# Bias-based traitor tracing codes

Boris Škorić

Eindhoven University of Technology

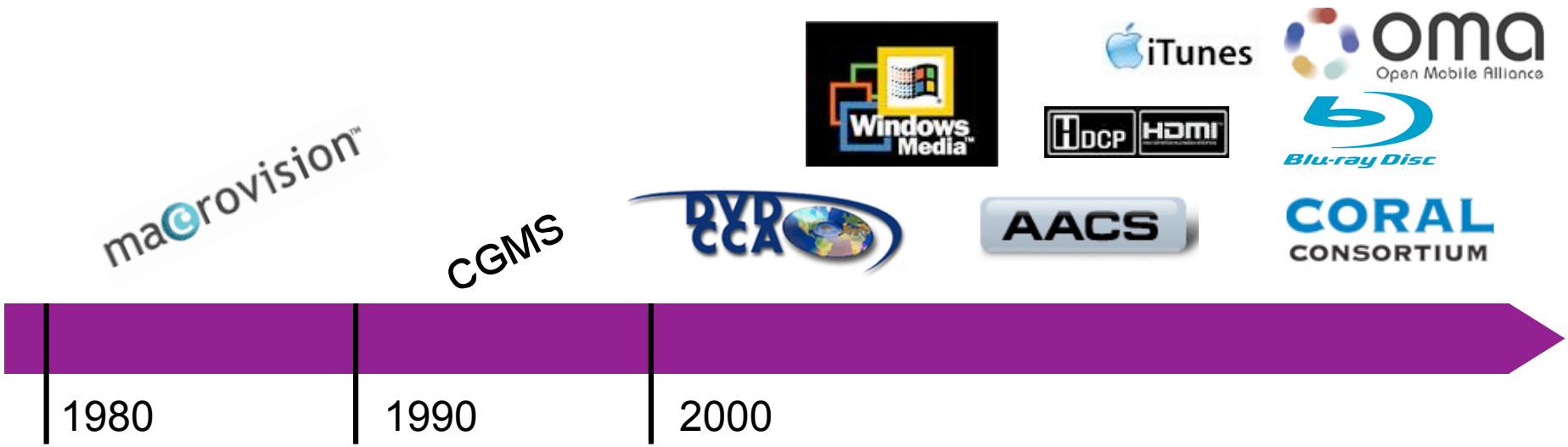
Guest lecture

3 December 2013

# Outline

- Collusion attacks on watermarks
- Bias-based codes (Tardos codes)
- Information-theoretic maxmin game
  - Saddlepoint
  - Capacity
- Capacity-achieving score function

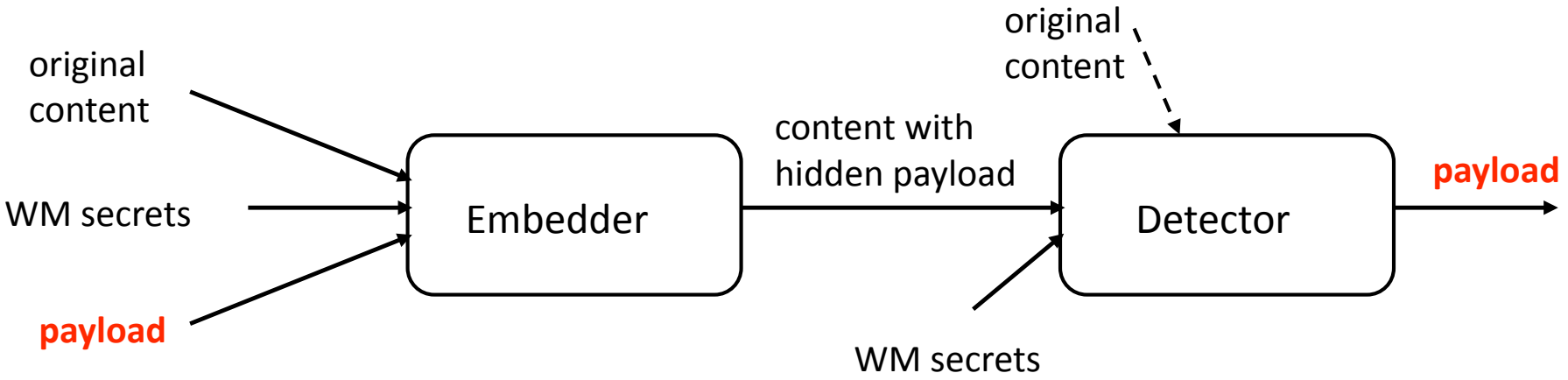
# Trends in content protection (old slide)




- Consumers increasingly dislike DRM
- Vista content protection spec

  "longest suicide note in history" (Gutmann 2006)
- "Disembodied" distribution ⇒ Hard to DRM-protect; Easy to watermark
- April 2007: EMI announces DRM-free music
- Gradual shift from copy prevention to distribution tracking

