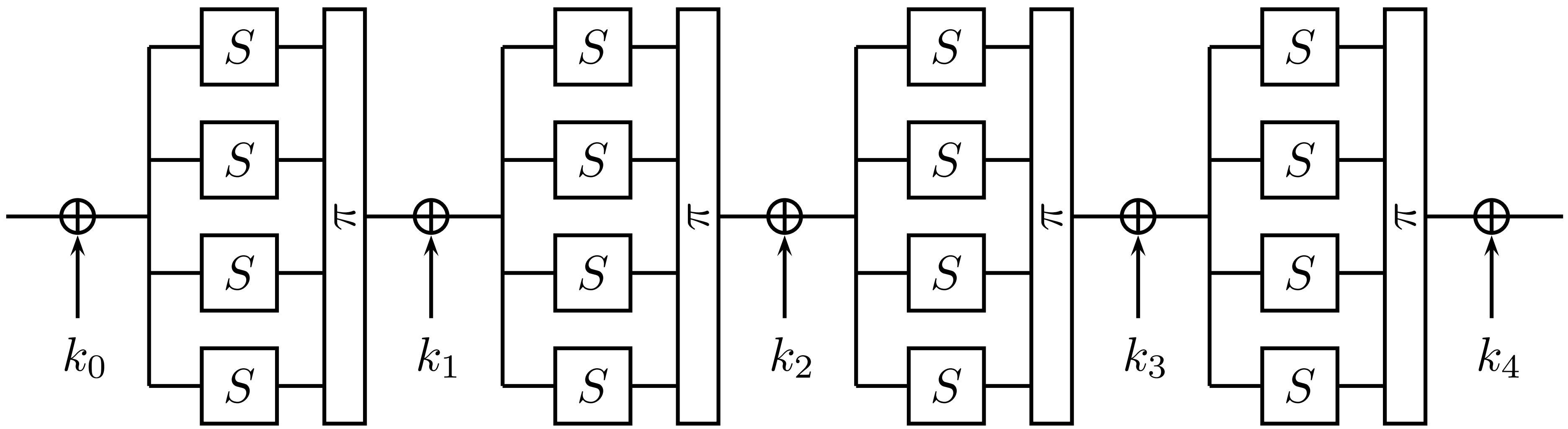
Indifferentiability of Confusion-Diffusion Networks

Yevgeniy Dodis (NYU), Martijn Stam (Bristol),
John Steinberger (Tsinghua), Tianren Liu (MIT)

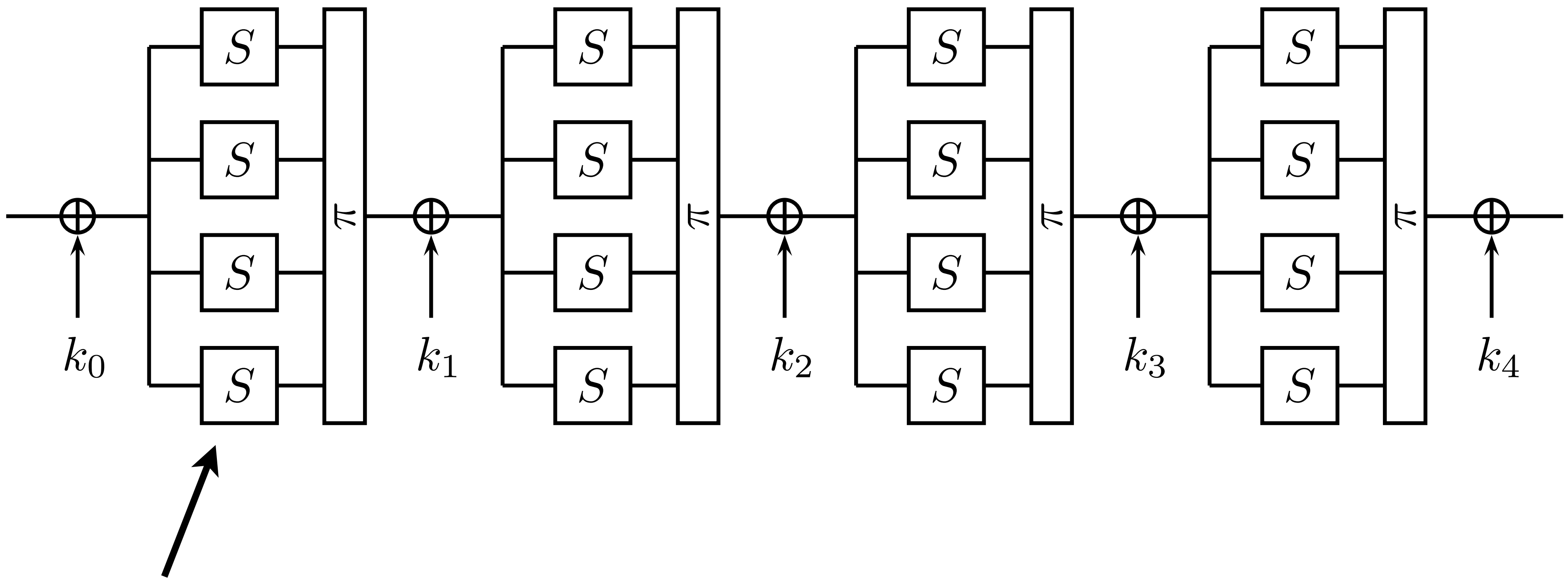
Indifferentiability of Confusion-Diffusion Networks

Yevgeniy Dodis (NYU), Martijn Stam (Bristol),
John Steinberger (Tsinghua), Liu Tianren (MIT)

Substitution-Permutation Network (ex: AES):

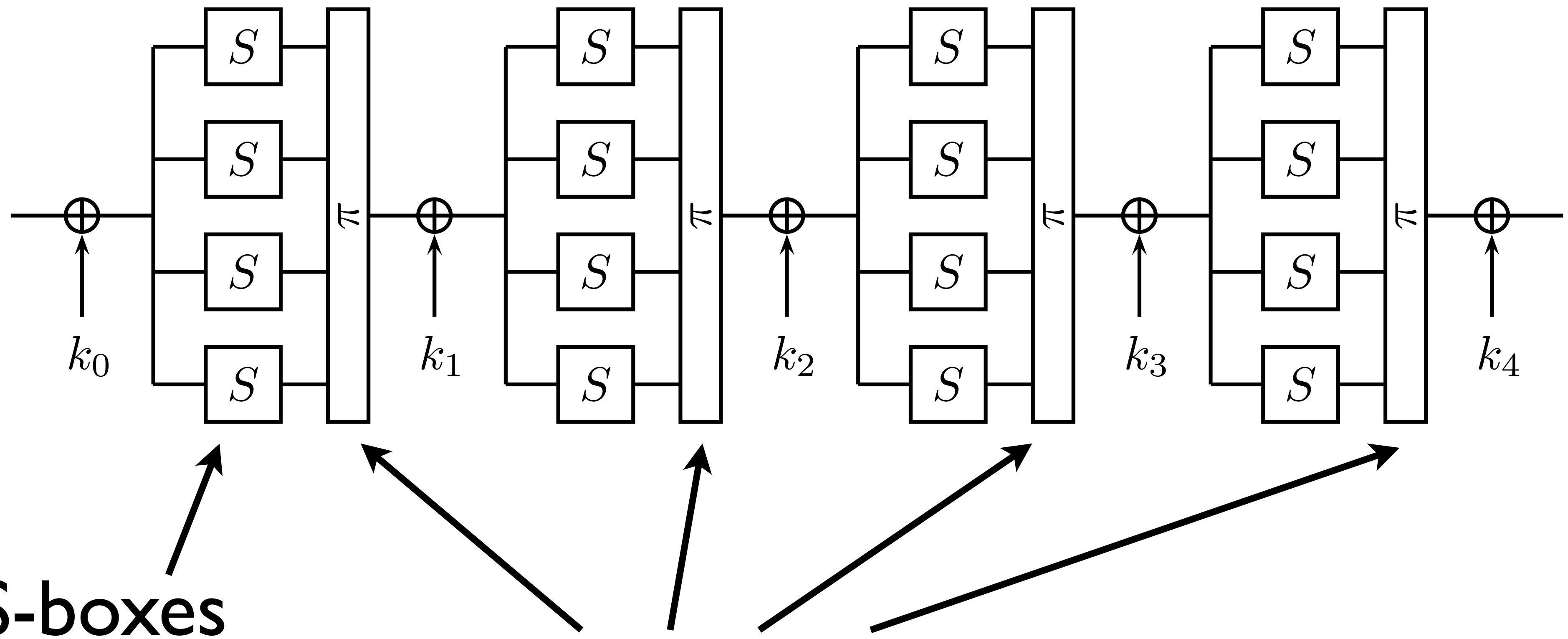


Substitution-Permutation Network (ex: AES):



S-boxes
(small input size)
(permutations!)

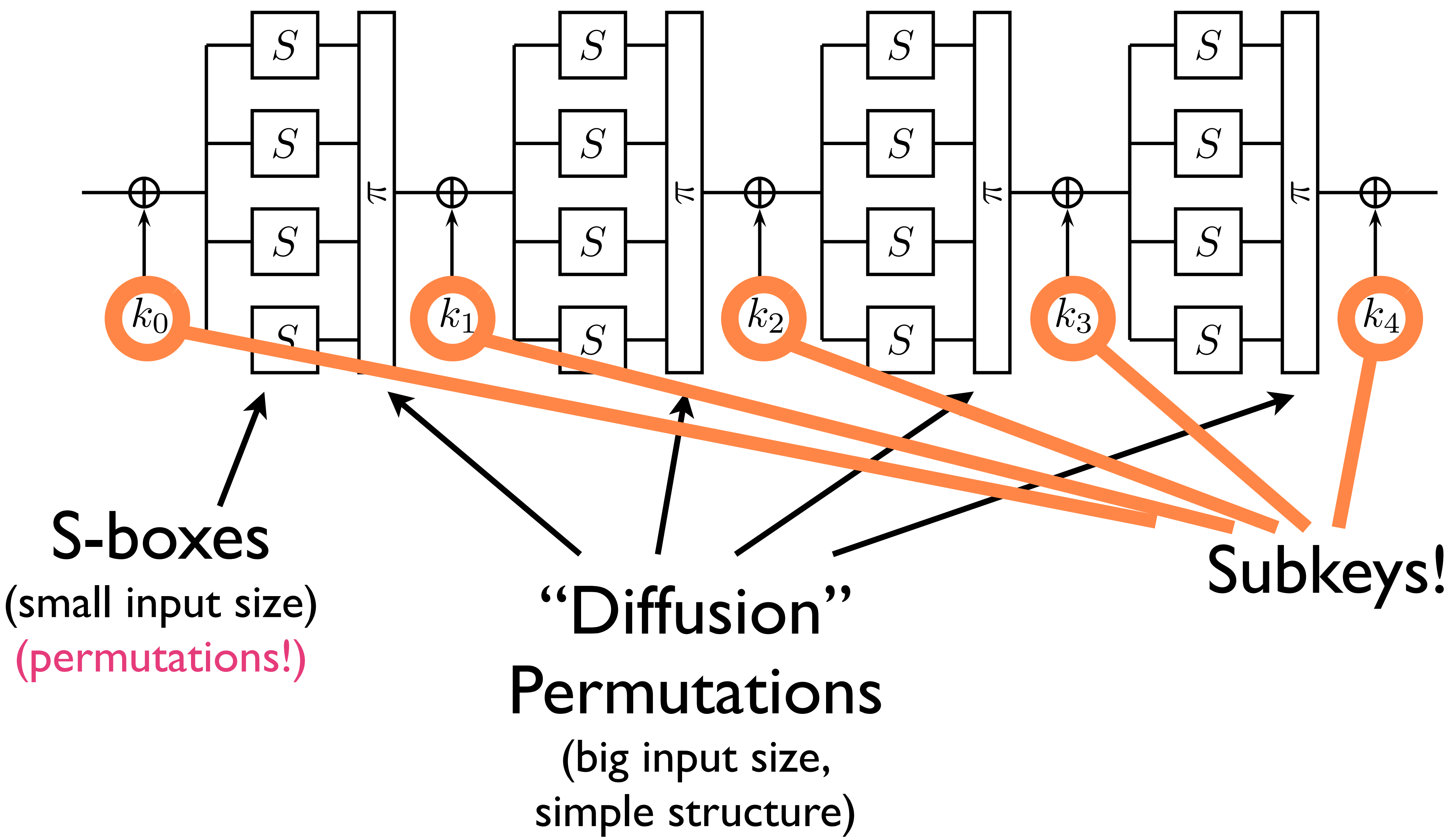
Substitution-Permutation Network (ex: AES):



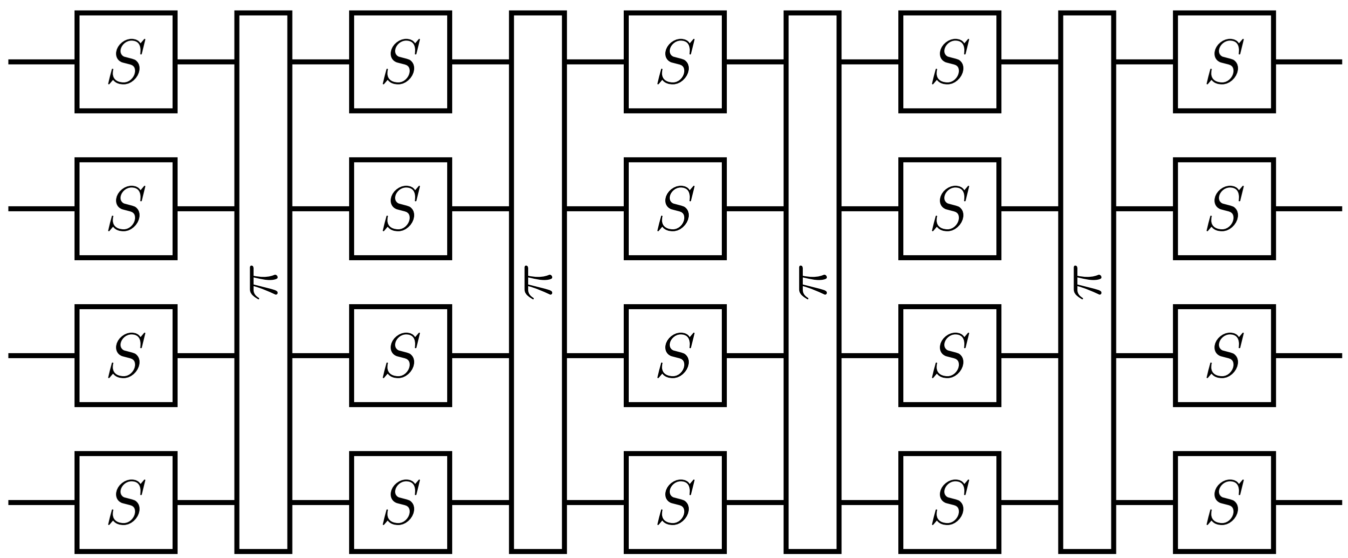
S-boxes
(small input size)
(permutations!)

