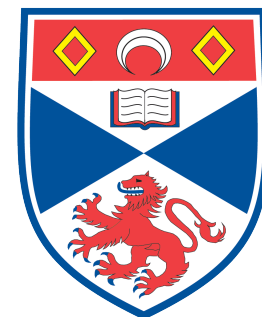


Privacy-enhanced social network routing in opportunistic networks

Iain Parris, Greg Bigwood, Tristan Henderson

{ip,gjb,tristan}@cs.st-andrews.ac.uk

<<http://www.cs.st-andrews.ac.uk/~ip/>>



University
of
St Andrews

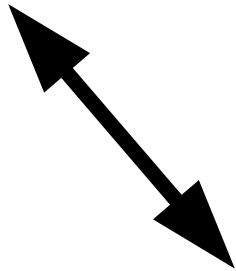
Opportunistic networks



Can **directly exchange** messages between the mobile devices that people already carry, forming a disconnected, store-and-forward **opportunistic network**.

One example application: micro-blogging.

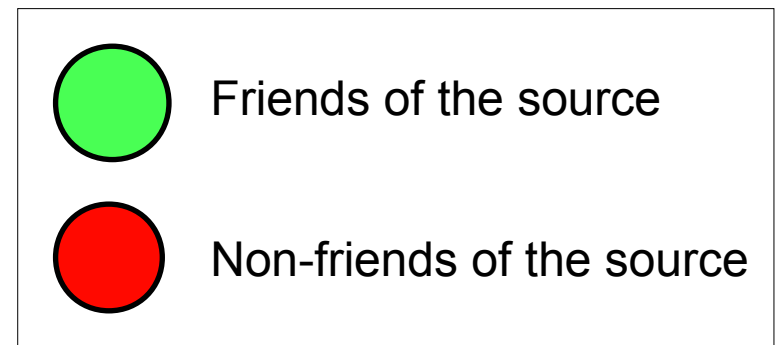
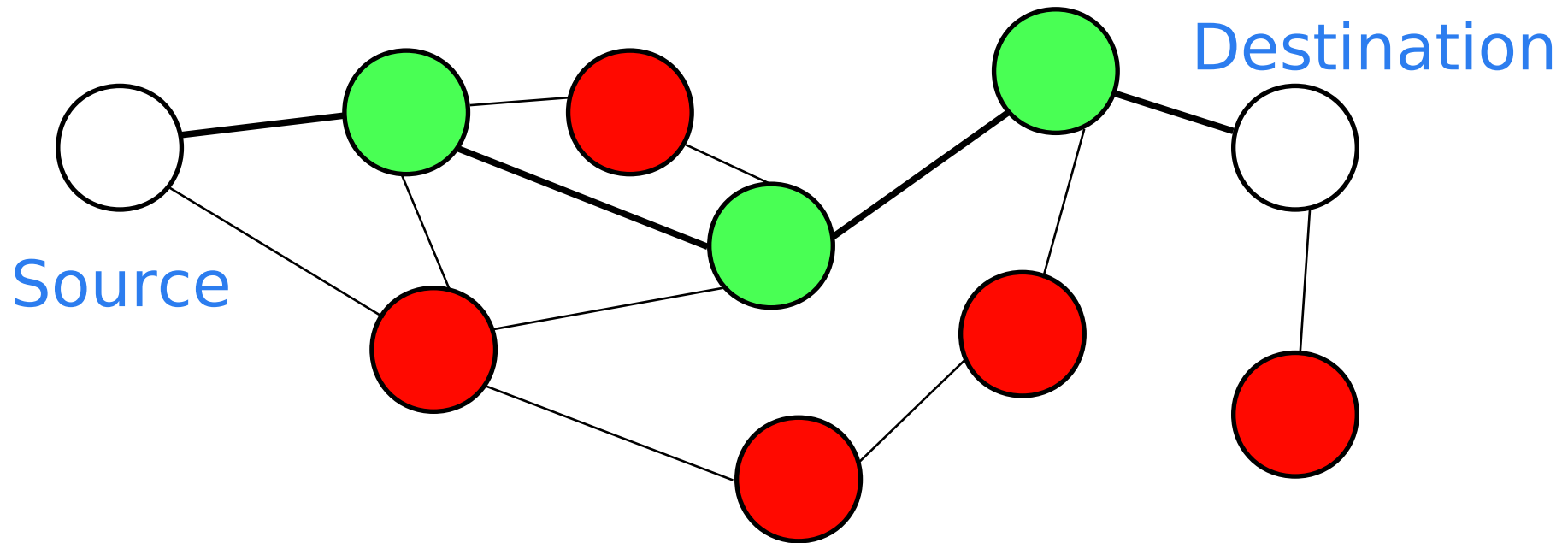
“Hello, I am in the Beer Garden.”



Routing?

- Epidemic routing – flooding all links drains batteries quickly
- **Social network routing** – better for battery life, but requires known social network

Routing Example



Social network routing

Simple social network routing

- Message includes the **sender's complete social network** as a message header
- Sender, and intermediate people, forward message to any person who is in the sender's social network



Privacy problem

Everyone who sees the message must know who else is in the sender's social network, even if there existed a PKI

Privacy

How can we protect users' social network privacy?

If we get it wrong...

- Social networks exposed
- Linkability to other sites – including pseudonymous websites

twitter → flickr



Google admits Buzz social network testing flaws

By Jonathan Fildes
Technology reporter, BBC News

Google has admitted to BBC News that testing of its controversial social network Buzz was insufficient.

