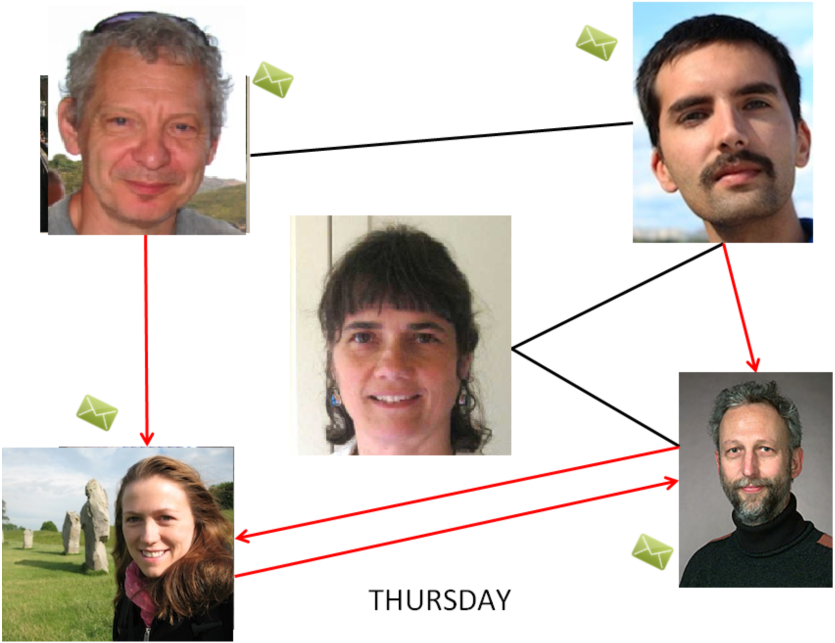
SUNDAY

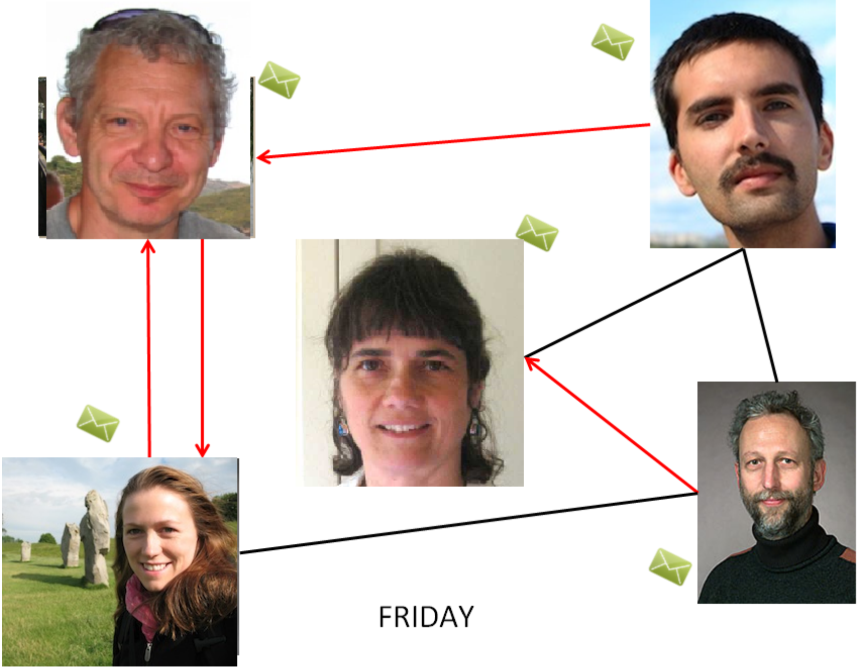

MONDAY



TUESDAY


WEDNESDAY


THURSDAY



FRIDAY

# Protocol definition

Demers, Gealy, Greene, Hauser, Irish, Larson, Manning, Shenker, Sturgis, Swinehart, Terry, Woods'87

1. The ground is a simple connected graph.
2. At time 0, one vertex knows a rumour.
3. At each time-step $1, 2, \ldots,$ every informed vertex tells the rumour to a random neighbour.

# Protocol definition

Demers, Gealy, Greene, Hauser, Irish, Larson, Manning, Shenker, Sturgis, Swinehart, Terry, Woods'87

1. The ground is a simple connected graph.
2. At time 0, one vertex knows a rumour.
3. At each time-step $1, 2, \ldots$, every informed vertex tells the rumour to a random neighbour.

Remark 1. Informed vertex may call a neighbour in consecutive steps.
Remark 2. If a vertex receives the rumour at time $t$, it starts passing it from time $t+1$.

# Protocol definition

Demers, Gealy, Greene, Hauser, Irish, Larson, Manning, Shenker, Sturgis, Swinehart, Terry, Woods'87

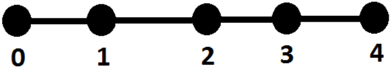
1. The ground is a simple connected graph.
2. At time 0, one vertex knows a rumour.
3. At each time-step $1, 2, \ldots$, every informed vertex tells the rumour to a random neighbour.

