Bayesian Inference and Traffic Analysis

Carmela Troncoso
George Danezis

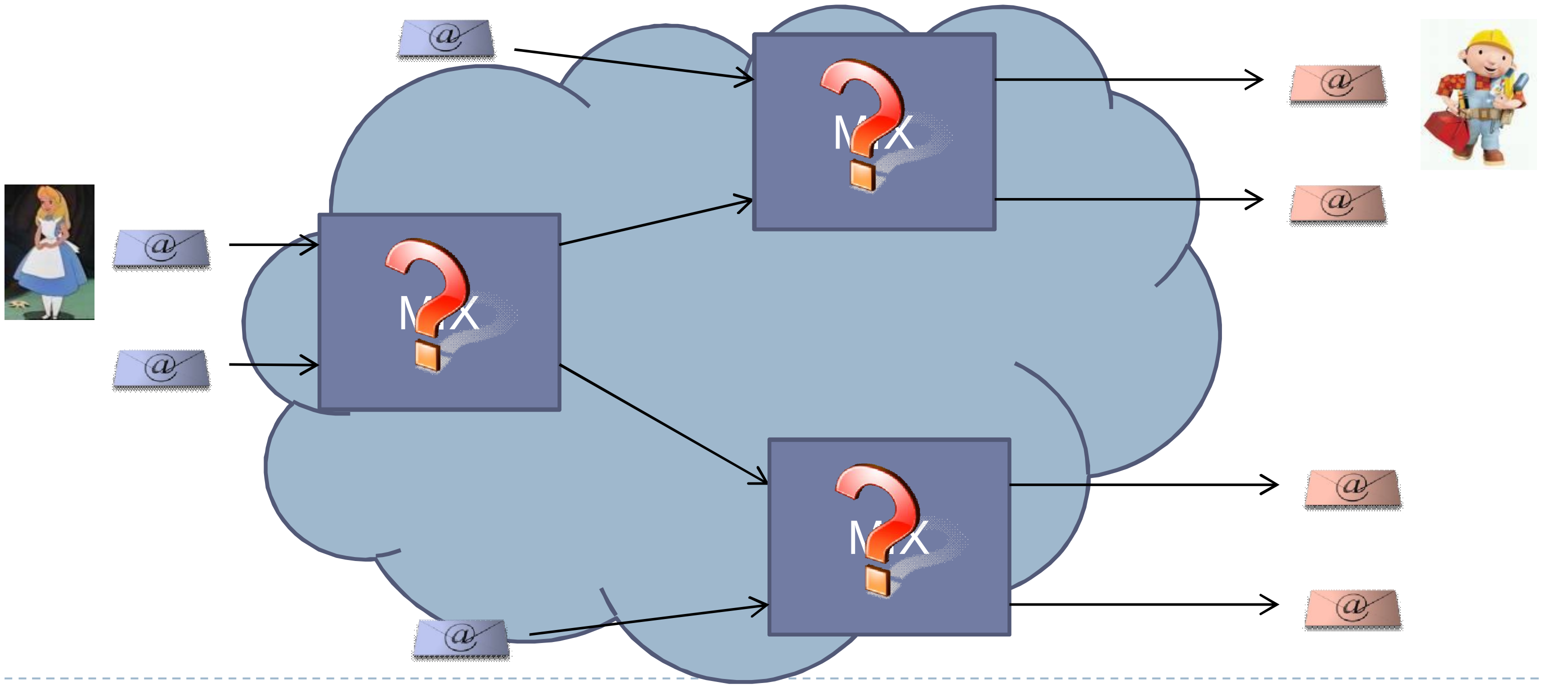
University of Texas at Arlington 16/11/2009
Microsoft Research Cambridge/ KU Leuven(COSIC)

Outline

- The traffic analysis problem and a bayesian approach
  - What is this all about
- Modelling mix networks
  - The boring part of the talk
- Markov Chain Monte Carlo methods and traffic analysis
  - The exciting part of the talk
- Evaluation and results
  - It actually works!
- Conclusions

Mix networks

- Mixes are combined in networks in order to
  - Distribute trust (one good mix is enough)
  - Load balancing (no mix is big enough)



Attacks against mix networks

- Uncover who speaks to whom
  - Observe all links (*Global Passive Adversary)*

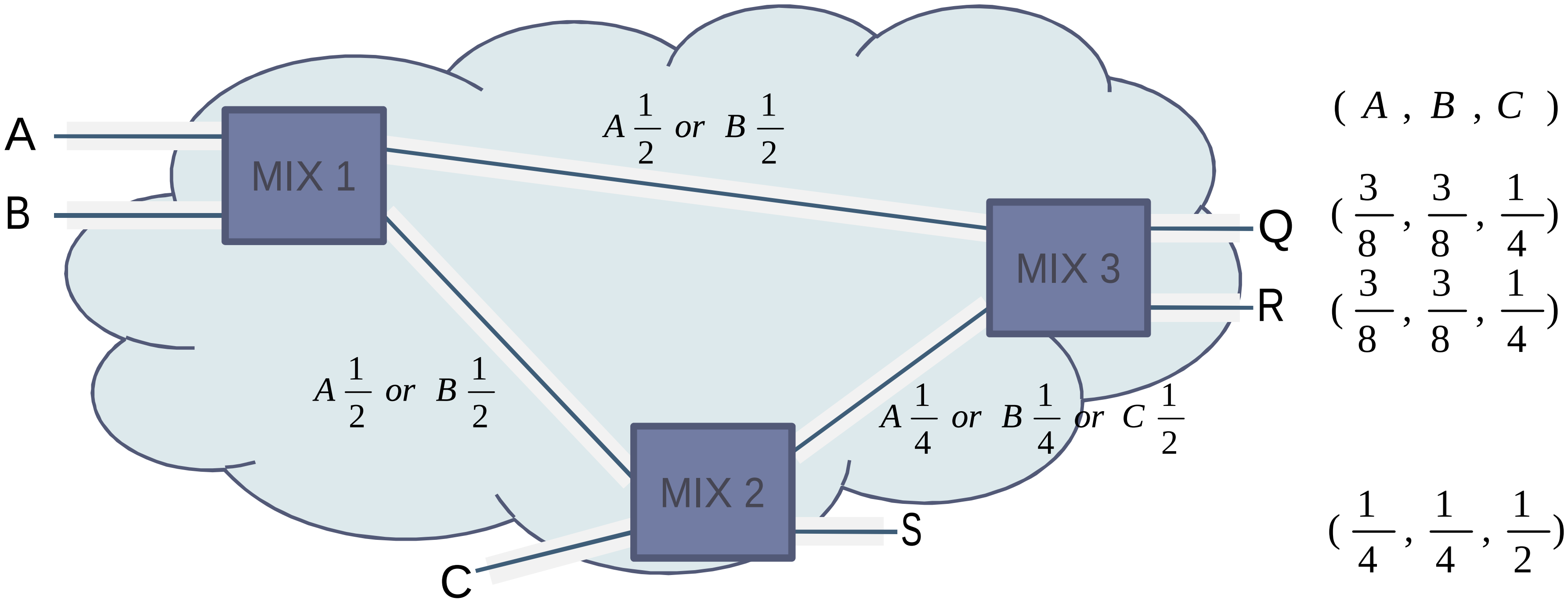
  - Restricted routes [Dan03]
    - Messages cannot follow any route
  - Bridging and Fingerprinting [DanSyv08]
    - Users have partial knowledge of the network