"Diffusion" Permutations
(big input size,
simple structure)

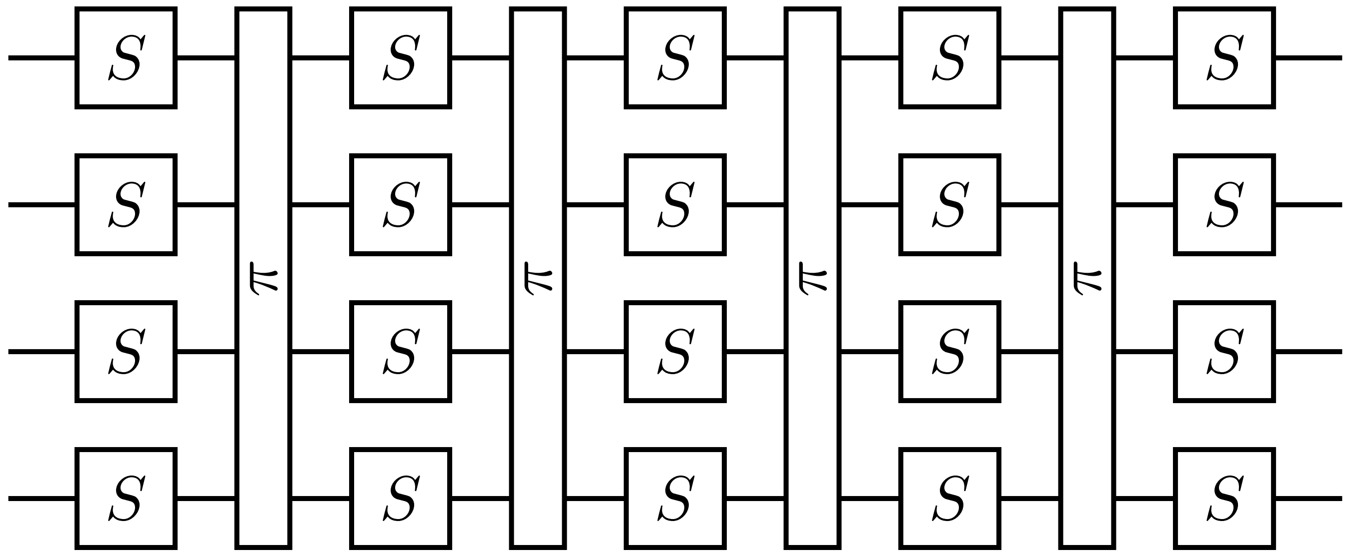
Substitution-Permutation Network (ex: AES):



Confusion-Diffusion Network:

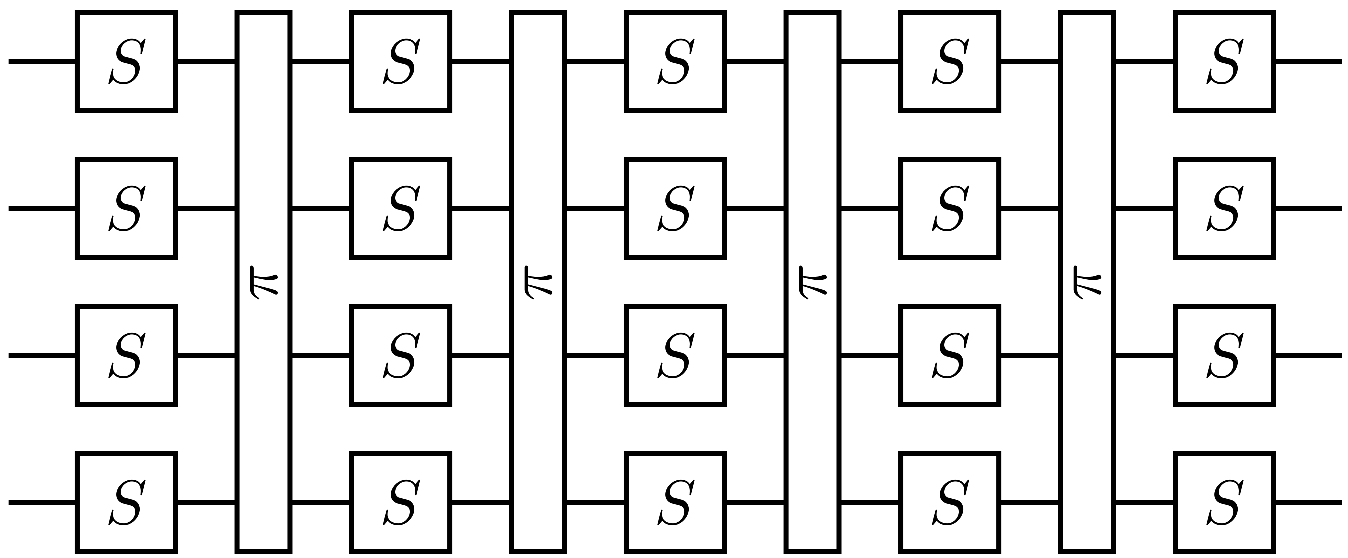


Confusion-Diffusion Network:



...same, but no keys!

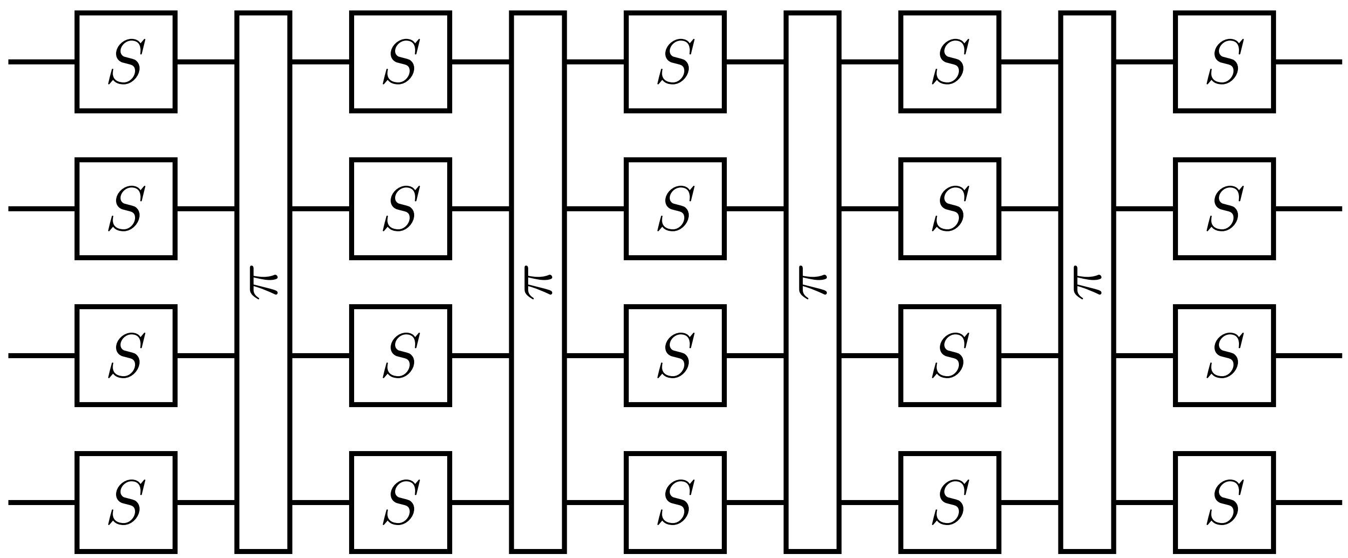
Confusion-Diffusion Network:



...same, but no keys!

- can be seen as a domain extension mechanism for permutations (from S-box to “full” permutation)

Confusion-Diffusion Network:



...same, but no keys!

- can be seen as a domain extension mechanism for permutations (from S-box to "full" permutation)
- terminology goes back to Shannon (1949), but the design paradigm seems to be Feistel's (1970)

This work's high-level goals

- Investigate the theoretical soundness of CD (confusion-diffusion!) networks as a design paradigm for cryptographic permutations

This work's high-level goals

- Investigate the theoretical soundness of CD (confusion-diffusion!) networks as a design paradigm for cryptographic permutations
- Fundamental question: (efficient) domain extension of a public random permutation

This work's high-level goals

- Investigate the theoretical soundness of CD (confusion-diffusion!) networks as a design paradigm for cryptographic permutations
- Fundamental question: (efficient) domain extension of a public random permutation
- Work in an ideal model (S-boxes are independent random permutations, D-boxes are fixed, explicit permutations)

This work's high-level goals

- Investigate the theoretical soundness of CD (confusion-diffusion!) networks as a design paradigm for cryptographic permutations
- Fundamental question: (efficient) domain extension of a public random permutation
- Work in an ideal model (S-boxes are independent random permutations, D-boxes are fixed, explicit permutations)
- Does the network "emulate" a random permutation? How many rounds are necessary, and what kinds of D-boxes do we need??

vaguely related work

- Miles & Viola prove an indistinguishability result for SPN networks where the S-boxes are secret (part of the key) and one-way (so not really an SPN network after all)

vaguely related work

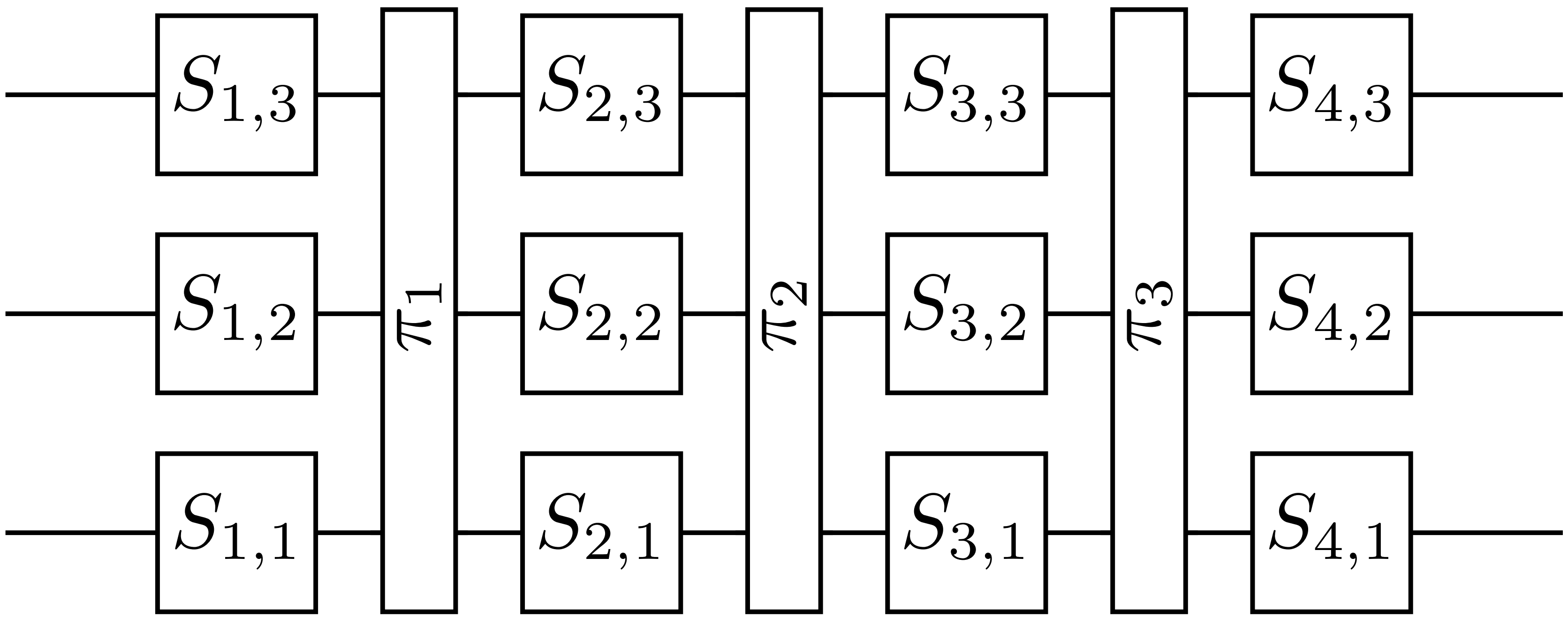
CD

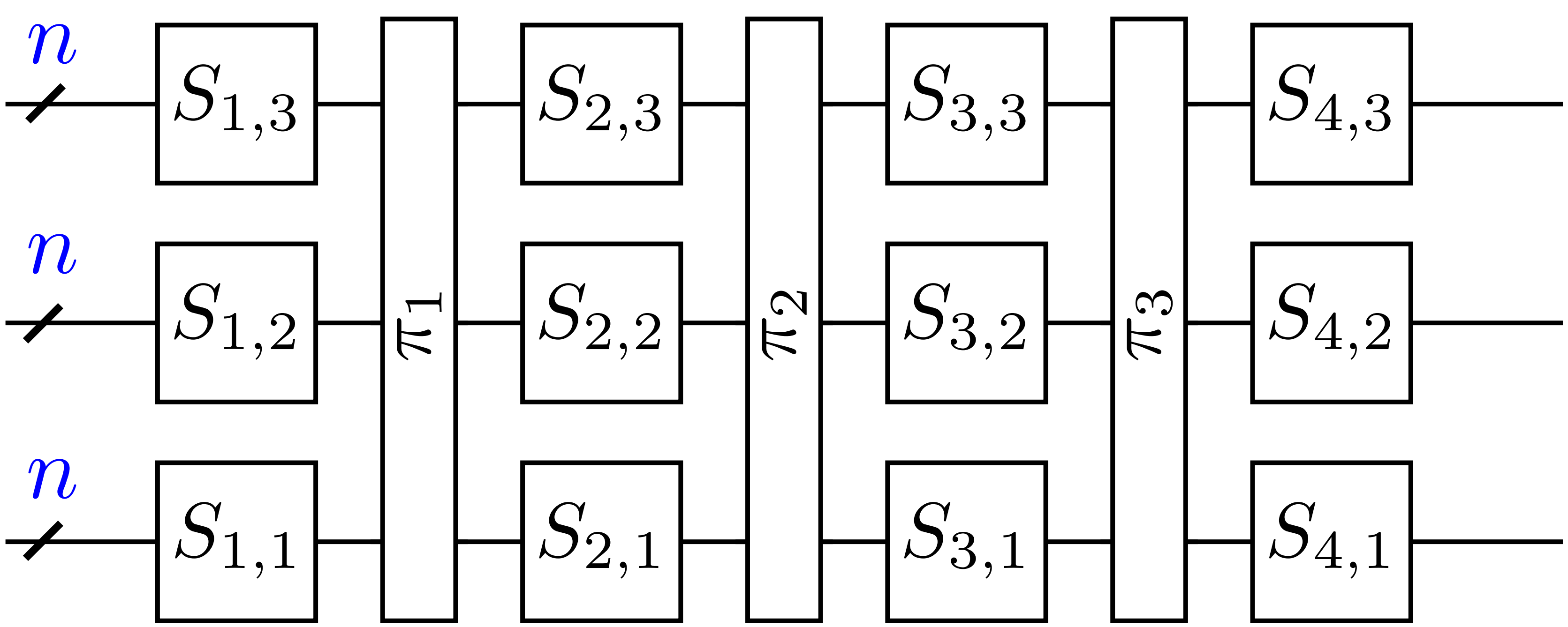
indifferentiability

- Miles & Viola prove an ~~indistinguishability~~ result for ~~SPN~~ networks where the S-boxes are ~~secret (part of the key~~) and ~~one-way~~ (so not really an SPN network after all)

