Radboud University

Rank estimation for large key

Vincent Grosso

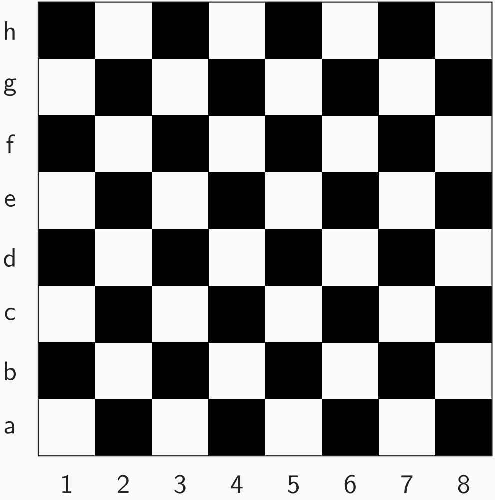
November 15, 2017

Institute for Computing and Information Sciences – Digital Security
Radboud University Nijmegen

Table of contents

Key enumeration/Rank estimation

Generous King



Generous King

h	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2
g	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2
f	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2
e	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2
d	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2
c	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2
b	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2
a	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2
	1	2	3	4	5	6	7	8

Each cell contains 2 piles of coins.

Generous King

h	$P_1(1)$	$P_1(2)$	$P_1(3)$	$P_1(4)$	$P_1(5)$	$P_1(6)$	$P_1(7)$	$P_1(8)$
g	$P_1(1)$	$P_1(2)$	$P_1(3)$	$P_1(4)$	$P_1(5)$	$P_1(6)$	$P_1(7)$	$P_1(8)$
f	$P_1(1)$	$P_1(2)$	$P_1(3)$	$P_1(4)$	$P_1(5)$	$P_1(6)$	$P_1(7)$	$P_1(8)$
e	$P_1(1)$	$P_1(2)$	$P_1(3)$	$P_1(4)$	$P_1(5)$	$P_1(6)$	$P_1(7)$	$P_1(8)$
d	$P_1(1)$	$P_1(2)$	$P_1(3)$	$P_1(4)$	$P_1(5)$	$P_1(6)$	$P_1(7)$	$P_1(8)$
c	$P_1(1)$	$P_1(2)$	$P_1(3)$	$P_1(4)$	$P_1(5)$	$P_1(6)$	$P_1(7)$	$P_1(8)$
b	$P_1(1)$	$P_1(2)$	$P_1(3)$	$P_1(4)$	$P_1(5)$	$P_1(6)$	$P_1(7)$	$P_1(8)$
a	$P_1(1)$	$P_1(2)$	$P_1(3)$	$P_1(4)$	$P_1(5)$	$P_1(6)$	$P_1(7)$	$P_1(8)$
	1	2	3	4	5	6	7	8

Each cell contains 2 piles of coins.

- For each column the first pile contains the same amount of coins $P_1(a,1) = P_1(b,1) = P_1(1)$

Generous King

h	$P_2(h)$	$P_2(h)$	$P_2(h)$	$P_2(h)$	$P_2(h)$	$P_2(h)$	$P_2(h)$	$P_2(h)$
g	$P_2(g)$	$P_2(g)$	$P_2(g)$	$P_2(g)$	$P_2(g)$	$P_2(g)$	$P_2(g)$	$P_2(g)$
f	$P_2(f)$	$P_2(f)$	$P_2(f)$	$P_2(f)$	$P_2(f)$	$P_2(f)$	$P_2(f)$	$P_2(f)$
e	$P_2(e)$	$P_2(e)$	$P_2(e)$	$P_2(e)$	$P_2(e)$	$P_2(e)$	$P_2(e)$	$P_2(e)$
d	$P_2(d)$	$P_2(d)$	$P_2(d)$	$P_2(d)$	$P_2(d)$	$P_2(d)$	$P_2(d)$	$P_2(d)$
c	$P_2(c)$	$P_2(c)$	$P_2(c)$	$P_2(c)$	$P_2(c)$	$P_2(c)$	$P_2(c)$	$P_2(c)$
b	$P_2(b)$	$P_2(b)$	$P_2(b)$	$P_2(b)$	$P_2(b)$	$P_2(b)$	$P_2(b)$	$P_2(b)$
a	$P_2(a)$	$P_2(a)$	$P_2(a)$	$P_2(a)$	$P_2(a)$	$P_2(a)$	$P_2(a)$	$P_2(a)$
	1	2	3	4	5	6	7	8