  - Long term disclosure attacks:
    - Exploit persistent patterns
    - Disclosure Attack [Kes03], Statistical Disclosure Attack [Dan03], Perfect Matching Disclosure Attacks [Tron-et-al08]

- Based on heuristics and specific models, not generic

Mix networks and traffic analysis

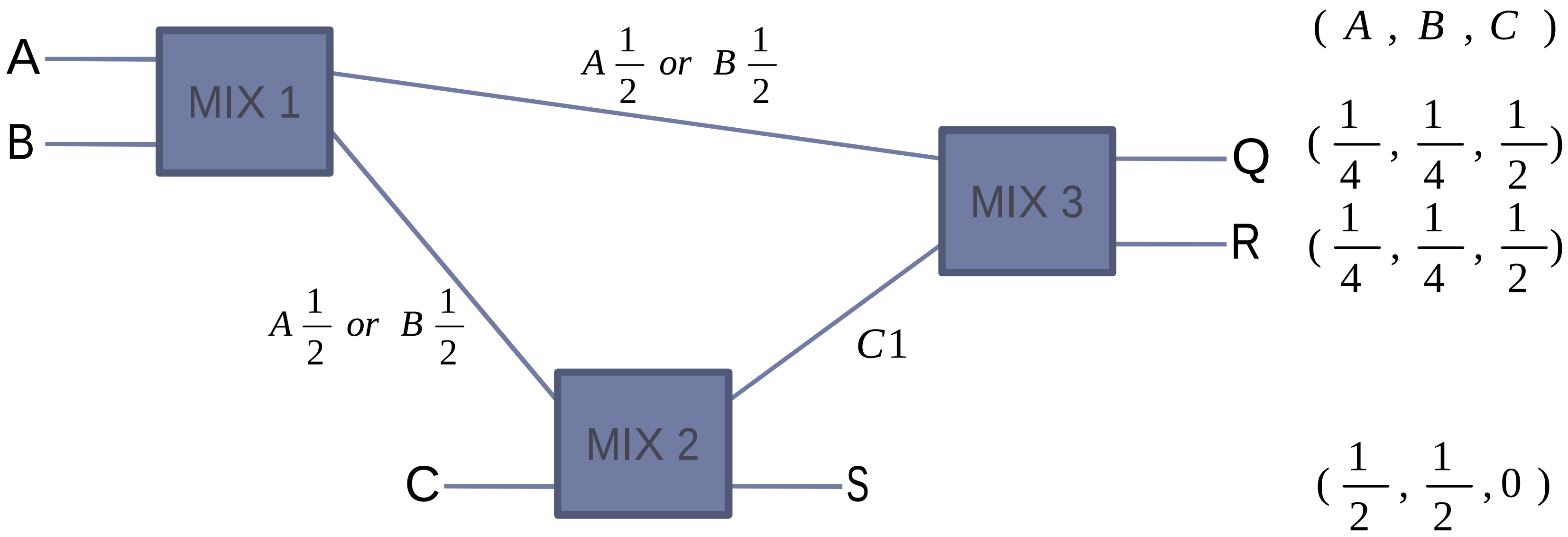
- Determine probability distributions input-output



- Threshold mix: collect t messages, and outputs them changing their appearance and in a random order

