# On the Effectiveness of Secret Key Extraction from Wireless Signal Strength in Real Environments

Suman Jana, Sriram Nandha Premnath
Mike Clark, Sneha K. Kasera, Neal Patwari
University of Utah

Srikanth V. Krishnamurthy
University of California, Riverside

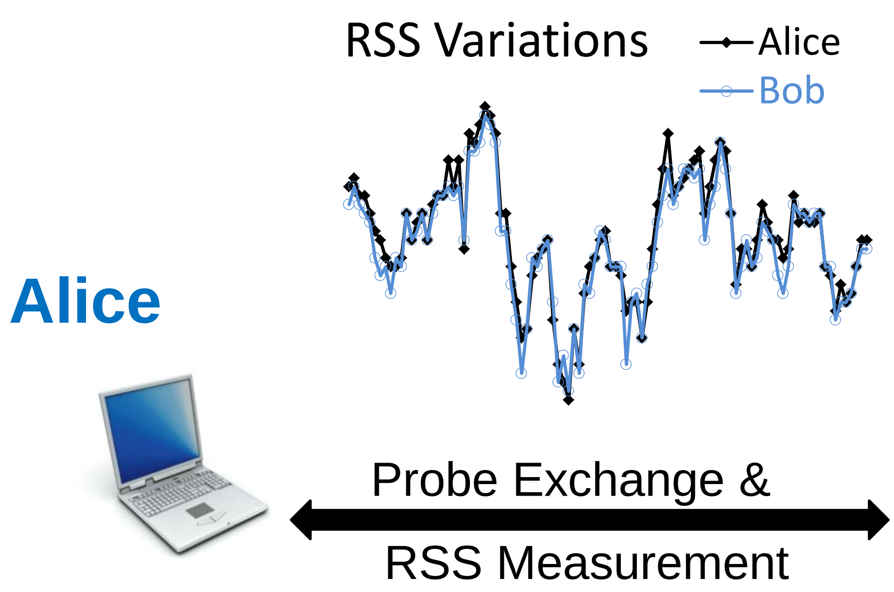
# Problem Definition

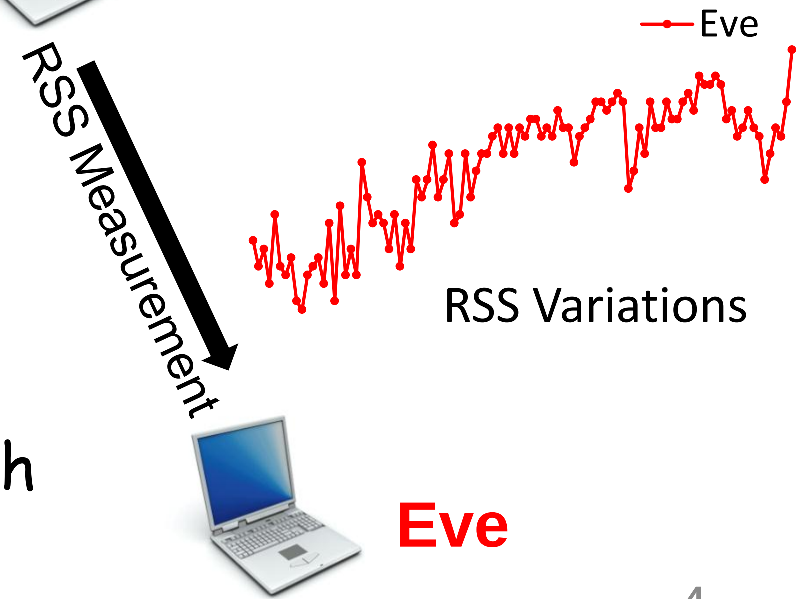
- wireless nodes, Alice & Bob, need to share secret key
- concerns with public key cryptography
- quantum cryptography – too expensive
- less expensive solution - use inherent randomness in wireless channel to extract secret key bits

# Wireless Channel Characteristics

- measured reciprocally at Alice, Bob
- when away by more than few multiples of wavelength, Eve cannot measure same channel
- channel varies with time

# Use of Received Signal Strength (RSS)




- Alice, Bob expected to see "*identical*" RSS variations [Stutzman '82]
- realistically, they must deal with lack of perfect reciprocity

# Related Work

Mathur '08, Li '06, Aono '05

- extract single bit per measurement
- experimental results from limited indoor settings
- Alice, Bob do not communicate to handle mismatches

  **will result in key disagreement in large number of cases**

Azimi-Sadjadi '07

- suggested using 2 stages from quantum cryptography - information reconciliation, privacy amplification
- did not implement!

# Our Contributions

- adaptive key extraction
  **increases secret bit rate 4-fold**
- implement information reconciliation to handle bit mismatch
- implement privacy amplification to reduce correlation between successive bits
- through extensive real world measurements, identify settings (un)suitable for key extraction
- expose new predictable channel attack in static settings

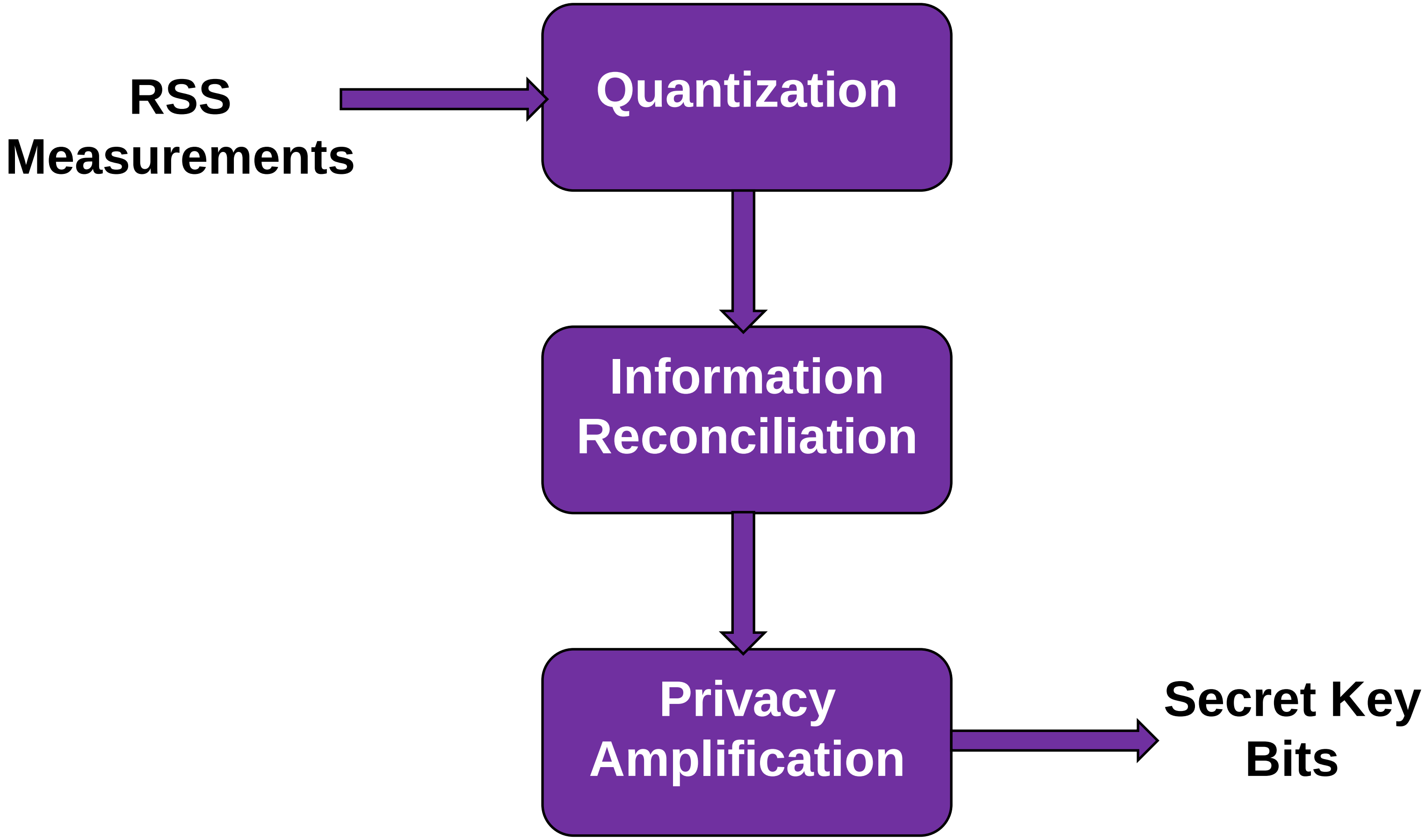
# Overview

- adversary model
- secret key extraction
- real world measurements, results
- summary