The firm has had to make a series of changes to the service after a ferocious backlash from users concerned about intrusions of privacy.

The BBC understands that Buzz was only tested internally and bypassed more extensive trials with external testers - used for many other Google services.


Google said that it was now working "extremely hard" to fix the problems.

"We're very early in this space. This was one of our first big attempts," Todd Jackson, Buzz product manager, told BBC News.

"We've been testing Buzz internally at Google for a while. Of course, getting feedback from 20,000 Googlers isn't quite the same as letting Gmail users play with Buzz in the wild."

Many of the firm's new services are tested by the so-called Google Trusted Tester program, a network of friends and family of Google employees who are given confidential access to products before they launch.

Buzz was not tested by this program.



SEE ALSO

- ▶ Google unveils new social network 09 Feb 10 | Technology
- ▶ Google invites users to join Wave 30 Sep 09 | Technology
- ▶ Strong reception for Google Wave 01 Jun 09 | Technology
- ▶ US calls for China Google probe 21 Jan 10 | Americas
- ▶ Google drops Gmail address in UK 19 Oct 05 | Business
- ▶ Google mobile phone expected 05 Jan 10 | Technology

RELATED INTERNET LINKS

- ▶ Google
- ▶ Buzz
- ▶ Facebook
- ▶ Evgeny Morozov' post

The BBC is not responsible for the content of external internet sites

FROM OTHER NEWS SITES

- ▶ Guardian.co.uk Green light: Cruel meat, Copenhagen conspiracies and iPhone climate war - 1 hr ago
- ▶ CIO CEO denies Google wants to take on mobile operators - 2 hrs ago
- ▶ Times Online Google forced into Buzz revamp over privacy row - 3 hrs ago
- ▶ New Statesman* Google advised to make Buzz opt-in service - 4 hrs ago

<http://news.bbc.co.uk/1/hi/8517613.stm>

Social network obfuscation

Our approaches

Statisticulated Social Network Routing (SSNR)

- On a message by message basis, sender **adds or removes random nodes from their social network** when sending each new message, → deniability.

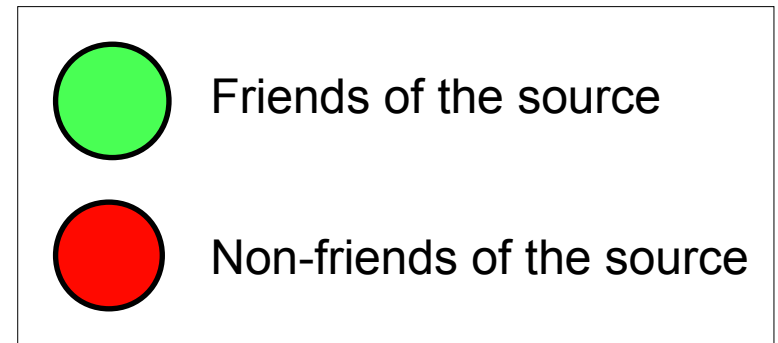
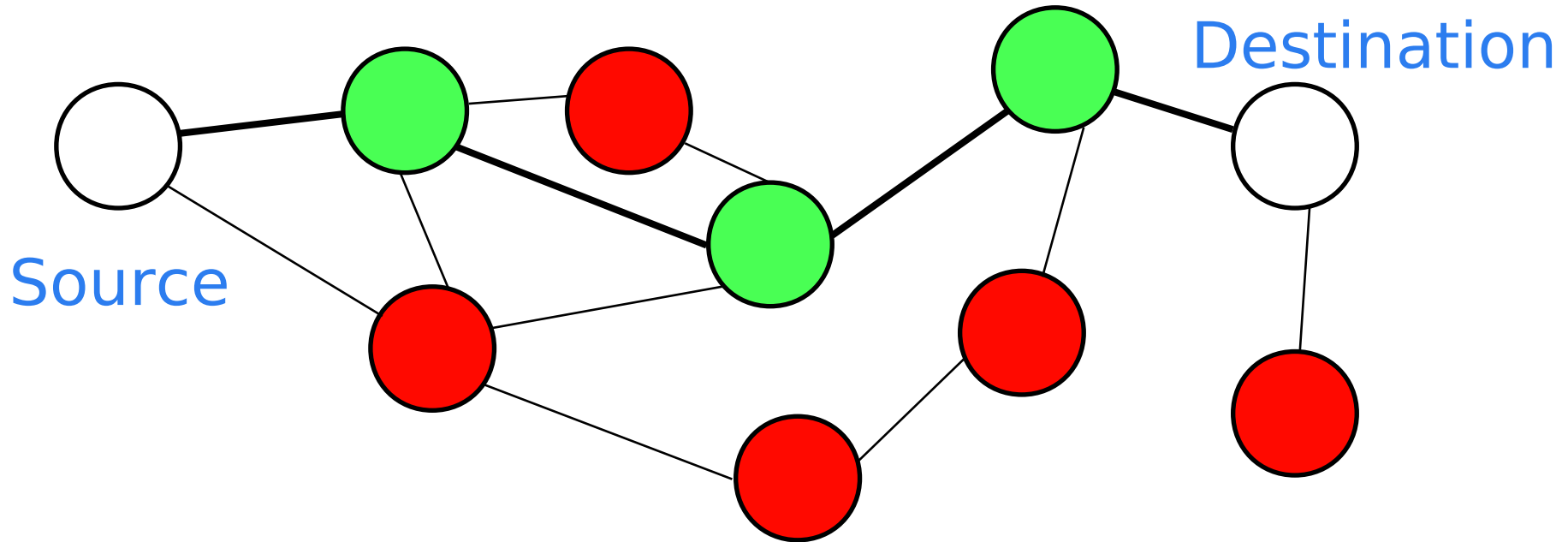
Obfuscated Social Network Routing (OSNR)

- Sender **includes only a one-way hash** (Bloom filter) of their social network.