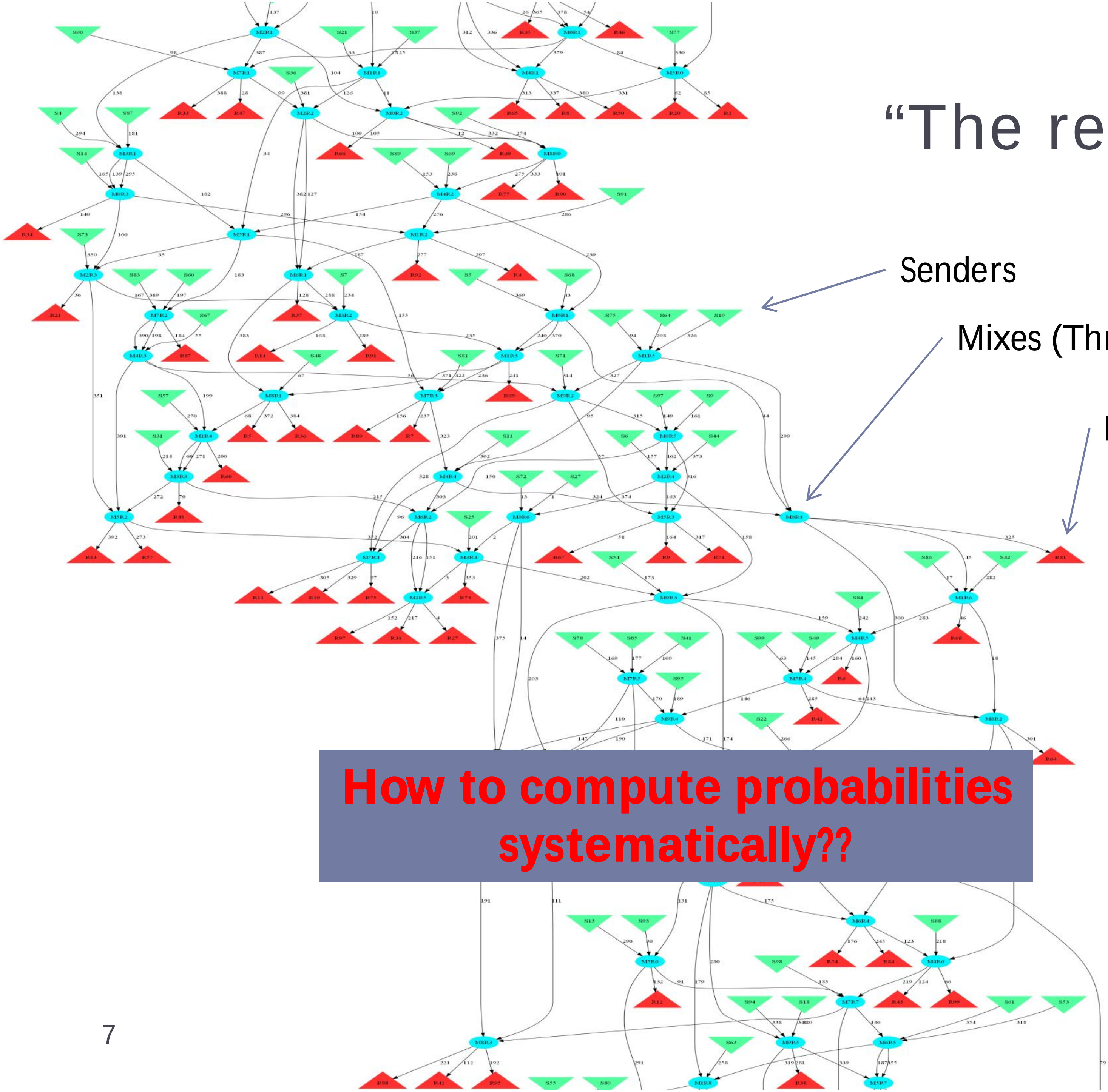
Mix networks and traffic analysis

- Constraints, e.g. length=2



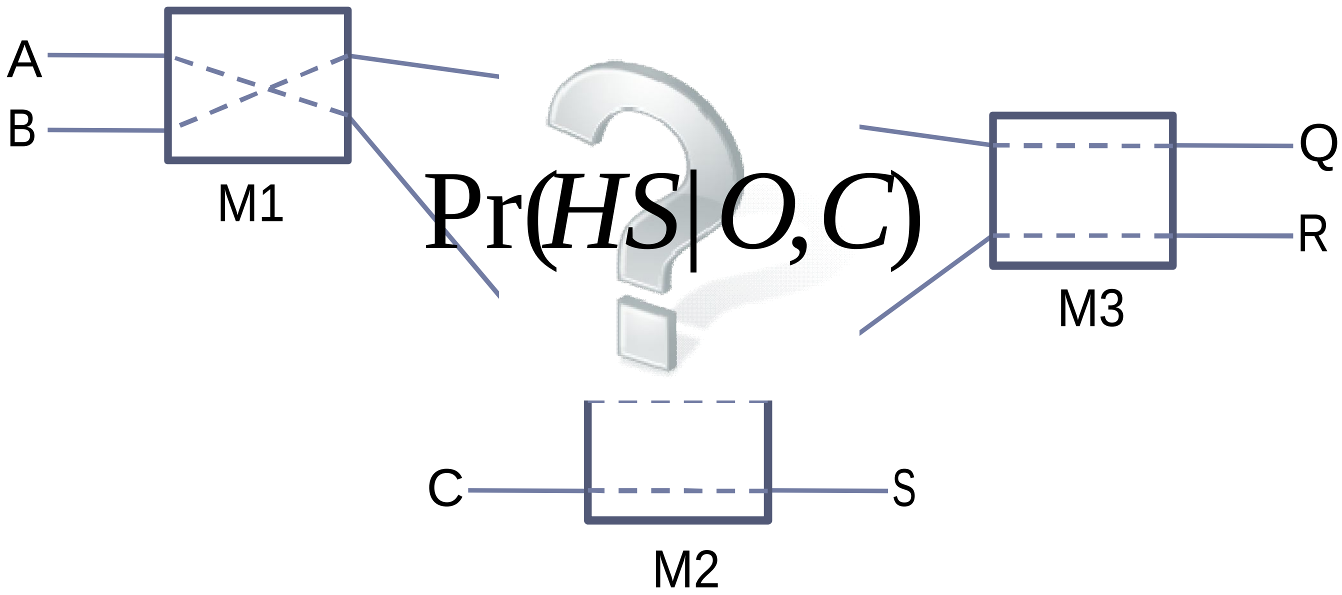
Non trivial given observation!!

"The real thing"



Redefining the traffic analysis problem

- Find *"hidden state"* of the mixes



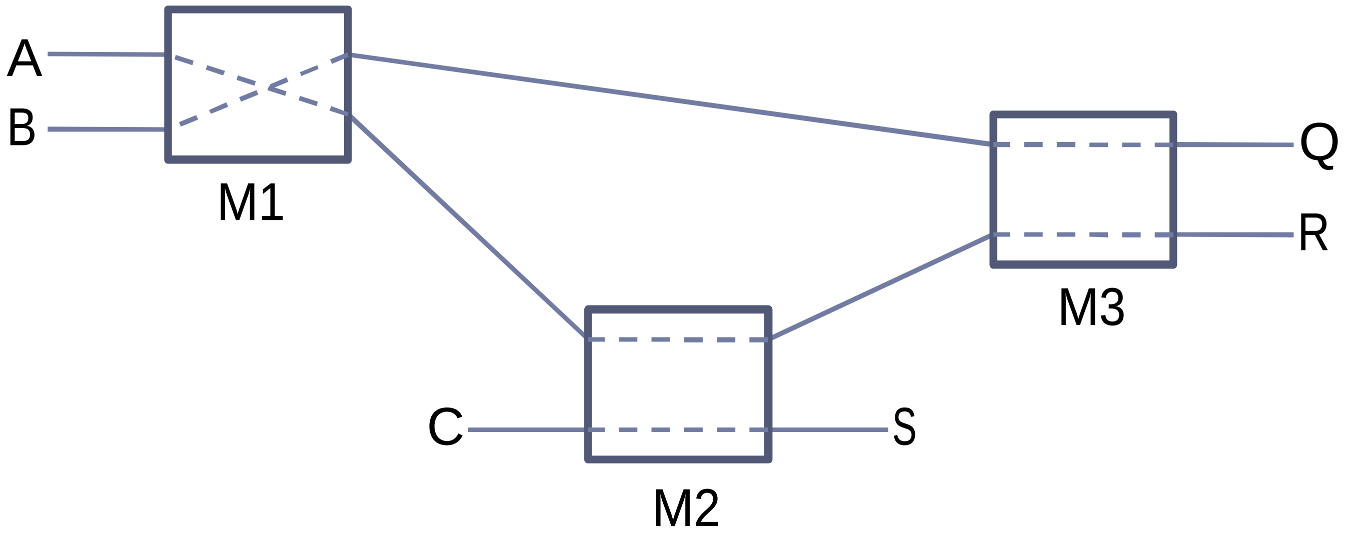
$$\Pr(HS \mid O, C) = \frac{\Pr(O \mid HS, C)\,\Pr(HS \mid C)}{\sum_{HS} \Pr(HS, O \mid C)} = \frac{\Pr(O \mid HS, C) \cdot K}{Z}$$

Prior information

Too large to enumerate!!

Redefining the traffic analysis problem

- *"Hidden State" + Observation = Paths*



$P_1 \rightarrow$ A — M1 — M2 — M3 — R

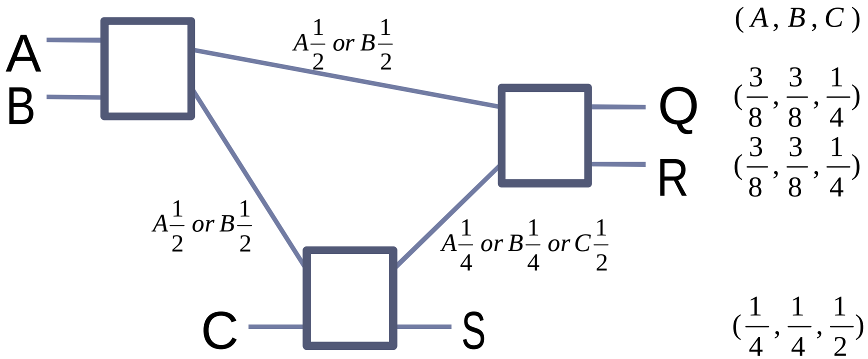
$P_2 \rightarrow$ B — M1 — M3 — Q

$P_3 \rightarrow$ C — M2 — S

$$\Pr(HS \mid O, C) = \frac{\Pr(O \mid HS, C) \cdot K}{Z} = \frac{\Pr(Paths \mid C)}{Z}$$

