Practical quantum-key-distribution post-processing

Xiongfeng Ma

IQC, University of Waterloo

Chi-Hang Fred Fung, Jean-Christian Boileau, Hoi Fung Chau


Hoi-Kwong Lo, Norbert Lütkenhaus
Outline

- Introduction
- Finite key analysis
- Post-processing
- Conclusion
Introduction
A little bit history

- Prepare-and-measure protocols
  - BB84, six-state…
- Entanglement based protocols
  - Ekert91, BBM92
- Unconditional security proof
  - Mayers (1996)
  - Lo and Chau (1999)
  - Shor and Preskill (2000)
- Security analysis for QKD with imperfect devices
  - E.g. Mayers, Lütkenhaus, ILM
  - Koashi-Preskill
  - Gottesman-Lo-Lütkenhaus-Preskill (GLLP)
Basis independent setup

**BBM92 (Bennett, Brassard & Mermin 1992)**

- One can assume Eve has a full control of the source.
- The basis choice is independent of the state in the channel.
- The source can be treated as a perfect single photon (qubit) or EPR source.

Objectives

- Link between security analyses and experiments
  - Infinite-key analysis to finite-key analysis
- Post-processing
  - From raw key to final key, step by step
- Parameter optimization
  - Biased basis choice
  - Security parameter vs. secure-key cost
- Towards a security analysis standard
Finite key analysis
Finite-key issues

Initial key
- Authentication: man-in-the-middle attack
- Other uses: encryption of classical communication in the post-processing

Composability
- Generated key may be used for the next round of QKD
- Key growth rather than key distribution

No perfect key in real-life
- Identical: Alice and Bob share the same key
- Real-life: no perfect error correction code, i.e., any error correction code can only guarantee with a certain confidence interval
- Private: Eve knows nothing about the final key
- Real-life: Eve can just guess the bit values of the initial key with a successful probability, $\varepsilon=2^{-k}$
Composable security

Confidence interval / Failure probability
- Confidence interval: the probability that the final key is identical and private, $1-\varepsilon$
- The failure probability for $n$ rounds: $n\varepsilon$, linear dependence
- Exponentially decreasing with key cost $k$
- A rough guess: $\varepsilon=2^{-O(k)}$ per round
- Smaller than a certain threshold determined by some practical use of the key, say $10^{-10}$
- Remark: it does not mean Eve knows $n\varepsilon$-bit information about the final key

Composable security definition
- Failure probability is smaller than the pre-determined threshold for all rounds of QKD system usages

M. Ben-Or et al, in TCC (2005), 386.
To-do list

- Re-exam security proofs
  - Underlying assumptions
  - Formulas from asymptotic security analyses should be re-derived
  - Note: all the security analyses are about privacy amplification
- Calculate failure probability and secure-key cost
  - For each step: basis sift, error correction, authentication, error verification, and phase error rate estimation
  - Efficiency of privacy amplification
- Parameter optimization
  - Biased basis choice
  - Secure-key cost for each step
Post-processing
Underlying assumptions

- A single photon (qubit) source or a basis-independent source
  - Coherent state (e.g., with decoy state) QKD: in progress
- A detection system: compatible with the squashing model
- Classical communication
- Random number generator and key management
Procedures I

- Key sift [neither authenticated nor encrypted]
  - Discard all no-clicks and randomly assign double-clicks
  - Other sift scheme might be applied, e.g., Ma, Moroder and Lütkenhaus, arXiv:0812.4301 (2008)

- Basis sift [authenticated but not encrypted]
  - Use a 2k-bit secure key to generate a Toeplitz matrix
  - Calculate the tag by multiply the matrix with her/his basis string
  - Send each other the basis string with the tag
  - Discard those bits that used different bases
  - Other sift scheme might be applied, e.g., SARG’04

\[ \varepsilon_{bs} = n2^{-k_{bs}} + 1 \]
Procedures II

- **Error correction [not authenticated but encrypted]**
  - Any error correction scheme can be applied here
  - Count the number of bits in the classical communication

\[
k_{ec} = n_x f(e_{bx}) H(e_{bx}) + n_z f(e_{bz}) H(e_{bz})
\]