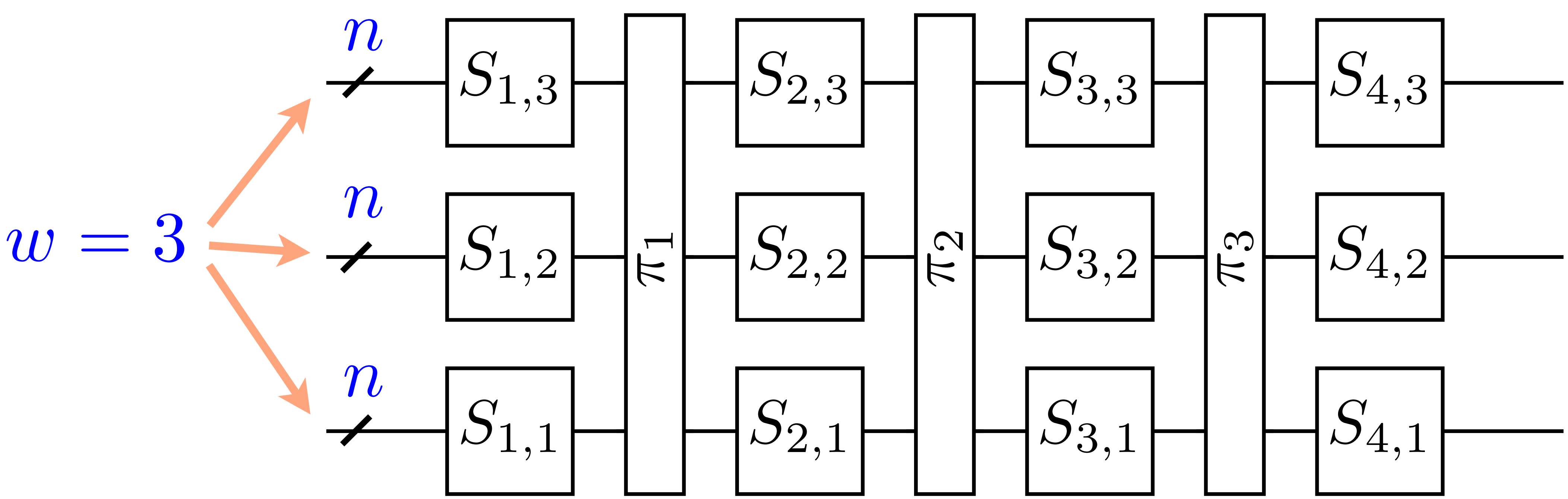
public

two-way


$S_{1,3}$
$S_{2,3}$
$S_{3,3}$
$S_{4,3}$
$S_{1,2}$
π_1
$S_{2,2}$
π_2
$S_{3,2}$
π_3
$S_{4,2}$
$S_{1,1}$
$S_{2,1}$
$S_{3,1}$
$S_{4,1}$

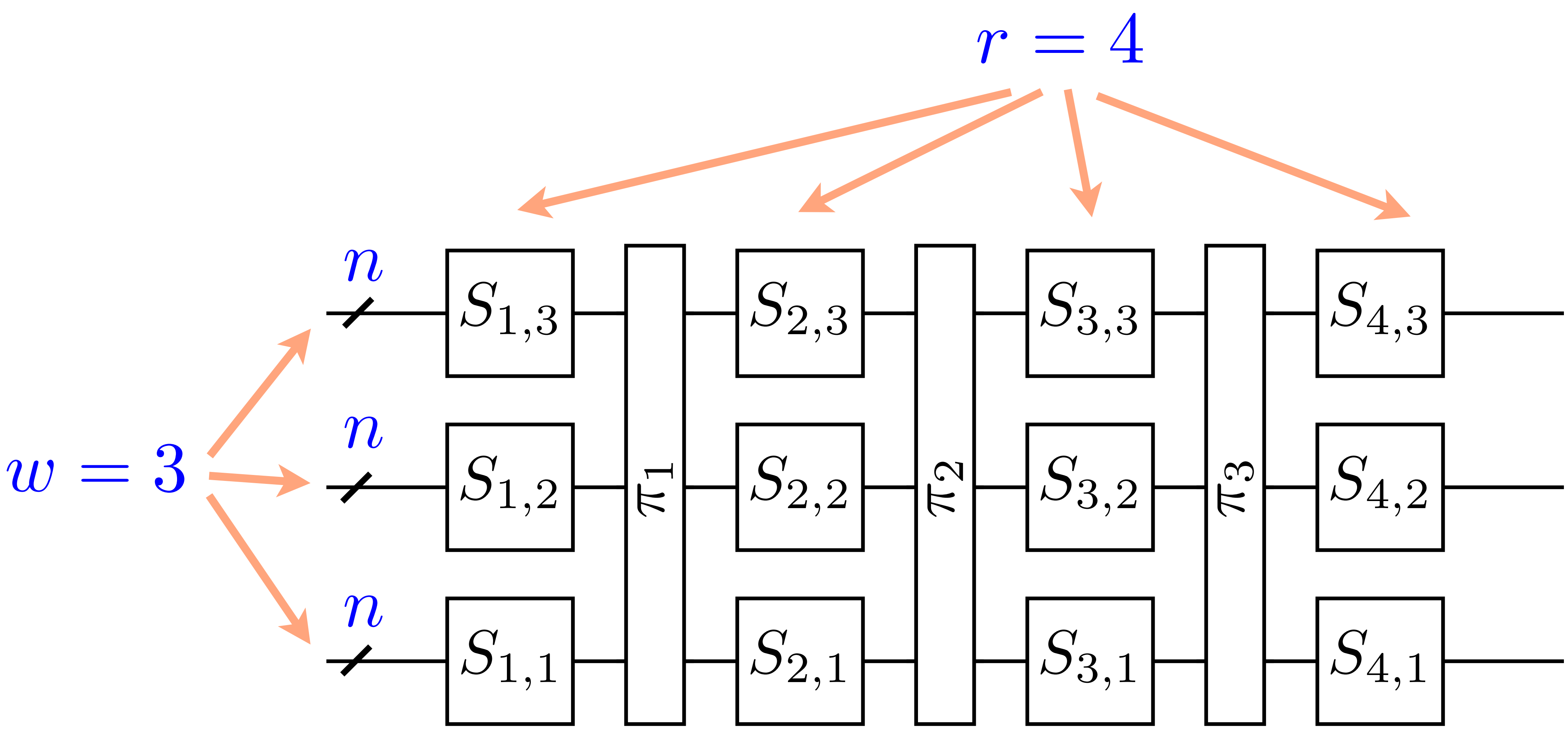


$n =$ wire length



$n =$ wire length

$w =$ "width" (no. S-boxes per round)

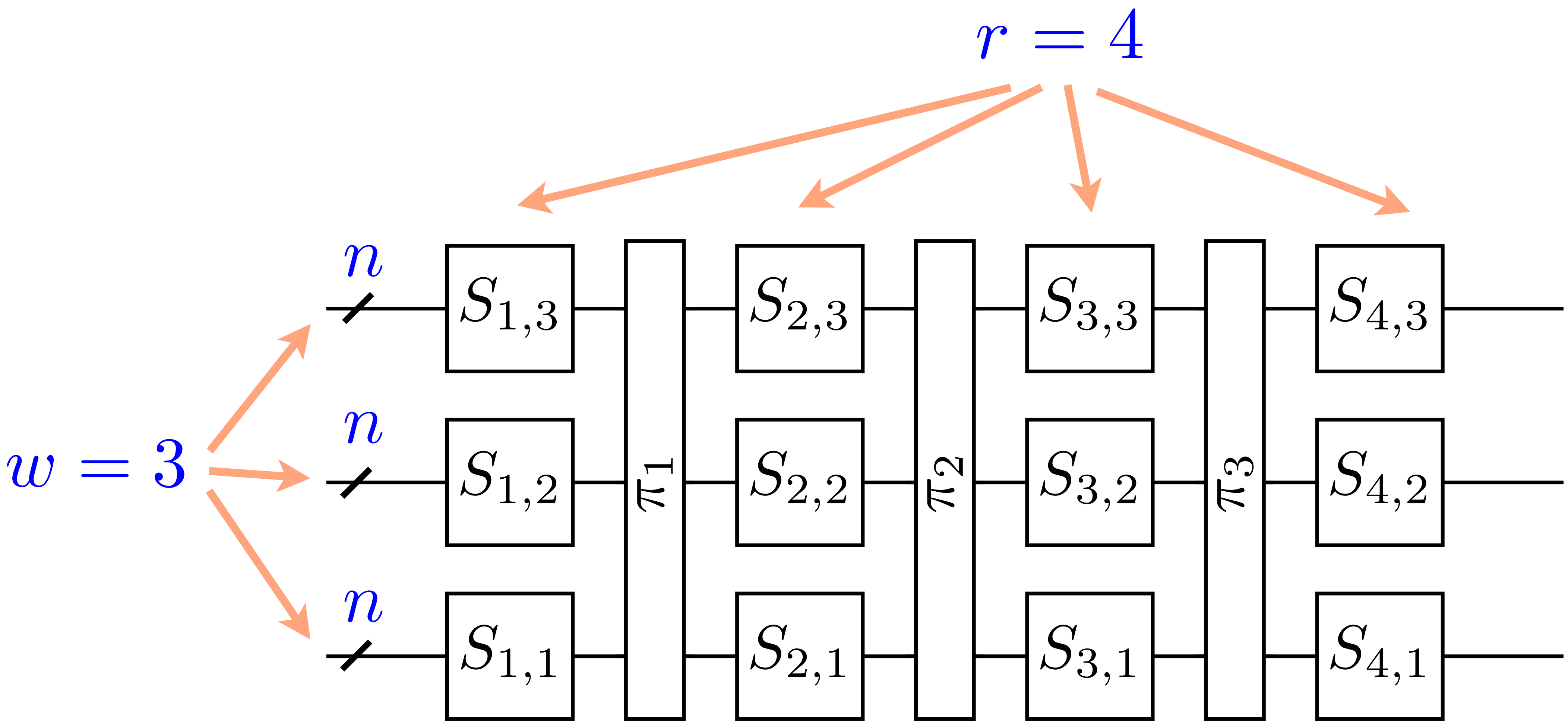


$n =$ wire length

$w =$ "width" (no. S-boxes per round)

$r =$ number of rounds

$\{0,1\}^{wn} =$ domain of CD network



$n =$ wire length

$w =$ "width" (no. S-boxes per round)

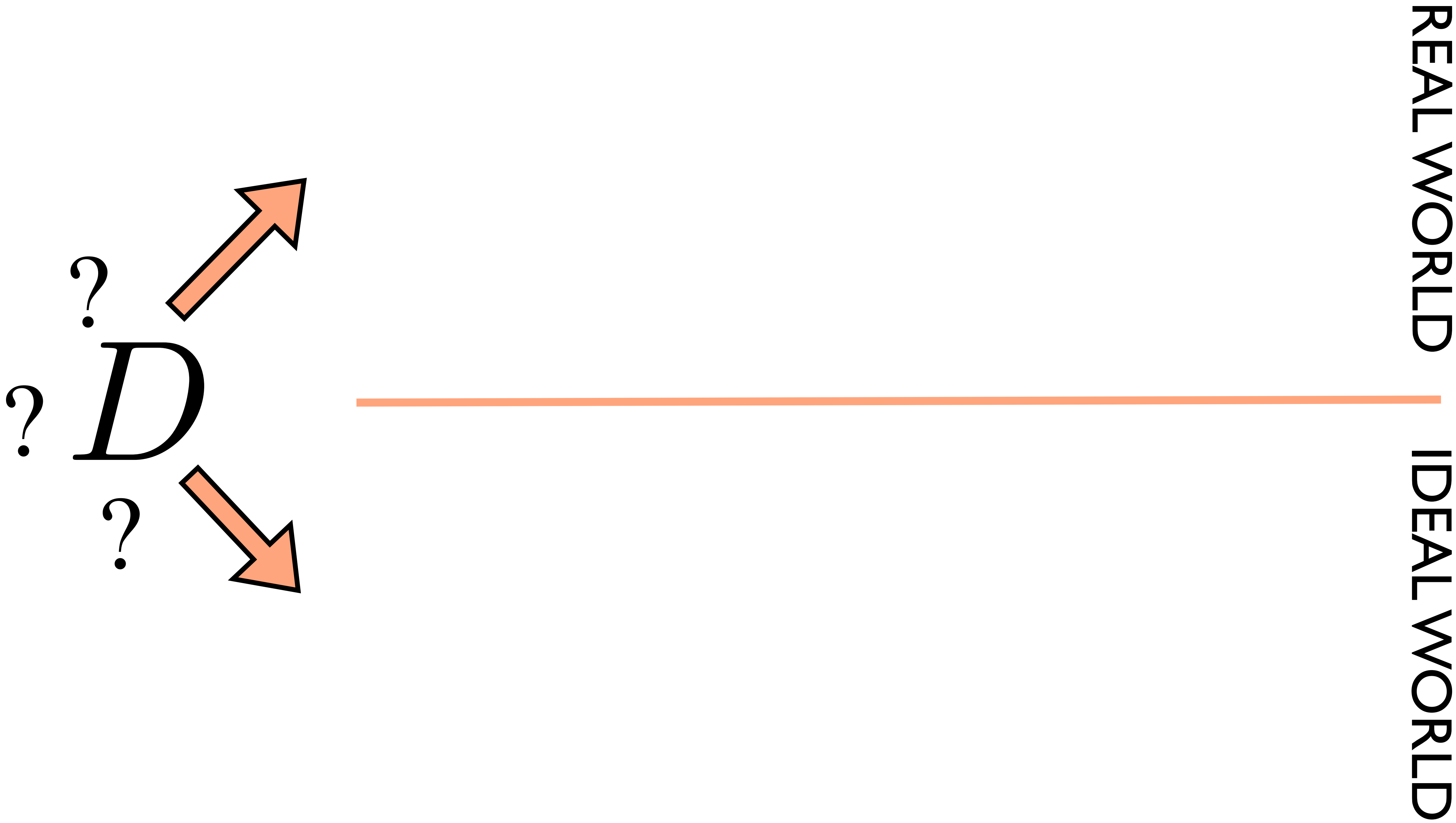
$r =$ number of rounds

Security Model

(indifferentiability)

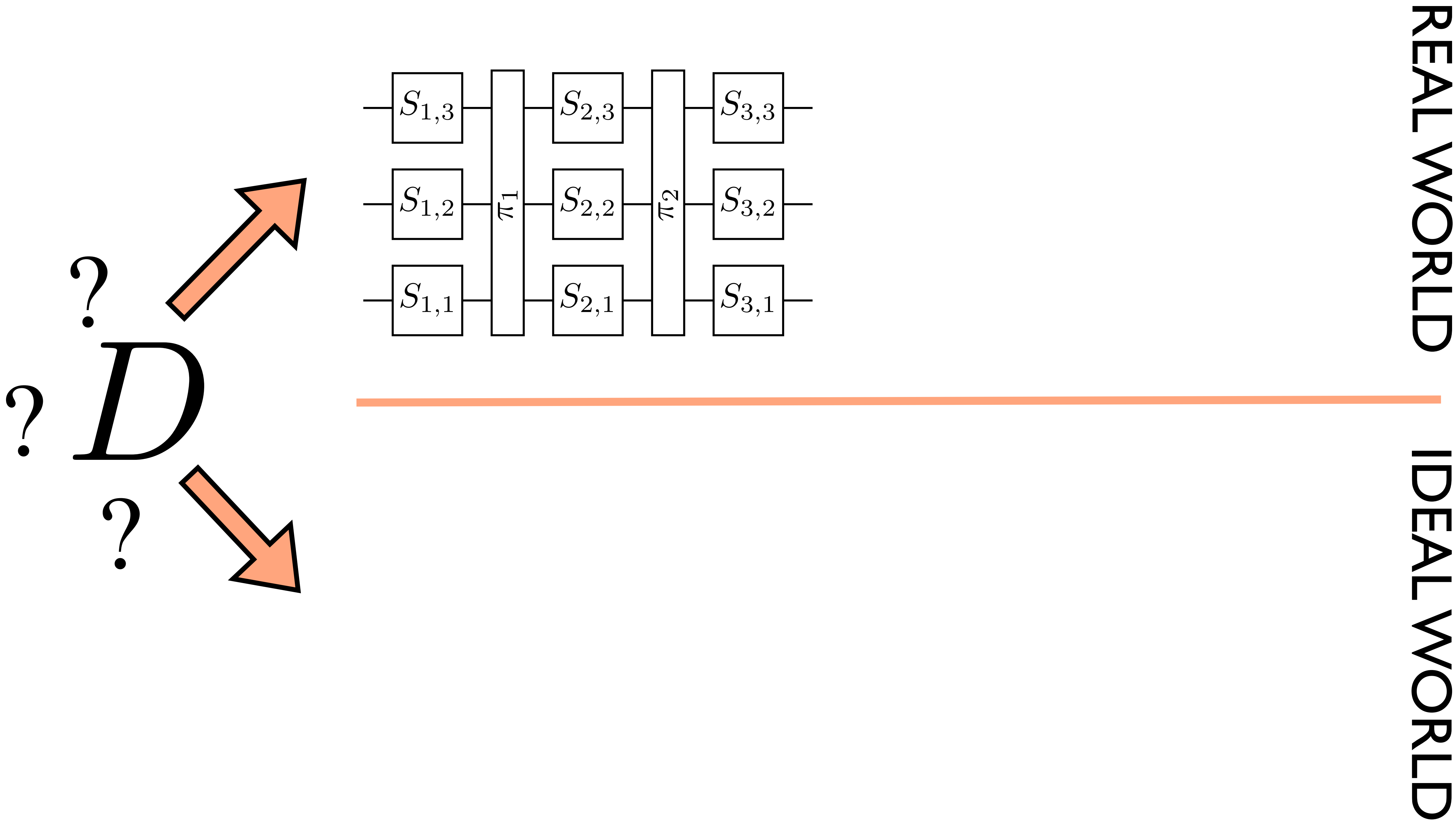
Security Model

(indifferentiability)



