Introduction to Traffic Analysis

George Danezis
University of Cambridge,
Computer Laboratory

Outline

- Introduction to anonymous communications
- Macro-level Traffic Analysis
- Micro-level Traffic Analysis
- P2P Traffic Analysis
- `Extreme' Traffic Analysis
- Conclusions

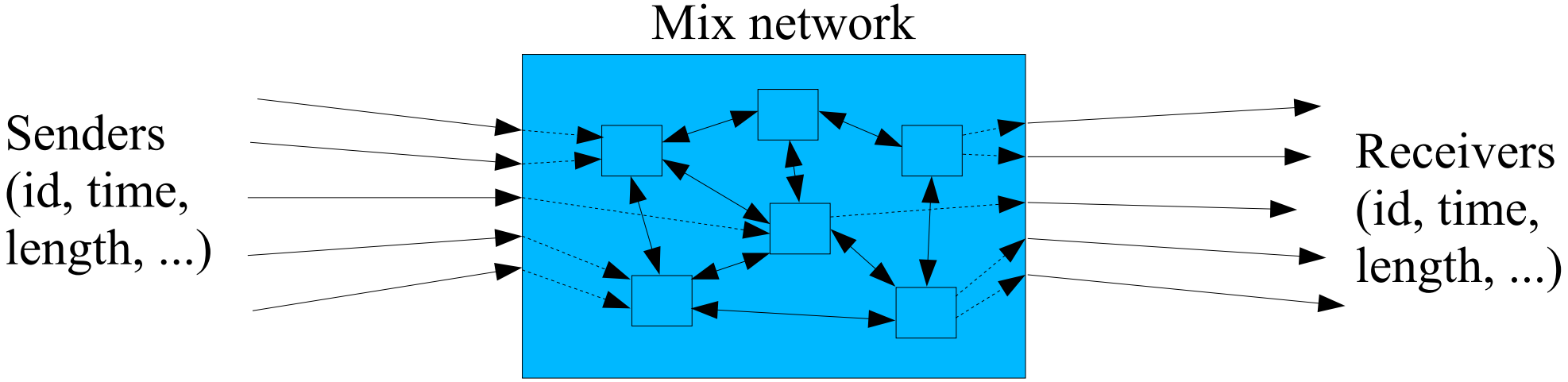
Anonymous communications – Abstract model

- Channel Objective: hide the identity of the sender (or receiver, or both)

- Make the bit patterns of inputs and outputs different (bitwise unlinkability)
- Destroy the timing characteristics (traffic analysis resistance).

Anonymous communications – (not so) Intimate Details

- Mix networks are an efficient solution to build such a blue box:



- Use many mixes to relay messages or streams
- Distribute load and trust
- Each hides correspondence between input and output

Meet the Adversary

- Objective 1: **Identification**.
  For a given message find out who sent it.
- Objective 2: **Linking/profiling**.
  Two messages – same (even unknown) sender?
- **How:** Observe all traffic, modify and inject any message, control some nodes (misbehave), destroy some node.
- Objective 3: **Disrupt** the network.
  (useability = security in anonymity)

The cryptanalysis of anon. comms.

- Input and output messages have to look different.
- Public key crypto used for mix packet formats: key distribution and distributed decoding.
- Adversary can try to replay, observe, modify (tag) messages to extract info about destination.
- As close as it gets to adaptive chosen ciphertext attacks: message is ciphertext, mix is decryption oracle.
- Don't reinvent: Mixmaster, Mixminion, Minx, Tor,...