Sampling to estimate probabilities (I)

- Actually… we want marginal probabilities $\Pr(A \rightarrow Q \mid O, C)$



$$\Pr(A \rightarrow Q \mid O, C) = \frac{\sum_{HS} I_{A \rightarrow Q}(HS_j)}{j}$$

- But… we cannot obtain them directly

Sampling to estimate probabilities (II)

- If we obtain samples

$$HS_1, HS_2, HS_3, HS_4, \ldots, HS_N \sim \Pr(HS \mid O, C)$$

$$\begin{array}{lcccccc} & \downarrow & \downarrow & \downarrow & \downarrow & & \downarrow \\ (A \rightarrow Q)? & 0 & 1 & 0 & 1 & \ldots & 1 \end{array}$$

$$\Pr(A \rightarrow Q \mid O, C) \approx \frac{\sum_j I_{A \rightarrow Q}(HS_j)}{N}$$

- Markov Chain Monte Carlo Methods

$$\Pr(HS \mid O, C) = \frac{\Pr(Paths \mid C)}{Z}$$

How does *Pr(Paths|C)* look like?

Modelling mix networks

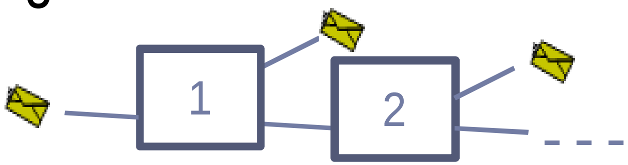
(The boring part of the talk)

Probabilistic model – Basic Constraints

- Users decide independently

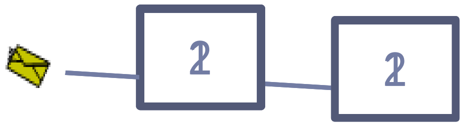
$$\Pr(Paths \mid C) = \prod_x \Pr(P_x \mid C)$$

- Length restrictions $\Pr(L = l \mid C)$ with any distribution



  - e.g. uniform (L_{min}, L_{max}) $\Pr(L = l \mid C) = \dfrac{1}{L_{\max} - L_{\min}}$

- Node choice restrictions

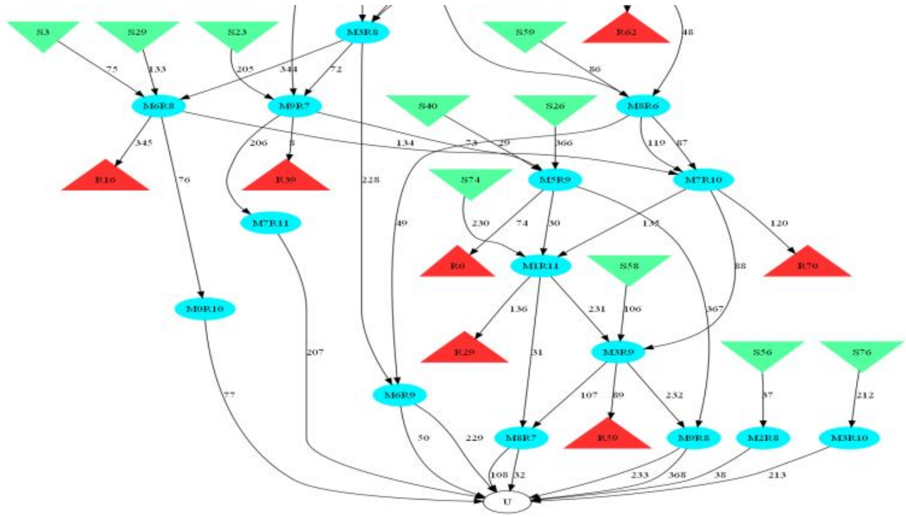


  - Choose l out of the N_{mix} node available $\Pr(M_x \mid L = l, C) = \dfrac{1}{P(N_{mix}, l)}$
  - Choose a set $I_{set}(M_x)$

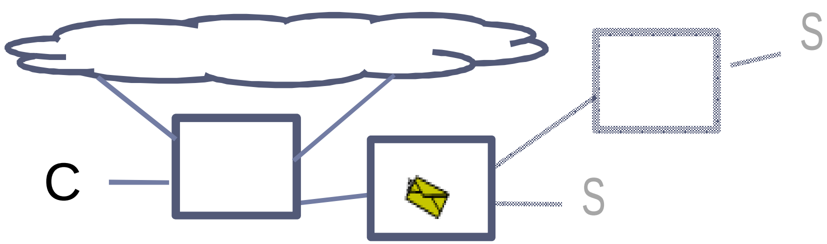
$$\Pr(P_x \mid C) = \Pr(L = l \mid C) \cdot \Pr(M_x \mid L = l, C) \cdot I_{set}(M_x)$$

Probabilistic model – Basic Constraints

- Unknown destinations



$$L_{\min} = 2 \quad L_{\max} = 3$$



$$\Pr(P_x \mid C) = \left[\sum_{l=L_{obs}}^{L_{\max}} \Pr(L = l \mid C) \cdot \Pr(M_x \mid L = l, C) \cdot I_{set}(M_x) \right]$$

Probabilistic model – More Constraints

- Bridging: users can only send through mixes they know

$$\Pr(P_x \mid C) = \Pr(L = l \mid C) \cdot \Pr(M_x \mid L = l, C) \cdot I_{set}(M_x) \cdot I_{bridging}(M_x)$$

- Non-compliant clients (with probability $p_{\overline{cp}}$)
  - Do not respect length restrictions $(L_{\min,\overline{cp}}, L_{\max,\overline{cp}})$
  - Choose l out of the N_{mix} node available, allow repetitions

$$\Pr(M_x \mid L = l, C, I_{\overline{cp}}(Path)) = \frac{1}{P_r(N_{mix}, l)}$$

$$\Pr(Paths \mid C) = \prod_x \Pr(P_x \mid C)$$

$$\Pr(Paths \mid C) = \left[\prod_{i \in P_{\overline{cp}}} p_{\overline{cp}} \Pr(P_i \mid C, I_{\overline{cp}}(P_i))\right] \cdot \left[\prod_{j \in P_{cp}} (1 - p_{\overline{cp}}) \Pr(P_j \mid C)\right]$$

Probabilistic model – More constraints

- Social network information
  - Assuming we know sending profiles $\Pr(\text{Sen}_x \rightarrow \text{Rec}_x)$

$$\Pr(P_x \mid C) = \Pr(L = l \mid C) \cdot \Pr(M_x \mid L = l, C) \cdot I_{set}(M_x) \cdot \Pr(\text{Sen}_x \rightarrow \text{Rec}_x)$$

- Other constraints
  - Unknown origin
  - Dummies
  - Other mixing strategies
  - ….