Each cell contains 2 piles of coins.

- For each column the first pile contains the same amount of coins
  $P_1(a,1) = P_1(b,1) = P_1(1)$
- For each row the second pile contains the same amount of coins
  $P_2(a,1) = P_2(a,2) = P_2(a)$

Generous King

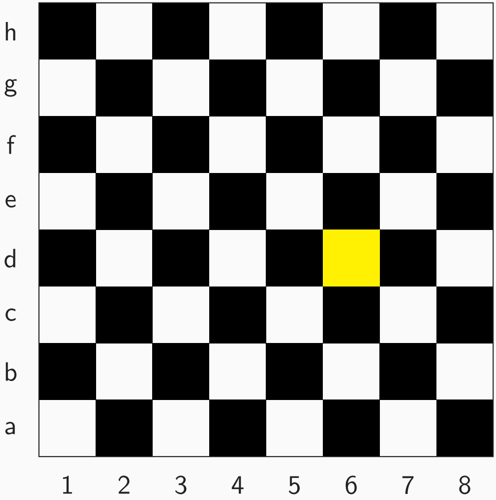
h	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2
g	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2
f	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2
e	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2
d	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2
c	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2
b	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2
a	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2	P_1, P_2
	1	2	3	4	5	6	7	8

Each cell contains 2 piles of coins.

- For each column the first pile contains the same amount of coins
  $P_1(a, 1) = P_1(b, 1) = P_1(1)$
- For each row the second pile contains the same amount of coins
  $P_2(a, 1) = P_2(a, 2) = P_2(a)$

You can take piles of n cells, how to maximize the profit?

How many better solution



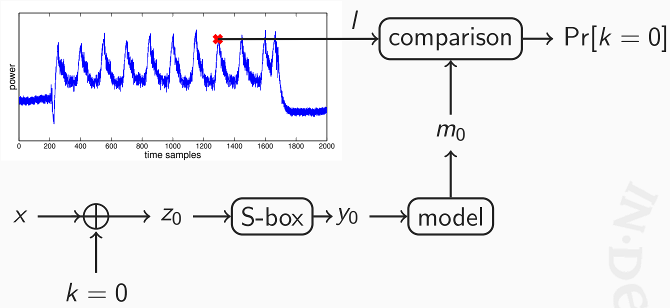
How to many cells have more coins than a specified cell?

Generalization

The problem can be generalized

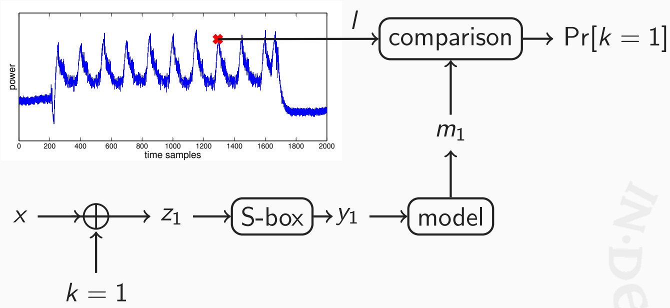
- with higher number of rows/columns
- higher dimension

Side-channel attacks



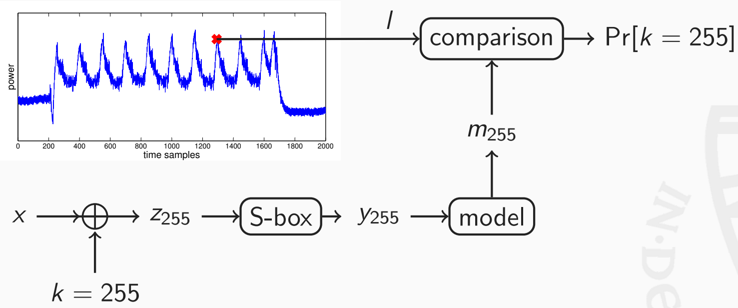
Divide-and-conquer strategy.