# Adversary Model

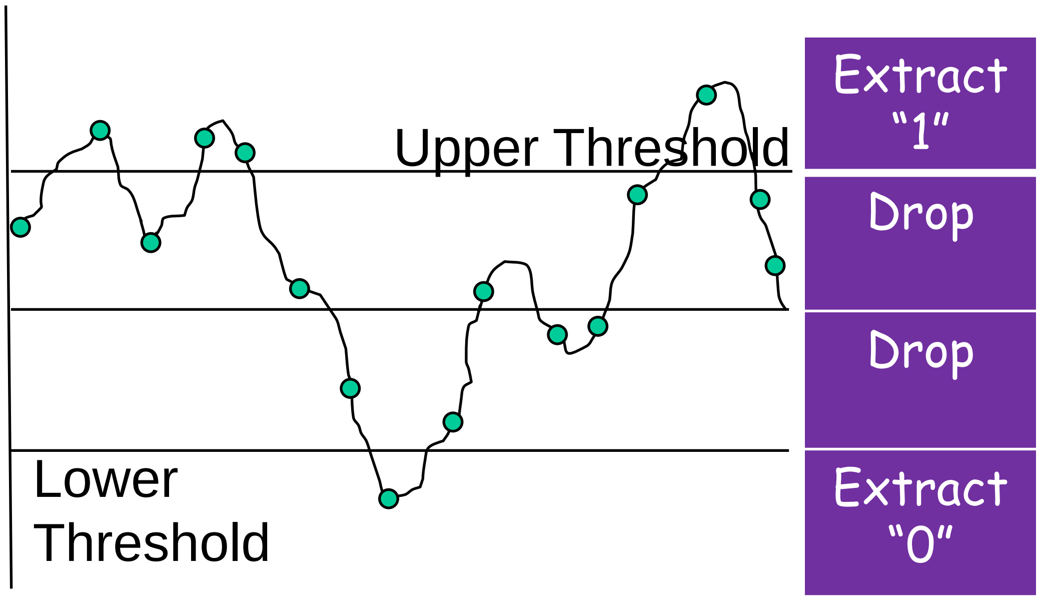
- adversary Eve
  - listens to all communication between Alice, Bob
  - can measure channel between herself and Alice, Bob
  - separated from both parties by distance >> wavelength
- Eve not interested in disrupting communication between Alice, Bob
- Alice, Bob are not authenticated

# Secret Bit Extraction



# Adaptive Quantization

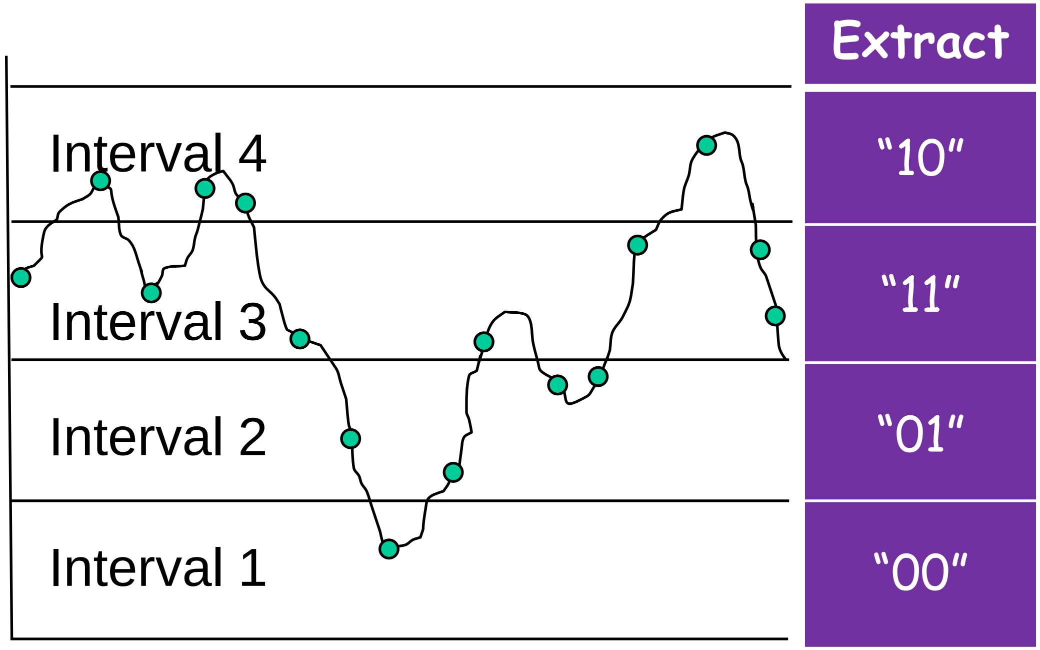
how to generate bits from RSS measurements?




adapt threshold for small blocks of measurements

Extracted Bits – 1 1 1 0 1 …

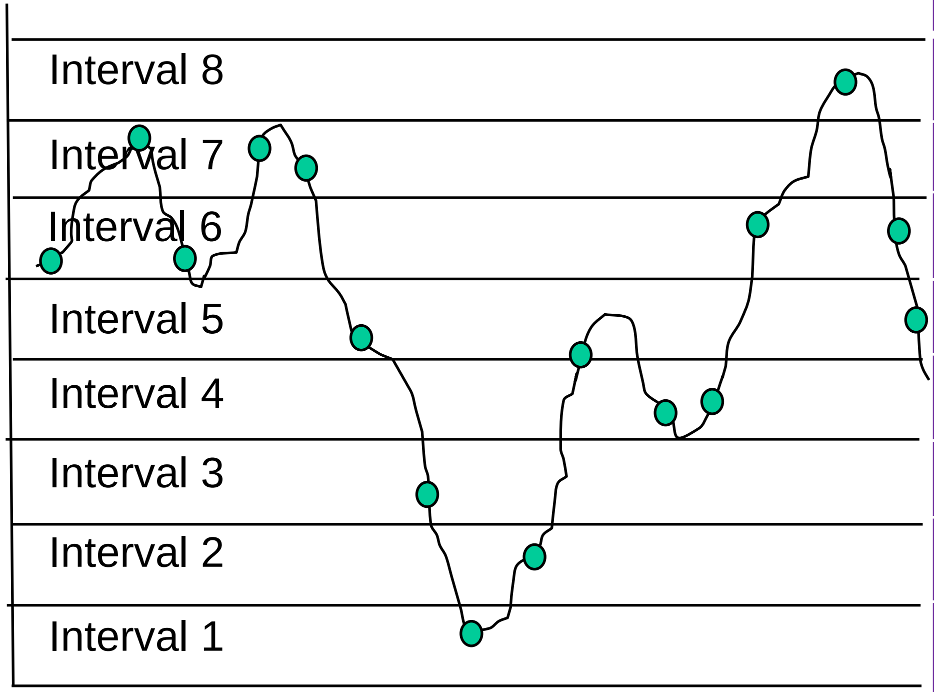
# Adaptive Quantization




adapt # intervals depending on range

Extracted Bits - 11 10 11 10 10 11 01 00 01 …

# Adaptive Quantization




| Extract |
|---|
| "111" |
| "110" |
| "101" |
| "100" |
| "011" |
| "010" |
| "001" |
| "000" |

adapt #intervals depending on range

limit: N ≤ log [Range]