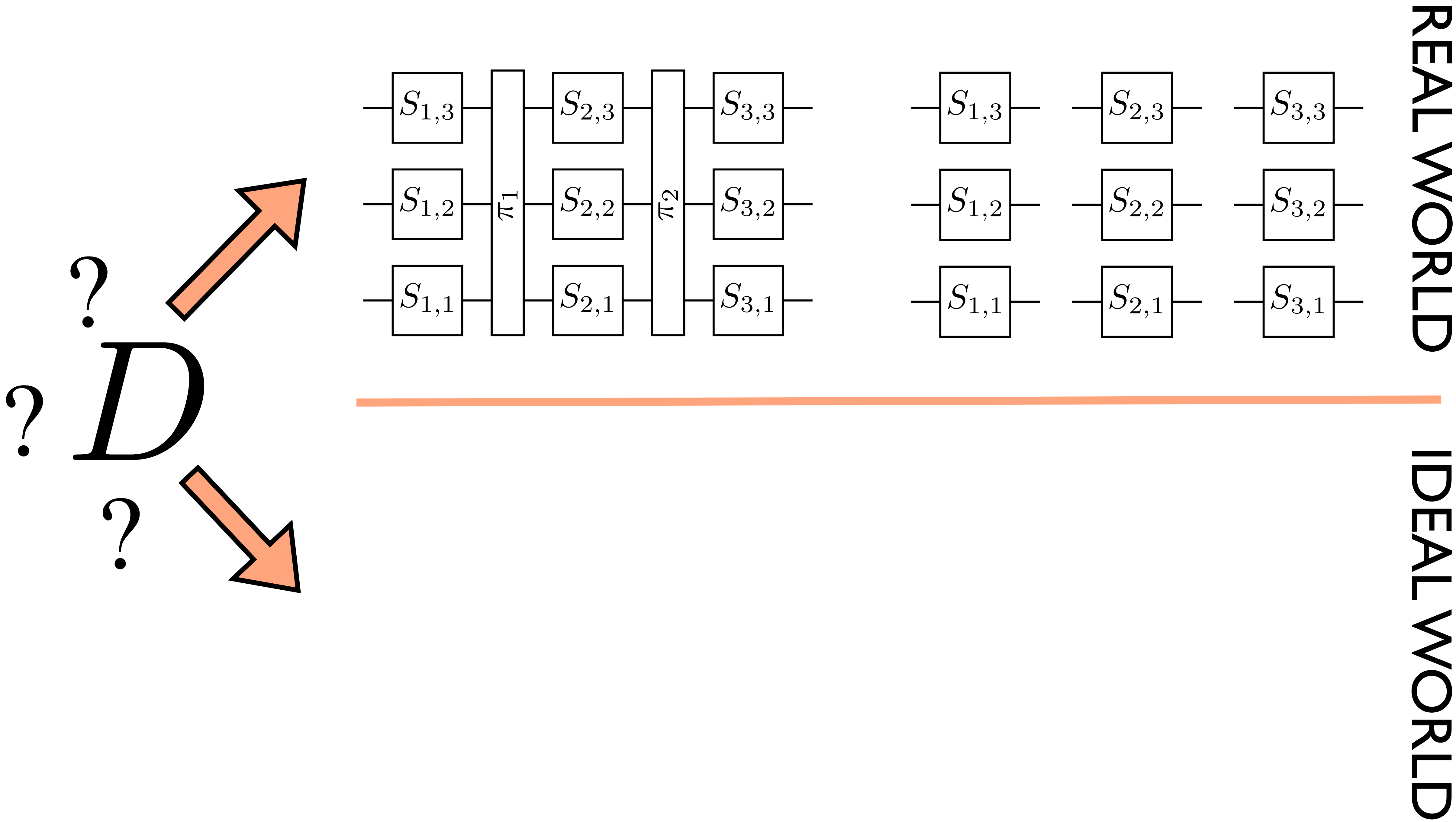
Security Model

(indifferentiability)



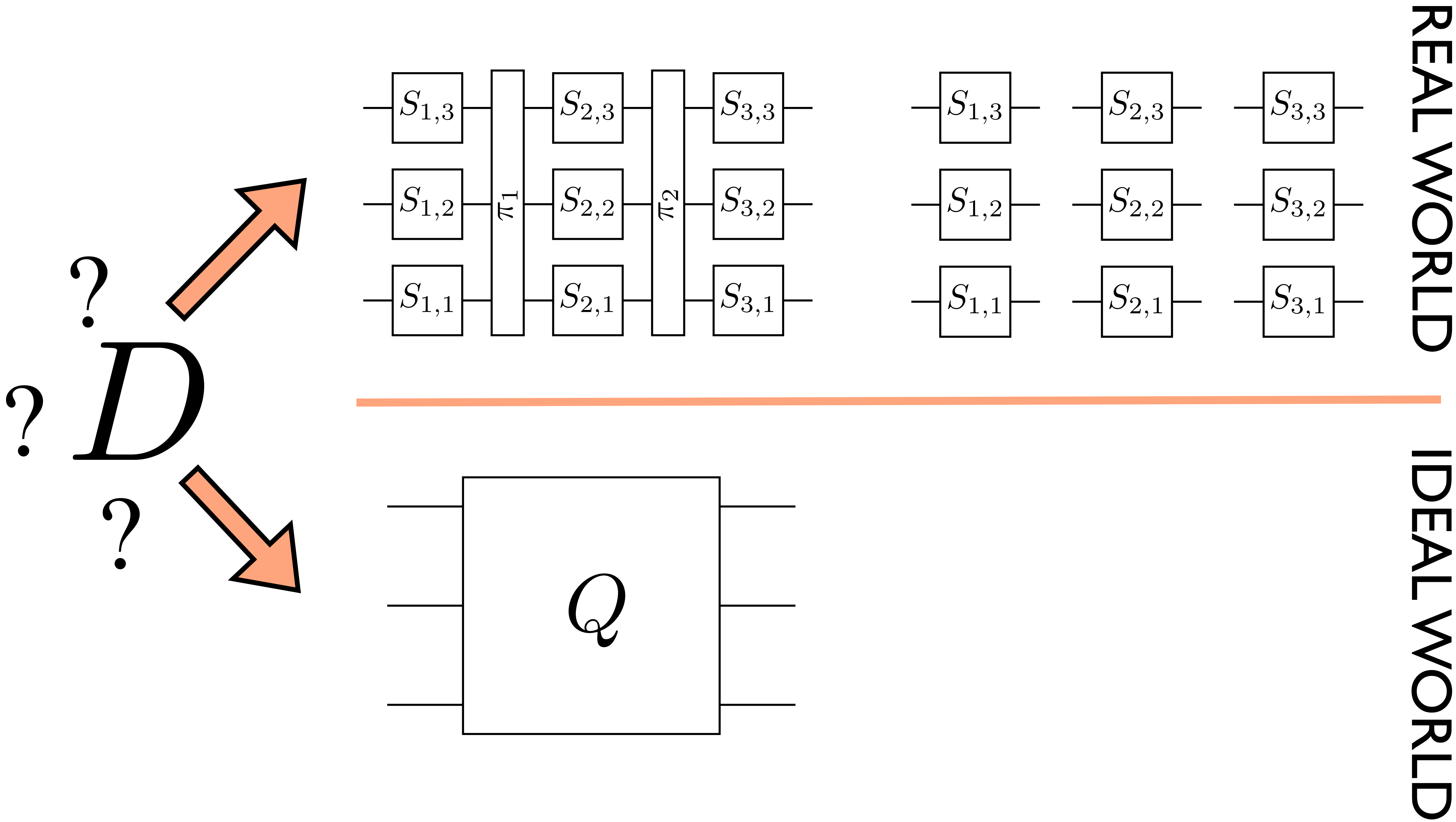
Security Model

(indifferentiability)



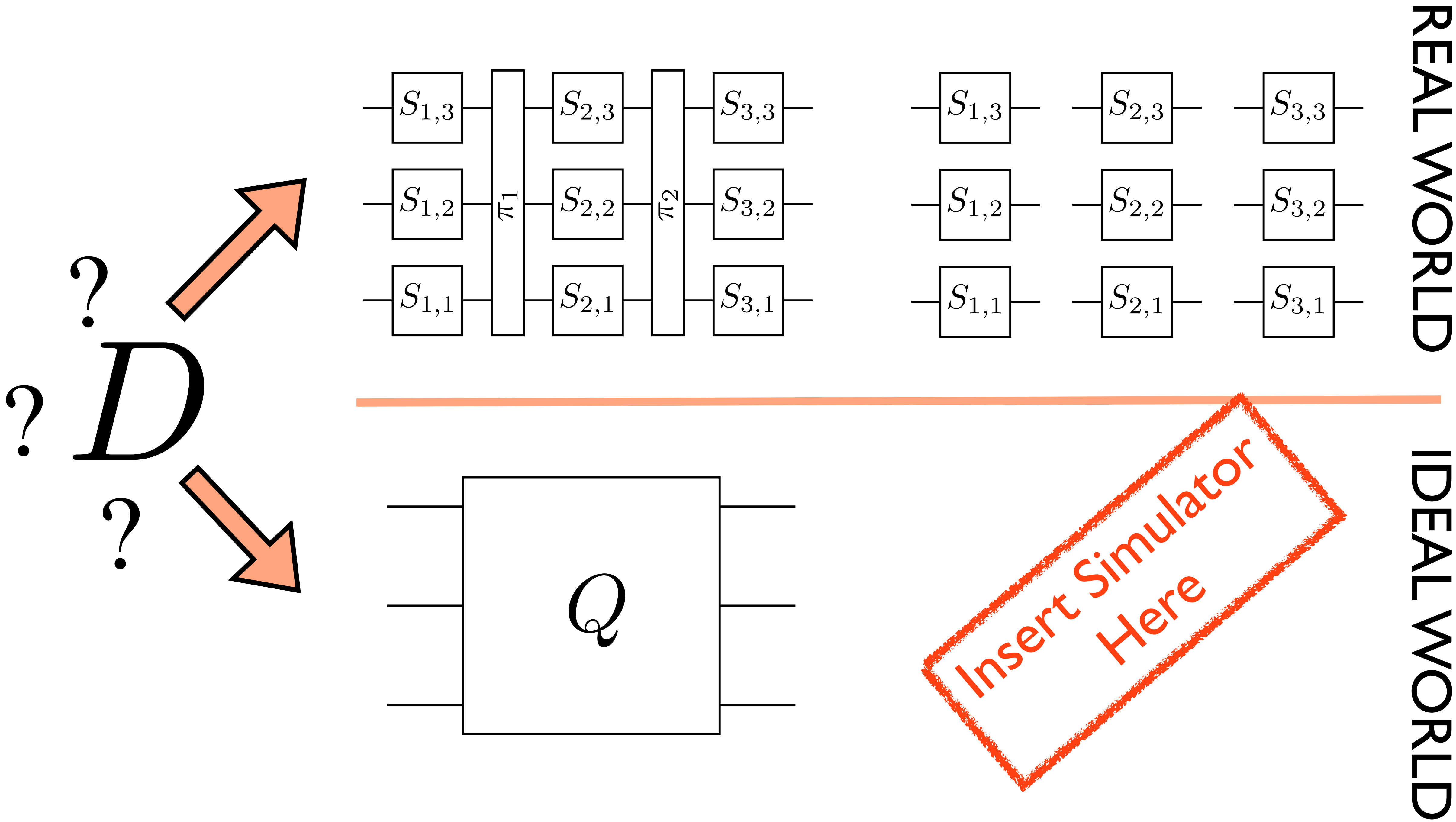
Security Model

(indifferentiability)



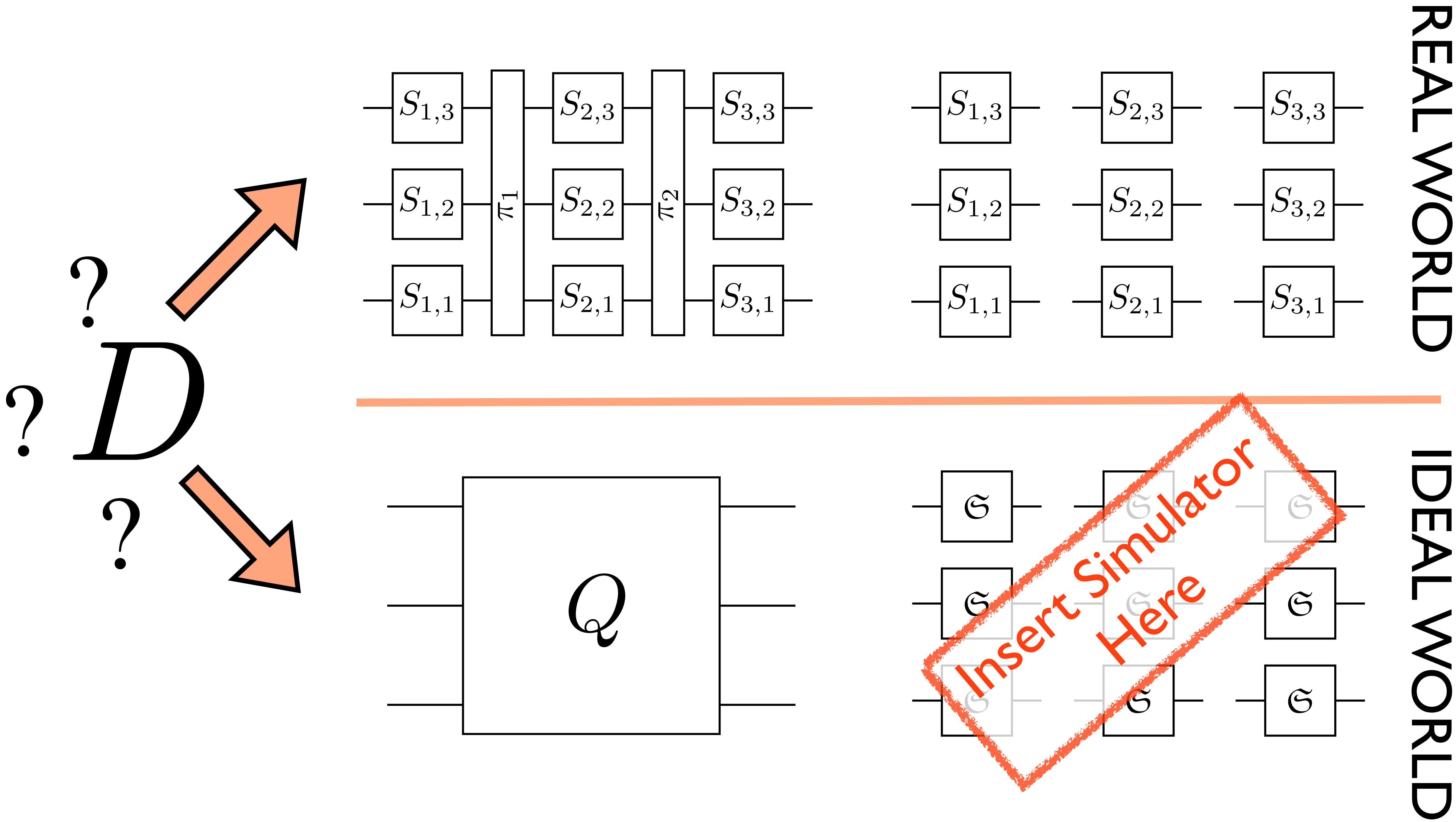
Security Model

(indifferentiability)