What is traffic analysis

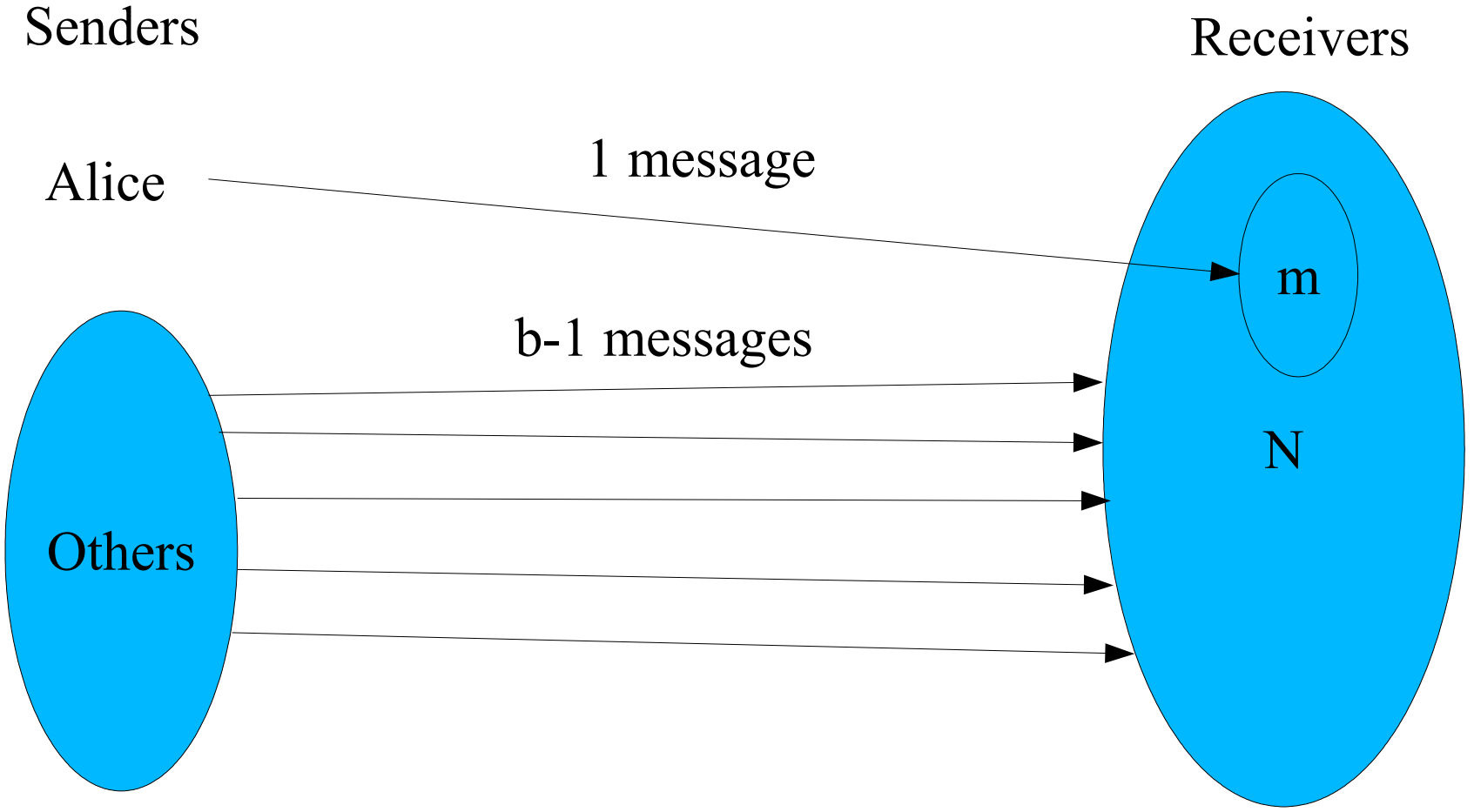
- Bit-patterns can be made not to leak any info.
- “The art of using communication meta-data to infer information about network nodes identities or characteristics, their relations or actions in the network.”
- Can be used: Radio comms, Trace IP traffic, HTTP log analysis, Spam filtering, ...
- Compromise the security properties of hardened systems.

Statistical Disclosure Attacks - Intro.

- Lets first attack the abstract model – no details.
- We consider an anonymous channel as a black box that works in rounds: it takes a batch of b messages, shuffles them and outputs them.
- Each round Alice sends a message to one of her friends with equal probability (prob 1/|m|).
- Others send to anyone with equal probability (1/N).
- Attacker: Who are Alice's friends?