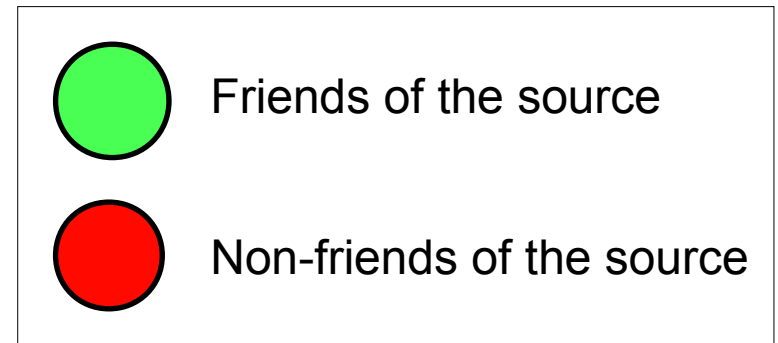
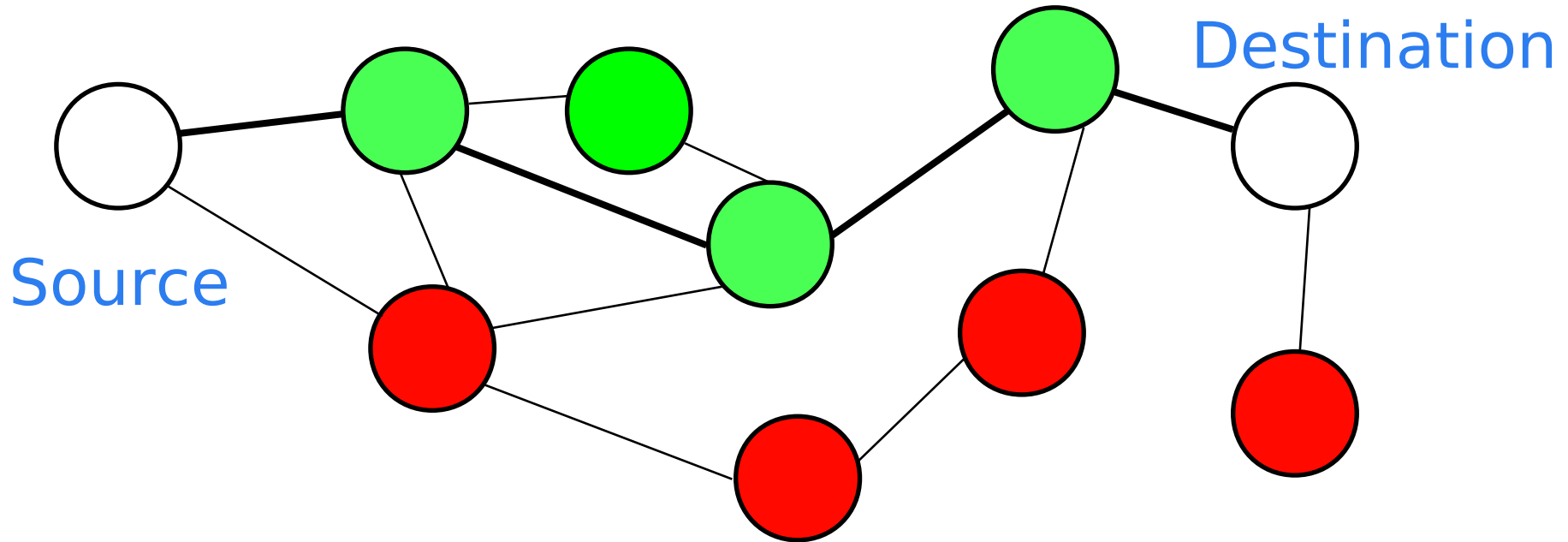
Combined SSNR-OSNR

- First apply SSNR, then apply OSNR.

SSNR Example



SSNR Example



OSNR – Bloom filter

A Bloom filter is used to obfuscate the sender's social network; it is essentially a **one-way hash of the sender's social network**.

Sender puts **node identifiers of friends** into a newly generated message's Bloom filter.

To decide whether to forward a message to an encountered node, the message's Bloom filter is **queried with the encountered node's identifier**.

