An Information Reconciliation Protocol for Secret-Key Agreement with Small Leakage

Christoph Pacher[1†], Philipp Grabenweger[1], Jesús Martínez-Mateo[2], Vicente Martin[2]

† `Christoph.Pacher@AIT.ac.at`

[1] *Digital Safety & Security Department, AIT Austrian Institute of Technology GmbH, 1220 Vienna, Austria*
[2] *Madrid Center for Computational Simulation, Universidad de Madrid, Spain*

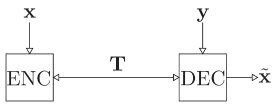
SUMMARY

We report on our information-theory based tuning approach (partly discussed in [3]) of the so-called Cascade [1] protocol to achieve very small leakage: We prove that powers of two are optimal values for the number of bits in the initial blocks (Box 3). This confirms and explains results of recent numerical optimizations [2]. Bits are corrected separately according to their individual error probability in the second pass (round); corrected bits are fully taken into account (Box 4). Simulation results for efficiency and throughput of these optimizations are shown in Box 5&6. A significant improvement for the efficiency is obtained, although at a highly increased number of exchanged messages. Also in Box 5&6 variants with still very high efficiency but also high throughput are shown.
The leakage is for block sizes of 2^{16} typically only 2.5% above the Shannon limit, and notably, this holds for an error rate p between 1% and 50%. For p between 1% and 6% the leakage is only 2% above the Shannon limit. As comparison, the leakage of the original Cascade algorithm is 20% (40%) above the Shannon limit for a p of 10% (35%).

1 INFORMATION RECONCILIATION (IR)

Description

- Alice and Bob hold raw keys $\boldsymbol{x}, \boldsymbol{y} \in \{0,1\}^n$ distributed according to $(P_{XY})^{\times n}$.
  Here we assume that $\boldsymbol{y}$ can be seen as the result of transmitting $\boldsymbol{x}$ over n identical binary symmetric channels with cross-over probability p, i.e. BSC(p).
- Two-way IR: Alice and Bob exchange a series of messages $\mathbf{T} = \{T_1, \ldots, T_m\}$, Bob constructs an estimate $\tilde{\boldsymbol{x}}$ of $\boldsymbol{x}$.



Efficiency of Information Reconciliation

- Two different definitions for efficiency are in use: $h_2(p) = -p\log_2 p - (1-p)\log_2(1-p)$
  - Efficiency β_{IR} is defined as ratio of the maximal length of the secret key (taking into account the leakage) and the capacity of the BSC(p):
    $$\beta_{IR} = \frac{n - \mathrm{leak}_{IR}}{n(1 - h_2(p))}$$
  - Efficiency η_{IR} is defined as ratio of number of transmitted bits and the (asymptotic) minimum of the leakage (Shannon limit):
    $$\eta_{IR} = \frac{\mathrm{leak}_{IR}}{nh_2(p)}$$

Motivation for this work

Reducing the leakage leak_{IR} increases the length of the secret key in key agreement. (The leakage is hard to calculate but we can use m, the number of transmitted bits, as an upper bound.)

2 INFORMATION THEORETIC APPROACH FOR IMPROVING THE EFFICIENCY

Main Idea

Transmit only bits without or with little redundancy (entropy close to one). This means, the conditional probability of (most) transmitted bits to be a one given the values of all previously transmitted bits must be (close to) $\frac{1}{2}$.

A-priori probabilities that Alice's and Bob's parity bits are equal or not equal are given by

$$p_{\underset{\text{notequal}}{\text{equal}}}(k_i, p) := \Pr\left\{\boldsymbol{h}\cdot\boldsymbol{x} \overset{=}{\neq} \boldsymbol{h}\cdot\boldsymbol{y}\right\} = \frac{1 \overset{+}{-} (1-2p)^{k_i}}{2} \overset{>}{<} \frac{1}{2}.$$

$\boldsymbol{h}$ ... parity check row, see Box 3

Information per parity bit: $H(\boldsymbol{h}\cdot\boldsymbol{x}|\boldsymbol{h}\cdot\boldsymbol{y}) = h_2(p_{\mathrm{equal}}(k_i, p))$.
To get efficient coding: $H(\boldsymbol{h}\cdot\boldsymbol{x}|\boldsymbol{h}\cdot\boldsymbol{y}) \approx 1 \iff p_{\mathrm{equal}}(k_i, p) \approx \frac{1}{2} \iff k_i$ must be "large enough".