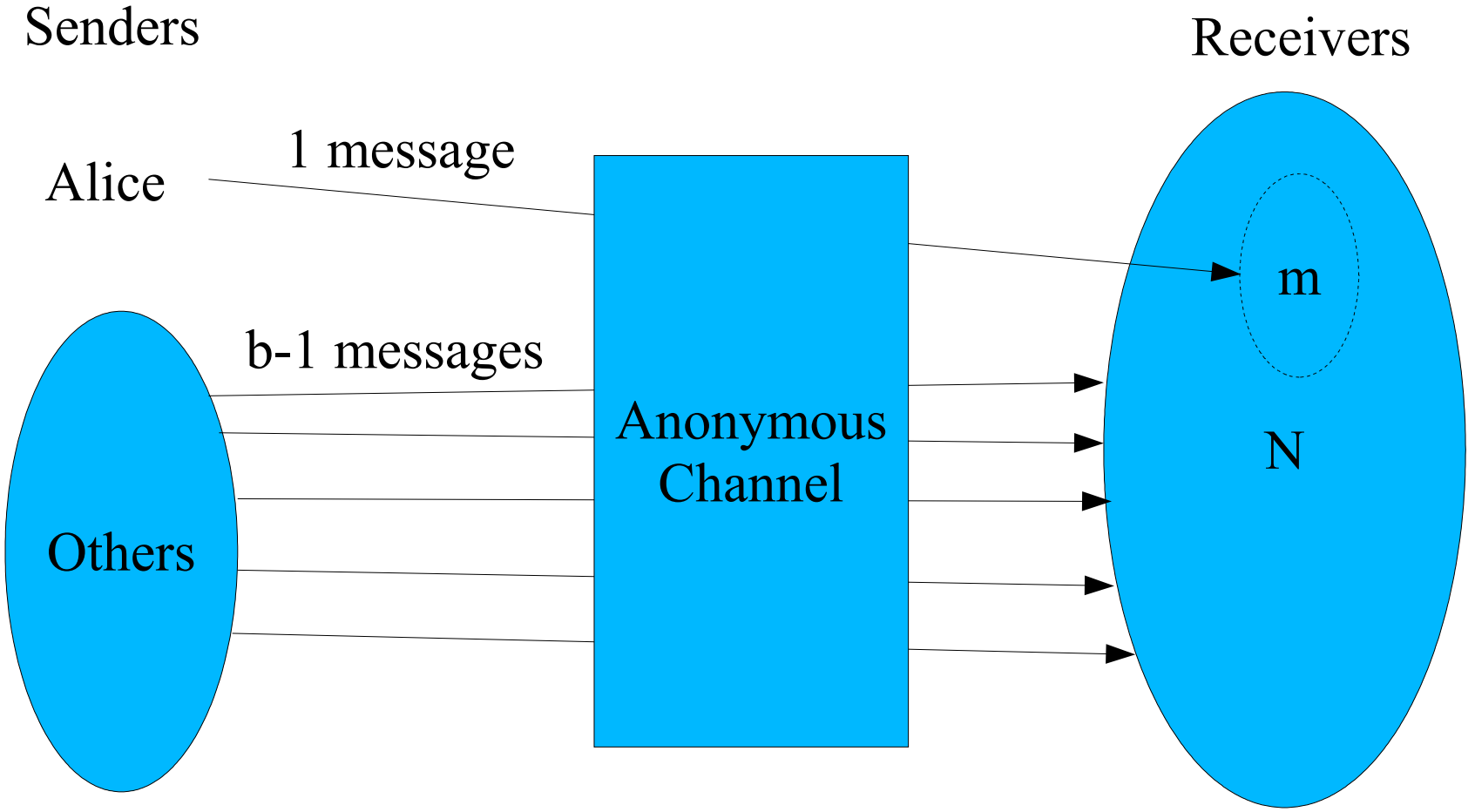
SDA – Usage model

- Alice and other correspondents:



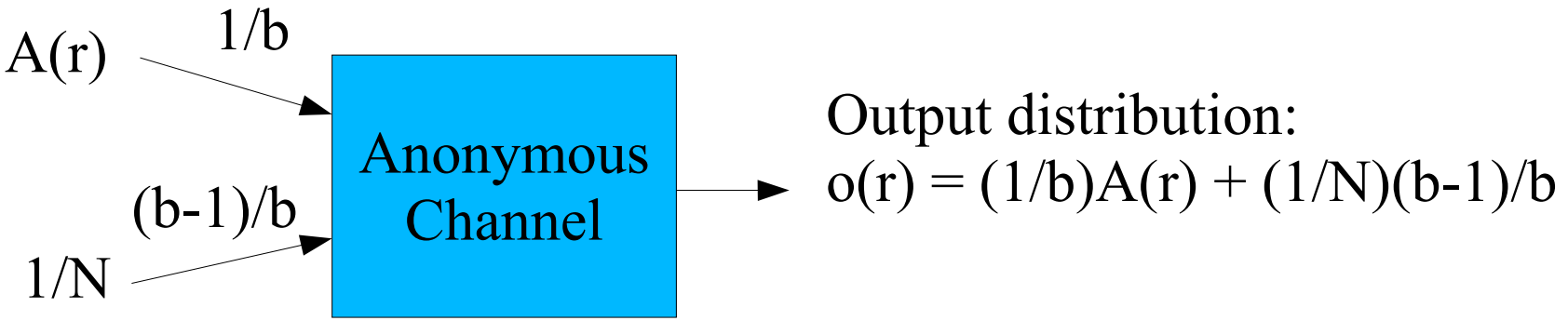
SDA – Usage model 2

- Alice and other correspondents + anon. channel:



SDA – How to do it

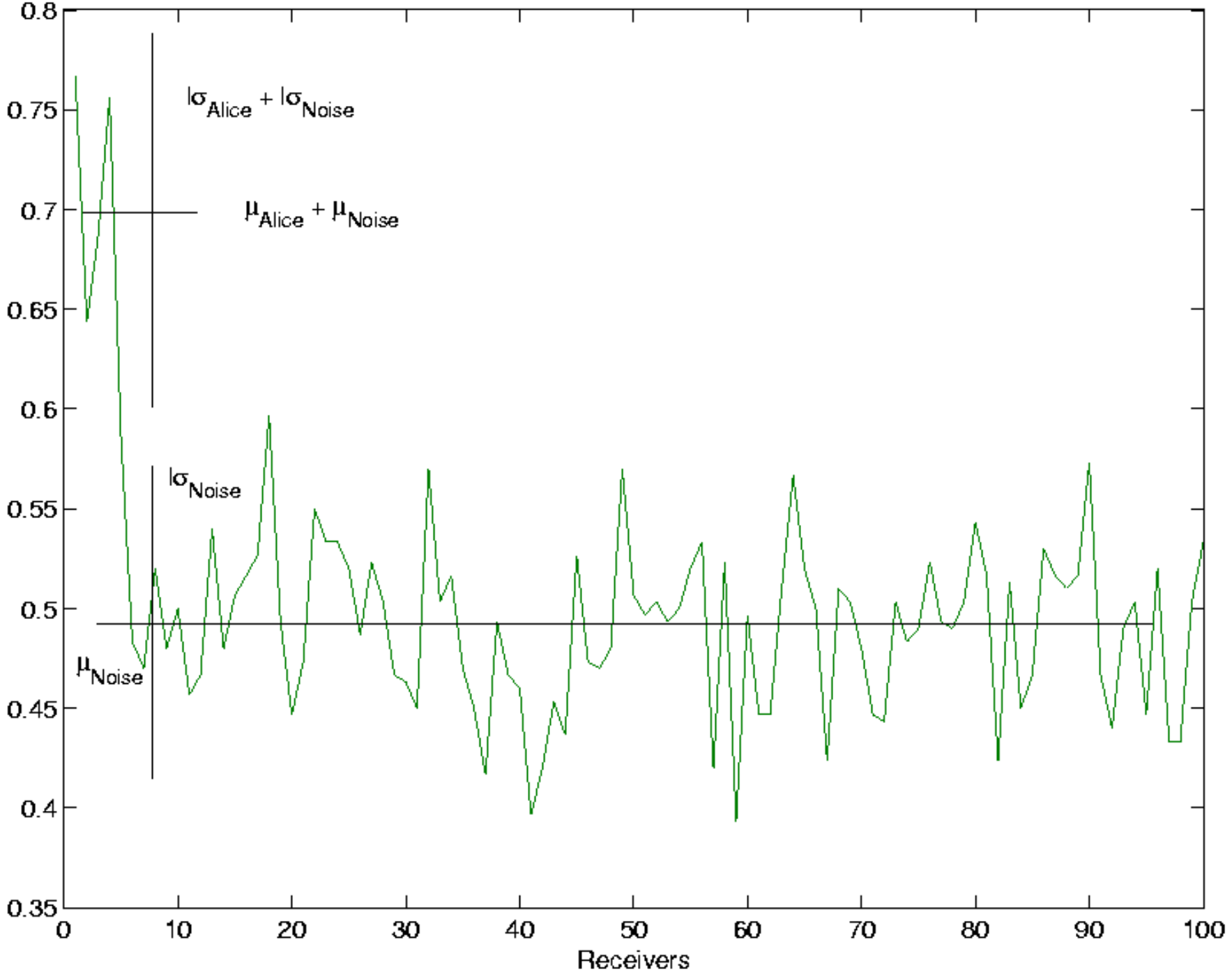
- Attacker objective: Determine the set of Alices friends m.
- How: Observe many rounds – note that Alice's friends will appear more often.
- Quick attack relies on approximations:



- Treat all outputs as samples from that distribution. Estimate A(r)