- **False positives** are possible, with low probability
- False negatives are not possible
- **Reversing** the Bloom filter is **not practical**

Performance evaluation

Trace-driven simulation, using two datasets:

- **SASSY** – Collected in **St Andrews, UK**
 - Movement trace → encounters when in proximity (10m)
 - Social network information: Facebook friends
 - 25 people
 - **Dense** (many encounters; large social networks)
- **Reality Mining** – Well-known dataset from **MIT, USA**
 - Bluetooth encounters
 - Social network information: from the address books
 - 52 people (originally ~100, but not all had social network info)
 - **Sparse** (few encounters; small social networks)

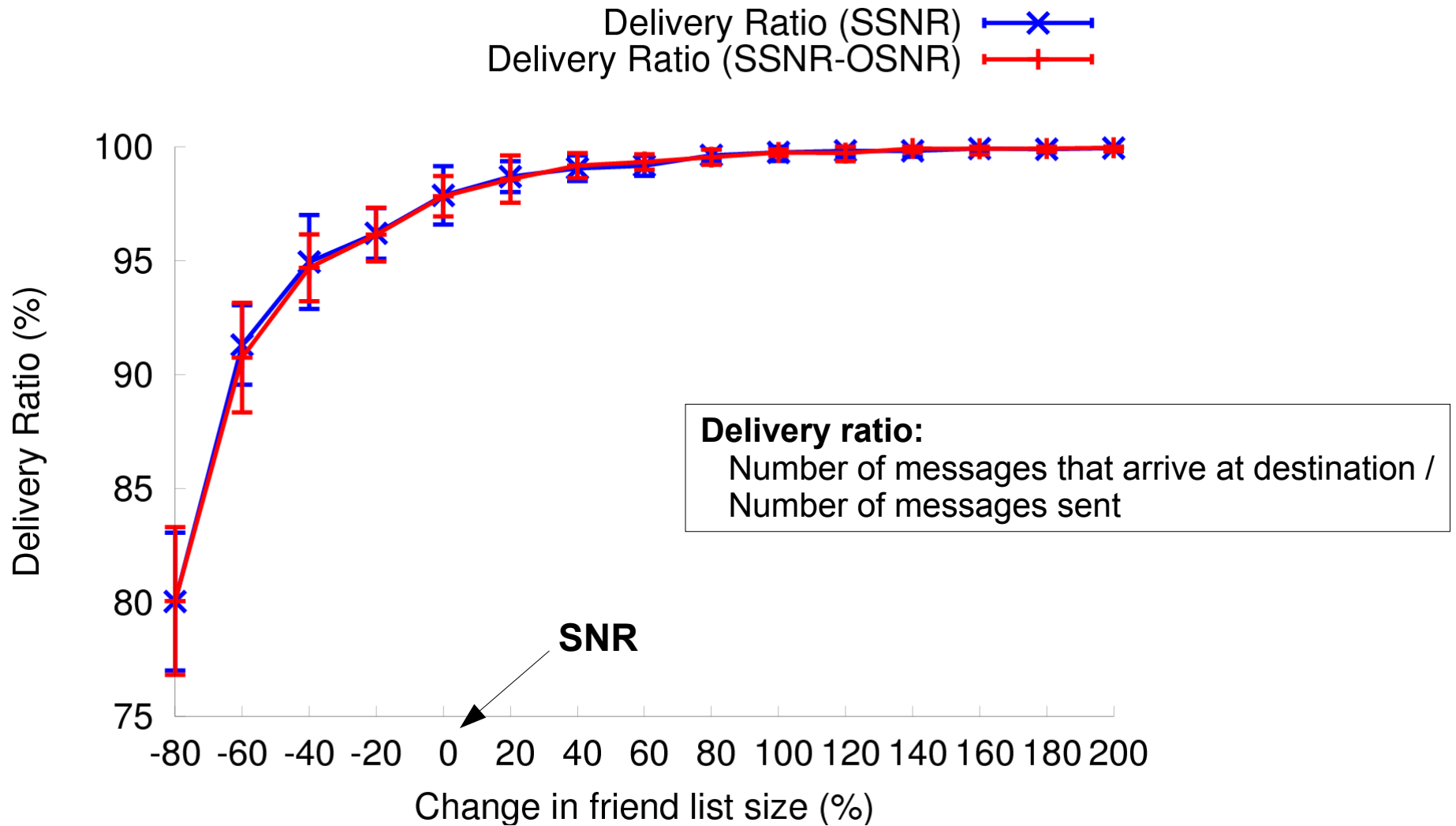
Performance evaluation

Simulation parameters

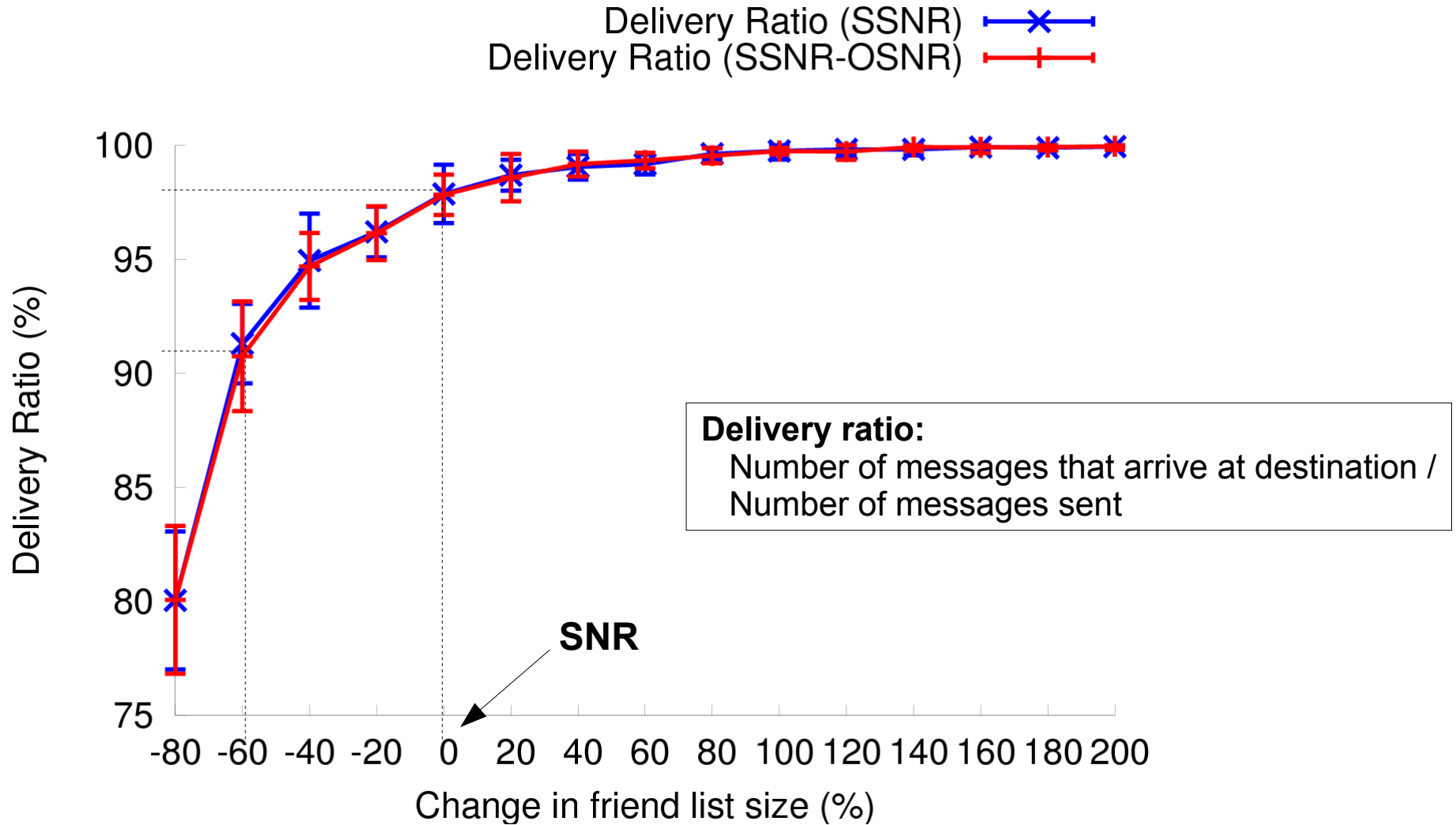
- **10 runs**
- **30 days** simulated
- 30 messages/day, total: **900 messages**
- TTL of messages: **1 day**
- **Infinite buffers**

Full simulation details in the paper...