Markov Chain Monte Carlo methods and traffic analysis

(The exciting part of the talk)

Markov Chain Monte Carlo

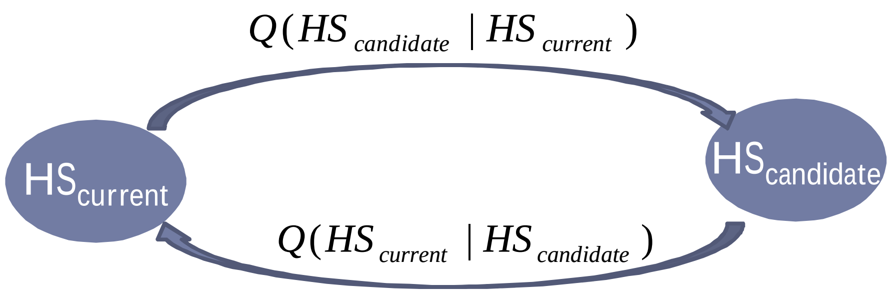
- Sample from a distribution difficult to sample from directly

$$\Pr(HS \mid O, C) = \frac{\Pr(O \mid HS, C) \cdot \Pr(HS \mid C)}{\sum_{HS} \Pr(HS, O \mid C)} = \frac{\Pr(O \mid HS, C) \cdot K}{Z} = \frac{\Pr(Paths \mid C)}{Z}$$

- 3 Key advantages:
  - Requires generative model (we know how to compute it!)
  - Good estimation of errors
    - Not false positives and negatives
  - Systematic

Metropolis Hastings Algorithm

- Constructs a Markov Chain with stationary distribution $\Pr(HS \mid O, C)$
  - Current state $\xrightarrow{Q}$ Candidate state



1. Compute $\alpha = \dfrac{\Pr(HS_{candidate})Q(HS_{candidate} \mid HS_{current})}{\Pr(HS_{current})Q(HS_{current} \mid HS_{candidate})}$
2. If $\alpha \geq 1$

   $HS_{current} = HS_{candidate}$ **Go!**

   else $u \sim U(0,1)$

   if $u \leq \alpha$

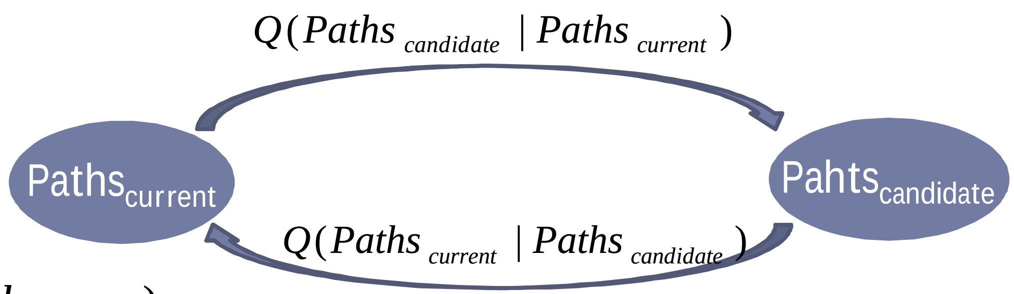
   $HS_{current} = HS_{candidate}$ **Go with probability α**

   else

   $HS_{current} = HS_{current}$

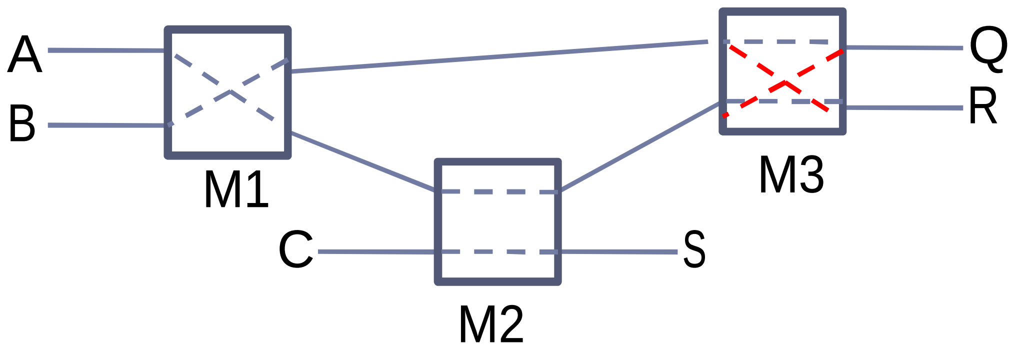
Our sampler: Q transition

$$\Pr(HS \mid O, C) = \frac{\Pr(Paths \mid C)}{Z}$$



$$\alpha = \frac{\Pr(Paths_{candidate})Q(Paths_{candidate} \mid Paths_{current})}{\Pr(Paths_{current})Q(Paths_{current} \mid Paths_{candidate})}$$

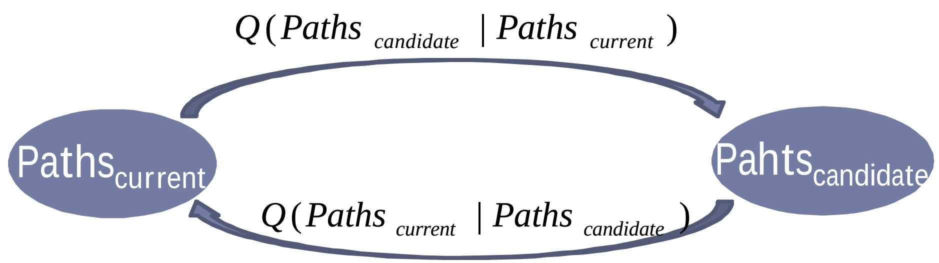
- Transition Q: swap operation



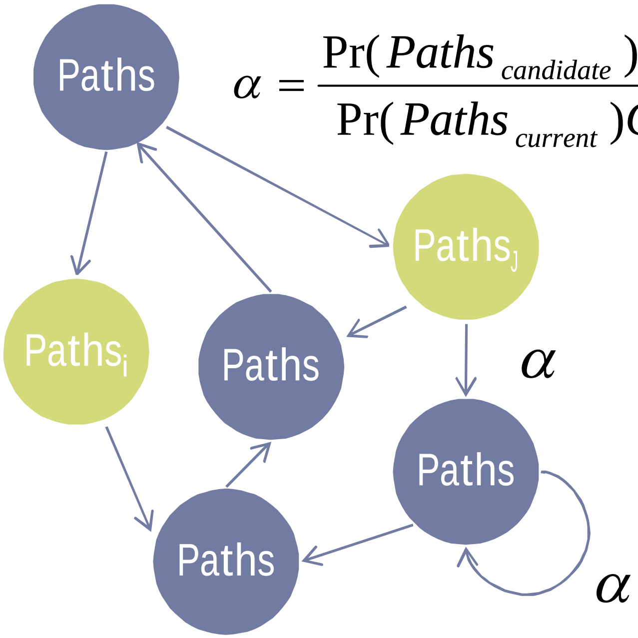
- More complicated transitions for non-compliant clients