Extracted Bits - 101 110 101 110 110 100 010 …

Adaptive Secret Bit Generation (ASBG)

# Information Reconciliation [Brassard '06]

differences in between bit streams of Alice, Bob arise due to

- noise/interference, wireless hardware limitations
- half-duplex nature of channel

solution:

- exchange parity information of small blocks of bits
- locate, correct mismatches using binary search
- permute, iterate until probability [success] > threshold

# Privacy Amplification [Impagliazzo '89]

- short-term correlation between subsequent bits when probing rate > $(\text{coherence time})^{-1}$
- need to remove bits leaked during information reconciliation
- solution:
  - apply 2-universal hash function h: {1…M} → {1…m}
  - for inputs x, y, probability [h(x) = h(y)] upper bounded by 1/m
  - decreases output length, but increases entropy

# Implementation

laptops - Alice, Bob

- equipped with Intel PRO/Wireless 3945 ABG cards
- monitor mode for collecting RSS measurements
- use *ipwraw* driver for raw packet injection

probes – IEEE 802.11g beacon frames

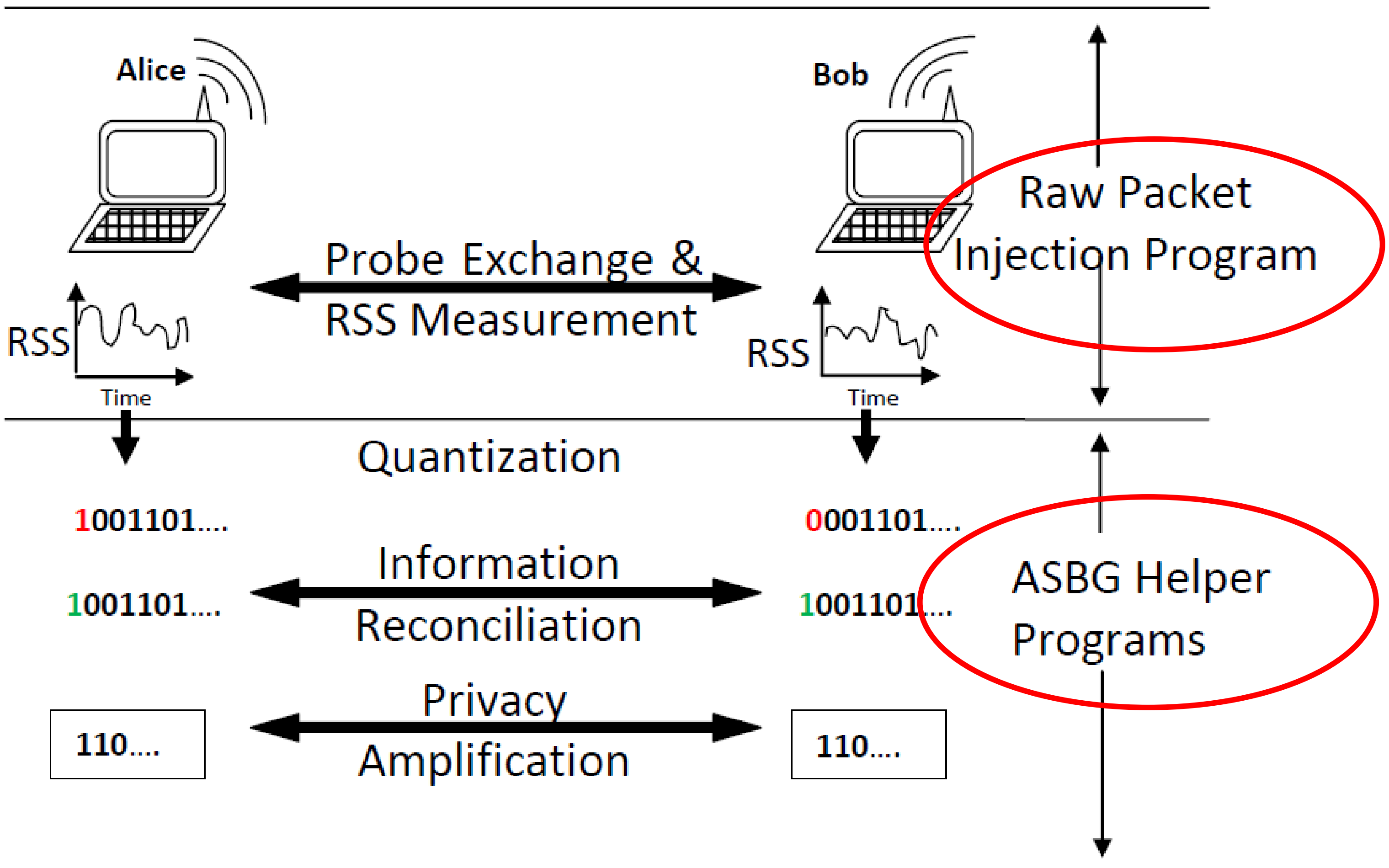
- management frames prioritized over data frames
- allows better control over probing rate
- probing rate ~20 packets per second

# Implementation

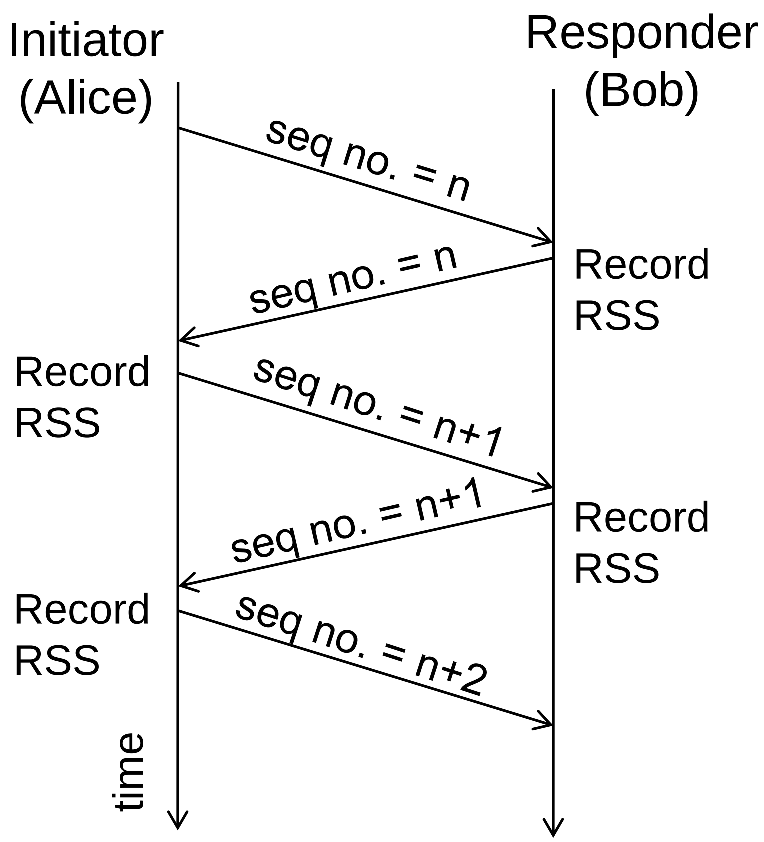
privacy amplification

- 2-universal hash functions
- use *BigNumber OpenSSL* routines

# Implementation



# RSS Measurement Protocol




- packet losses handled by initiator
- 20 ms timeout for detecting packet loss
- responder discards last RSS if duplicate beacon sequence #

# Measurement Goals

- in what kind of settings, key extraction *"works"*?
- how does device heterogeneity affect key extraction?