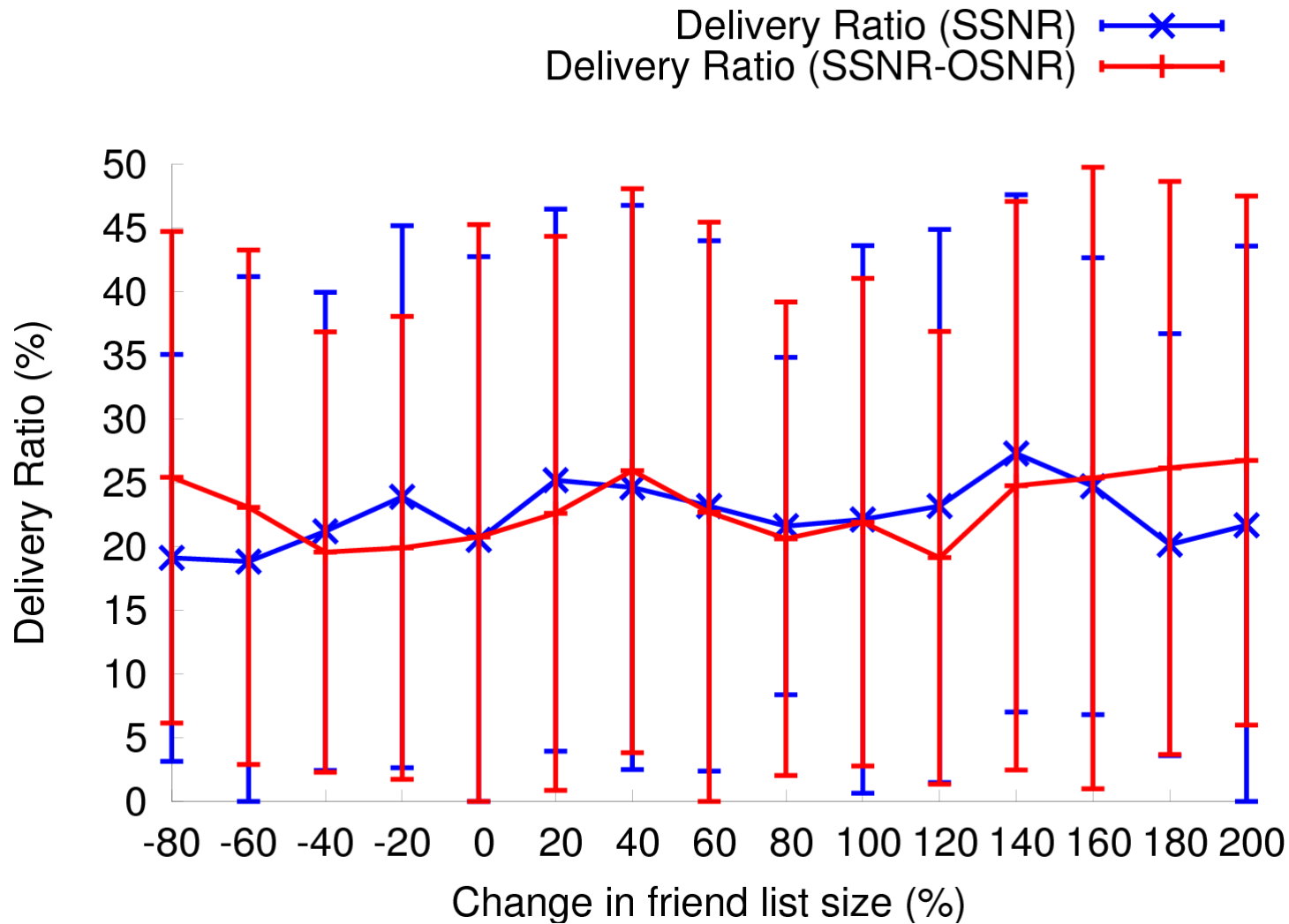
Results: Delivery ratio – SASSY



Results: Delivery ratio – SASSY



Results: Delivery ratio – Reality Mining



Take home messages

- We can obfuscate a sender's social network by **removing up to 60% of the nodes**, while only reducing delivery ratio by **10%**.
- **Bloom filters** can prevent trivial eavesdropping of social network information with only a **minimal impact on delivery ratio**.

What next?

Future refinement

- Other **data sets**