3 DICHOTOMIC SEARCHES IN BLOCKS WITH DIFFERENT PARITY

A single parity (bisection step)

$$\boldsymbol{h} = \Big(\ldots \overbrace{0\ldots0}^{k_i}\ \overbrace{1\ldots1}^{k_i}\ \overbrace{0\ldots0}^{k_i} \ldots\Big) \quad \longrightarrow \quad \boldsymbol{h}' = \Big(\ldots \overbrace{0\ldots0}^{k_i}\ \overbrace{1\ldots1}^{\lfloor k_i/2\rfloor}\ \overbrace{0\ldots0}^{\lceil k_i/2\rceil}\ \overbrace{0\ldots0}^{k_i} \ldots\Big)$$

k_i is even, different parity

$$\Pr\left\{\boldsymbol{h}'\cdot\boldsymbol{x} = \boldsymbol{h}'\cdot\boldsymbol{y}|\boldsymbol{h}\cdot\boldsymbol{x}\neq\boldsymbol{h}\cdot\boldsymbol{y}\right\} = \Pr\left\{\boldsymbol{h}'\cdot\boldsymbol{x} \neq \boldsymbol{h}'\cdot\boldsymbol{y}|\boldsymbol{h}\cdot\boldsymbol{x}\neq\boldsymbol{h}\cdot\boldsymbol{y}\right\} = \frac{1}{2}$$
$$\implies H(\boldsymbol{h}'\cdot\boldsymbol{x}|\boldsymbol{h}\cdot\boldsymbol{x}\neq\boldsymbol{h}\cdot\boldsymbol{y}, \boldsymbol{h}'\cdot\boldsymbol{y}) = 1.$$

A complete dichotomic search to find a faulty bit

$$k_i \text{ is a power of two} \overset{\text{by induction}}{\iff} \textbf{all} \text{ parities fulfill } H = 1.$$

k_i is even, same parity

$$H(\boldsymbol{h}'\cdot\boldsymbol{x}|\boldsymbol{h}\cdot\boldsymbol{x} = \boldsymbol{h}\cdot\boldsymbol{y}, \boldsymbol{h}'\cdot\boldsymbol{y}) < 1 \text{ because } \Pr\left\{\boldsymbol{h}'\cdot\boldsymbol{x} \overset{=}{\neq} \boldsymbol{h}'\cdot\boldsymbol{y}|\boldsymbol{h}\cdot\boldsymbol{x} = \boldsymbol{h}\cdot\boldsymbol{y}\right\} = \frac{p^2_{\underset{\text{notequal}}{\text{equal}}}(n,p)}{p_{\mathrm{equal}}(2n,p)} \overset{>}{<} \frac{1}{2}.$$

k_i is odd
For odd k_i and same or different parity and general p always $H < 1$.

Optimality condition for block size

To obtain parity bits without redundancy it is necessary and sufficient to divide blocks with different parity that have a size k_i which is a power of two.

4 CATEGORISATION AND FURTHER EFFICIENCY IMPROVEMENTS

Dichotomic Searches

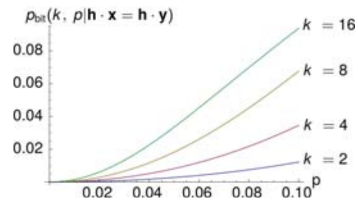
- Recursively dividing only blocks with odd parity we localize one faulty bit.
- What about blocks with same parity? Not used so far.