- **Error verification [essentially an authentication problem]**
  - Alice sends an encrypted tag to Bob
  - Bob verify the tag
  - If failed, they can go back to error correction again

\[
\epsilon_{ev} = (n_x + n_z)2^{-k_{ev}+1}
\]
Procedures III

Phase error rate estimation

- \( \text{Prob.\{phase error\}} = \text{Prob.\{bit error\}} \)
- Asymptotically, rates \( \rightarrow \) probabilities
- A simple random sampling problem

\[
P_{\theta x} < \frac{\sqrt{n_x + n_z}}{\sqrt{n_x n_z} e_{bx} (1 - e_{bx})} 2^{-(n_x + n_z) \xi_x(\theta_x)}
\]

\[
\xi_x(\theta_x) \equiv H(e_{bx} + \theta_x - q_x \theta_x) - q_x H(e_{bx}) - (1 - q_x) H(e_{bx} + \theta_x)
\]

\[
q_x = \frac{n_x}{n_x + n_z}
\]
Procedure IV

Privacy amplification [authenticated but not encrypted]

- Alice generates an \( n_x + n_z + l - 1 \) bit random bit string and sends it to Bob through an authenticated channel.
- They use this random bit string to generate a Toeplitz matrix to do privacy amplification.

\[
l = n_x [1 - H(e_{bz} + \theta_z)] + n_z [1 - H(e_{bx} + \theta_x)] - t_{oe}
\]

\[
\varepsilon_{pa} = (n_x + n_z + l - 1)2^{-k_{pa}+1} + 2^{-t_{oe}}
\]
Final key rate

Key rate formula

\[ NR \geq n_x [1 - f(e_{bx})H(e_{bx}) - H(e_{bz} + \theta_z)] \]

\[ + n_z [1 - f(e_{bz})H(e_{bz}) - H(e_{bx} + \theta_x)] - k_3 \]

\[ k_3 \equiv 2k_{bs} + k_{ev} + k_{pa} + t_{oe} \]

Finite-key effect

- Phase error rate estimation, which determines the biased basis choice
- \( k_3 \): cost of authentication, error verification, efficiency of privacy amplification
Finite-key effects

- Error correction [fixed]
- Phase error rate estimation:
  - The amount of privacy amplification
  - Optimal basis bias
- Other parts
  - Authentication
  - Error verification
  - Efficiency of privacy amplification
Quick results

An extreme example
- Raw key size $n=10^{30}$ and $\varepsilon=10^{-30}$;
- $k_3=947$ bits and $\varepsilon_3<10^{-32}$

A typical example
- $n=10^7$, $\varepsilon=10^{-7}$;
- $k_3=202$ bits and $\varepsilon_3<10^{-9}$

Observation
- Main effect comes from phase error estimation
- The efficiency of privacy amplification is close to 1
- The costs of authentication and error verification are negligible in a normal (say, data size $>10k$) experiment
$$e_x = e_z = 4\%$$
$$\epsilon = 10^{-7}$$
100% error correction efficiency

- **$p_x$** set by Alice
- **$q_x$** obtained by Bob
$e_x = e_z = 4\%$

$\varepsilon = 10^{-7}$

100% error correction efficiency
Conclusion

- A practical post-processing scheme with failure probability as the security definition
- Error verification: essentially an authentication scheme
- Phase error estimation: a strict bound
- Efficiency of privacy amplification: high
- Parameter optimization: main finite-key effect comes from phase error rate estimation
Further discussion

Further improvement in the privacy amplification step: no classical communication needed?
  - Efficiency: failure probability, secure-key cost

Detector efficiency mismatch
  - Squashing model

Finite-key analysis for other protocols, such as decoy-state, COW and DPS

Compare with other approaches, such as the one by Scarani and Renner
References

- M. Ben-Or et al, in TCC (2005), 386.
Thank you!

Xiongfeng Ma, ICQFT’09