# Experiments

1. Stationary Endpoints, Intermediate objects

A. Underground concrete tunnel
B. Ed Catmull Gallery
C. Lawn

---

2. Mobile Endpoints

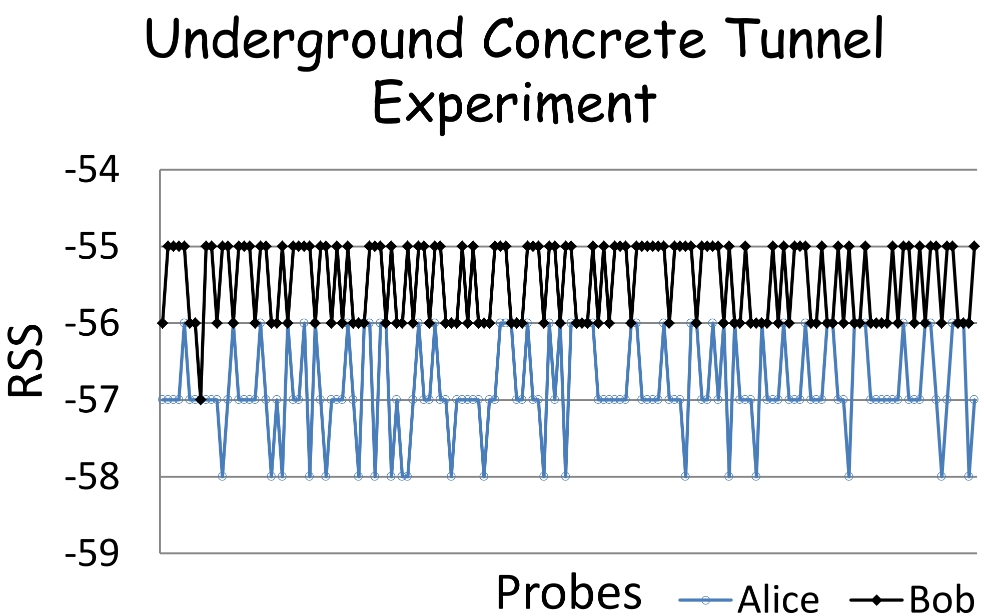
D. Walk Indoors
E. Walk Outdoors
F. Bike Ride

---

3. Stationary Endpoints, Mobile Intermediate objects

G. Crowded Cafeteria
H. Across busy road

# Stationary Endpoints & Intermediate Objects

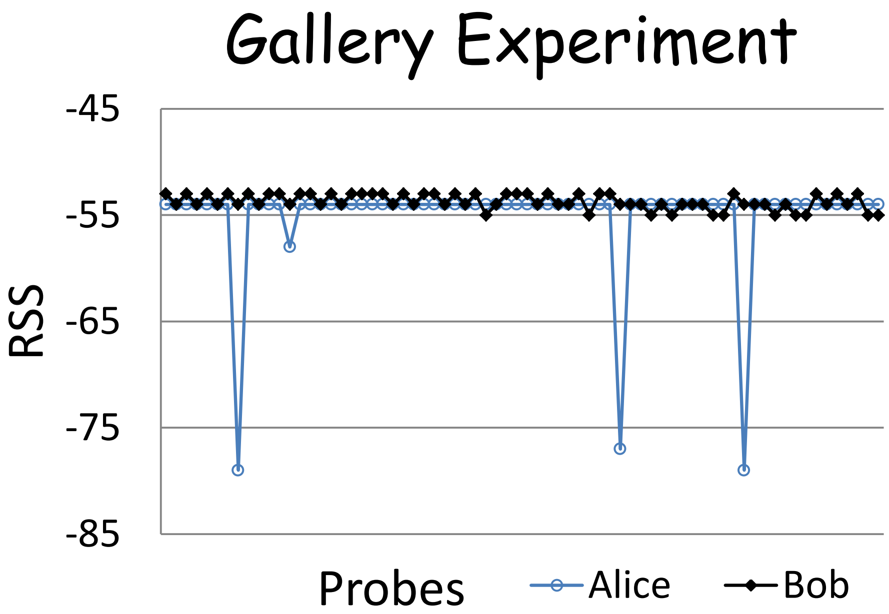



snapshot of data collected for few seconds

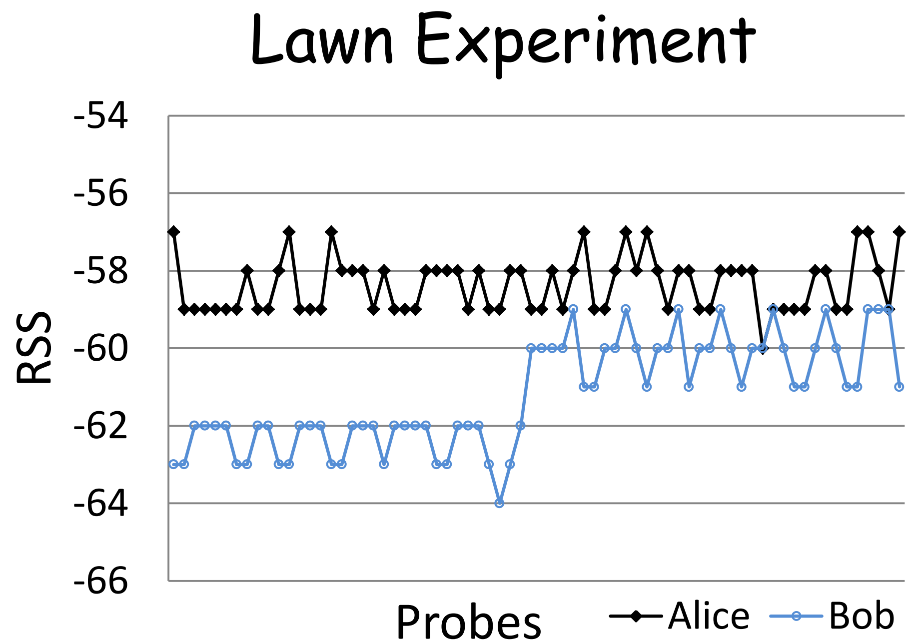
distance between Alice, Bob = 10 feet

- variations very small (range: ~2 dB), exhibit poor reciprocity
- expect Alice's & Bob's bit streams to have very high mismatch
- small scale variations represent noise