Remark 1. Informed vertex may call a neighbour in consecutive steps.
Remark 2. If a vertex receives the rumour at time $t$, it starts passing it from time $t + 1$.
inform-time($v$): the first time $v$ learns the rumour.
Spread Time: the first time everyone knows the rumour.

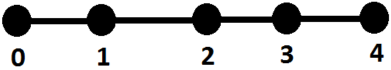
# Applications

- ✓ Replicated databases
- ✓ Broadcasting algorithms
- ✓ News propagation in social networks
- ✓ Spread of viruses on the Internet

# Example: a path

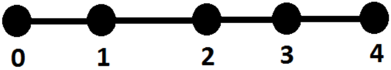


# Example: a path




$$\mathrm{inform} - \mathrm{time}(0) = 0$$

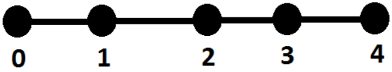
# Example: a path




$$\text{inform} - \text{time}(0) = 0$$

$$\text{inform} - \text{time}(1) = 1$$

# Example: a path

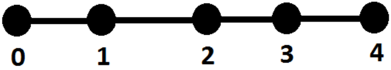



$$\text{inform} - \text{time}(0) = 0$$

$$\text{inform} - \text{time}(1) = 1$$

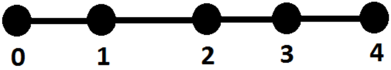
$$\text{inform} - \text{time}(2) = 1 + \text{Geo}(1/2)$$

# Example: a path




$$\mathrm{inform}-\mathrm{time}(0) = 0$$
$$\mathrm{inform}-\mathrm{time}(1) = 1$$
$$\mathrm{inform}-\mathrm{time}(2) = 1 + \mathrm{Geo}(1/2)$$
$$\mathrm{inform}-\mathrm{time}(3) = 1 + \mathrm{Geo}(1/2) + \mathrm{Geo}(1/2)$$

# Example: a path




$$\mathrm{inform} - \mathrm{time}(0) = 0$$

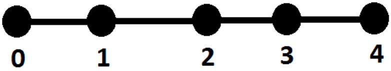
$$\mathrm{inform} - \mathrm{time}(1) = 1$$

$$\mathrm{inform} - \mathrm{time}(2) = 1 + \mathrm{Geo}(1/2)$$

$$\mathrm{inform} - \mathrm{time}(3) = 1 + \mathrm{Geo}(1/2) + \mathrm{Geo}(1/2)$$

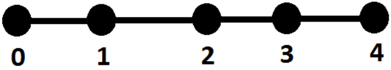
$$\mathrm{inform} - \mathrm{time}(4) = 1 + \mathrm{Geo}(1/2) + \mathrm{Geo}(1/2) + \mathrm{Geo}(1/2)$$

# Example: a path



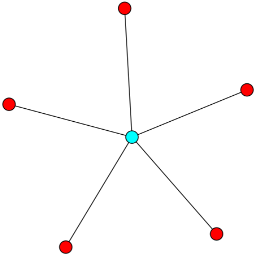

$$\mathrm{inform} - \mathrm{time}(0) = 0$$
$$\mathrm{inform} - \mathrm{time}(1) = 1$$
$$\mathrm{inform} - \mathrm{time}(2) = 1 + \mathrm{Geo}(1/2)$$
$$\mathrm{inform} - \mathrm{time}(3) = 1 + \mathrm{Geo}(1/2) + \mathrm{Geo}(1/2)$$
$$\mathrm{inform} - \mathrm{time}(4) = 1 + \mathrm{Geo}(1/2) + \mathrm{Geo}(1/2) + \mathrm{Geo}(1/2)$$
$$\mathbb{E}[\text{Spread Time}] = 1 + 3 \times 2 = 7$$

# Example: a path

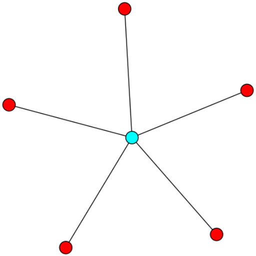



$$
\begin{aligned}
\text{inform} - \text{time}(0) &= 0 \\
\text{inform} - \text{time}(1) &= 1 \\
\text{inform} - \text{time}(2) &= 1 + \text{Geo}(1/2) \\
\text{inform} - \text{time}(3) &= 1 + \text{Geo}(1/2) + \text{Geo}(1/2) \\
\text{inform} - \text{time}(4) &= 1 + \text{Geo}(1/2) + \text{Geo}(1/2) + \text{Geo}(1/2) \\
\mathbb{E}[\text{Spread Time}] &= 1 + 3 \times 2 = 7 \\
&= 2n - 3
\end{aligned}
$$

# Example: a star

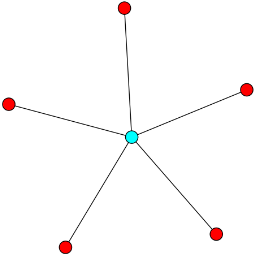


# Example: a star



When $k+1$ vertices are informed and $n-1-k$ uninformed, after $\frac{n-1}{n-1-k}$ more rounds a new vertex will be informed.