- **Formal analysis**: how much deniability do our schemes provide, formally?

The big picture

- Is social network privacy really a big deal?
- What threats are people **worried about**? How worried? Willing to make a performance trade-off?
- To find out: **user studies!**

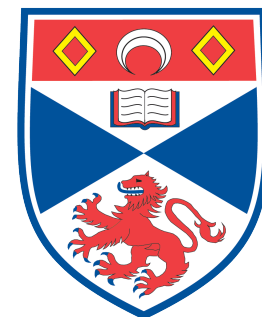


Privacy-enhanced social network routing in opportunistic networks

Iain Parris, Greg Bigwood, Tristan Henderson

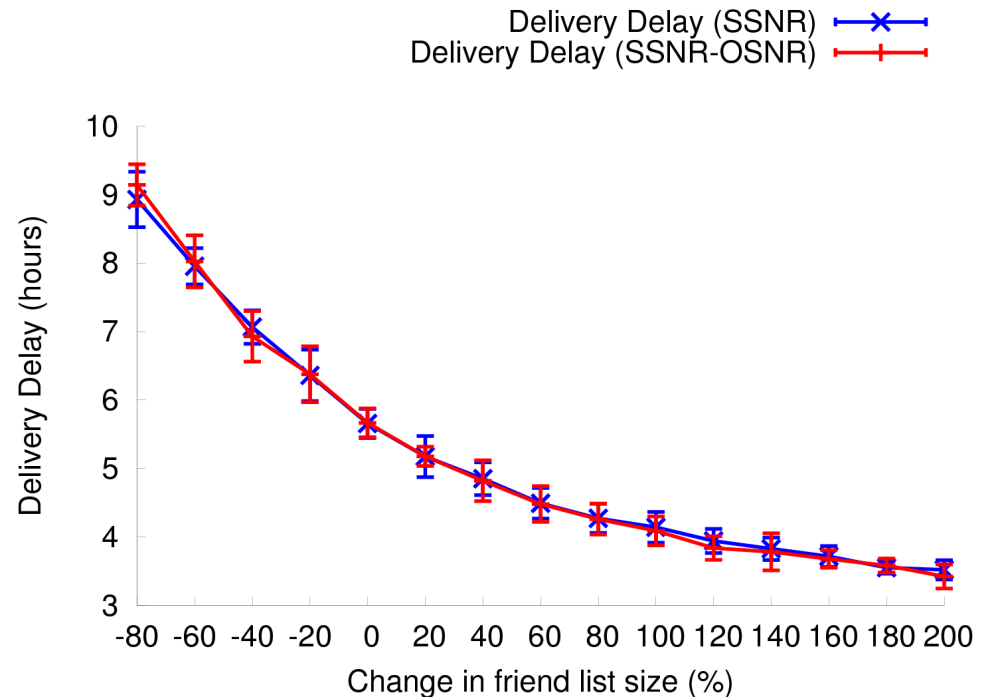
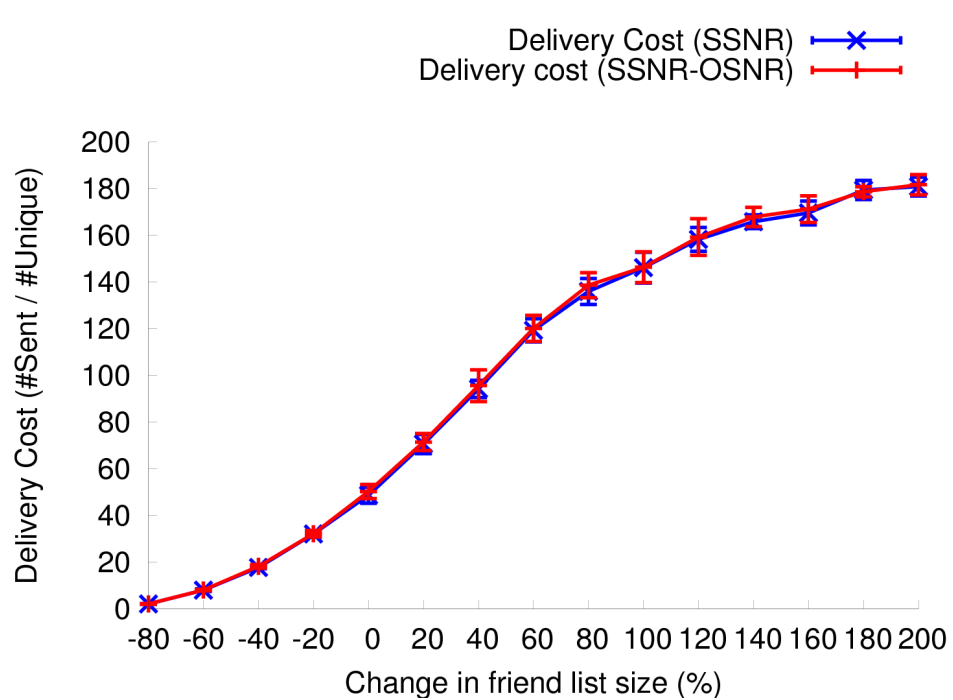
{ip,gjb,tristan}@cs.st-andrews.ac.uk