# Stationary Endpoints & Intermediate Objects



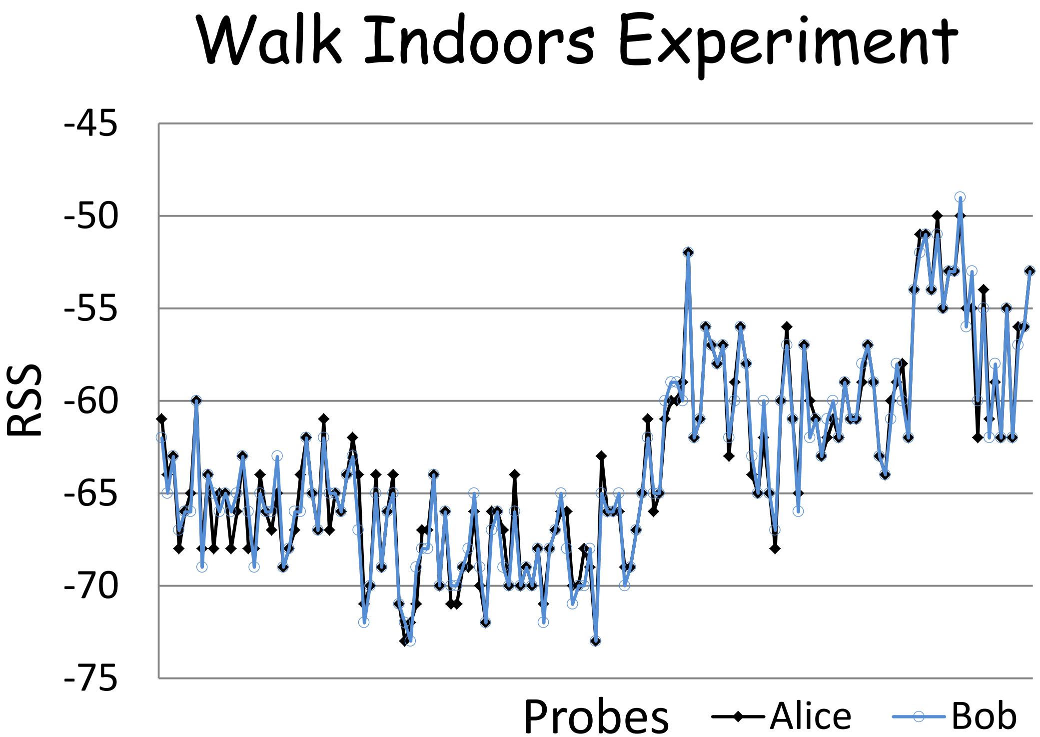

distance = 30 ft




distance = 10 ft

- even typical stationary settings are no different from underground concrete tunnel!

# Mobile Endpoints



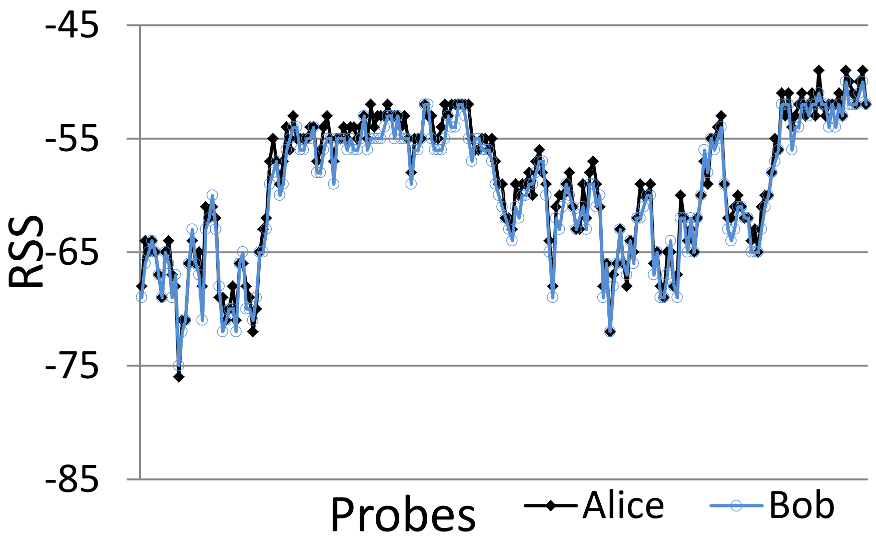

normal walk speed

distance = 10-15 ft

- large variations
  - range ~25 dB
  - highly reciprocal
- hints that Alice's & Bob's bit streams will have very low mismatch

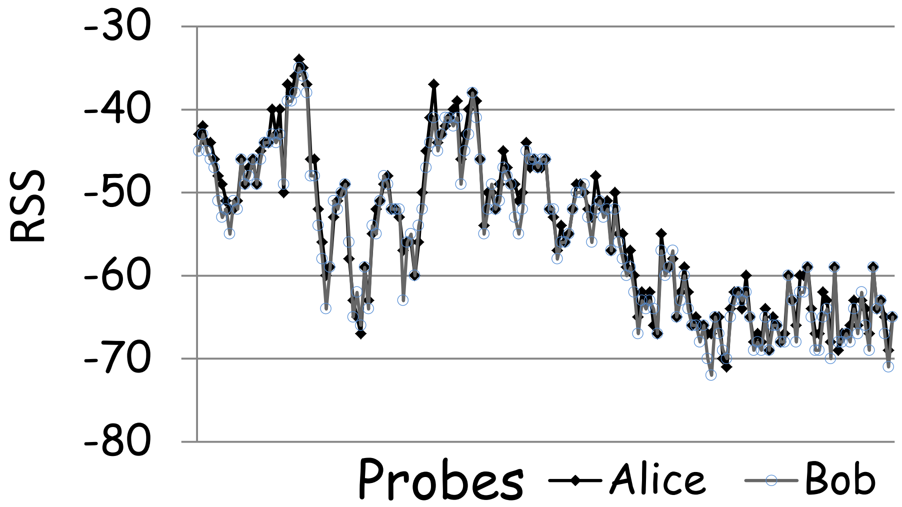
# Mobile Endpoints

Walk Outdoors Experiment




normal walk speed
20-25 ft distance

Bike Ride Experiment

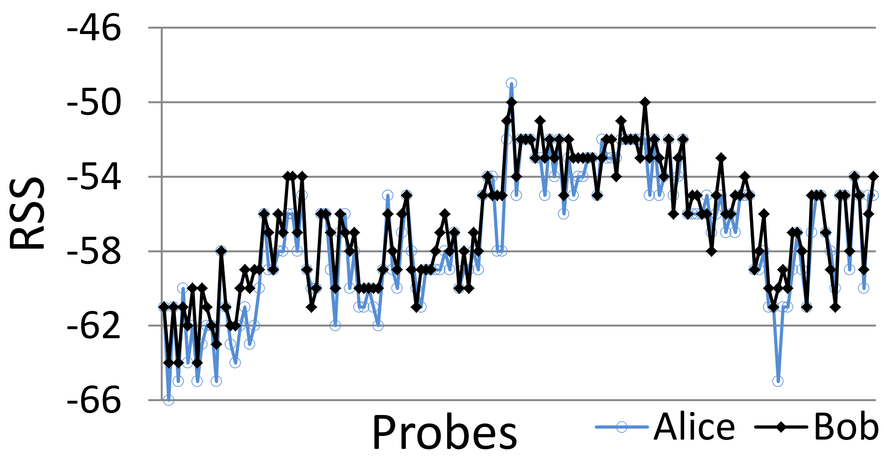



slow bike ride
10 ft or more distance

- more evidence - mobile devices likely to have very low mismatch
- effects of noise diminished by large scale variations