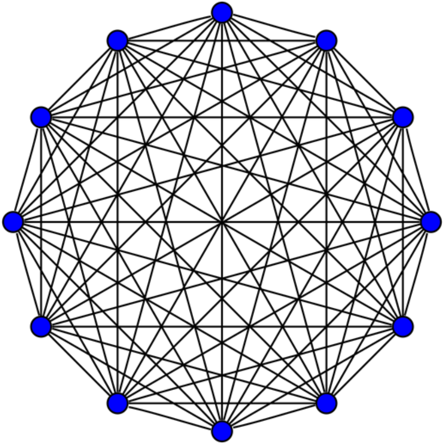
# Example: a star



When $k + 1$ vertices are informed and $n - 1 - k$ uninformed, after $\frac{n-1}{n-1-k}$ more rounds a new vertex will be informed.

$$\mathbb{E}[\text{Spread Time}] = \frac{n-1}{n-1} + \frac{n-1}{n-2} + \cdots + \frac{n-1}{2} + \frac{n-1}{1} \approx n \ln n$$

# Example: a complete graph



$$\mathbb{E}[\text{Spread Time}] \approx \log_2 n + \ln n \qquad [\text{Pittel}'87]$$

# Other known results

With probability close to 1, for any starting vertex,

1. $\max\{\text{diameter}(G)/2, \log_2 n\} \leq \text{Spread Time} \leq n \ln n$

   [Elsässer and Sauerwald'06]

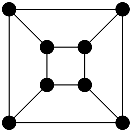
# Other known results

With probability close to 1, for any starting vertex,

1. $\max\{\text{diameter}(G)/2, \log_2 n\} \leq$ Spread Time $\leq n \ln n$

   [Elsässer and Sauerwald'06]

2. Spread Time of $\mathcal{H}_d = \Theta(d)$ [Feige, Peleg, Raghavan, Upfal'90]



$\mathcal{H}_3$

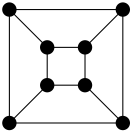
# Other known results

With probability close to 1, for any starting vertex,

1. $\max\{\text{diameter}(G)/2, \log_2 n\} \leq$ Spread Time $\leq n \ln n$

   [Elsässer and Sauerwald'06]

2. Spread Time of $\mathcal{H}_d = \Theta(d)$ [Feige, Peleg, Raghavan, Upfal'90]



$\mathcal{H}_3$

3. If $pn \geq (1+\varepsilon)\ln n$ then Spread Time of $G(n,p) = \Theta(\ln n)$ [Feige et al.'90]

# Improving the protocol

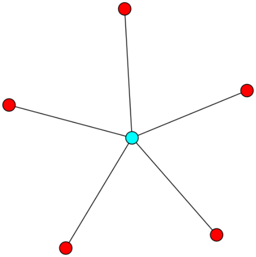
Uninformed vertices ask the informed ones...

# The push-pull protocol

Demers, Gealy, Greene, Hauser, Irish, Larson, Manning, Shenker, Sturgis, Swinehart, Terry, Woods'87

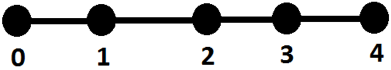
1. The ground is a simple connected graph.
2. At time 0, one vertex knows a rumour.
3. At each time-step $1, 2, \ldots,$
   every informed vertex sends the rumour to a random neighbour (PUSH);
   and every uninformed vertex queries a random neighbour about the rumour (PULL).

# Example: a star

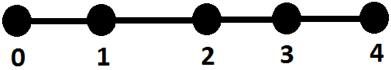


push protocol: $n \ln n$ rounds
push-pull protocol: 1 or 2 rounds

# Example: a path

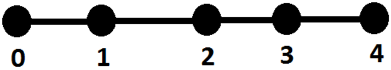


# Example: a path




$$\mathrm{inform-time}(0) = 0$$

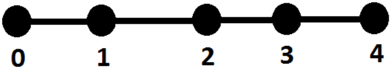
# Example: a path




$$\text{inform} - \text{time}(0) = 0$$

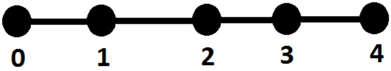
$$\text{inform} - \text{time}(1) = 1$$

# Example: a path




$$\begin{aligned}
\text{inform} - \text{time}(0) &= 0 \\
\text{inform} - \text{time}(1) &= 1 \\
\text{inform} - \text{time}(2) &= 1 + \min\{\text{Geo}(1/2), \text{Geo}(1/2)\} \\
&= 1 + \text{Geo}(3/4)
\end{aligned}$$