Iterations

$$\Pr(HS \mid O, C) = \frac{\Pr(Paths \mid C)}{Z}$$



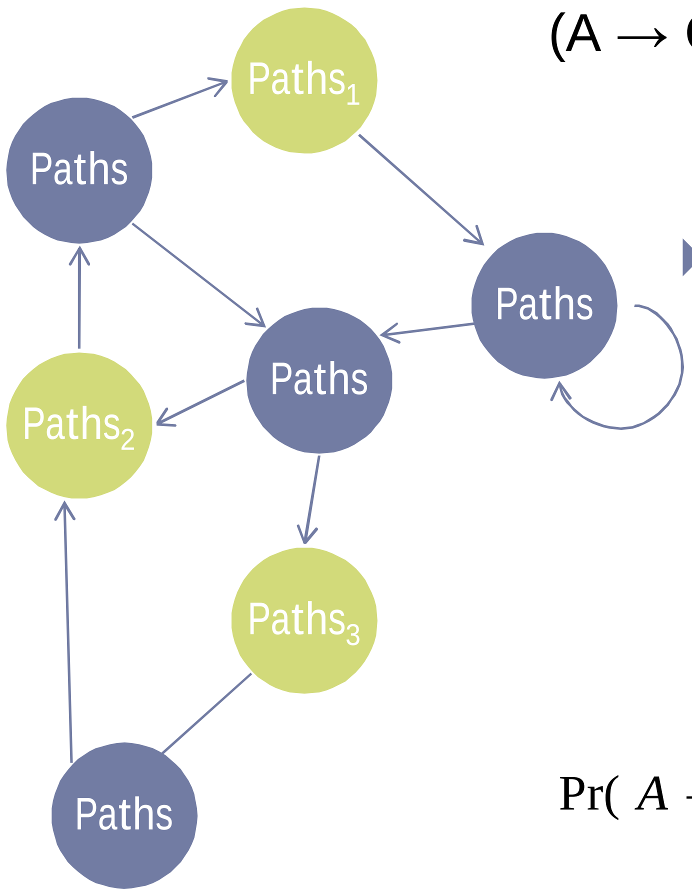
$$\alpha = \frac{\Pr(Paths_{candidate})Q(Paths_{candidate} \mid Paths_{current})}{\Pr(Paths_{current})Q(Paths_{current} \mid Paths_{candidate})}$$



- Consecutive samples dependant
- Sufficiently separated

Sampling error estimation

$$\begin{array}{lcccc} & P_1 & P_2 & P_3 & \dots \\ & \downarrow & \downarrow & \downarrow & \\ (A \to Q)? & 1 & 0 & 1 & \dots \end{array} \longrightarrow \Pr(A \to Q) = \frac{\sum_i I_{A\to Q}(Paths_i)}{j}$$



- Error estimation
  - Bernouilli distribution

$$\Pr[\, I_{A\to Q}(P_1), I_{A\to Q}(P_2), I_{A\to Q}(P_3), \dots \mid \Pr(A \to Q)]$$

$$\Pr[\Pr(A \to Q) \mid I_{A\to Q}(P_1), I_{A\to Q}(P_2), I_{A\to Q}(P_3), \dots]$$

  - Prior Beta(1,1) ~ uniform

$$\Pr(A \to Q) \sim Beta\left(\sum_{Paths} I_{A\to Q}(P_i) + 1, \sum_{Paths} \bar{I}_{A\to Q}(P_i) + 1\right)$$

Evaluation and results

(It actually works!)

Evaluation

Events should happen with the predicted probability

1. Create an instance of a network
2. Run the sampler and obtain $P_1, P_2, \ldots$
3. Choose a target sender and a receiver
4. Predict probability

$$\Pr(\mathrm{Sen} \to \mathrm{Rec}) \approx \frac{\sum_j I_{\mathrm{Sen} \to \mathrm{Rec}}(Paths_j)}{N}$$

5. Check if actually Sen chose Rec as receiver $I_{\mathrm{Sen} \to \mathrm{Rec}}(network)$
6. Choose new network and go to 2

Example

- Studying events with $\Pr(\text{Sen} \rightarrow \text{Rec}) = 0.4$
  - Network 1

$$I_{A\rightarrow B}(P_1) = 0;\ I_{A\rightarrow B}(P_2) = 1;\ I_{A\rightarrow B}(P_3) = 0;\ I_{A\rightarrow B}(P_4) = 0;\ I_{A\rightarrow B}(P_5) = 1$$

$$\Pr(A \rightarrow B) \approx \frac{\sum_j I_{A\rightarrow B}(P_j)}{5} = 0.4; \quad I_{A\rightarrow B}(Network_1) = 0$$

  - Network 2 $\Pr(X \rightarrow Y) = 0.4; \quad I_{X\rightarrow Y}(Network_2) = 0$
  - Network 3 $\Pr(X \rightarrow Y) = 0.4; \quad I_{X\rightarrow Y}(Network_3) = 1$
  - Network 4 $\Pr(X \rightarrow Y) = 0.4; \quad I_{X\rightarrow Y}(Network_4) = 1$
  - Network 5 $\Pr(X \rightarrow Y) = 0.4; \quad I_{X\rightarrow Y}(Network_5) = 0$

$$\Pr_{sampled}(X \rightarrow Y) = 0.4$$

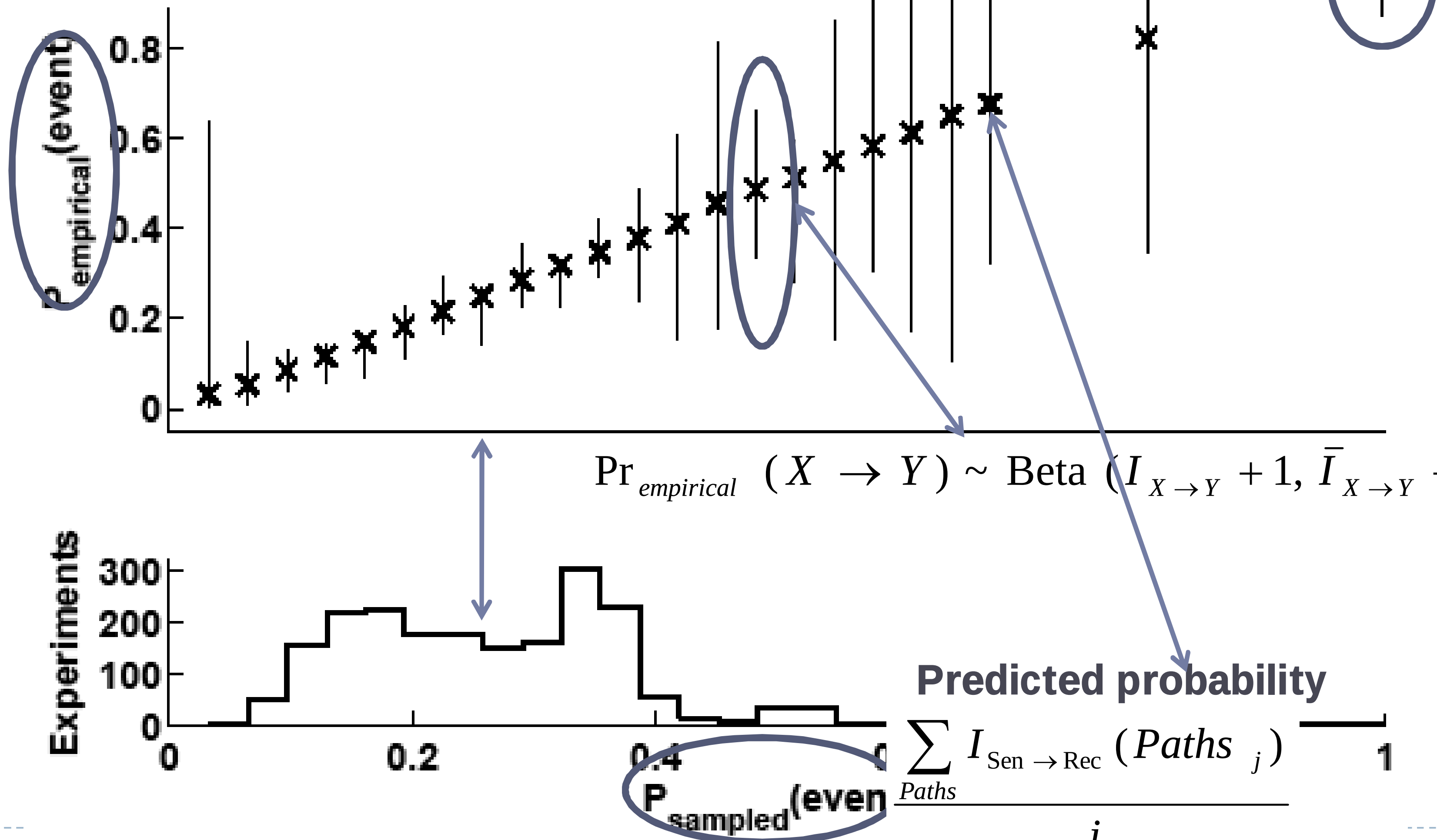
$$\Pr_{empirical}(X \rightarrow Y) \sim \text{Beta}(2+1, 3+1)$$