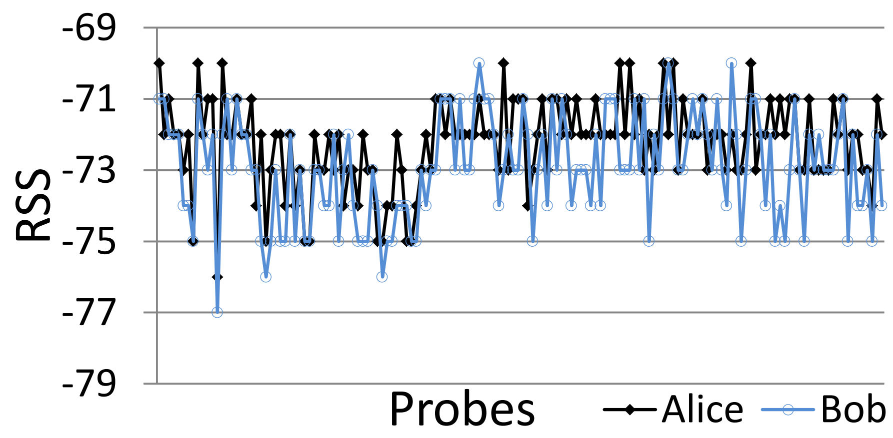
# Mobile Intermediate Objects & Stationary Endpoints

Crowded Cafeteria Experiment




low speed mobility; distance = 10 ft

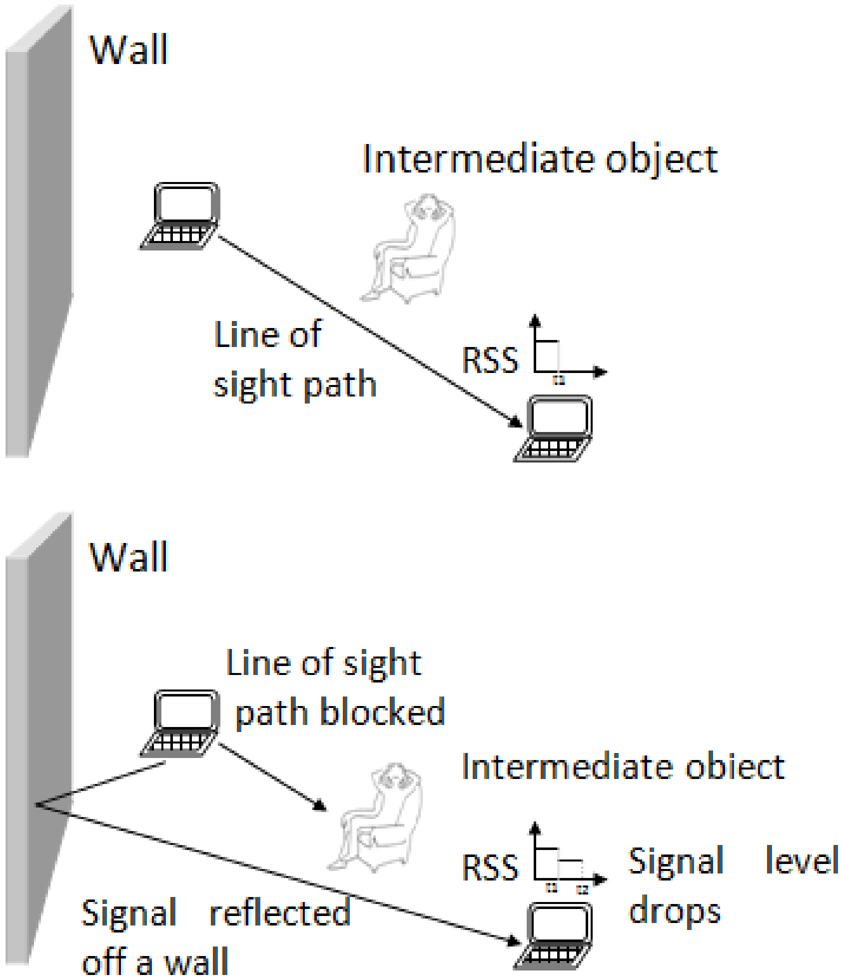
Experiment Across Busy Road




high speed mobility; distance = 25 ft

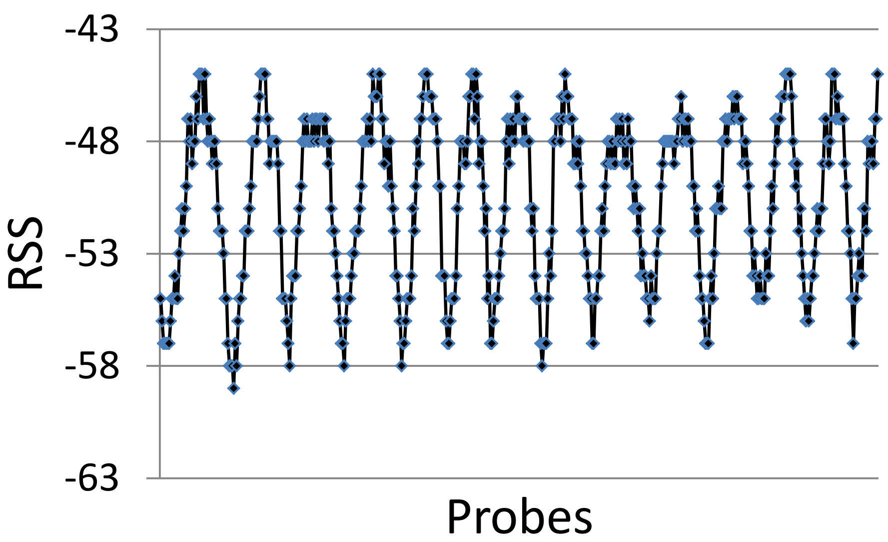
- intermediate variation range (~8-16 dB), reciprocity
- hints - Alice's, Bob's bit streams will have moderate mismatch

# Predictable Channel Attack




- novel attack
- in 'all stationary' settings Eve can cause **predictable channel variations**
  - by controlling movements of intermediate objects
- break key extraction schemes without spending compute power

# Predictable Channel Attack

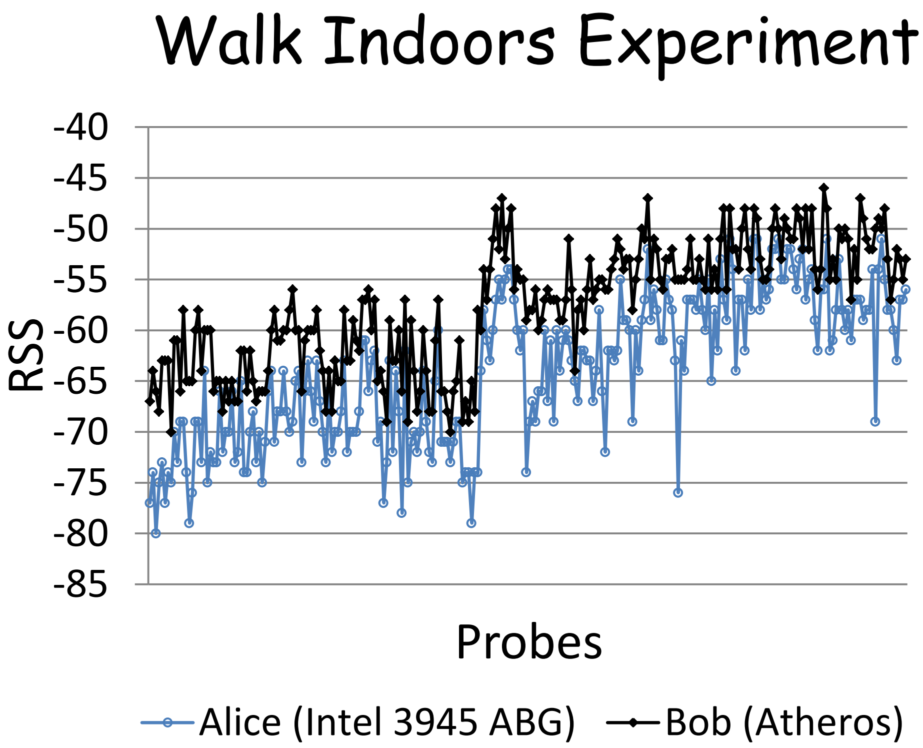



bits extracted:

**0000 1111 0000 1111 …**

- no precision machinery required
- Eve can produce zig-zag patterns, or any other pattern by controlling movements
- no post processing will ensure security of extracted key!