# Example: a path



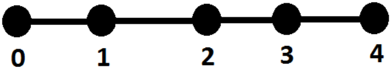

$$\text{inform} - \text{time}(0) = 0$$

$$\text{inform} - \text{time}(1) = 1$$

$$\text{inform} - \text{time}(2) = 1 + \text{Geo}(3/4)$$

$$\text{inform} - \text{time}(3) = 1 + \text{Geo}(3/4) + \text{Geo}(3/4)$$

# Example: a path




$$\mathrm{inform-time}(0) = 0$$

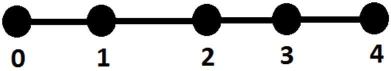
$$\mathrm{inform-time}(1) = 1$$

$$\mathrm{inform-time}(2) = 1 + \mathrm{Geo}(3/4)$$

$$\mathrm{inform-time}(3) = 1 + \mathrm{Geo}(3/4) + \mathrm{Geo}(3/4)$$

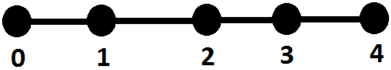
$$\mathrm{inform-time}(4) = 1 + \mathrm{Geo}(3/4) + \mathrm{Geo}(3/4) + 1$$

# Example: a path



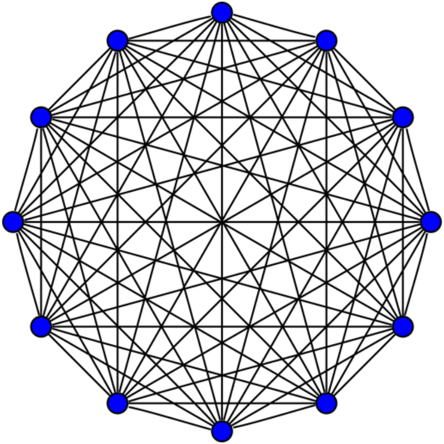

$$\mathrm{inform-time}(0) = 0$$
$$\mathrm{inform-time}(1) = 1$$
$$\mathrm{inform-time}(2) = 1 + \mathrm{Geo}(3/4)$$
$$\mathrm{inform-time}(3) = 1 + \mathrm{Geo}(3/4) + \mathrm{Geo}(3/4)$$
$$\mathrm{inform-time}(4) = 1 + \mathrm{Geo}(3/4) + \mathrm{Geo}(3/4) + 1$$
$$\mathbb{E}[\text{Spread Time}] = 2 + 2 \times 4/3 = 14/3$$

# Example: a path




$$
\begin{aligned}
\mathrm{inform}-\mathrm{time}(0) &= 0\\
\mathrm{inform}-\mathrm{time}(1) &= 1\\
\mathrm{inform}-\mathrm{time}(2) &= 1+\mathrm{Geo}(3/4)\\
\mathrm{inform}-\mathrm{time}(3) &= 1+\mathrm{Geo}(3/4)+\mathrm{Geo}(3/4)\\
\mathrm{inform}-\mathrm{time}(4) &= 1+\mathrm{Geo}(3/4)+\mathrm{Geo}(3/4)+1\\
\mathbb{E}[\text{Spread Time}] &= 2+2\times 4/3 = 14/3\\
&= \frac{4}{3}n-2 \qquad (\text{versus } 2n-3 \text{ for push})
\end{aligned}
$$

# Example: a complete graph



push: $\log_2 n + \ln n$ [Pittel'87]

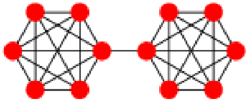
push-pull: $\log_3 n$ [Karp, Schindelhauer, Shenker, Vöcking'00]

# Other results on push-pull protocol

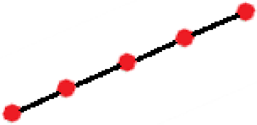
1. Barabasi-Albert preferential attachment graph has
   Spread Time $\Theta(\ln n)$,
   PUSH alone has Spread Time $poly(n)$.
   [Doerr, Fouz, Friedrich '11]
2. Random graphs with power-law expected degrees
   (a.k.a. the Chung-Lu model) with exponent $\in (2, 3)$ has
   Spread Time $\Theta(\ln n)$.
   [Fountoulakis, Panagiotou, Sauerwald '12]
3. If $\alpha$ is the vertex expansion (vertex isoperimetric number),
   and $\Phi$ is the conductance,
   Spread Time $\le C \max\{\Phi^{-1} \ln n, \alpha^{-1} \ln^2 n\}$. [Giakkoupis '11, '14]
   No bottleneck $\Rightarrow$ fast rumour spreading

# An extremal question

What's the maximum spread time of an $n$-vertex graph?