Side-channel attacks



Divide-and-conquer strategy.

Side-channel attacks



Divide-and-conquer strategy.

SCA result

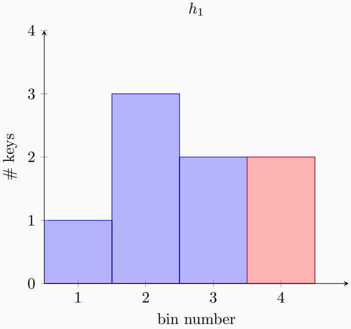
k_0	k_1	k_2	$\cdots$	k_{15}
0X2b,0.125	0X23,0.128	0X23,0.325		0X45,0.347
0Xcd,0.100	0X51,0.045	0Xde,0.204		0Xdc,0.210
0Xae,0.050	0Xff,0.035	0Xfe,0.036		0X83,0.151
0X12,0.025	0X2b,0.025	0X21,0.029		0X13,0.435
$\cdots$	$\cdots$	$\cdots$		$\cdots$

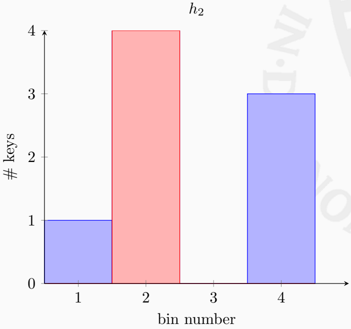
Real key.

Previous solution

The histogram

	k_1			k_2		
Candidate	Pr	log	bin	Pr	log	bin
0	0.6643	-0.5901	1	0.0012	-9.7027	3
1	0.2588	-1.9501	1	0.0011	-9.8283	3
2	0.0313	-4.9977	2	0.3588	-1.4787	1
3	0.0412	-4.6012	2	0.0713	-3.8100	1
4	0.0001	-13.2877	4	0.5643	-0.8255	1
5	0.0020	-8.9658	3	0.0012	-9.7027	3
6	0.0013	-9.5873	3	0.00005	-14.2877	4
7	0.0010	-9.9658	3	0.00205	-8.9302	3