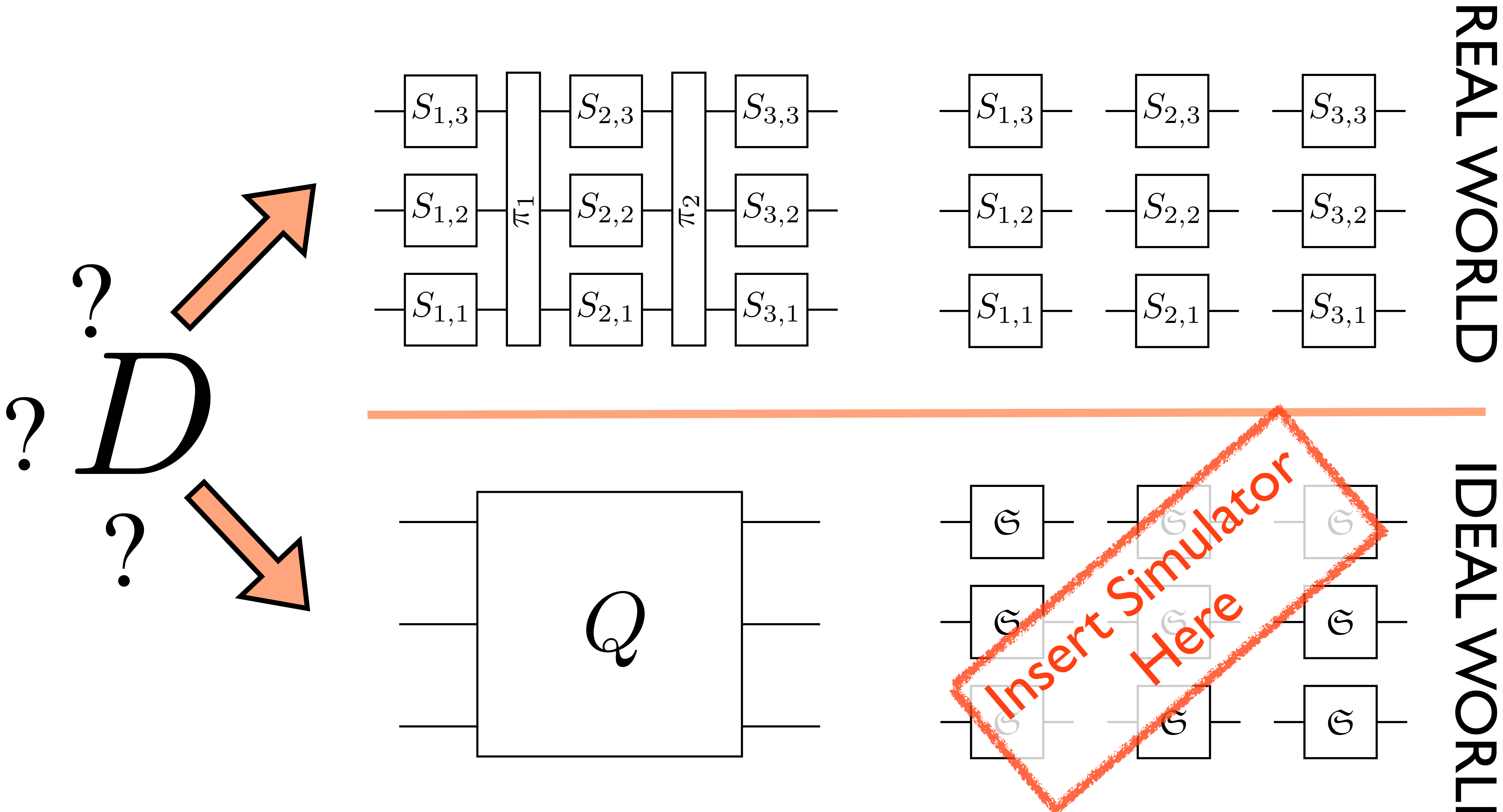
Security Model

(indifferentiability)



Security Model

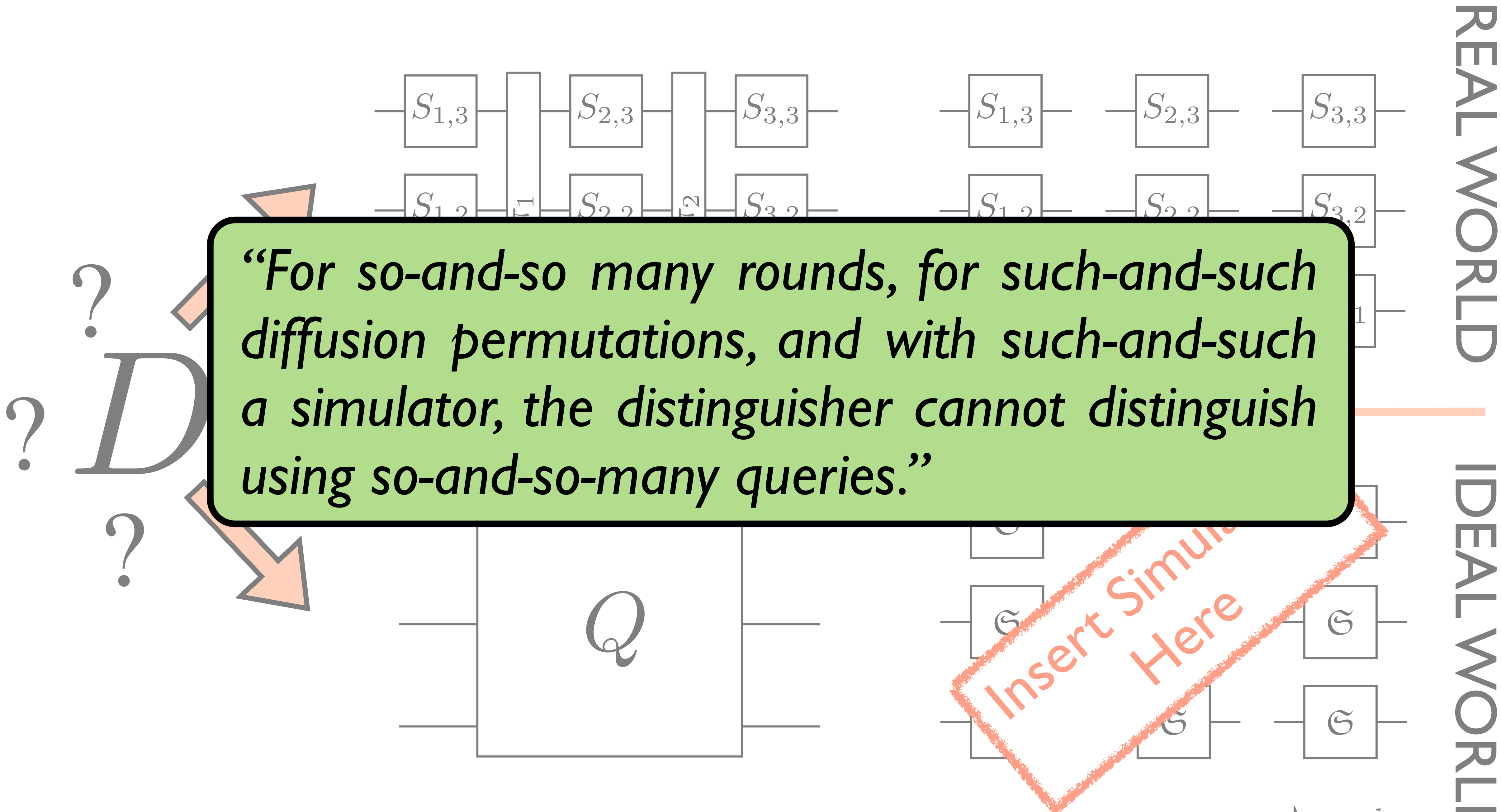
(indifferentiability)



Goal: By using oracle access to Q the simulator $\mathfrak{S}$ has to make up answers that look "consistent" with Q

Security Model

(indifferentiability)

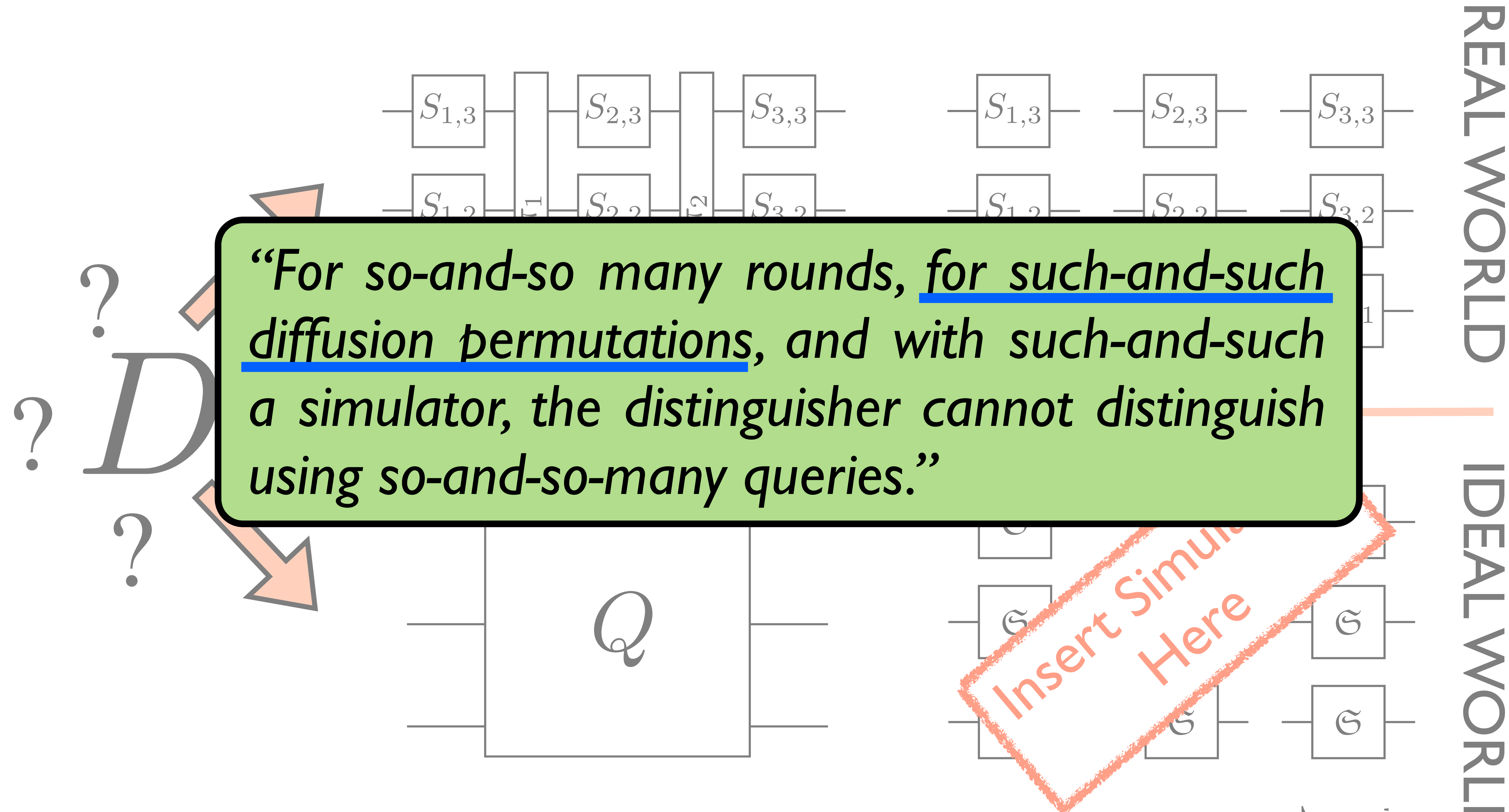


"For so-and-so many rounds, for such-and-such diffusion permutations, and with such-and-such a simulator, the distinguisher cannot distinguish using so-and-so-many queries."

Goal: By using oracle access to Q the simulator has to make up answers that look "consistent" with Q

Security Model

(indifferentiability)



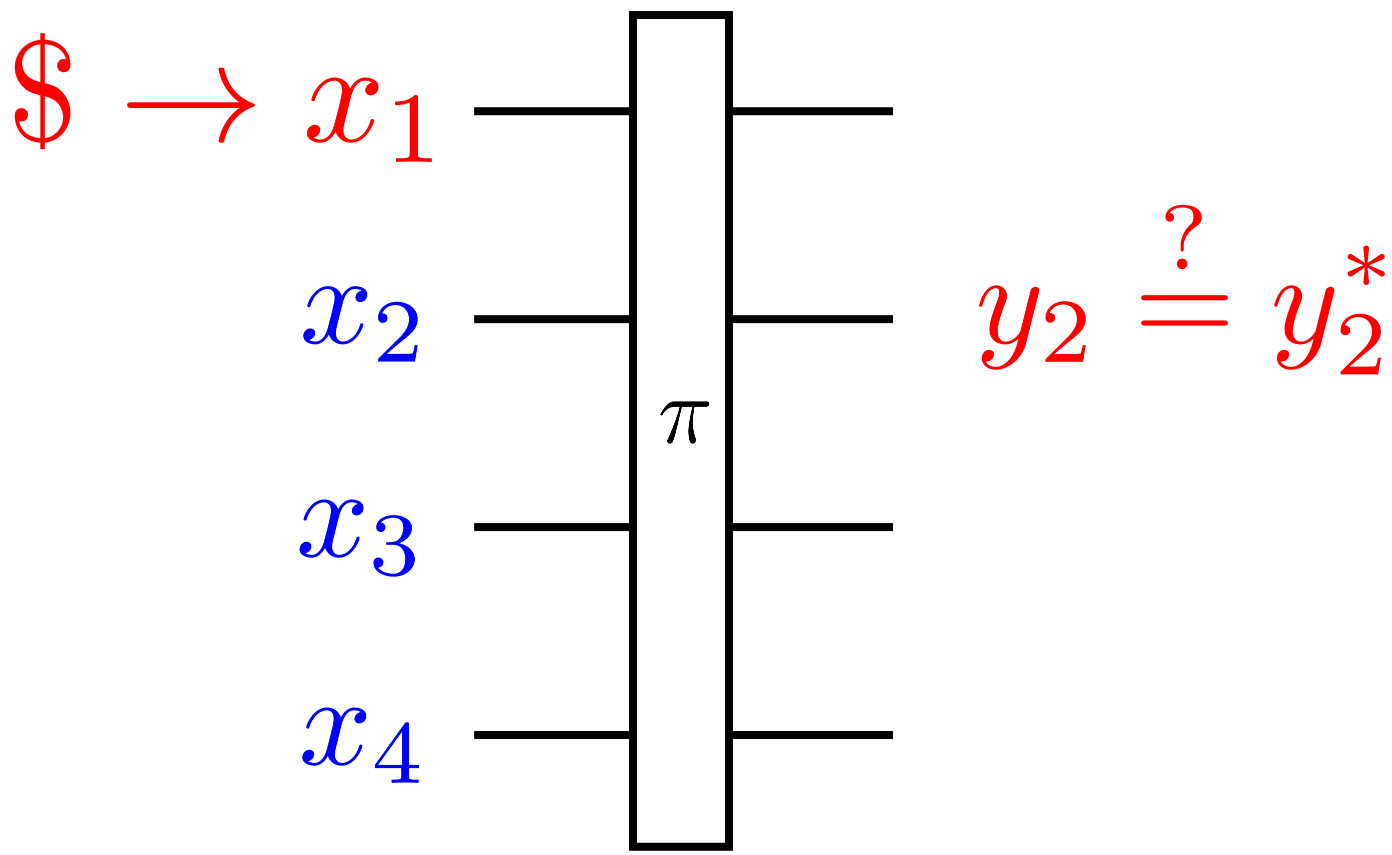
"For so-and-so many rounds, for such-and-such diffusion permutations, and with such-and-such a simulator, the distinguisher cannot distinguish using so-and-so-many queries."