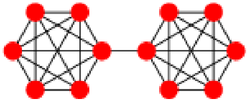
$n/4$



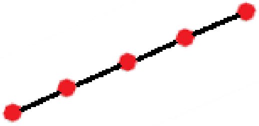
$4n/3$

# An extremal question

What's the maximum spread time of an $n$-vertex graph?



$n/4$



$4n/3$

**Theorem (Acan, Collevecchio, M, Wormald'14+)**

*On any $n$-vertex graph,* $\mathbb{E}\,[\text{Spread Time}] \leq 5n$

## ...


"Very well, Then — If there's no more 'old gossip', lets move on to 'new gossip'."

Push-Pull on Random $k$-trees

# Random $k$-trees

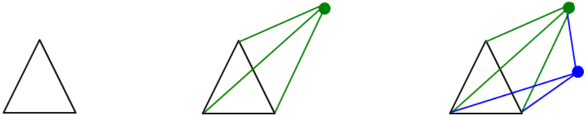
**Example ($k = 3$)**

# Random $k$-trees

Example ($k = 3$)

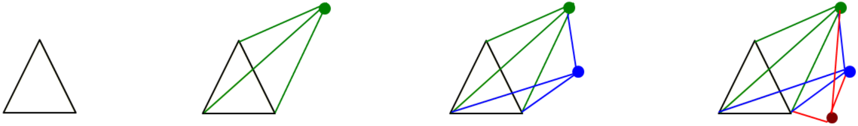
# Random $k$-trees

**Example ($k = 3$)**

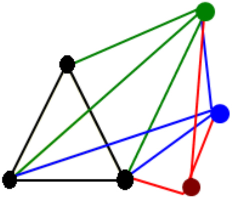


# Random $k$-trees

**Example ($k = 3$)**



# Some properties of random $k$-trees



- ✓ Logarithmic diameter
- ✓ Power law degree sequence:
  fraction of vertices with degree $d \approx d^{-2-\frac{1}{k-1}}$
- ✓ Very small conductance and vertex expansion
  Lots of bottlenecks

# Our results

Push-Pull protocol on random $k$-trees ($k > 1$ fixed):

**Theorem (M, Pourmiri'14, upper bound)**

*If initially a random vertex knows the rumor,*
*after* $(\ln n)^{1+3/k}$ *rounds, 99 percent of vertices will know it.*

# Our results

Push-Pull protocol on random $k$-trees ($k > 1$ fixed):

**Theorem (M, Pourmiri'14, upper bound)**

*If initially a random vertex knows the rumor, after $(\ln n)^{1+3/k}$ rounds, 99 percent of vertices will know it.*

Push-Pull is efficient on a poorly connected random network, if informing almost all vertices suffices.

## Our results

Push-Pull protocol on random $k$-trees ($k > 1$ fixed):

**Theorem (M, Pourmiri'14, upper bound)**

*If initially a random vertex knows the rumor, after $(\ln n)^{1+3/k}$ rounds, 99 percent of vertices will know it.*

Push-Pull is efficient on a poorly connected random network, if informing almost all vertices suffices.

**Theorem (M, Pourmiri'14, lower bound)**

*The time required to inform all vertices is $> n^{1/3k}$*

# Our results

Push-Pull protocol on random $k$-trees ($k > 1$ fixed):

Theorem (M, Pourmiri'14, upper bound)

*If initially a random vertex knows the rumor, after* $(\ln n)^{1+3/k}$ *rounds, 99 percent of vertices will know it.*

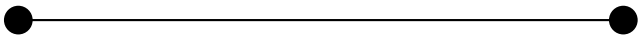
Push-Pull is efficient on a poorly connected random network, if informing almost all vertices suffices.

Theorem (M, Pourmiri'14, lower bound)

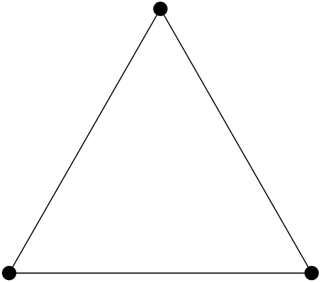
*The time required to inform all vertices is* $> n^{1/3k}$

Exponential blow up if informing each and every vertex is required.