Analysis of Blocks with Same Parity

- Starting with $k_1 = 2^K$, approximately in 50% of the cases Alice's and Bob's blocks have the same parity.
- During the dichotomic search in each 2^K-block with different parity we learn $K-1$ blocks that have same parity (with sizes $2^{K-1}, 2^{K-2}, \ldots, 2^1$).
- Conditional bit error probability in a block of size k with same parity

$$p_{\mathrm{bit}}(k, p|\boldsymbol{h}\cdot\boldsymbol{x} = \boldsymbol{h}\cdot\boldsymbol{y}) = p\frac{p_{\mathrm{notequal}}(k-1,p)}{p_{\mathrm{equal}}(k,p)}.$$

k	$p_{\mathrm{bit}}(k, 0.06\|\boldsymbol{h}\cdot\boldsymbol{x} = \boldsymbol{h}\cdot\boldsymbol{y})$
2	0.004
4	0.012
8	0.026
16	0.045



Using the Information in Blocks with Same Parity (Categorisation)

- The bit error probability is non-uniformly distributed before the second pass!
- The original Cascade protocol performs random shuffling in each pass.
- We put bits in different categories $\mathcal{C}_K$ according to $p_{\mathrm{bit}}(2^K, p|\boldsymbol{h}\cdot\boldsymbol{x} = \boldsymbol{h}\cdot\boldsymbol{y})$.
- Instead of using one block size k_2, we choose the block size $k_{2,K}$ individually in each category $\mathcal{C}_K$ ($k_{2,K}$ is always a power of two).

Keeping a Record of All Corrected Bits

- For each bisection step in a block with different parity, first check if one of the halves consists only of already corrected bits. If this is the case, the parity for this half is equal for Alice and Bob, and that for the other half is different $\implies$ need not transmit parity of neither half.
- Same argument for initialisation of a new pass: If a block consists only of previously corrected bits, its parity must be the same for Alice and Bob, so it need not be exchanged.

5 IMPROVING THROUGHPUT

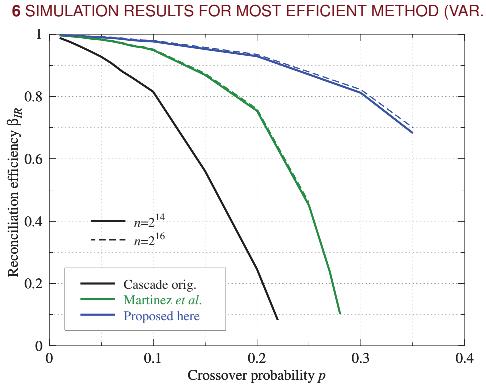
Different variants of the cascade protocol were tested. The most efficient variant (var. (1), see Box 6) was modified to achieve higher throughput at the cost of little lower efficiency β_{IR}. In var. (2) to (4), number of passes was 14. For var. (3) and (4), $k_2 = \min\left(2^{\lceil(\lceil\log_2(1/p)-0.5\rceil+12)/2\rceil}, n/2\right)$. Other k_i values as for var. (1) (see Box 6).

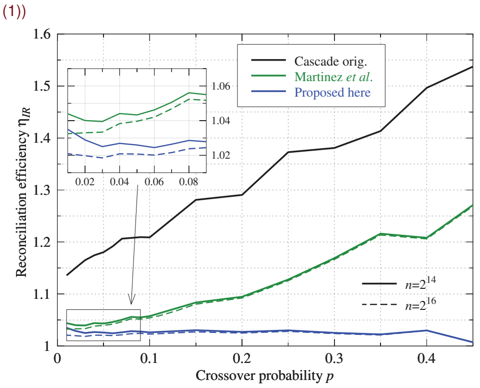
protocol	record corrected bits	simultaneous bisection	categorisation	permutations
var. (1)	all	no	yes	Fisher-Yates shuffle
var. (2)	only current pass	yes	yes	Knuth shuffle
var. (3)	none	yes	no	Knuth shuffle
var. (4)	only current pass	yes	no	$i \mapsto (ai+b) \mod n$