Goal: By using oracle access to Q the simulator has to make up answers that look "consistent" with Q

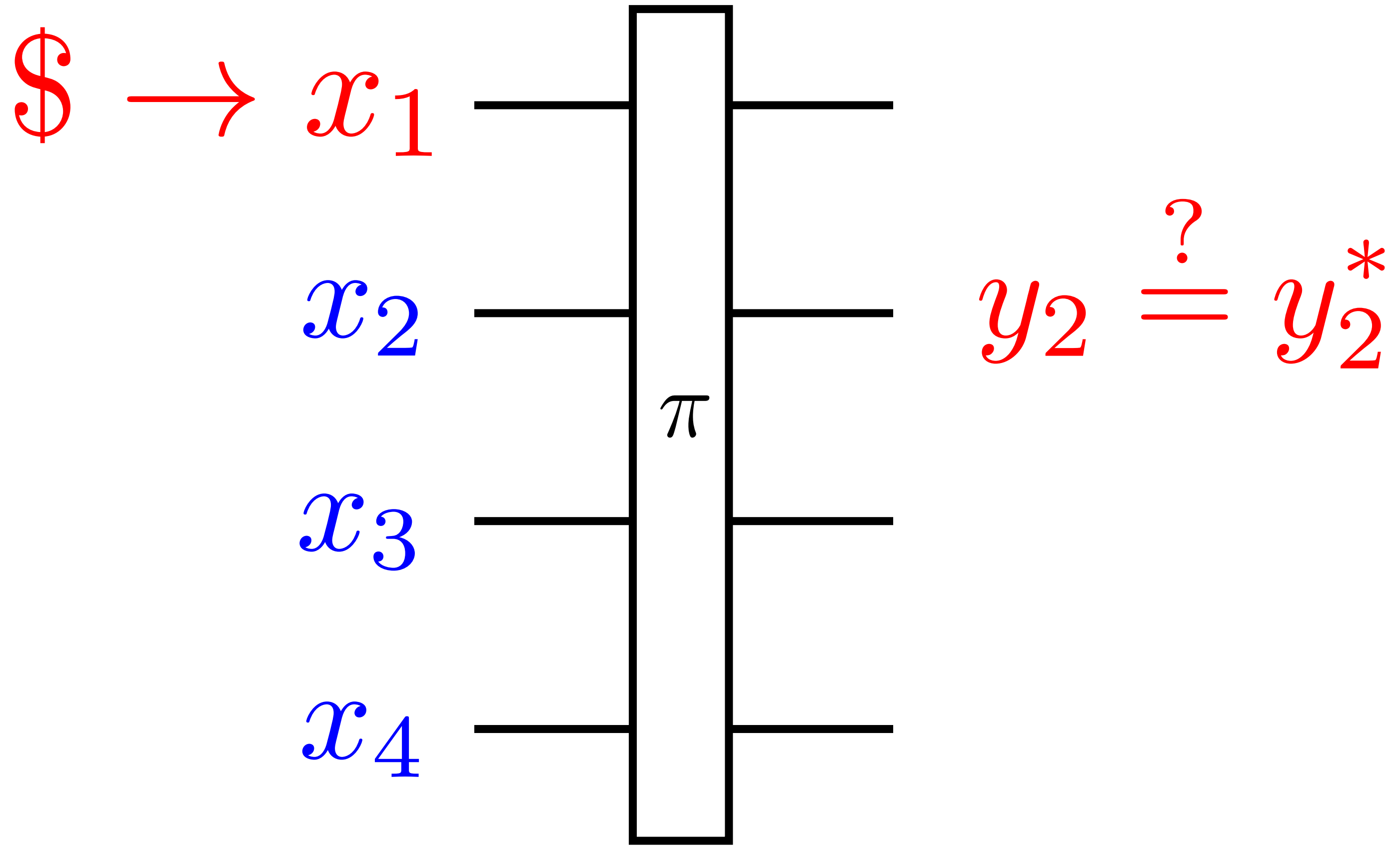
Combinatorial Properties of the Diffusion Permutations, by name:

1. Entry-Wise Randomized Preimage Resistance (RPR)
2. Entry-Wise Randomized Collision Resistance (RCR)
3. Conductance (& “all-but-one Conductance”)

RPR

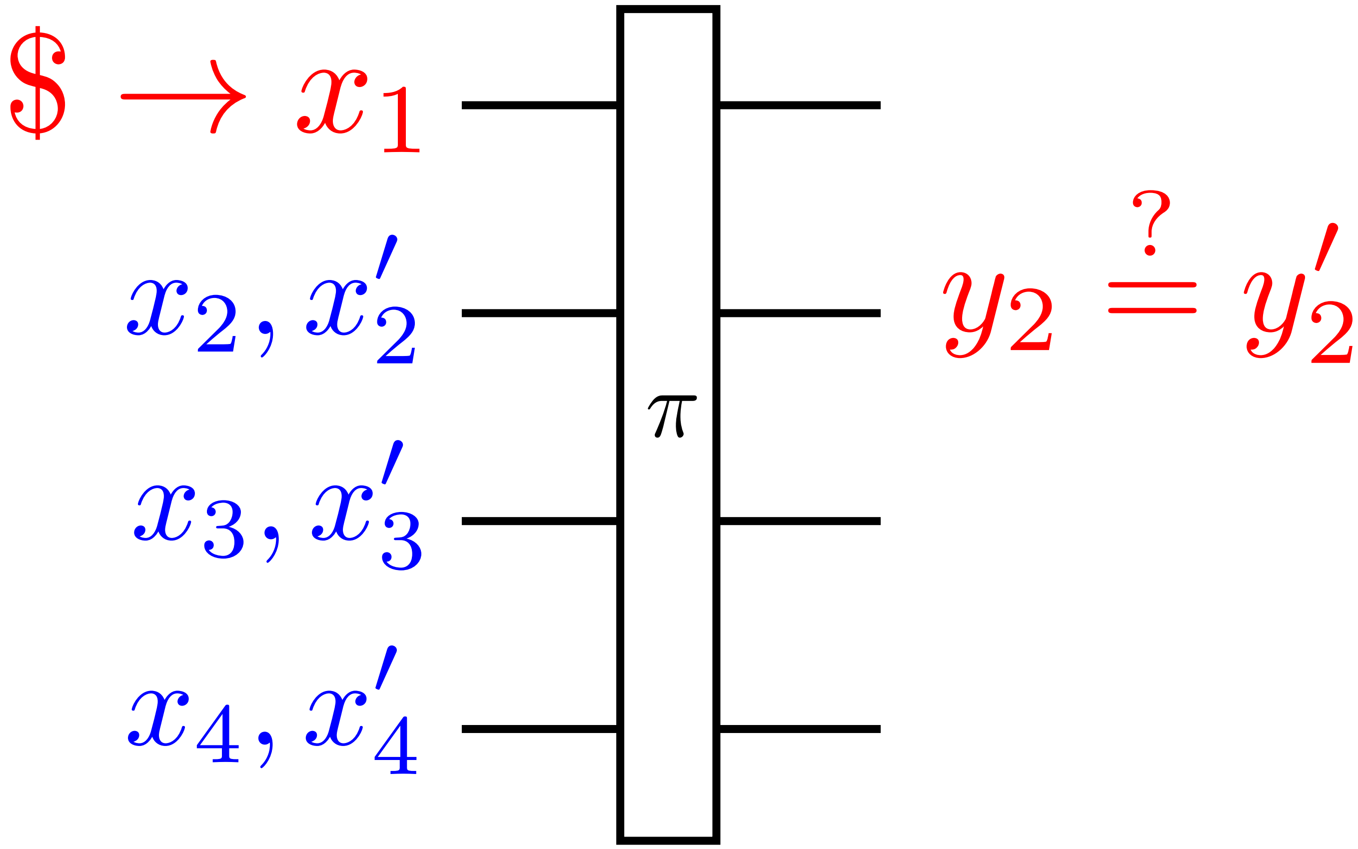


RPR

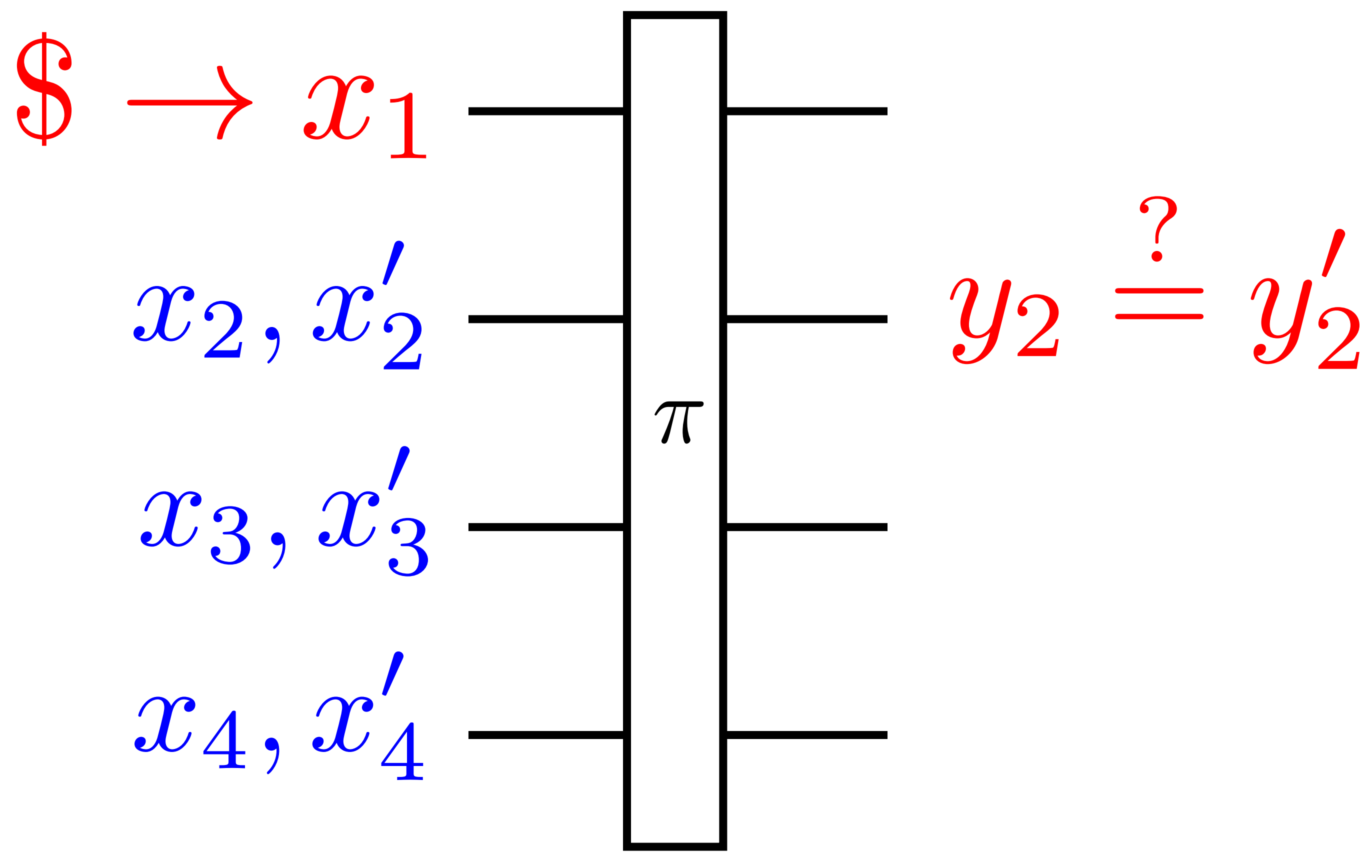


For any fixed values of x_2, x_3 and x_4, and for any y_2^*, there is low probability that $y_2 = y_2^*$ over the randomness in x_1

RCR



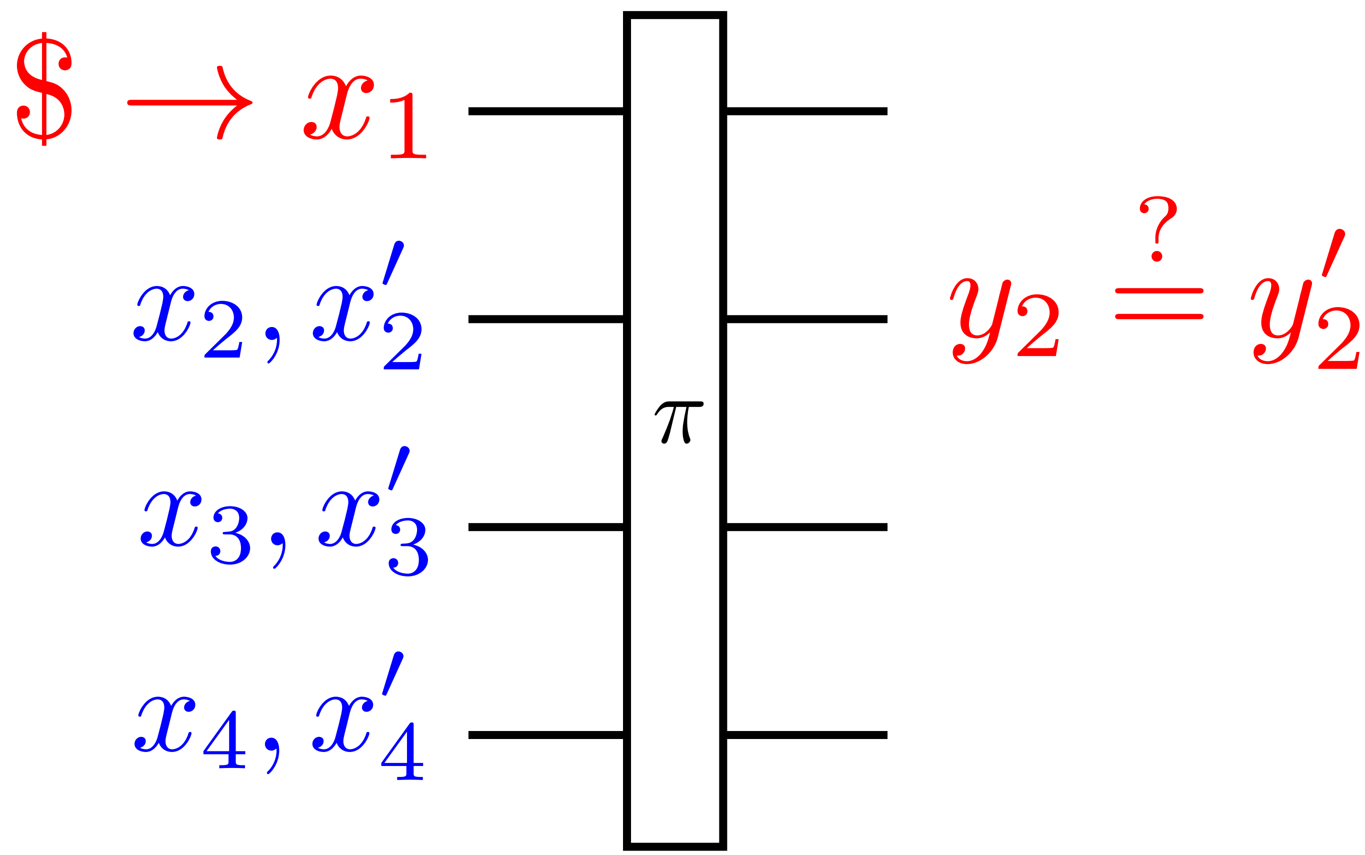
RCR



For any $x_2, x_3, x_4, x_2', x_3', x_4'$ such that $(x_2, x_3, x_4) \neq (x_2', x_3', x_4')$ there is low probability that $y_2 = y_2'$ over the random choice of $x_1 (= x_1')$.