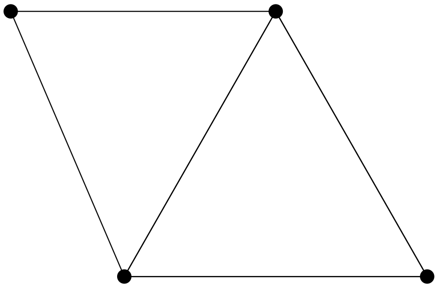
## Self-similarity of random $k$-trees



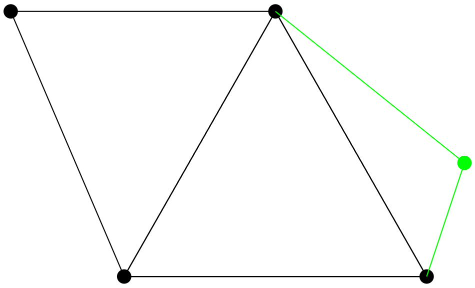
# Self-similarity of random $k$-trees



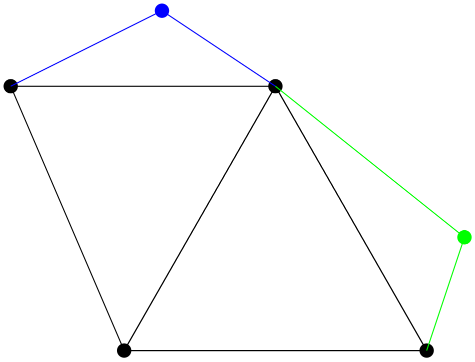
# Self-similarity of random $k$-trees



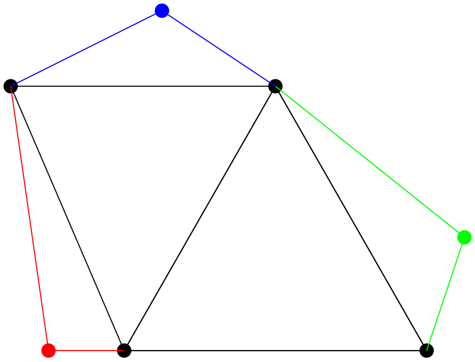
# Self-similarity of random $k$-trees



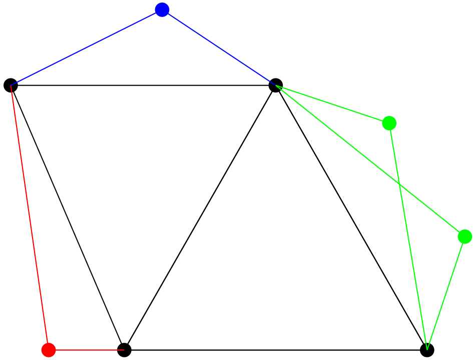
# Self-similarity of random $k$-trees



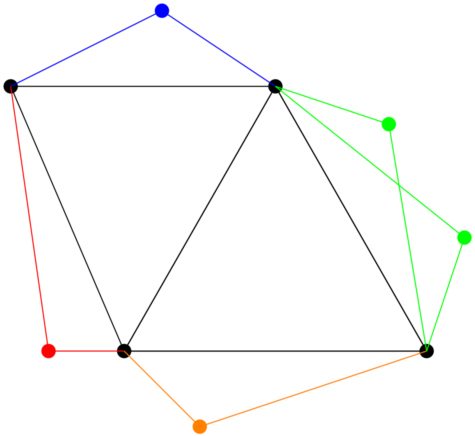
# Self-similarity of random $k$-trees



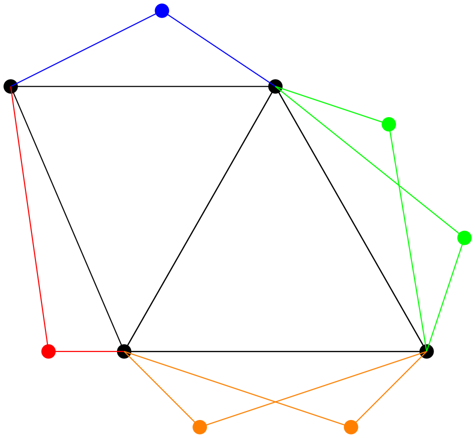
# Self-similarity of random $k$-trees



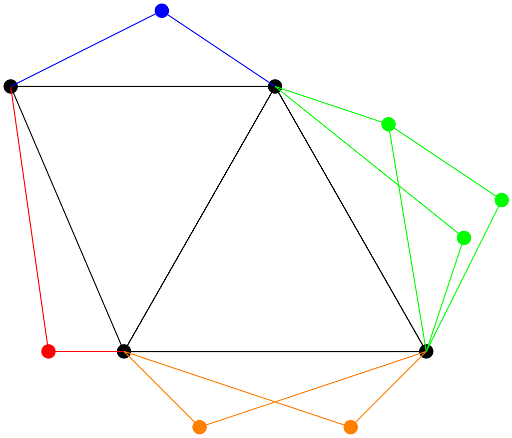
# Self-similarity of random $k$-trees



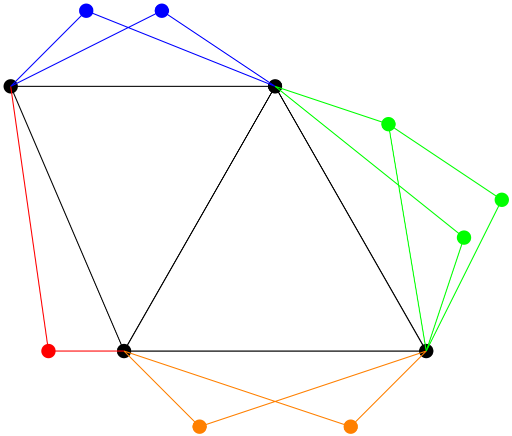
# Self-similarity of random $k$-trees



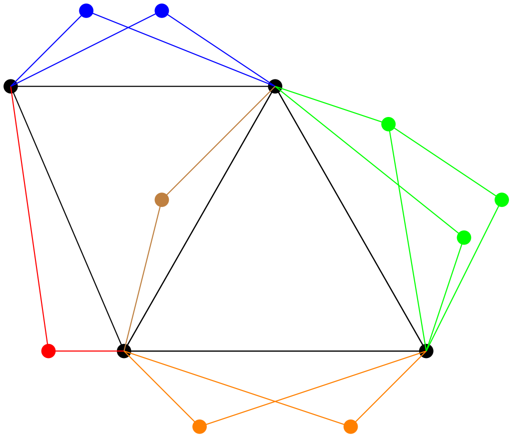
# Self-similarity of random $k$-trees



# Self-similarity of random $k$-trees



# Self-similarity of random $k$-trees



Proof sketch of the upper bound

# Fast edges

Let $d$ be some parameter.

**Definition (fast edge)**

An edge $uv \in E(H)$ is fast if $\deg(u) \leq d$ or $\deg(v) \leq d$ or $u$ and $v$ have a common neighbour with degree $\leq d$.

# Fast edges

Let $d$ be some parameter.

**Definition (fast edge)**

An edge $uv \in E(H)$ is fast if $\deg(u) \le d$ or $\deg(v) \le d$ or $u$ and $v$ have a common neighbour with degree $\le d$.