# Watermarking (a.k.a. Fingerprinting)




Forensic tracing

- Payload = unique identifier of recipient
- Redistribution traced back to source

Examples

- Jan.2004: Man arrested for distributing oscar screeners.
- Digital cinema

# Collusion attacks

"Coalition of pirates"


- Users pool their content
- Differences point to watermark
- Attackers remove watermark

■ = "detectable positions"

| | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| *pirate #1* | 1 | 1 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 1 |
| *#2* | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 1 |
| *#3* | 1 | 0 | 1 | 0 | 1 | 0 | 1 | 0 | 0 | 0 | 1 | 1 |
| *#4* | 1 | 1 | 1 | 0 | 0 | 0 | 1 | 1 | 0 | 0 | 0 | 1 |
| Attacked Content | 1 | 0/1 | 1 | 0 | 0/1 | 0 | 1 | 0/1 | 0/1 | 0 | 0/1 | 1 |

# Collusion-resistant watermarking

Requirements

- Resistance against $c \leq c_0$ attackers
- Low False Positive error rate
- Low False Negative error rate
- ... and all that with small watermark payload! (7bits/min. video)

Attack model

- "**Marking assumption**": Modification only at detectable positions
- Several options
  - **Restricted digit model**: Choice from available symbols only
  - Unreadable digit model: Erasure allowed
  - Arbitrary digit model: Arbitrary symbol (but not erasure)
  - General digit model

} equivalent for binary symbols

# History of collusion resistance: Code length

Construction

Boneh and Shaw 1998: $m = \mathcal{O}(c_0^4 \ln[n/\eta] \ln[1/\eta]), \quad q = 2$

Tardos 2003: $m = 100 c_0^2 \ln(1/\varepsilon_1), \quad q = 2, \quad \varepsilon_2 = \varepsilon_1^{c_0/4}$

Chor et al 2000: $m = 4 c_0^2 \log n, \quad q = 2 c_0^2$

Staddon et al 2001: $m = c_0^2 \log_q(n), \quad q > m - 2$

Huang + Moulin; Amiri + Tardos 2009: $m = 2 \ln 2 \cdot c_0^2 \ln[1/\varepsilon_1], \quad q = 2$

Tardos 2003:

$m = \Omega(c_0^2 \ln[1/\varepsilon_1]), \quad q$ arbitrary

Boneh and Shaw 1998:

$m = \Omega(c_0 \ln[1/c_0\eta]), \quad q = 2$

Lower bound

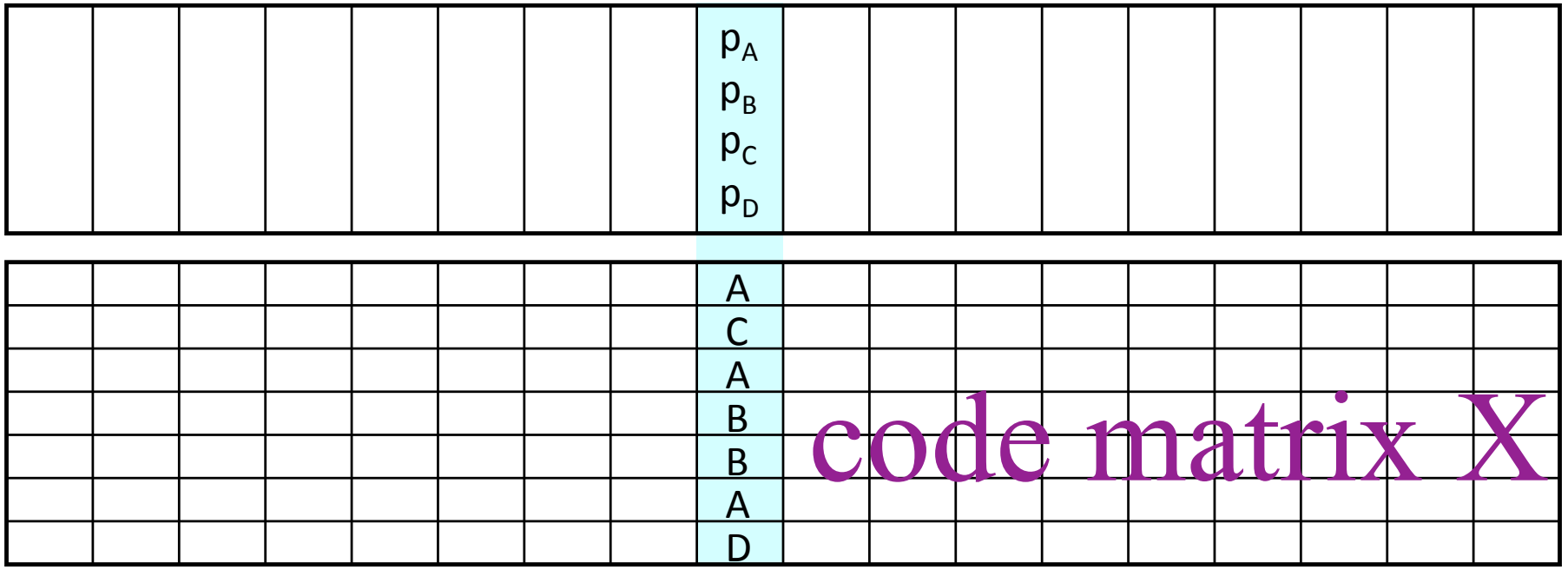
n = #users
m = code length in symbols
q = alphabet size
$\varepsilon_1$ = Prob[accuse specific innocent]
$\eta$ = Prob[not all accused are guilty]
$\varepsilon_2$ = False Negative prob.