‘C’ stands for ‘CANNOT’ be linear

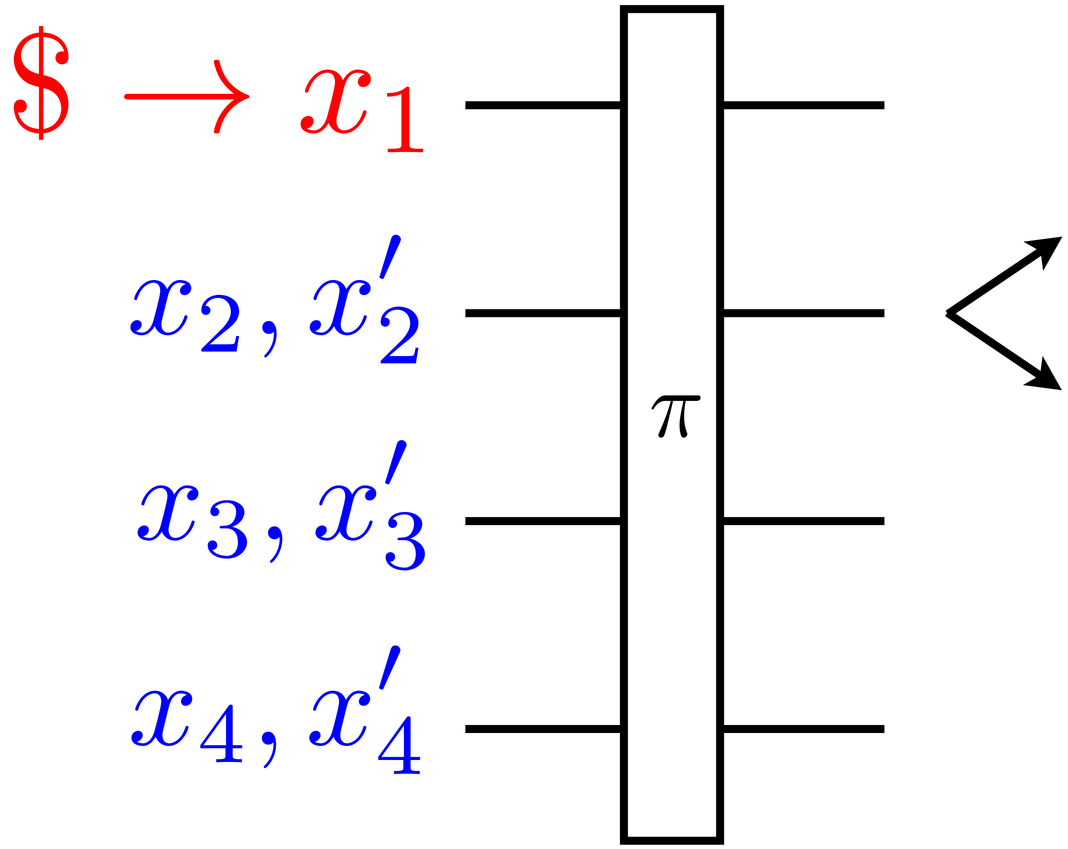
RCR



For any $x_2, x_3, x_4, x_2', x_3', x_4'$ such that $(x_2, x_3, x_4) \neq (x_2', x_3', x_4')$ there is low probability that $y_2 = y_2'$ over the random choice of $x_1 (= x_1')$.

‘C’ stands for ‘CANNOT’ be linear

RCR



$$y_2 = ax_1 + bx_2 + cx_3 + dx_4$$

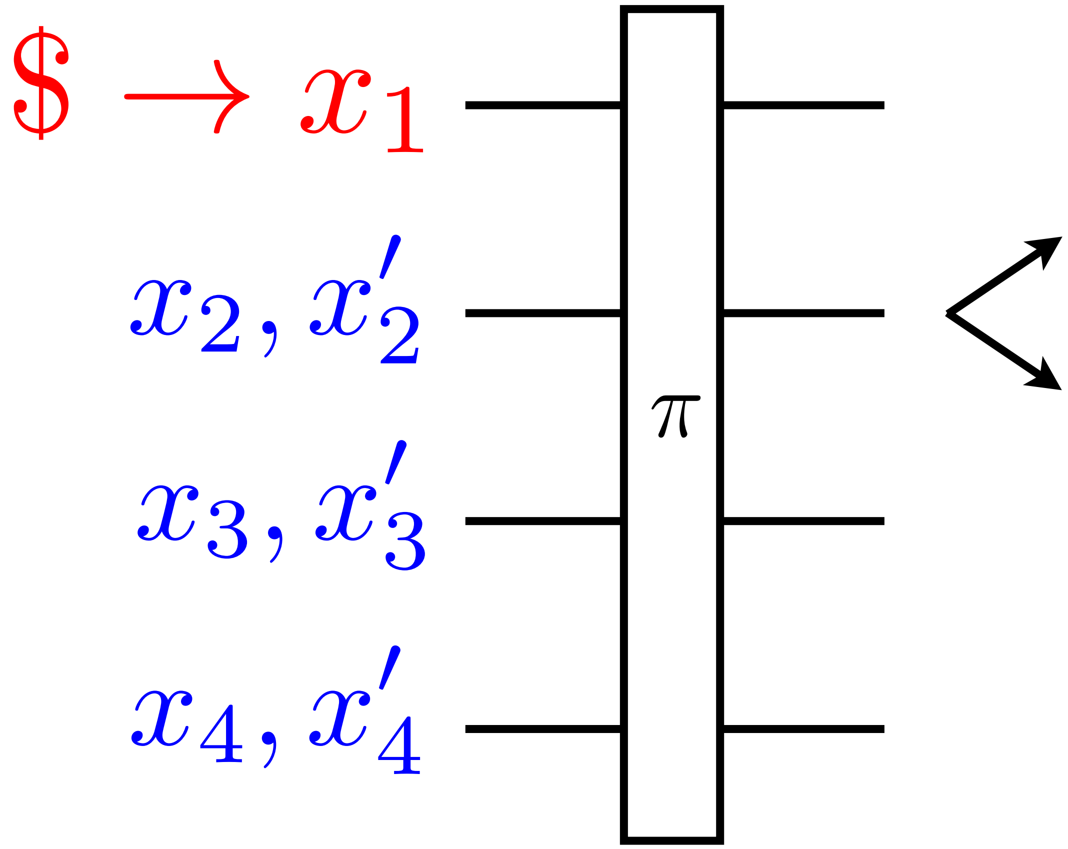
$$y_2' = ax_1 + bx_2' + cx_3' + dx_4'$$

For any $x_2, x_3, x_4, x_2', x_3', x_4'$ such that $(x_2, x_3, x_4) \neq (x_2', x_3', x_4')$ there is low probability that $y_2 = y_2'$ over the random choice of $x_1 (= x_1')$.

‘C’ stands for ‘CANNOT’ be linear

(for $w > 2$)

RCR



$$y_2 = a x_1 + b x_2 + c x_3 + d x_4$$

$$y_2' = a x_1 + b x_2' + c x_3' + d x_4'$$

For any $x_2, x_3, x_4, x_2', x_3', x_4'$ such that $(x_2, x_3, x_4) \neq (x_2', x_3', x_4')$ there is low probability that $y_2 = y_2'$ over the random choice of $x_1 (= x_1')$.

An RCR permutation:

$$\pi = \sigma^{-1} \circ \eta \circ \sigma$$

"Feistel polynomial"

suitably "full rank" linear permutation

An RCR permutation:

$$\pi = \sigma^{-1} \circ \eta \circ \sigma$$

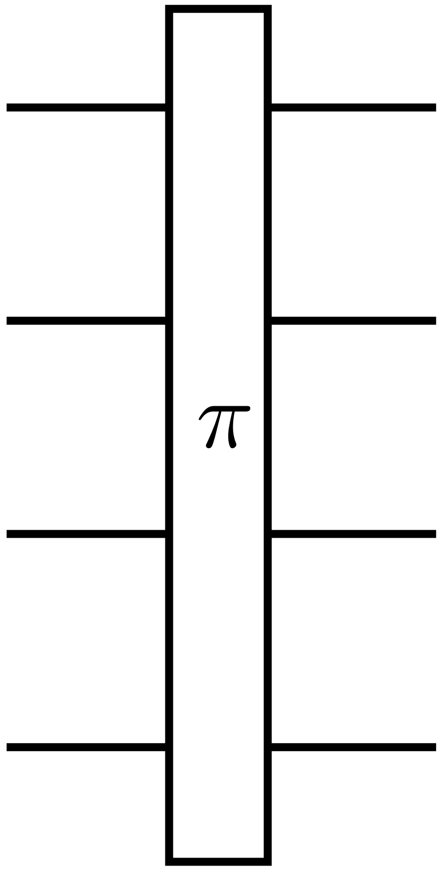
suitably "full rank" linear permutation

"Feistel polynomial":

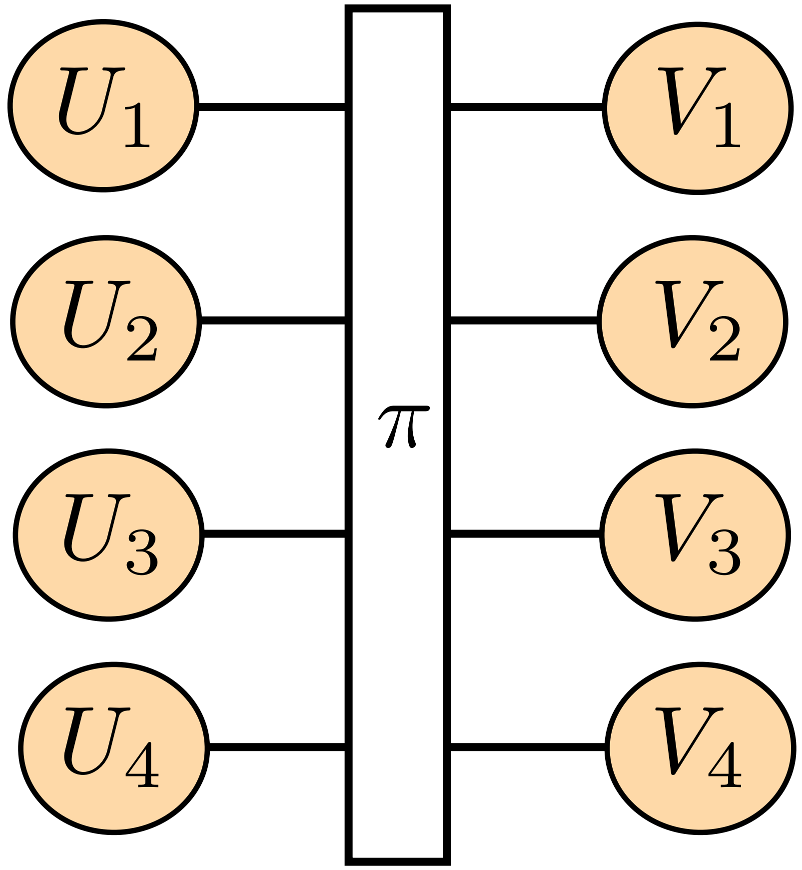
$$\eta(\overrightarrow{\mathbf{x}})[i] = \begin{cases} x_1 + \sum_{j=2}^{w} x_j^{2j+1} & \text{if } i = 1, \\ x_i & \text{if } i \neq 1 \end{cases}$$

(where $\overrightarrow{\mathbf{x}} = (x_1, \ldots, x_w)$)