SDA Analysis

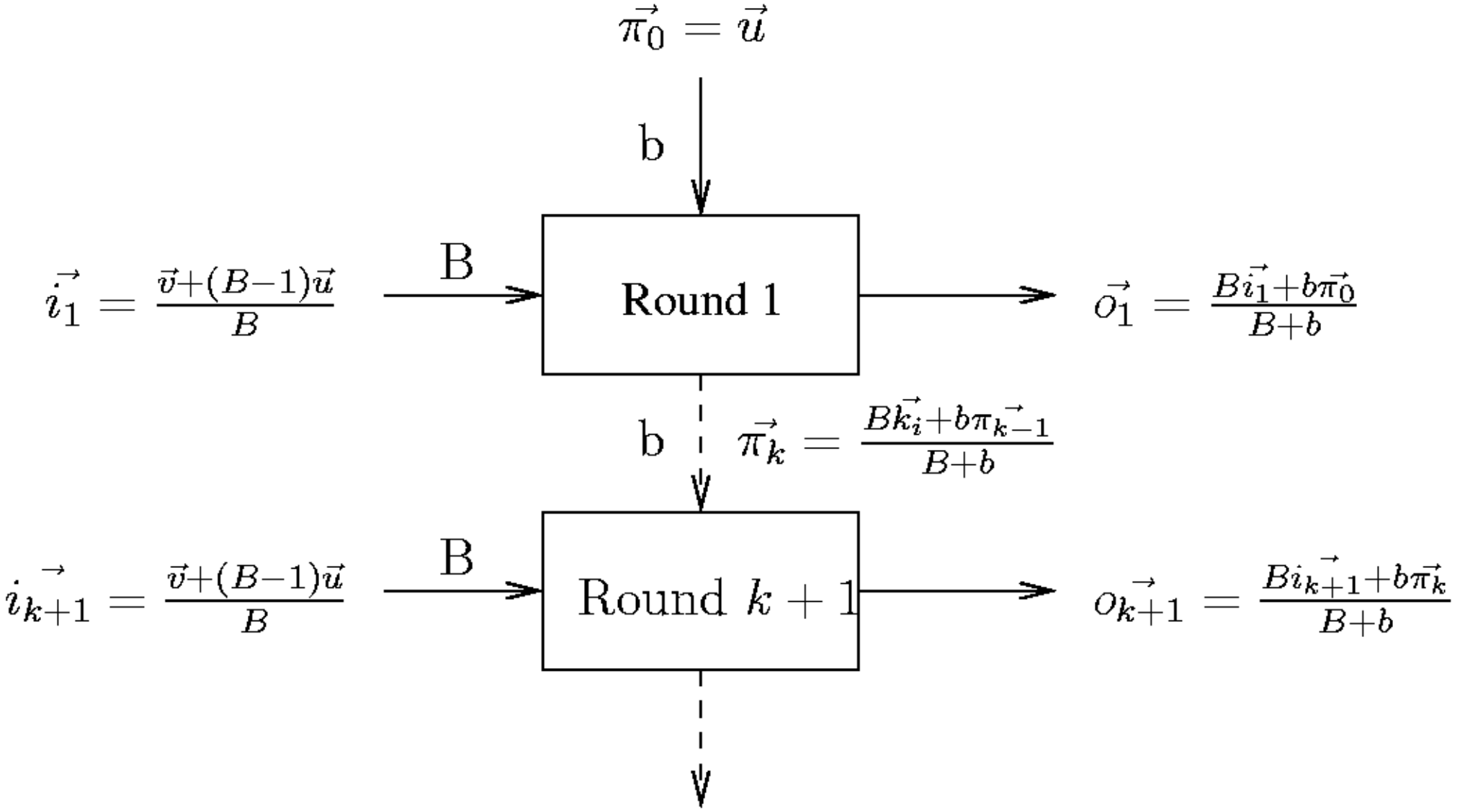
- Could use bayes theorem.
- Can also analyse each receiver separately and classify them:
  - Receiving from Alice: Binomial 1/bm + (b-1)/bN.
  - Receiving only from others: Binomial (b-1)/bN
- See how the number of messages received compares with the expected number.
- Use the standard deviation to calculate false positive or negative probability (binomial).


0.8
0.75
0.7
0.65
0.6
0.55
0.5
0.45
0.4
0.35
0
10
20
30
40
50
60
70
80
90
100
Receivers
lσAlice + lσNoise
μAlice + μNoise
lσNoise
μNoise

Advanced SDA

- Real networks don't work in batches, messages are always in.
- Better modelled as pool mixes – some fraction of the messages always stay in the mix.
- Again approximate the output of the pool mix as a sample from a distribution.
- Use bayes theorem to infer Alice's friends: analysis is much harder – open problem!

Advanced SDA – a peek



More on disclosure attacks

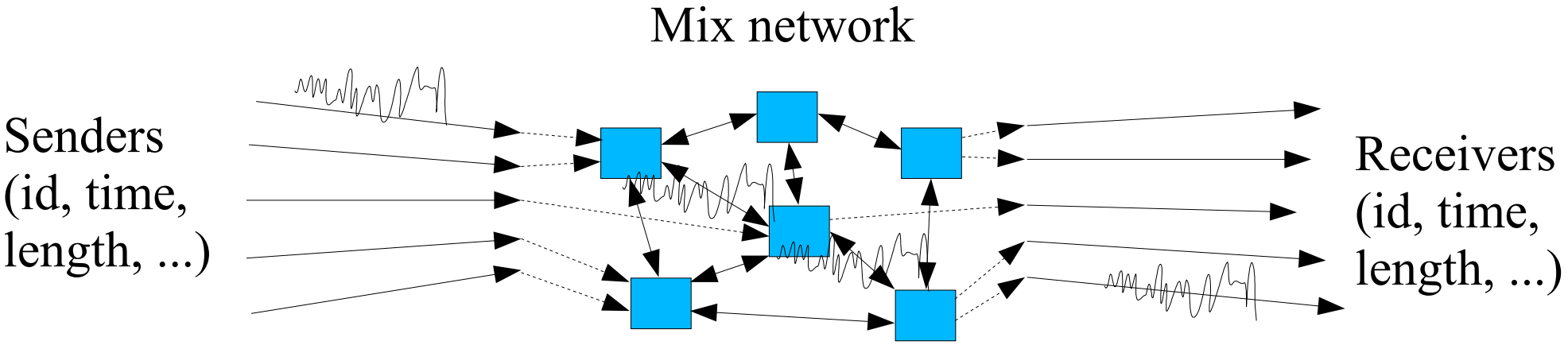
- The original disclosure attack (Kesdogan)
  - Original model – tied to the model.
  - Computationally expensive attack.
  - Hybrid statistical and combinatorial: hitting set.
- Statistical disclosure simulations (Mathewson)
  - SDA works even with partial data (sampling).
  - SDA works even with cover traffic.
  - SDA works better with pseudonyms.
  - Example of applying theory to practice (Mixminion)

SDA Conclusions

- Any anonymous communication channel will reveal long term relationships.
- Look at model carefully: unobservability might help (expensive, offline/online)
- A lot more to do:
  - Analysis of pool mix SDA
  - Collect real user profiles
- SDA takes time – anonymity is tactical!