# **Bias-based code** [Tardos 2003, ŠKC 2007]

Alphabet Q

Step 1:
For each position, generate bias vector $\mathbf{p}=(p_\alpha)_{\alpha\in Q}$. $|\mathbf{p}|=1$ $\mathbf{p} \sim F$

Step 2:
For each position and user, draw watermark symbol: Pr[symbol α] = $p_\alpha$.




pirated copy carries watermark y

Step 3:
Find attackers based on X and y

**Asymptotically optimal scaling: code length $\propto c_0^2$**

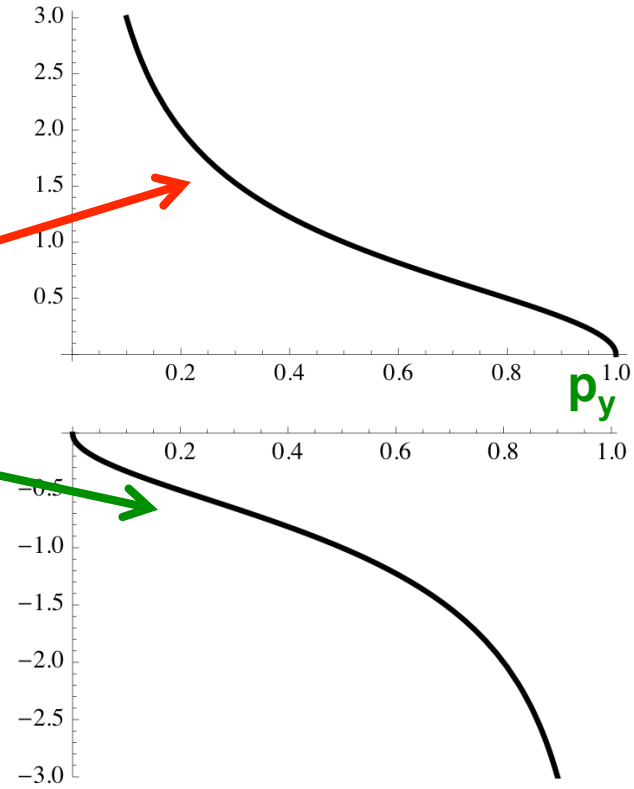
# De Tardos code

- Allereerste code met $m \propto c^2$ macht en klein alfabet
- In twee opzichten probabilistisch
  - staat kleine kans toe op valse beschuldiging en compleet missen van de aanvallers
  - constructie van de code is gerandomiseerd
- In 2003 verzonnen voor q=2, in 2007 uitgebreid naar algemene q

# Tardos code: Het traceren

- Er wordt een "ongeauthoriseerde" kopie gevonden
- Watermerk-detector ziet symbool $y_j$ in segment j
- Reken voor elke klant *i* de "score" $S_i$ uit
  - som van losse scores per segment: $S_i = \Sigma_j S_{ij}$

$$S_{ij} = \begin{cases} X_{ij} = y_j : & \sqrt{(1-p_y)/p_y} \\ X_{ij} \neq y_j : & -\sqrt{p_y/(1-p_y)} \end{cases}$$




- Klant *i* is verdacht als score $S_i$ boven een bepaalde grens uitkomt

# Tardos code: speciale eigenschappen

Scores van onschuldigen gedragen zich eenvoudig:

- Gemiddelde is nul in elk segment
- Variantie 1 in elk segment

$$\mathrm{E}[S_{ij}] = p_y \sqrt{\frac{1-p_y}{p_y}} - (1-p_y)\sqrt{\frac{p_y}{1-p_y}} = 0$$