<<http://www.cs.st-andrews.ac.uk/~ip/>>

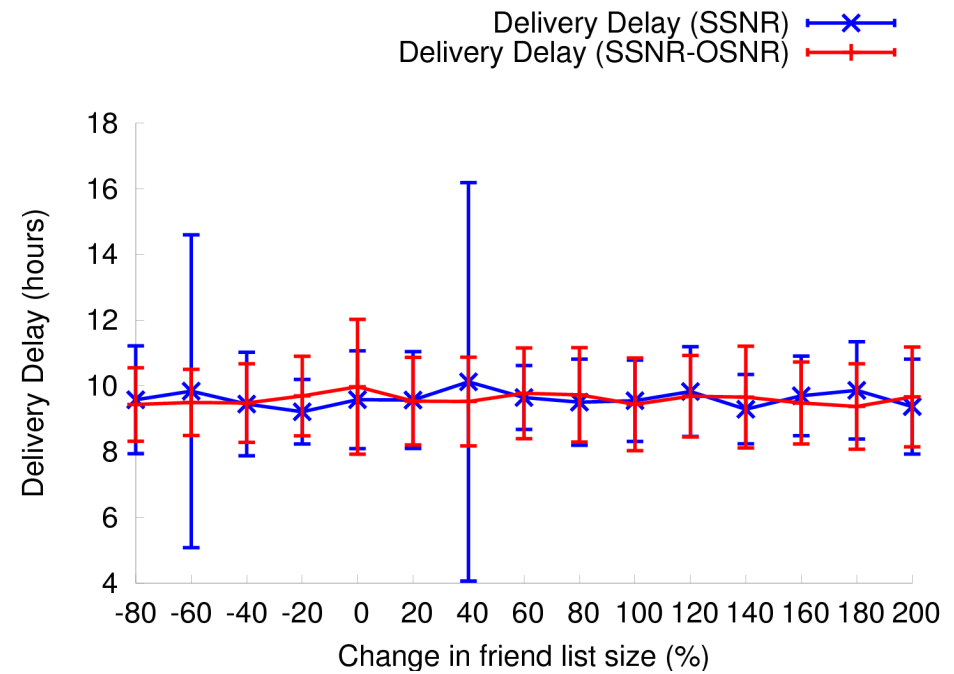
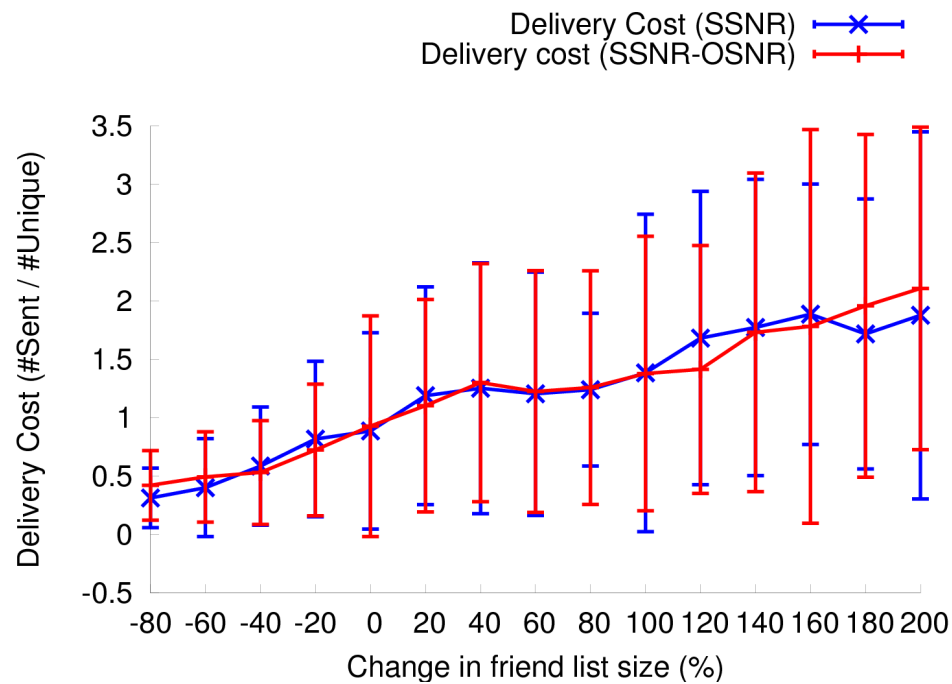