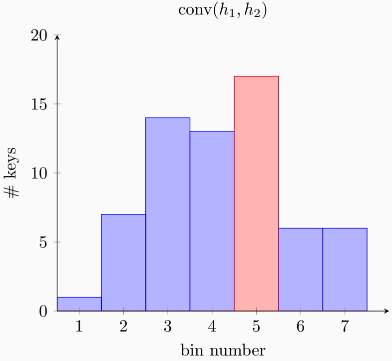




Mix result

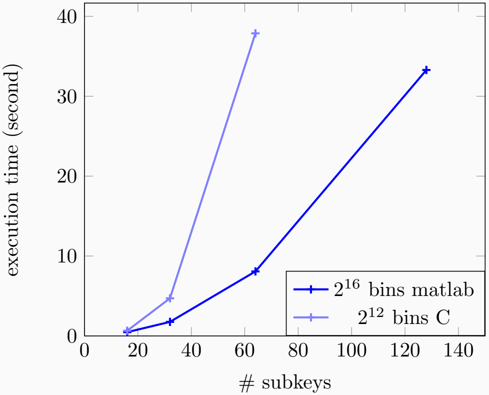
Perform convolution of histogram

$$conv(h_1, h_2)[i] = \sum_{j=0}^{i} h_1[j]h_2[i-j]$$



Limitation for larger keys

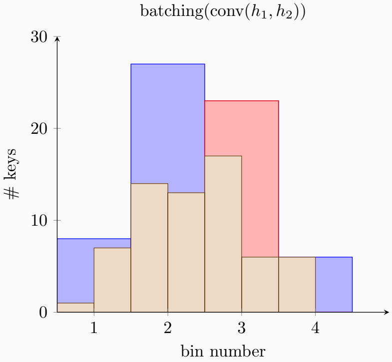
For large number of dimension we perform convolution on larger and larger histograms: could be costly.



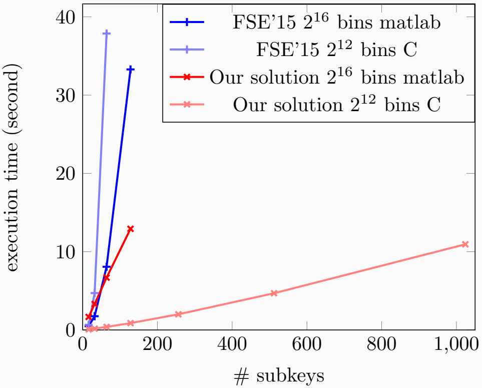
New solution

Batching

Keep the size of the histogram constant

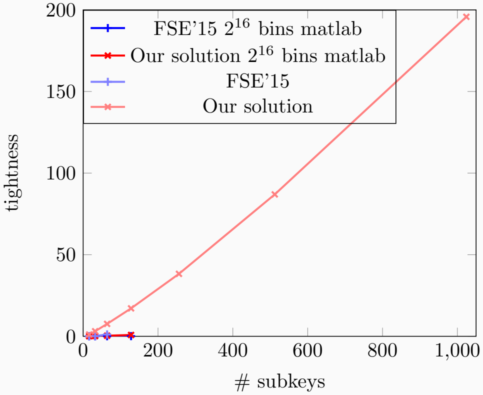


Efficiency



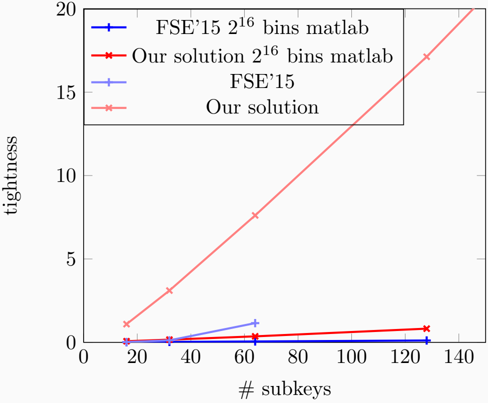
Our solution has a complexity linear in the number of subkeys

Tightness



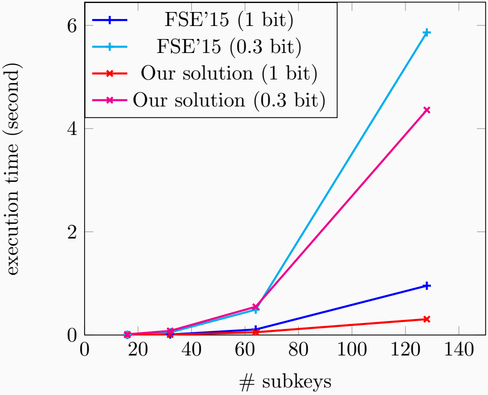
200
150
100
50
0
0
200
400
600
800
1,000
tightness
subkeys
FSE'15 2^{16} bins matlab
Our solution 2^{16} bins matlab
FSE'15
Our solution

Tightness



FSE'15 2^16 bins matlab
Our solution 2^16 bins matlab
FSE'15
Our solution
tightness
subkeys
0
5
10
15
20
0
20
40
60
80
100
120
140

Similar tightness



Thanks