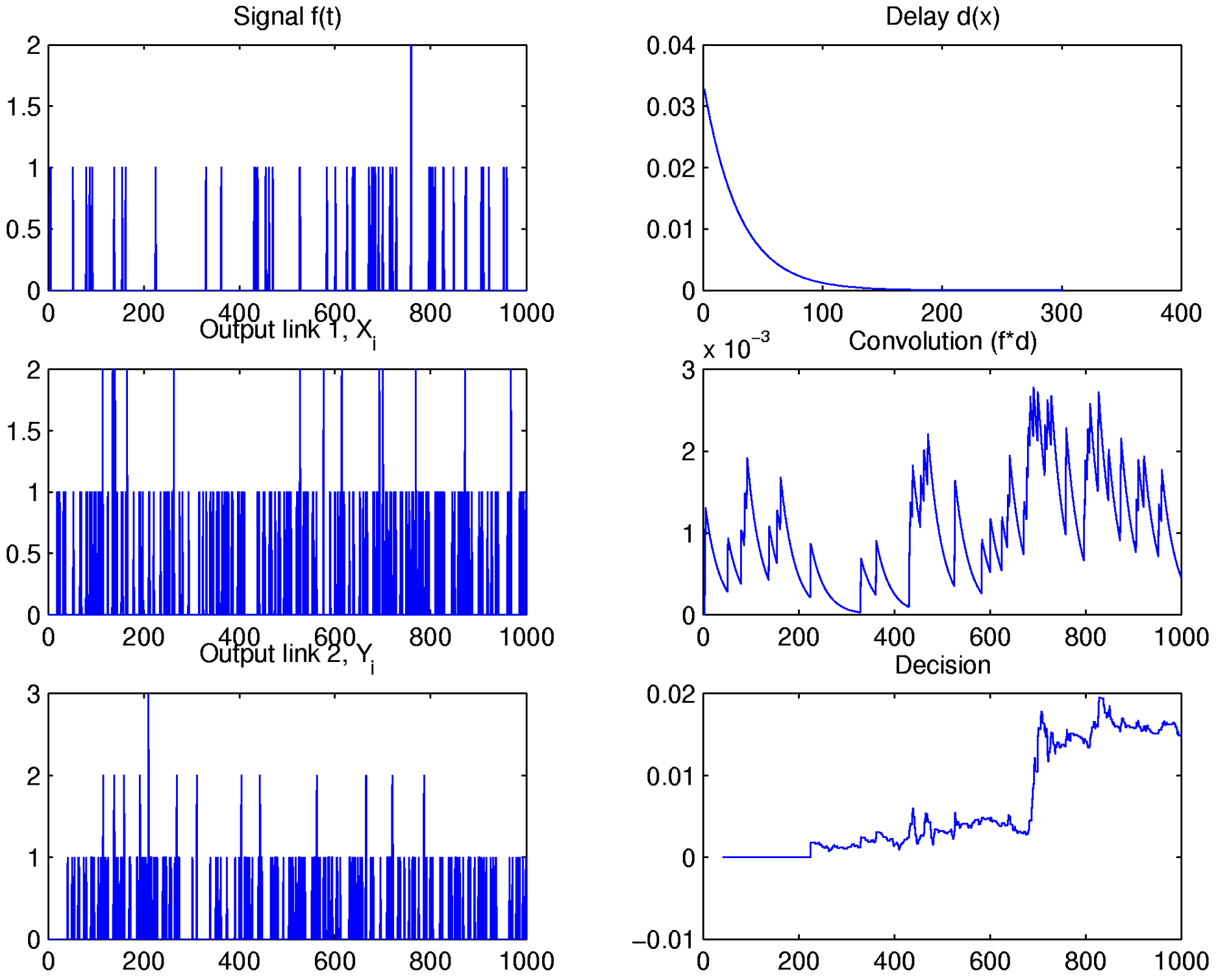
Tracing Streams of Data

- SDA looks at abstract model, a good network would not give more info.
- BUT difficult to mix streams of data without introducing a lot of latency.
- The `shape' of data streams is not changed much.



Modelling anonymous streams

- Streams are not usually mixed amongst them, but delayed individually.
- Optimal strategy: exponential delay of each packet (mean is latency).
- Traffic analysis:
  - Create a model of the stream.
  - Try to detect it in the network.