# Effect of Device Heterogeneity



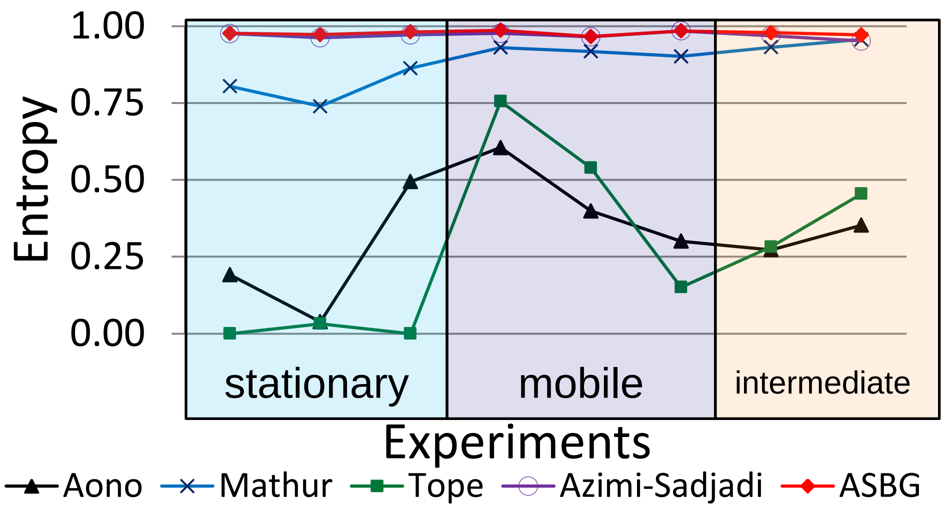

- greater mismatch *than* with homogeneous devices
- mismatch low enough to help establish secret key

# Comparison of Key Extraction Approaches in Various Settings

- performance metrics
  - entropy rate
  - mismatch rate
  - secret bit rate

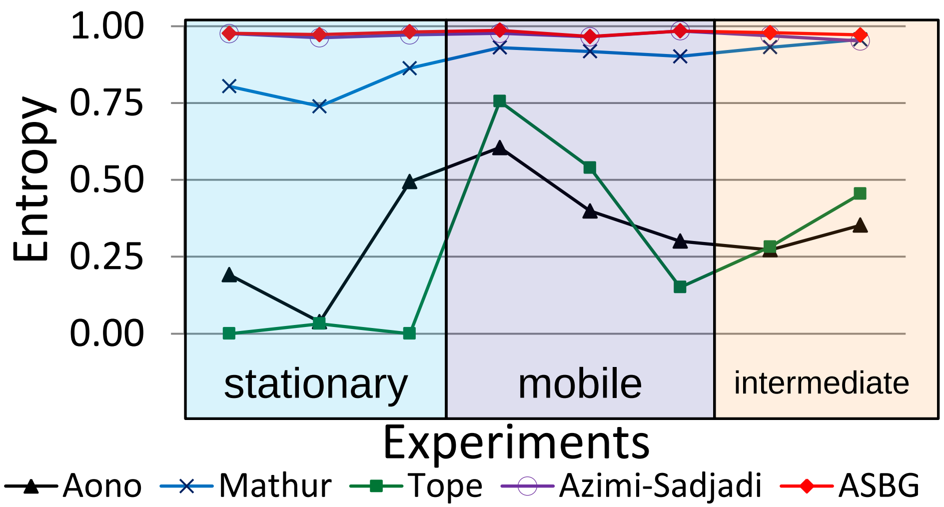
- single bit, multiple bit extraction

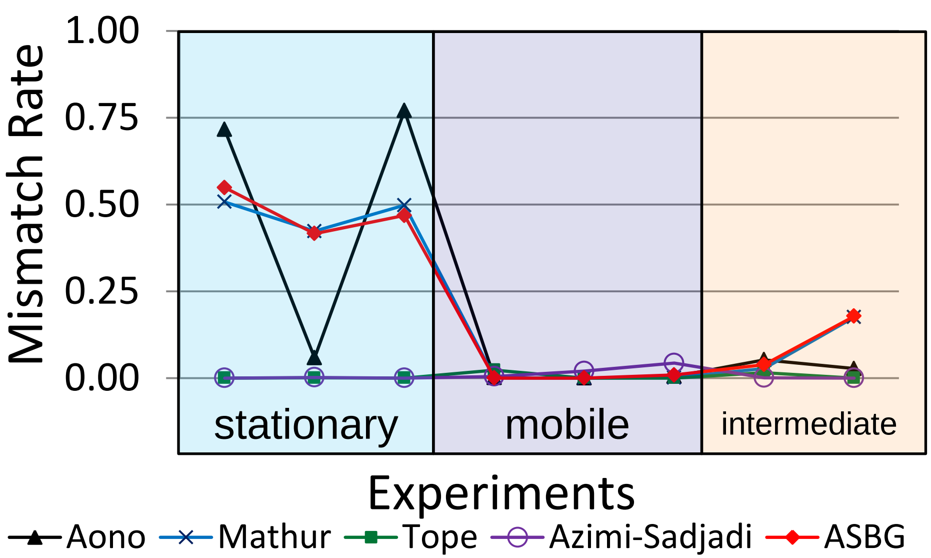
# Comparison of Key Extraction Approaches in various Settings




- secret bit stream from ASBG
  - entropy close to 1
  - passes randomness tests of NIST test suite we conduct

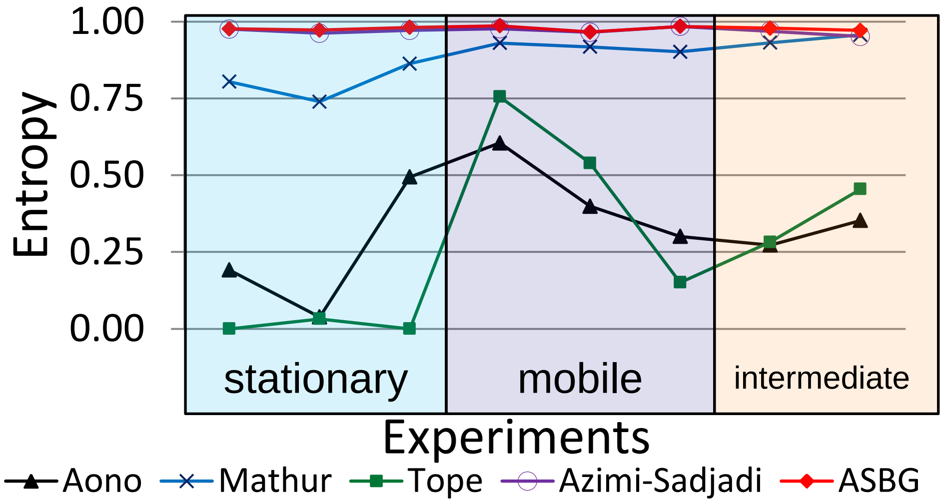
# Comparison of Key Extraction Approaches in various Settings