X% Confidence intervals

Results – compliant clients – 50 messages

Empirical probability

$$E(I_{\text{Sen}\rightarrow\text{Rec}}(network))$$

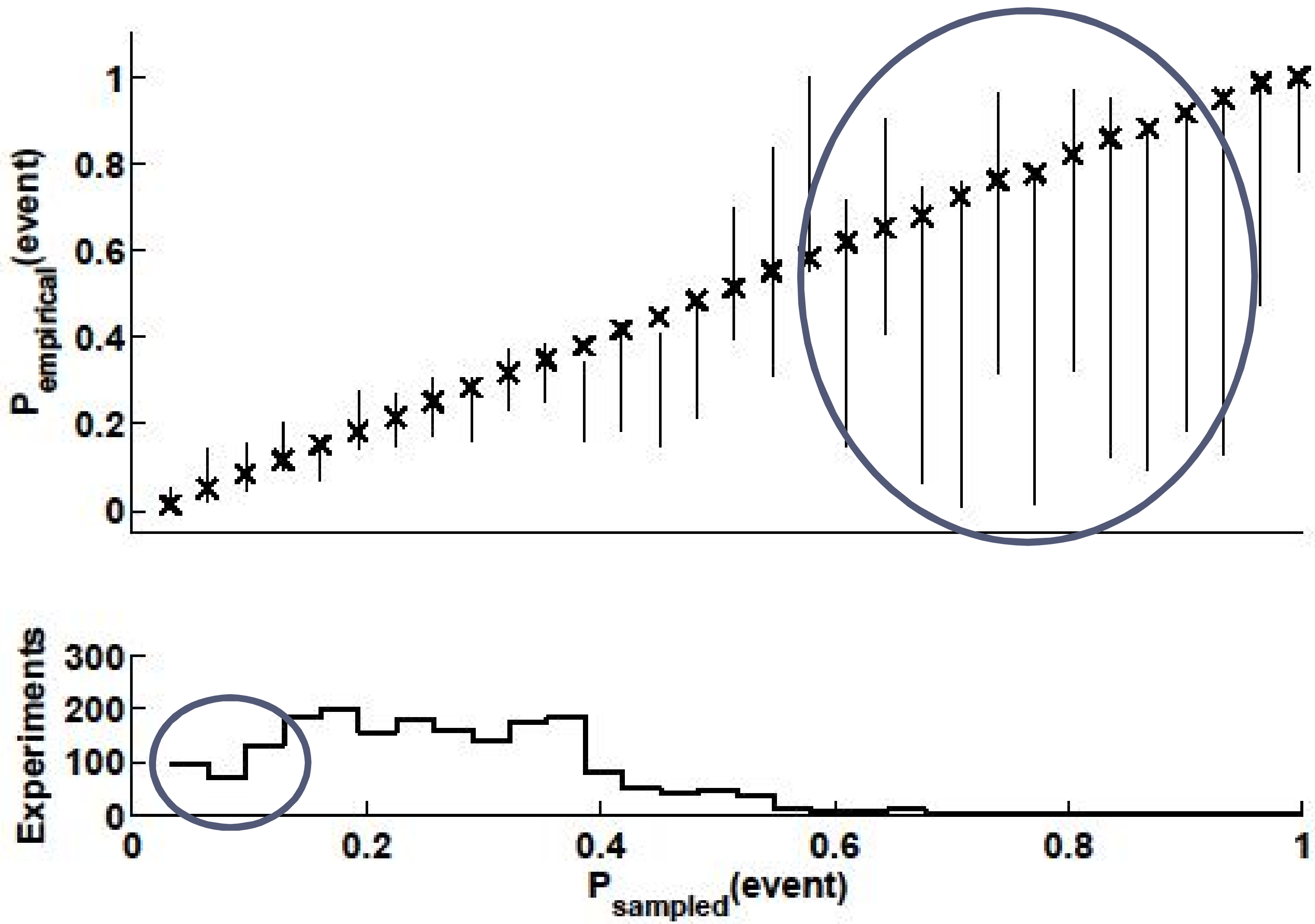


$$\Pr_{empirical}(X \rightarrow Y) \sim \text{Beta}(I_{X\rightarrow Y}+1, \bar{I}_{X\rightarrow Y}+1)$$