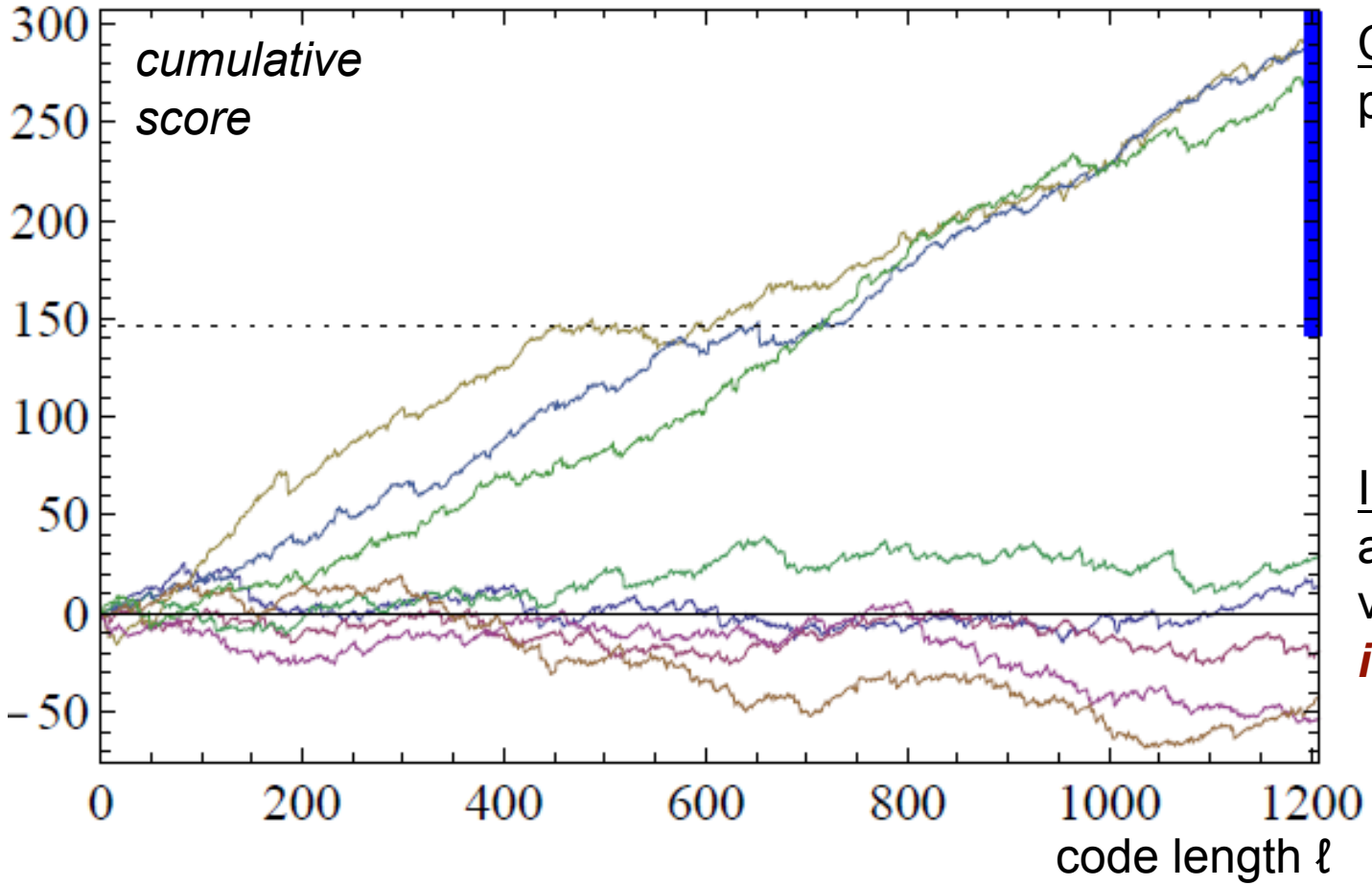
$$\mathrm{E}[S_{ij}^2] = p_y \frac{1-p_y}{p_y} + (1-p_y)\frac{p_y}{1-p_y} = (1-p_y) + p_y = 1$$

- De Tardos score-functie is de enige met deze eigenschap
- Aanvallers hebben geen invloed op gemiddelde en variantie van onschuldigen