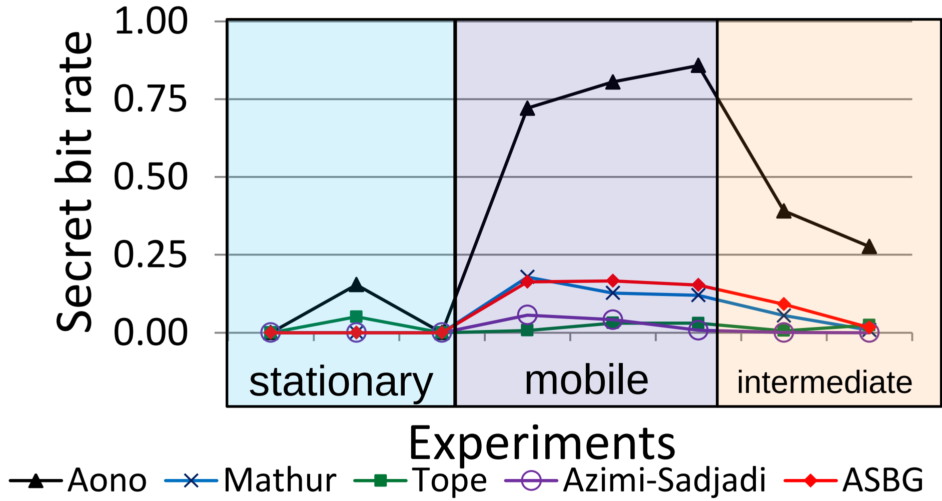


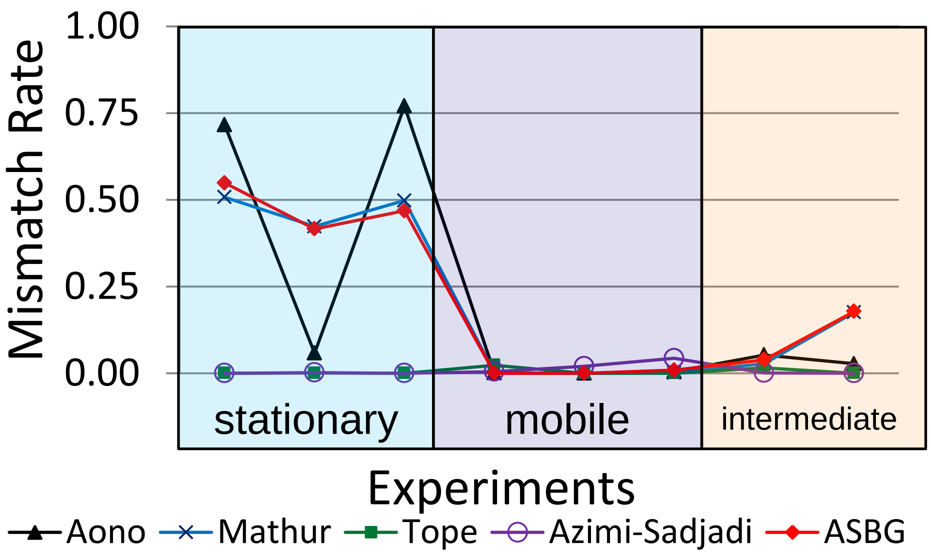




- mobile settings yield bits with low mismatch rates

# Comparison of Key Extraction Approaches in various Settings

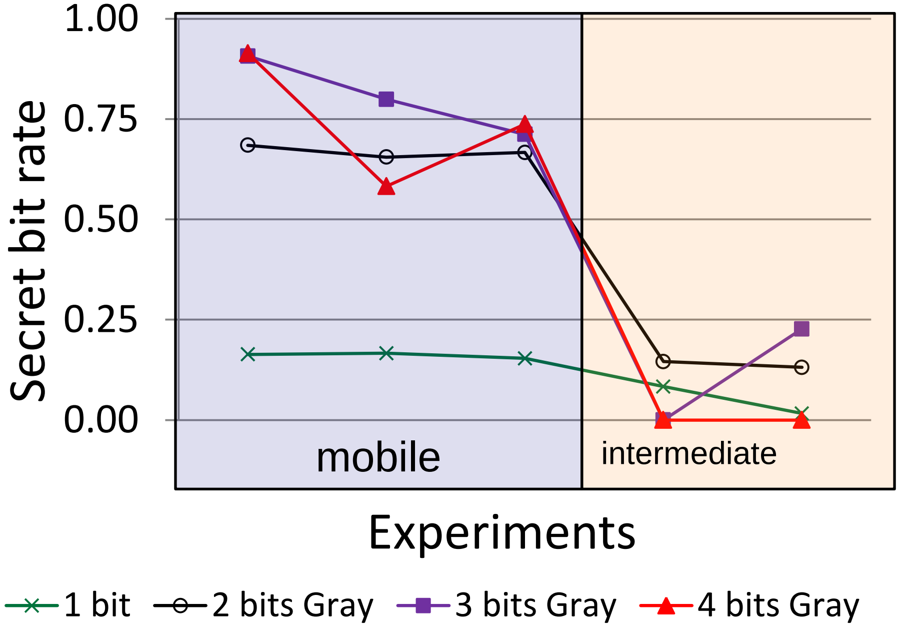


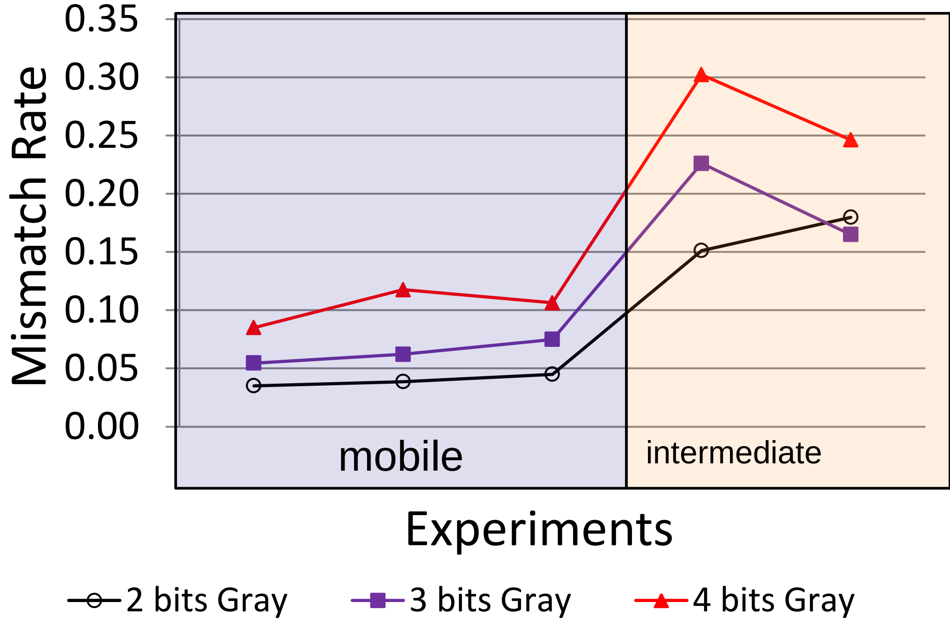




- ASBG exhibits highest secret bit rate among those with entropy > 0.7

# Multiple Bit Extraction




- significant increase in secret bit rate
  - at least 4 times (with N = 2) compared to single bit extraction
- bit mismatch rate increases with N
- Gray code assignment produces smaller mismatch rates

# Summary

- mobile settings best suited for key extraction due to low mismatch
- don't depend solely on movements of limited intermediate objects
  - beware of predictable channel attack!
- our environment adaptive quantization scheme + information reconciliation + privacy amplification generates high entropy bits at high rate