University
of
St Andrews

Results: SASSY



Results: Reality Mining



Routing performance after social network obfuscation

Dataset 1: **SASSY**

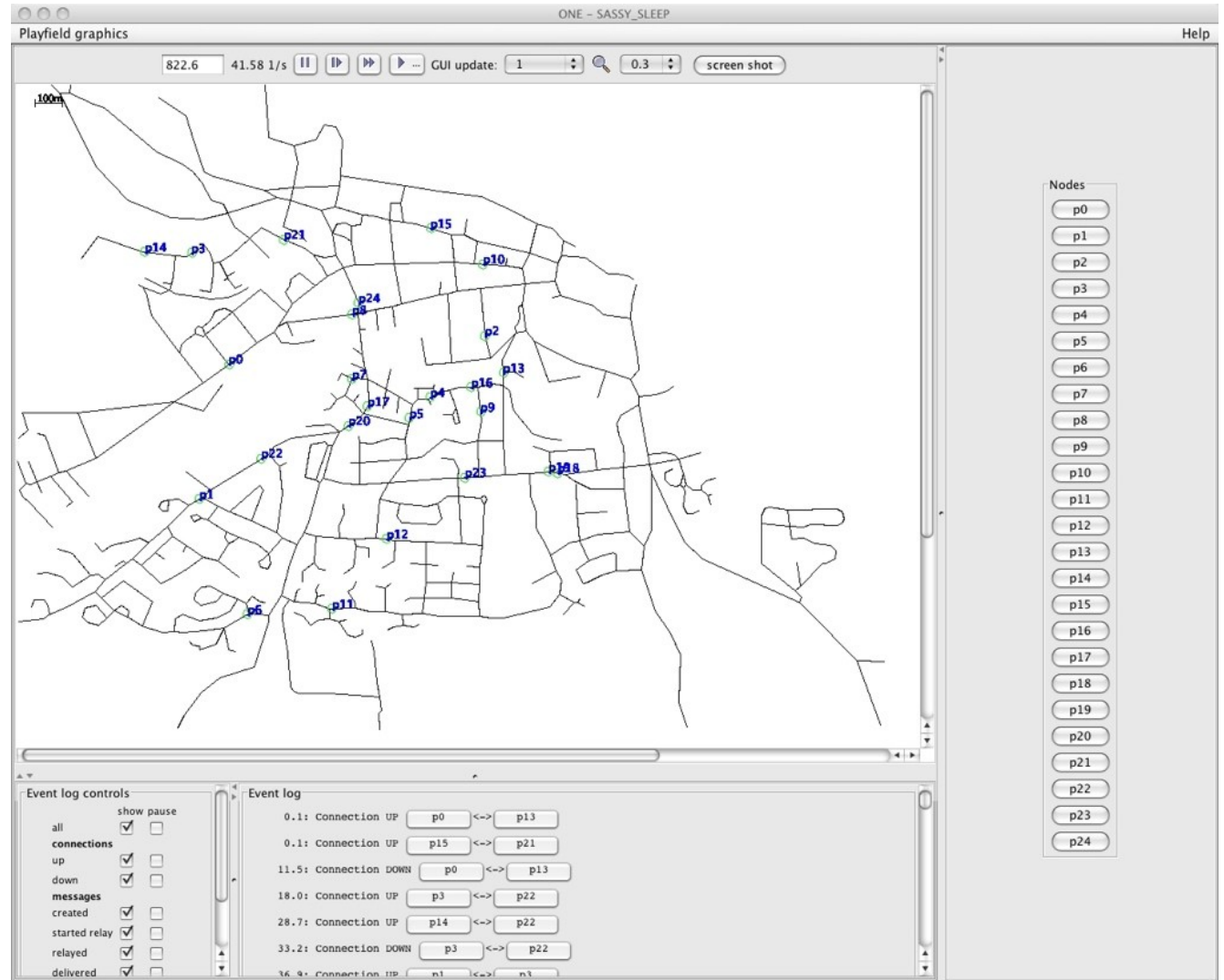
- Collected by us in a previous experiment
- 25 people carried T-motes
- Raw trace was sparse; augmented using a working-day & augmented random-waypoint model, → movement trace.
- **Encounters**: People in proximity (10 metres)
- **Social network**: Facebook friends
- Trace driven simulation using **ns-2**
- Final dataset: **dense** (many encounters, and most people knew each other)

Routing performance after social network obfuscation

Dataset 2: Reality Mining

- Well known dataset, collected at MIT
- **Encounters:** Bluetooth encounters
- **Social network:** From **address books**
- ~100 people carrying mobile phones; 52 had social network information from their address book that we could use
- No movement data, so parsed Bluetooth encounters using a Python program
- Final dataset: **sparse** (few encounters, and most people did not know each other)

Mobility Traces



Credits

Photos used under the Creative Commons license:

- cobalt123 - SamsungSCH-u740 Cellphone:
<<http://www.flickr.com/photos/cobalt/1339314355/>>
- mujitra - Cyber-shot cellphone "W61S" (2008):
<<http://www.flickr.com/photos/mujitra/2723986495/>>
- bogenfreund - The Mechanic Eye:
<<http://www.flickr.com/photos/bogenfreund/1808719569/>>

Logos

- Twitter, <<http://twitter.com/>>
- Facebook, <<http://www.facebook.com/>>
- Google Buzz, <<http://www.google.com/buzz>>