# Separating the attackers from the innocents

Binary alphabet; Tardos score function

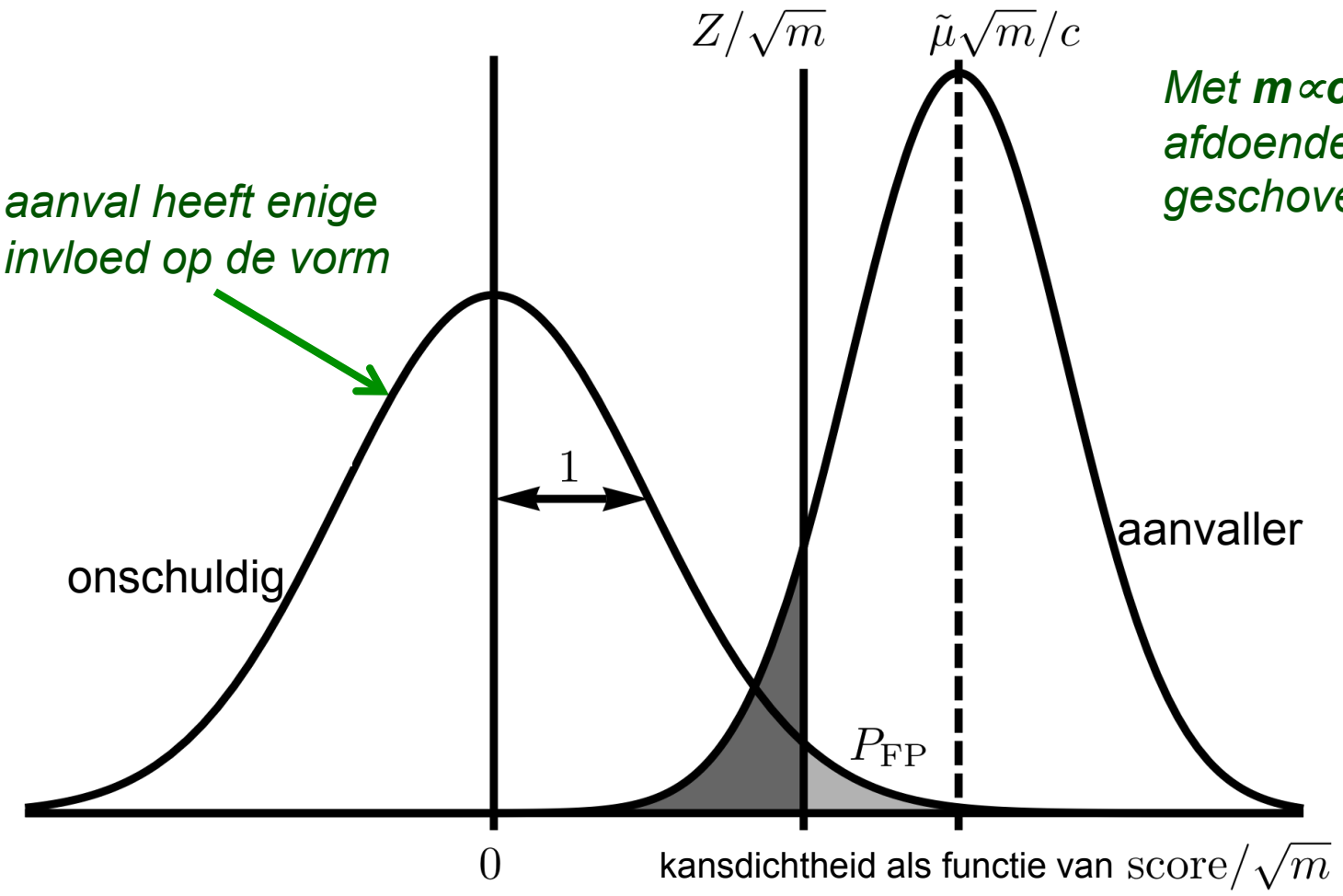
$$g(x,y,\boldsymbol{p}) = \begin{cases} \sqrt{(1-p_y)/p_y} & \text{if } x = y \\ -\sqrt{p_y/(1-p_y)} & \text{if } x \neq y \end{cases}$$