Signal f(t)
Delay d(x)
Output link 1, X_i
Convolution (f*d)
x 10^{-3}
Output link 2, Y_i
Decision

Analysis

- How well does this attack performs?
- Approximate again:
  - Probability the traffic was generated by template
  - Vs. it was random
- We use standard deviations to find out how likely false positives and negatives are.
- Can we do better?

Other Techniques

- Other stream analysis techniques:
  - Interpacket delay (Umass)
  - Packet Counting (Serjantov)
  - Stop and start of stream (Pfitzmann)
- Active attacks:
  - Induce pattern that is easy to detect/difficult to distort.
  - Like active sonar: send a pulse.
  - Very difficult to protect – cover traffic is expensive.
- Conclusion: against GPA – you lose.

P2P anonymizing systems

- P2P ethics: everyone is equal
  - Crowds: pass the request
  - Tarzan/MorphMix: everyone is a mix.
- Scaling issues:
  - Node discovery
  - Key distribution
- Attack these to reduce anonymity.

Predecessor attack

- Crowds security: you do not know if the previous node started the request.
- BUT the initiator is more likely to be on the path.
- Look for repeated communications again!
- Other systems relying on initiator anonymity can be attacked like that (Gnutella)
- The larger the Crowd the worse the attack!

P2P mixing

- Everyone is a mix.
- Good: large attack surface (no GPA?)
- Bad: Low traffic, key distribution, peer discovery.
- Attack 1: Route Capture
  - You connect to a node and ask it for more nodes.
  - As soon as you hit a bad node it only returns bad nodes.
  - MorphMix: intrusion detection – only works in the long run.

The young Tarzan

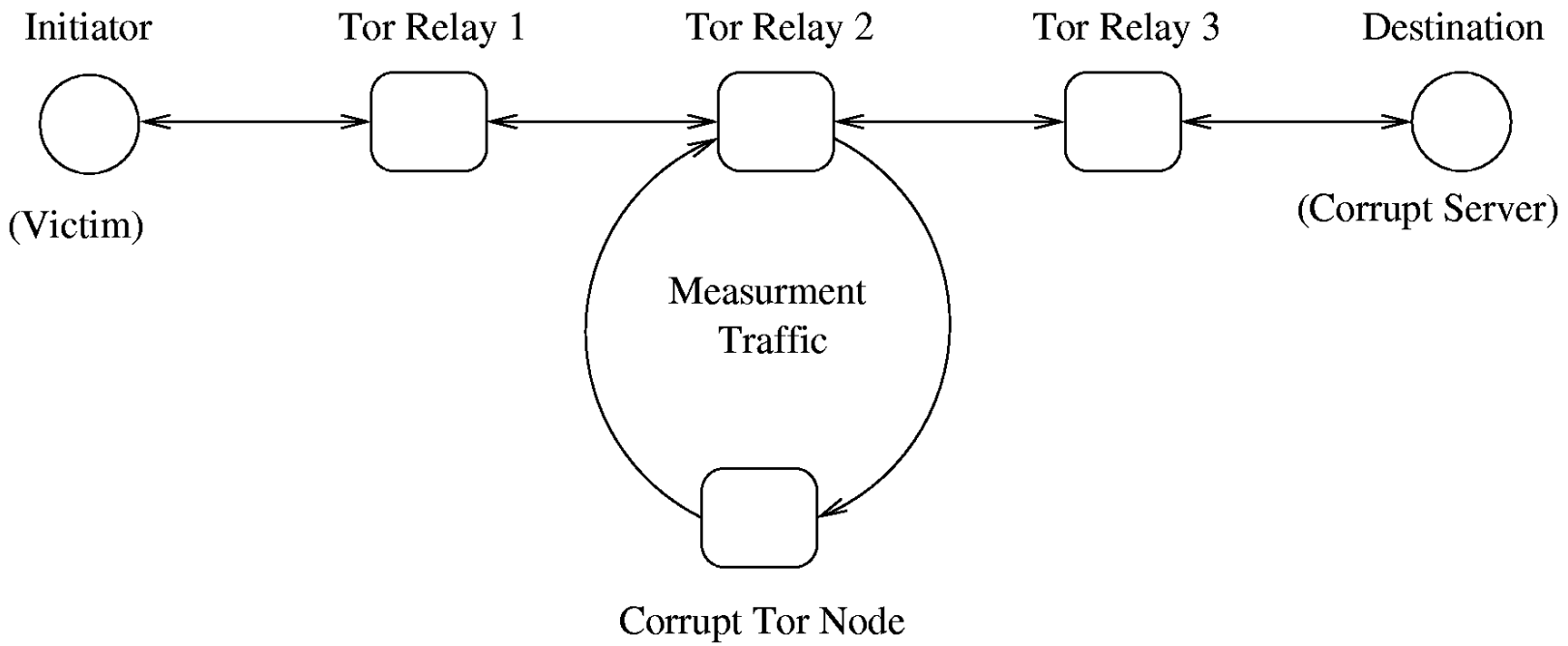
- Attack 2: Knowing only a subset of the peers
  - Each participant discovers a subset of the network.
  - Then chooses a number of nodes to relay messages.
  - If a bad node is on the path it knows 3 nodes (itself, previous and next node).
  - The probability more than one nodes know them is very small.
  - Attack works better with larger networks!
  - Analysis and countermeasures are available.