Observe: if $uv$ is fast, average transmission time from $u$ to $v \le 2d$.

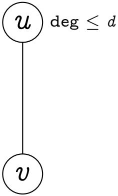
# Fast edges

Let $d$ be some parameter.

**Definition (fast edge)**

An edge $uv \in E(H)$ is fast if $\deg(u) \leq d$ or $\deg(v) \leq d$ or $u$ and $v$ have a common neighbour with degree $\leq d$.

Observe: if $uv$ is fast, average transmission time from $u$ to $v \leq 2d$.



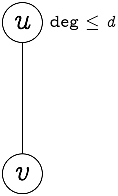
# Fast edges

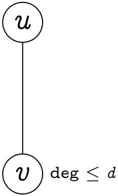
Let $d$ be some parameter.

**Definition (fast edge)**

An edge $uv \in E(H)$ is fast if $\deg(u) \leq d$ or $\deg(v) \leq d$ or $u$ and $v$ have a common neighbour with degree $\leq d$.

Observe: if $uv$ is fast, average transmission time from $u$ to $v \leq 2d$.



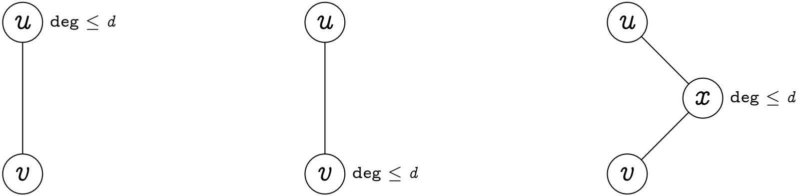
# Fast edges

Let $d$ be some parameter.

**Definition (fast edge)**

An edge $uv \in E(H)$ is fast if $\deg(u) \leq d$ or $\deg(v) \leq d$ or $u$ and $v$ have a common neighbour with degree $\leq d$.

Observe: if $uv$ is fast, average transmission time from $u$ to $v \leq 2d$.



# Fast edges

Let $d$ be some parameter.

**Definition (fast edge)**

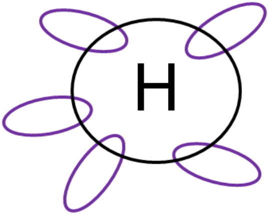
An edge $uv \in E(H)$ is fast if $\deg(u) \leq d$ or $\deg(v) \leq d$ or $u$ and $v$ have a common neighbour with degree $\leq d$.

Observe: if $uv$ is fast, average transmission time from $u$ to $v \leq 2d$.