Coalition score:
positive slope

Innocent:
average 0
variance 1
***in each position***

# Gevolgen van de speciale eigenschappen



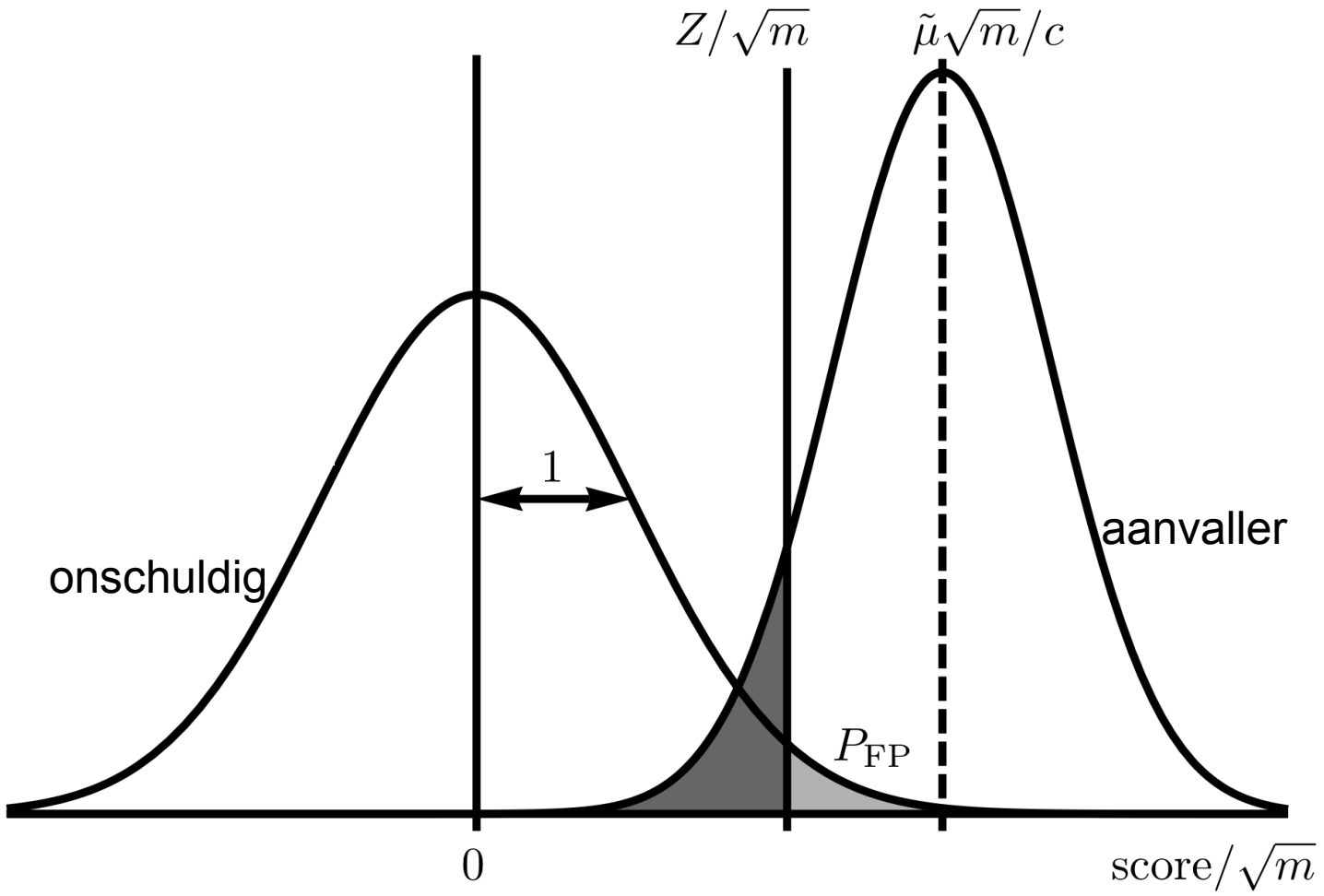

*Met **m**∝**c²** kunnen de curves afdoende uit elkaar geschoven worden*

$P_{FP}$ = kans dat (onschuldige) klant i onterecht wordt beschuldigd
m = #segmenten
c = #aanvallers
Z = grens

# "No framing"

Wat als er meer aanvallers zijn dan geanticipeerd?

- Rechtercurve schuift links van de grens Z ⇒ **aanvallers niet gepakt**
- Linkercurve verandert nauwelijks ⇒ **geen onschuldigen gepakt**



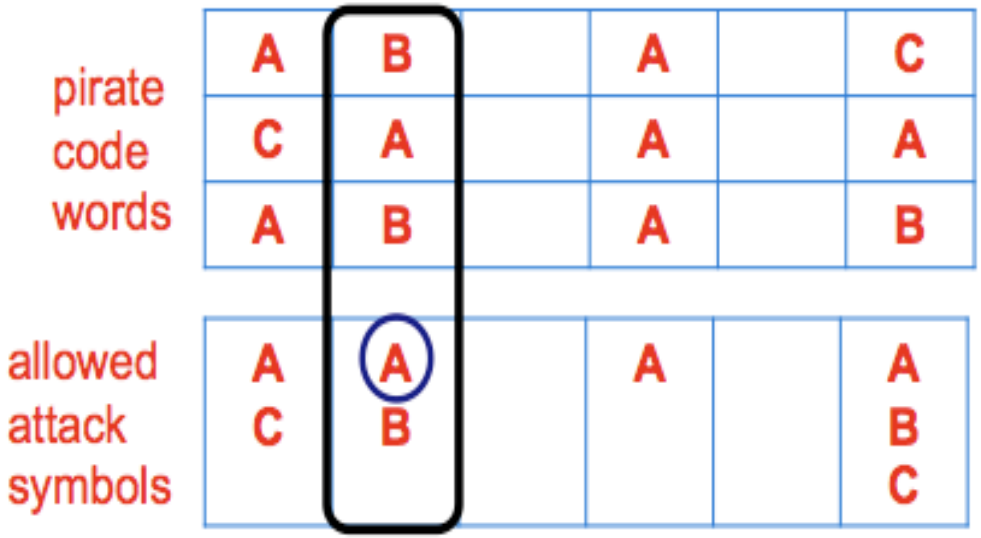
# Collusion channel (Restricted Digit Model)

"Tally" vector **m**:

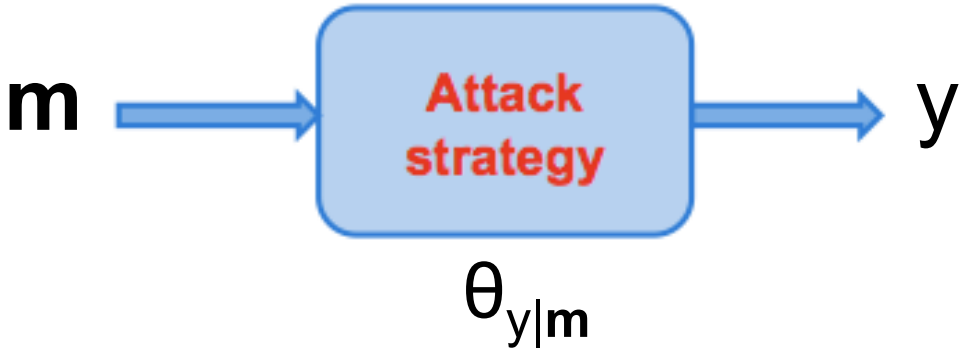
- #colluders = c
- $m_\alpha$= #α received by colluders
- |**m**|=c

**Attack**:

- same strategy in each position (asymptotically strongest)
- Choose y as a function of **m**: $\theta_{y|\mathbf{m}}$ = Prob[output y given **m**]




**m**=(1,2,0)

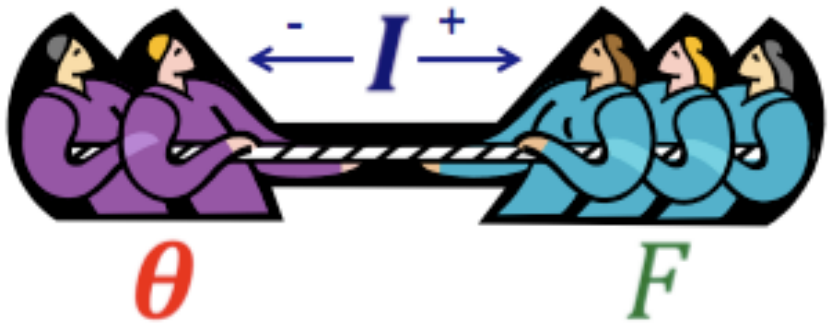


# Information theory: capacity

- Collusion attack can be seen as "malicious noise".
- Use techniques from channel coding!
  - How much does Y reveal about **M**? (**M** equivalent to colluder identities)
  - *Mutual information* I(**M**;Y)

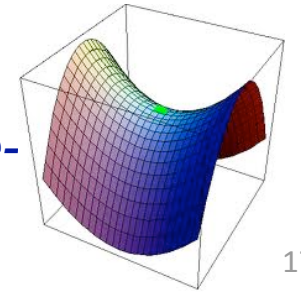
**Fingerprinting game**:

- Pay-off function I(**M**;Y|**P**) / c
- Tracer chooses bias distribution F(**p**)
- Colluders choose strategy θ




**Fingerprinting capacity** $$C = \frac{1}{c} \max_{F} \min_{\theta} I(\boldsymbol{M};Y \mid \boldsymbol{P})$$

*saddle-point*



# Fingerprinting Capacity

**Meaning of capacity C**:

- Max. achievable code rate
- Asymptotic error rate follows from C and R

DEF: $R = \frac{\log_q n}{\ell}$

n = #users
q = alphabet size
ℓ = code length

$$P_{\text{err}} \leq q^{-(C-R)\ell}$$

$$\ell_{\text{sufficient}} = \frac{1}{C \ \ln q} \ln \frac{n}{P_{\text{err}}}$$

# Asymptotic capacity and saddlepoint

Asymptotic: c → ∞

- [Huang & Moulin 2010]. Solution for binary alphabet.

$$C = \frac{1}{c^2\, 2\ln 2}$$

$$F(p_0, p_1) = \frac{1}{\pi} \frac{1}{\sqrt{p_0 p_1}}$$

*"arcsine distribution"*

$$\theta_{y|\boldsymbol{m}} = \frac{m_y}{c}$$

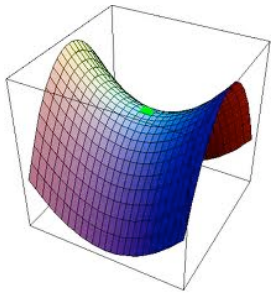
*interleaving attack (pick random attacker)*

- [Boesten & Skoric 2011]. **Capacity** for q-ary alphabet. ***Increases with q.***

$$C = \frac{q-1}{c^2\, 2\ln q}$$

- [Huang & Moulin 2012]. **Saddlepoint** for q-ary case.

$$F(\boldsymbol{p}) \propto \prod_{\alpha \in Q} p_\alpha^{-1/2}$$ vs. interleaving attack