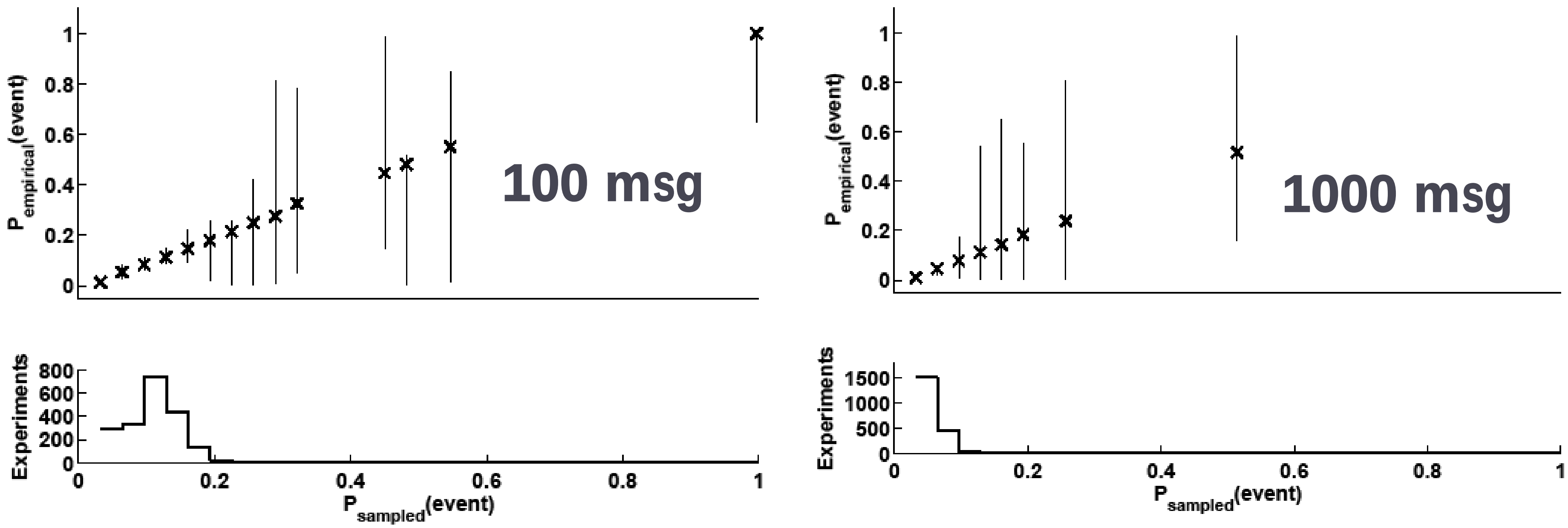
Predicted probability

$$\frac{\sum_{Paths} I_{\text{Sen}\rightarrow\text{Rec}}(Paths_j)}{j}$$

Results non compliant – 50 messages



Results – big networks



- It scales well as networks get larger
- As expected mix networks offer good protection

Performance – RAM usage

Nmix	t	Nmsg	Samples	RAM(Mb)
3	3	10	500	16
3	3	50	500	18
10	20	50	500	18
10	20	1 000	500	24
10	20	10 000	500	125

- Size of network and population
- Results are kept in memory during simulation

Performance – Running time

Nmix	t	Nmsg	iter	Full analysis (min)	One sample(ms)
3	3	10	6011	4.24	509.12
3	3	50	6011	4.80	576.42
5	10	100	7011	5.34	641.28
10	20	1 000	7011	5.97	706.12

- Operations should be O(1)
  - Writing of the results on a file
  - Different number of iterations

Applications

- Evaluation information theoretic metrics for anonymity

$$H = -\sum_{R_i} P(A \to R_i \mid O,C) \cdot \log P(A \to R_i \mid O,C)$$

- Estimating probability of arbitrary events
  - Input message to output message?
  - Alice speaking to Bob ever?
  - Two messages having the same sender?

- Accommodate new constraints
  - Key to evaluate new mix network proposals

Conclusions

- Traffic analysis is non trivial when there are constraints
- Probabilistic model: incorporates most attacks
  - Non-compliant clients
- Monte Carlo Markov Chain methods to extract marginal probabilities
  - Systematic
  - Only generative model needed
- Future work:
  - Model more constraints
  - Added value?

Time for questions

- More info
  - **Vida: How to use Bayesian inference to de-anonymize persistent communications.** George Danezis and Carmela Troncoso. Privacy Enhancing Technologies Symposium 2009
  - **The Bayesian analysis of mix networks.** Carmela Troncoso and George Danezis. 16th ACM Conference on Computer and Communications Security 2009
  - **The Application of Bayesian Inference to Traffic analysis.** Carmela Troncoso and George Danezis Microsoft Technical Report

Carmela.Troncoso@esat.kuleuven.be

http://homes.esat.kuleuven.be/~ctroncos/

Bayes theorem

$$\Pr(O, HS \mid C) = \Pr(HS \mid O, C) \cdot \Pr(O \mid C)$$

$$\Pr(O, HS \mid C) = \Pr(O \mid HS, C) \cdot \Pr(HS \mid C)$$

$$\Pr(HS \mid O, C) = \frac{\Pr(O \mid HS, C) \cdot \Pr(HS \mid C)}{\Pr(O \mid C)} = \frac{\Pr(O \mid HS, C) \cdot \Pr(HS \mid C)}{\sum_{HS} \Pr(HS, O \mid C)}$$

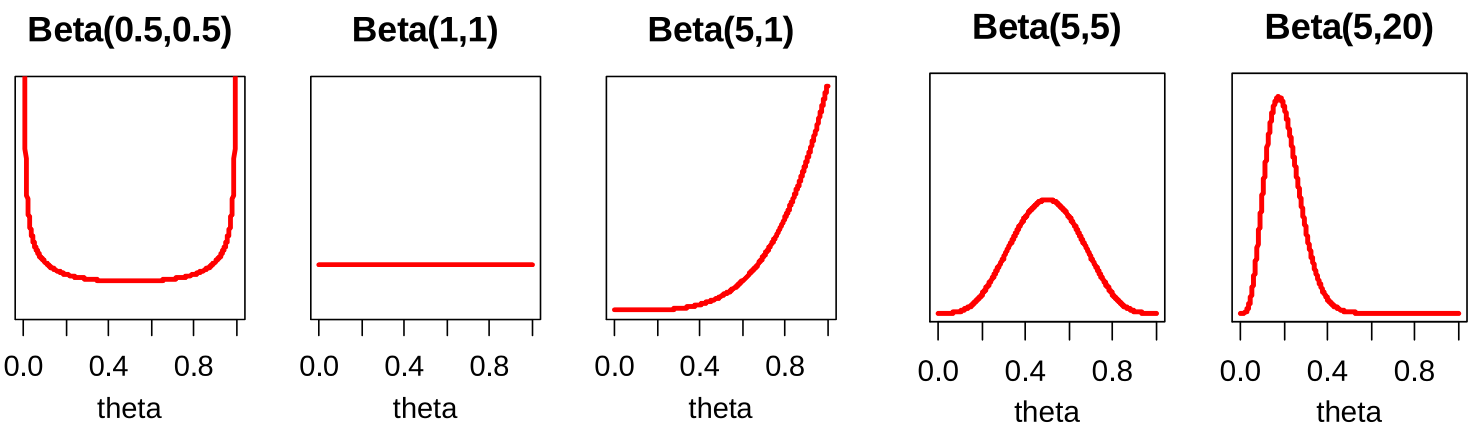
Joint probability:

$$\Pr(X, Y) = \Pr(X \mid Y) \cdot \Pr(Y) = \Pr(Y \mid X) \cdot \Pr(X)$$

Error estimation: the Beta function

- We need to specify the prior knowledge $\Pr(\theta)$ ($\Pr(HS \mid C)$)
  - expresses our uncertainty
  - conforms to the nature of the parameter, i.e. is continuous but bounded between 0 and 1
- A convenient choice is the Beta distribution

$$P(\theta) = Beta(a,b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}\theta^{a-1}(1-\theta)^{b-1}$$



Error estimation: the Beta function

- Combining a beta prior with the binomial likelihood gives a posterior distribution

$$p(\theta \mid total, successes) = p(successes \mid \theta, total) \cdot p(\theta)$$

$$\propto \theta^{successes+a-1}(1-\theta)^{total-successes+b-1}$$

$$\propto Beta(successes+a, total-successes+b)$$

Prior knowledge