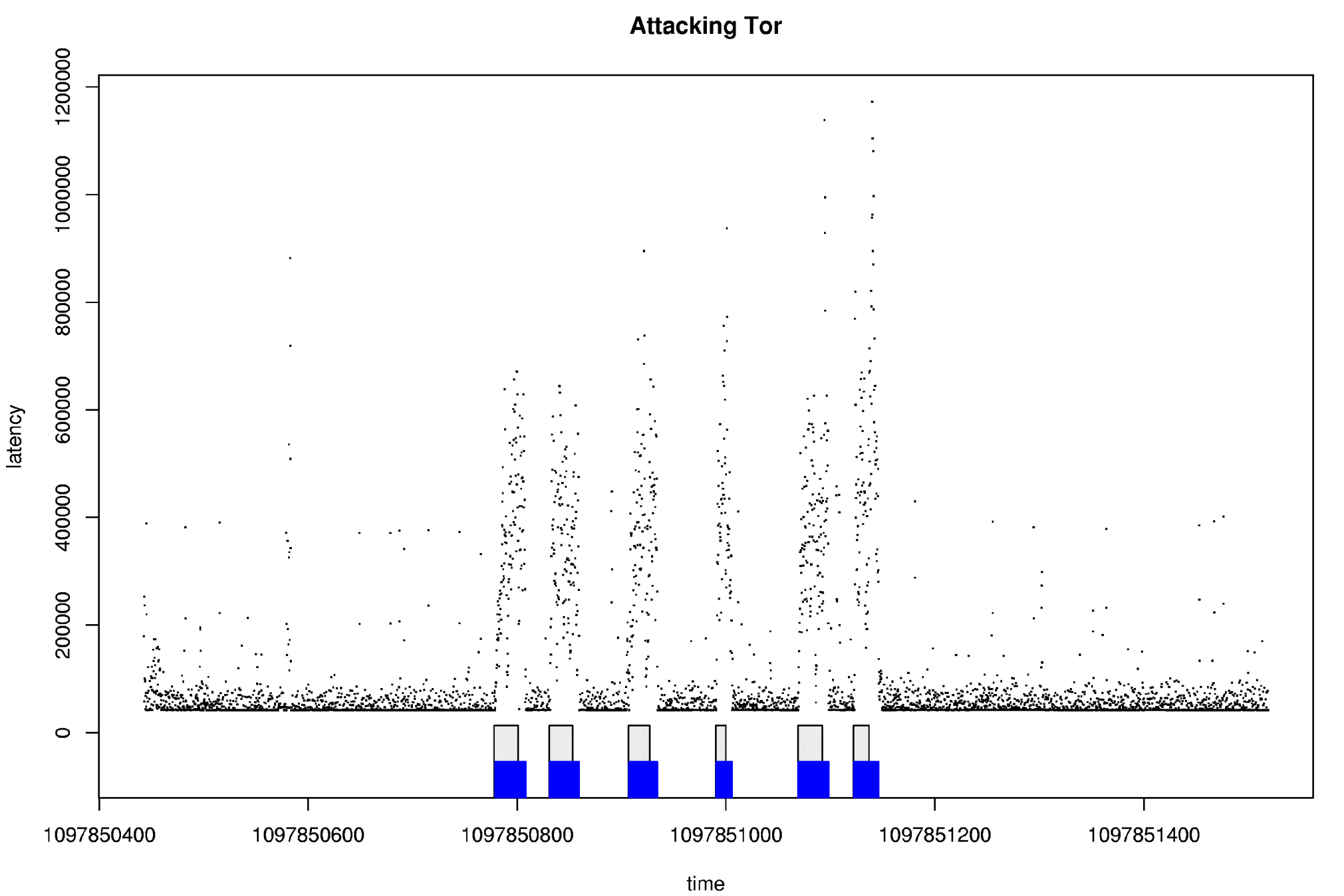
`Extreme' Traffic Analysis

- Impossible to protect against GPA.
- P2P systems make GPA very expensive.
- Tor: Moving beyond the GPA.
- Can an adversary with no access to any network links perform traffic analysis? Yes!
  (similar attacks would work on other low latency systems)

How to do TA from home

- The problem: how do you measure remotely the traffic volume on network links you have no access to?
- The setup:




Attacking Tor
latency
0
200000
400000
600000
800000
1000000
1200000
1097850400
1097850600
1097850800
1097851000
1097851200
1097851400
time

Overall conclusions

- There are fundamental limits to the anonymity of repeated communications.
- Stream based protocols: Trivial traffic analysis attack defeat them – cover traffic is expensive.
- P2P is no silver bullet: new attacks
- The GPA is everyone!
- Young field – in need of serious research.