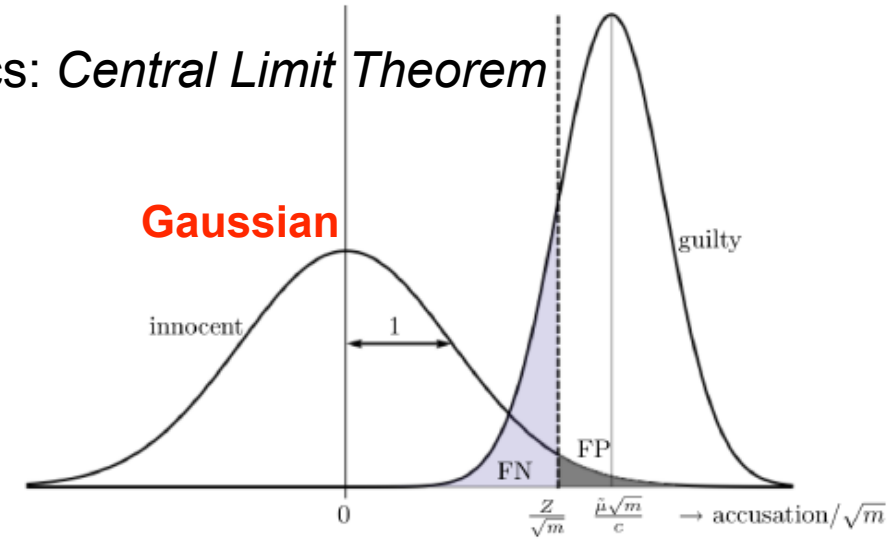
*Dirichlet distribution*

# Decoder

- In theory one can achieve sufficient code length $\ell_{\text{suff}} = \frac{2}{q-1} c^2 \ln \frac{n}{P_{\text{err}}}$
  - if it is known how to trace users!
- "Decoder" algorithm for finding colluders based on X,y,**p**.
  - **Simple** decoder: each user gets a score
  - **Joint** decoder: tuplets of users get a score
- [Tardos 2003, Skoric et al. 2007]: simple decoder
  - far away from capacity (at least factor 2.5)
- [Amiri & Tardos 2009]: Joint decoder for q=2
  - capacity-achieving but impractical
- [Huang & Moulin 2012]
  - "simple capacity = joint capacity" (asymptotically)
  - no recipe
- [Oosterwijk et al. 2013]
  - **simple decoder that achieves capacity**
  - **q-ary**

# Finding the optimal score function

Asymptotics: *Central Limit Theorem*




Optimize $\mu_{guilty}$ with constraints

- $\mu_{innocent}$ =0
- $\sigma_{innocent}$ =1

**Euler-Lagrange optimization, with F and θ *in saddlepoint***

$$L[h,\lambda_1,\lambda_2] = \mu_{\text{guilty}}[h] - \lambda_1\mu_{\text{inn}}[h] - \frac{1}{2}\lambda_2(\sigma^2_{\text{inn}}[h] - 1)$$

score function h(x,y,**p**)

x = symbol of user under scrutiny

# Optimal score function

$$h(x,y,\boldsymbol{p}) = \frac{1}{\sqrt{q-1}}\left(\frac{\delta_{xy}}{p_y} - 1\right)$$

- "strongly centered": $E_x[h(x,y,\boldsymbol{p})] = \sum_x p_x h(x,y,\boldsymbol{p}) = 0$
- If attack=interleaving, then <u>for all F</u>:

$$E_{xy|\boldsymbol{p}}[h_{\text{inn}}^2] = \frac{1}{q-1} E_{y|\boldsymbol{p}}[1/p_y - 1] = 1 \quad \textit{(normalized)}$$

$$\mu_{\text{coalition}} = \sqrt{q-1}$$

✓ Consistency check: indeed a code-rate saddlepoint

  - "ridge" at θ=interleaving

- $\sigma_{\text{inn}}$ depends on attack strategy

# Optimal score function (2)

"Tardos score": $$g(x,y,\boldsymbol{p}) = \begin{cases} \sqrt{\dfrac{1-p_y}{p_y}} & \text{if } x = y \\ -\sqrt{\dfrac{p_y}{1-p_y}} & \text{if } x \neq y \end{cases} = \sqrt{\frac{p_y}{1-p_y}}\left[\frac{\delta_{xy}}{p_y} - 1\right]$$

Tardos' score is "strongly normalized" version of optimal score!

- Guaranteed to have $E[g^2]=1$ (innocent) for *any* attack and *any* **p**.
- Demanding strong normalization over-constrains the problem.

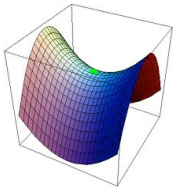
# Game over?


Did we kill the field?

NO!

**<u>Still to be done</u>**:

- Validation
  - simulations, provable bounds, etc.
- *Dynamic* traitor tracing
  - is the capacity the same?
  - different conditions, different solutions?
- Finite c; not just asymptotics
  - find the saddlepoint
  - joint decoder required?



- More realistic attack models
  - Combined Digit Model