**Lemma**

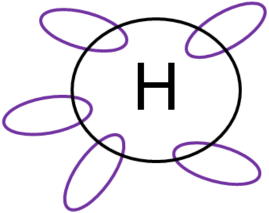
*Transmission time along a path of length $\ell$ of fast edges is $\leq O(d(\ell + \ln n))$ with prob. $\geq 1 - n^{-3}$.*

# Proof of upper bound



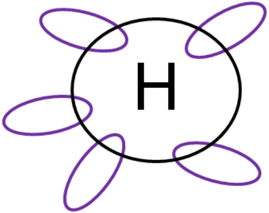

1. $H =$ graph at round $m \approx n/\ln n$

# Proof of upper bound



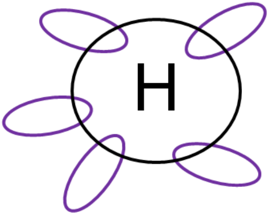

1. $H$ = graph at round $m \approx n/\ln n$
2. Almost all vertices in small pieces have degrees $\leq d = (\ln n)^{3/k}$

# Proof of upper bound




1. $H$ = graph at round $m \approx n / \ln n$
2. Almost all vertices in small pieces have degrees $\leq d = (\ln n)^{3/k}$
3. $\exists$ an almost-spanning tree of $H$ of height $O(\ln n)$ consisting fast edges.

# Proof of upper bound




1. $H$ = graph at round $m \approx n/\ln n$
2. Almost all vertices in small pieces have degrees $\leq d = (\ln n)^{3/k}$
3. $\exists$ an almost-spanning tree of $H$ of height $O(\ln n)$ consisting fast edges.

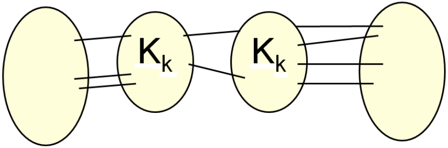
**Theorem (upper bound)**

*If initially a random vertex knows the rumor, after $O\left(\ln n \times (\ln n)^{3/k}\right)$ rounds, 99 percent of vertices will know it.*

# Proof sketch of the lower bound

# Proof of lower bound

Definition ($D$-barrier)

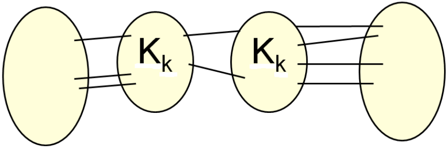



Vertices in the two $k$-cliques have degrees $\geq D$.

Observe: if a $D$-barrier exists, Spread Time is $\geq \Omega(D)$.

# Proof of lower bound

**Definition ($D$-barrier)**



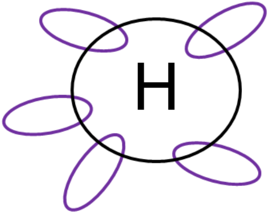

Vertices in the two $k$-cliques have degrees $\geq D$.

Observe: if a $D$-barrier exists, Spread Time is $\geq \Omega(D)$.

**Lemma**

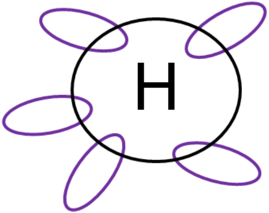
*A random $k$-tree has a $\Omega\left(n^{1-1/k}\right)$-barrier with prob. $\geq \Omega\left(n^{-k}\right)$*

# Proof of lower bound



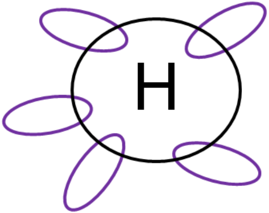

1. $H$ = graph at round $m \approx \sqrt{n}$

# Proof of lower bound



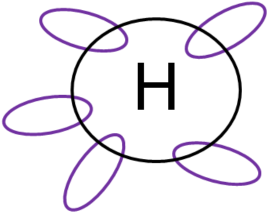

1. $H$ = graph at round $m \approx \sqrt{n}$
2. Each piece has a $(n/km)^{1-1/k}$-barrier with prob. $\Omega\left((n/km)^{-k}\right)$.

# Proof of lower bound




1. $H$ = graph at round $m \approx \sqrt{n}$
2. Each piece has a $(n/km)^{1-1/k}$-barrier with prob. $\Omega\left((n/km)^{-k}\right)$.
3. Since $km\left((n/km)^{-k}\right) \to \infty$ and by independence of pieces, w.h.p. there exists a $(n/km)^{1-1/k}$-barrier.

# Proof of lower bound




1. $H$ = graph at round $m \approx \sqrt{n}$
2. Each piece has a $(n/km)^{1-1/k}$-barrier with prob. $\Omega\left((n/km)^{-k}\right)$.
3. Since $km\left((n/km)^{-k}\right) \to \infty$ and by independence of pieces, w.h.p. there exists a $(n/km)^{1-1/k}$-barrier.

Hence, w.h.p. Spread Time$> n^{1/3k}$

# Future research directions

1. Design a (deterministic) approximation algorithm for finding the Spread Time of a given graph.
2. Study other graph classes, e.g. random geometric graphs.

# Future research directions

1. Design a (deterministic) approximation algorithm for finding the Spread Time of a given graph.
2. Study other graph classes, e.g. random geometric graphs.