Table: Cross-over probability p, efficiency β_{IR} and throughput for different protocol variants and selected block sizes n

	var. (2), $n = 2^{14}$		var. (3), $n = 2^{14}$		var. (4), $n = 2^{24}$		
p	β_{IR}	throughput (Mbit/s)	β_{IR}	throughput (Mbit/s)	β_{IR}	throughput (Mbit/s)	throughput (Mbit/s) latency 1ms
0.01	0.9959	12.46	0.9963	20.24	0.9976	5.47	2.98
0.02	0.9936	7.98	0.9935	12.89	0.9949	2.87	2.04
0.03	0.9913	6.18	0.9907	10.53	0.9921	2.15	1.56
0.04	0.9884	4.71	0.9860	8.41	0.9879	1.50	1.00
0.05	0.9856	4.32	0.9827	7.40	0.9836	1.35	1.18
0.06	0.9828	3.60	0.9776	6.40	0.9789	1.04	0.90
0.07	0.9797	3.08	0.9708	5.72	0.9726	0.85	0.72
0.08	0.9753	2.68	0.9623	5.13	0.9649	0.71	0.59
0.09	0.9702	2.85	0.9574	5.08	0.9577	0.78	0.69
0.10	0.9675	2.51	0.9492	4.62	0.9501	0.69	0.60

6 SIMULATION RESULTS FOR MOST EFFICIENT METHOD (VAR. (1))





We use 16 passes,

$$k_1 = \begin{cases} \min(2^{\lceil\log_2(1/p)-0.5\rceil}, n/2) & \text{if } p \le 0.25, \\ \min(\max(1, 2^{\lceil\log_2(1/p)-0.5\rceil-1}), n/2) & \text{if } p > 0.25, \end{cases}$$

$$k_{2,K} = \min(2^{\left\lceil\log_2\left(4/p_{\mathrm{bit}}(2^K, p|\boldsymbol{h}\cdot\boldsymbol{x}=\boldsymbol{h}\cdot\boldsymbol{y})\right)-0.5\right\rceil}, |\mathcal{C}_K|/2),$$

$k_3 = 2^{12}$, $k_4 = \ldots k_{16} = n/2$, and get a FER of $\epsilon \approx 10^{-4}$.

Table: Block size n, cross-over probability p, efficiency values β_{IR} and η_{IR}, frame error rate ϵ, bit error rate ϵ_b, number of messages, throughput (without latency).

n	p	β_{IR}	η_{IR}	ϵ	ϵ_b	$\#msg$	throughput (Mbit/s)
2^{10}	0.03	0.9747	1.105	1.6×10^{-4}	0.00146	116	2.564
2^{10}	0.1	0.9433	1.064	2.3×10^{-4}	0.00376	213	1.120
2^{10}	0.3	0.9223	1.0466	6×10^{-5}	0.00208	202	0.505
2^{14}	0.03	0.994	1.025	1.4×10^{-4}	0.0029	1386	2.617
2^{14}	0.1	0.9768	1.0263	4×10^{-5}	0.00466	2805	0.877
2^{14}	0.3	0.8116	1.0254	5×10^{-5}	0.0026	2878	0.320
2^{16}	0.03	0.9955	1.0185	1×10^{-4}	0.0002	5180	2.102
2^{16}	0.1	0.9798	1.023	0	0	10773	0.671
2^{16}	0.3	0.822	1.024	0	0	11412	0.238

References

[1] G. Brassard and L. Salvail, *Secret key reconciliation by public discussion*, Advances in Cryptology: Proc. Eurocrypt 93, pp. 410-423 (1993).

[2] J. Martinez-Mateo, et. al., *Demystifying the Information Reconciliation Protocol Cascade*, Quantum Information and Computation, Vol. 15, pp. 453-477 (2015).

[3] C. Pacher, P. Grabenweger, J. Martinez-Mateo, V. Martin, *An Information Reconciliation Protocol for Secret-Key Agreement with Small Leakage*, in 2015 IEEE International Symposium on Information Theory (ISIT), June 14-19, Hong Kong (2015).