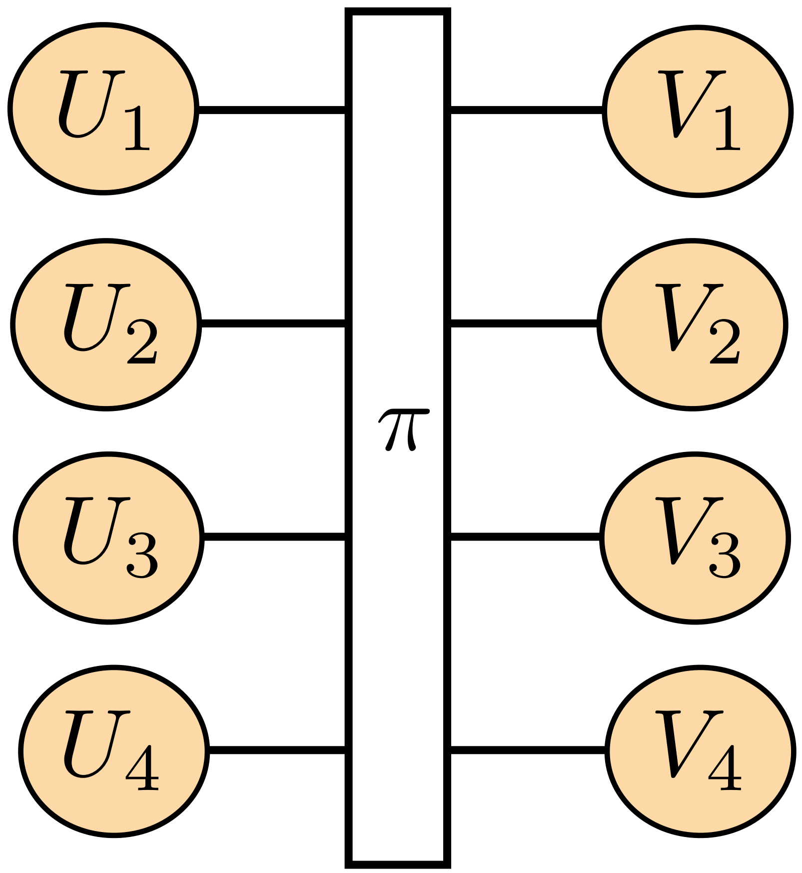
Conductance



Conductance



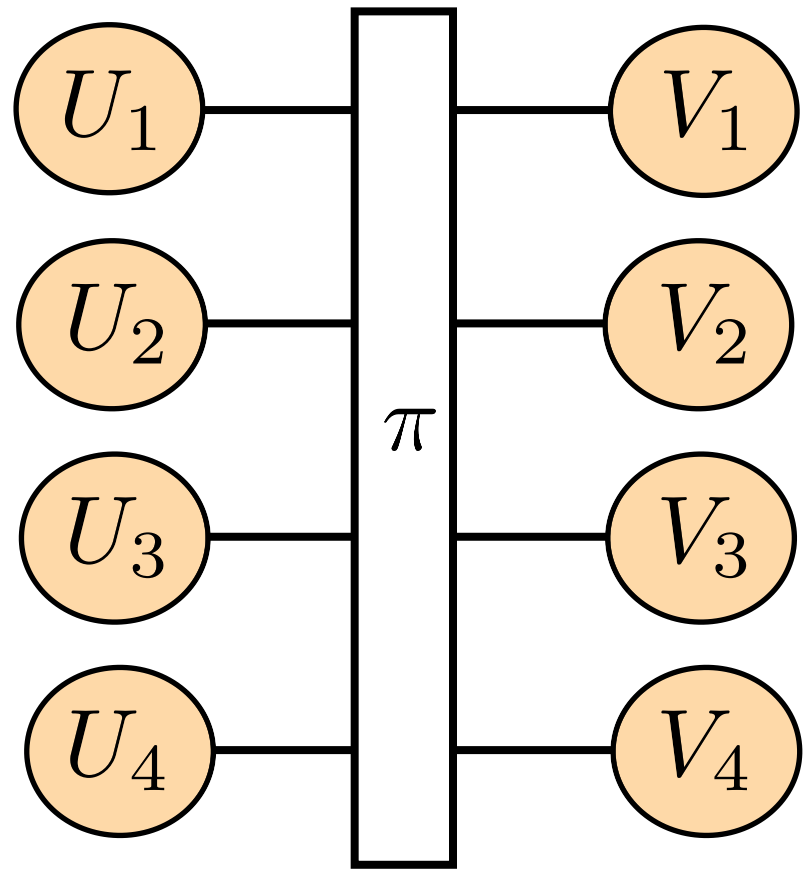
Conductance



$$|U_i| = |V_i| = q$$

$$U_i, V_i \subseteq \{0,1\}^n$$

Conductance



$|U_i| = |V_i| = q$

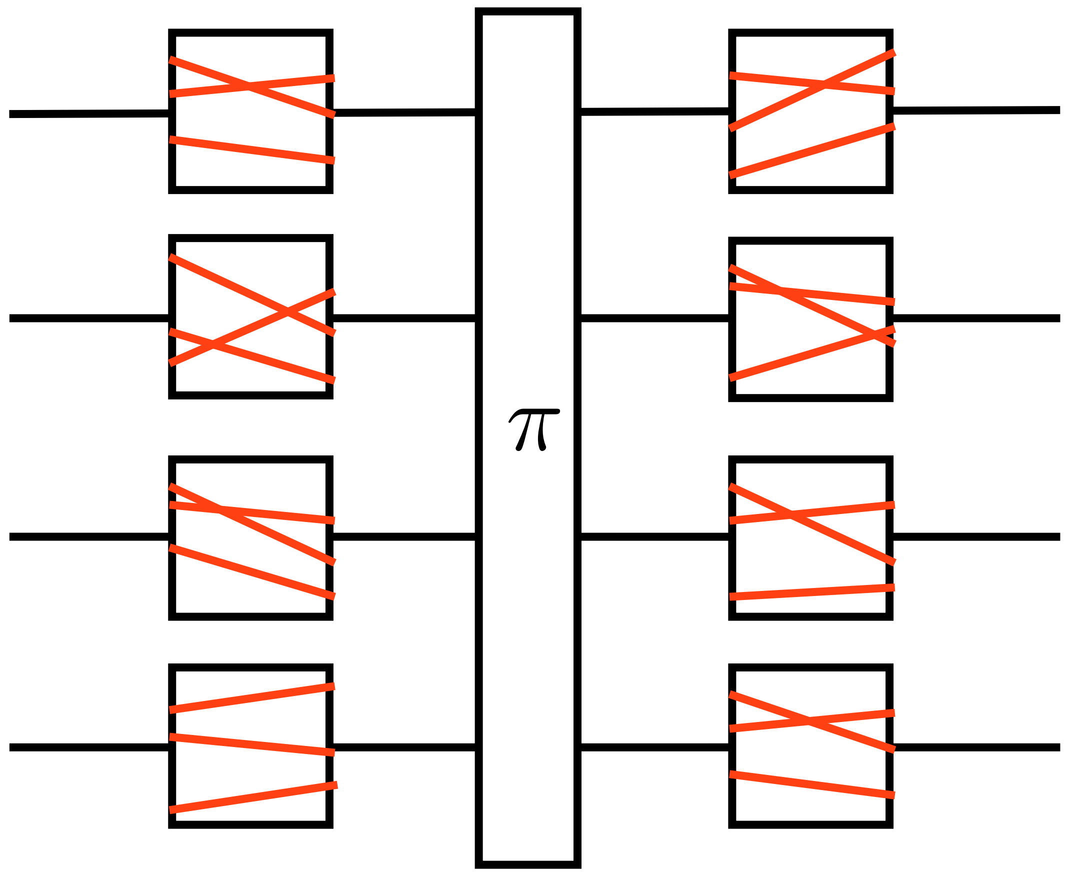
$U_i, V_i \subseteq \{0,1\}^n$

$$|\{(\overrightarrow{\mathbf{x}}, \overrightarrow{\mathbf{y}}) \, : \, \overrightarrow{\mathbf{x}} \in U_1 \times \cdots \times U_w, \, \overrightarrow{\mathbf{y}} \in V_1 \times \cdots \times V_w, \, \pi(\overrightarrow{\mathbf{x}}) = \overrightarrow{\mathbf{y}}\}|$$

conductance(q): maximum of this over all possible choices of $U_1, \ldots, U_w, V_1, \ldots, V_w$ of size q

Conductance

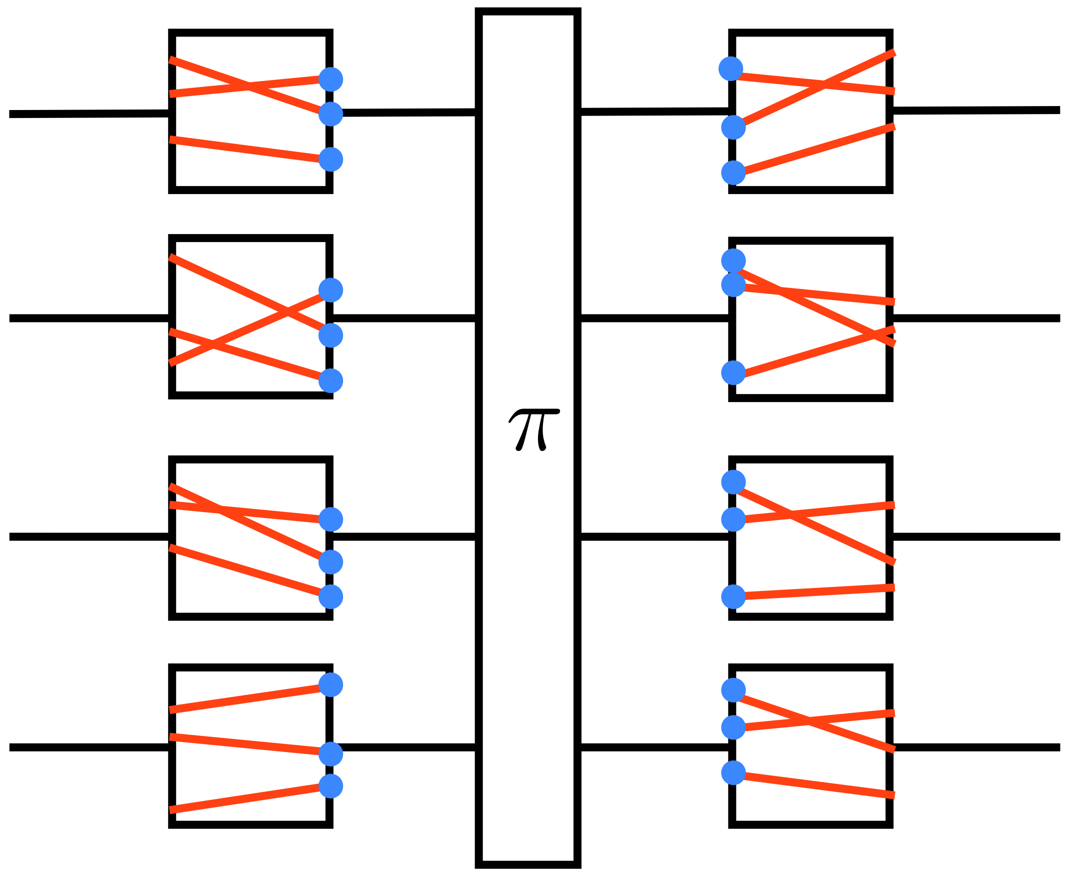
$$|U_i| = |V_i| = q$$
$$U_i, V_i \subseteq \{0,1\}^n$$



$$|\{(\overrightarrow{\mathbf{x}}, \overrightarrow{\mathbf{y}}) \,:\, \overrightarrow{\mathbf{x}} \in U_1 \times \cdots \times U_w,\ \overrightarrow{\mathbf{y}} \in V_1 \times \cdots \times V_w,\ \pi(\overrightarrow{\mathbf{x}}) = \overrightarrow{\mathbf{y}}\}|$$

conductance(q): maximum of this over all possible choices of $U_1, \ldots, U_w, V_1, \ldots, V_w$ of size q

Conductance



$|U_i| = |V_i| = q$

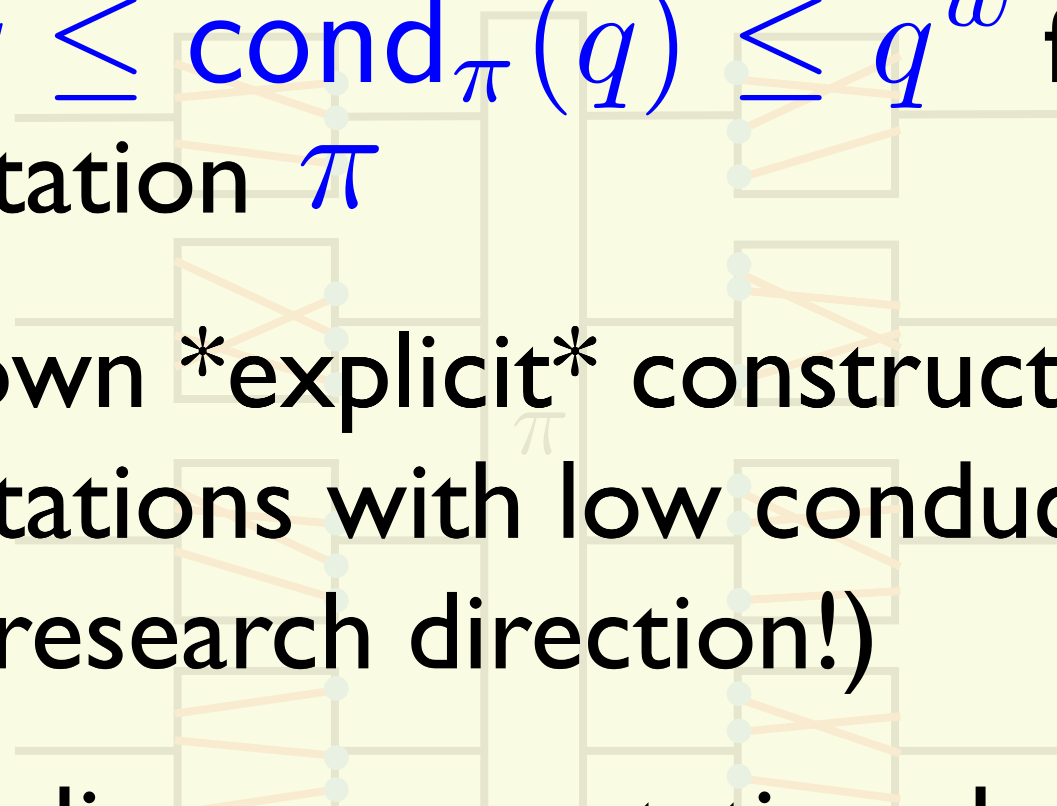
$U_i, V_i \subseteq \{0,1\}^n$

$$|\{(\overrightarrow{\mathbf{x}}, \overrightarrow{\mathbf{y}}) : \overrightarrow{\mathbf{x}} \in U_1 \times \cdots \times U_w,\ \overrightarrow{\mathbf{y}} \in V_1 \times \cdots \times V_w,\ \pi(\overrightarrow{\mathbf{x}}) = \overrightarrow{\mathbf{y}}\}|$$

conductance(q): maximum of this over all possible choices of $U_1, \ldots, U_w, V_1, \ldots, V_w$ of size q

Conductance

- have $q \leq \text{cond}_\pi(q) \leq q^w$ for any permutation π
- no known *explicit* constructions of permutations with low conductance (great research direction!)
- generic linear permutations have suboptimal conductance ($\approx q^2$, maybe worse)



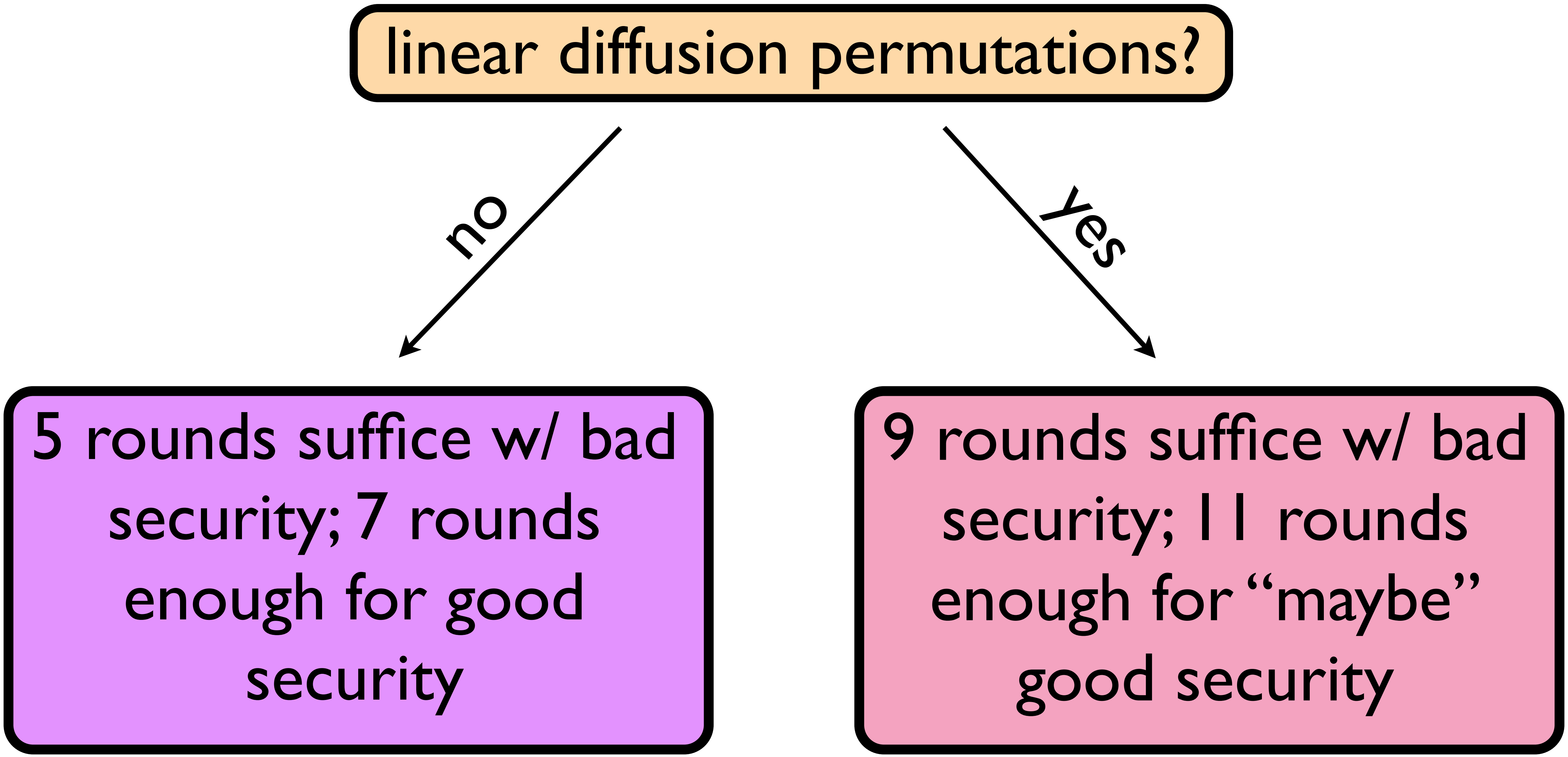
$|U_i| = |V_i| = q$

$\approx q^2$

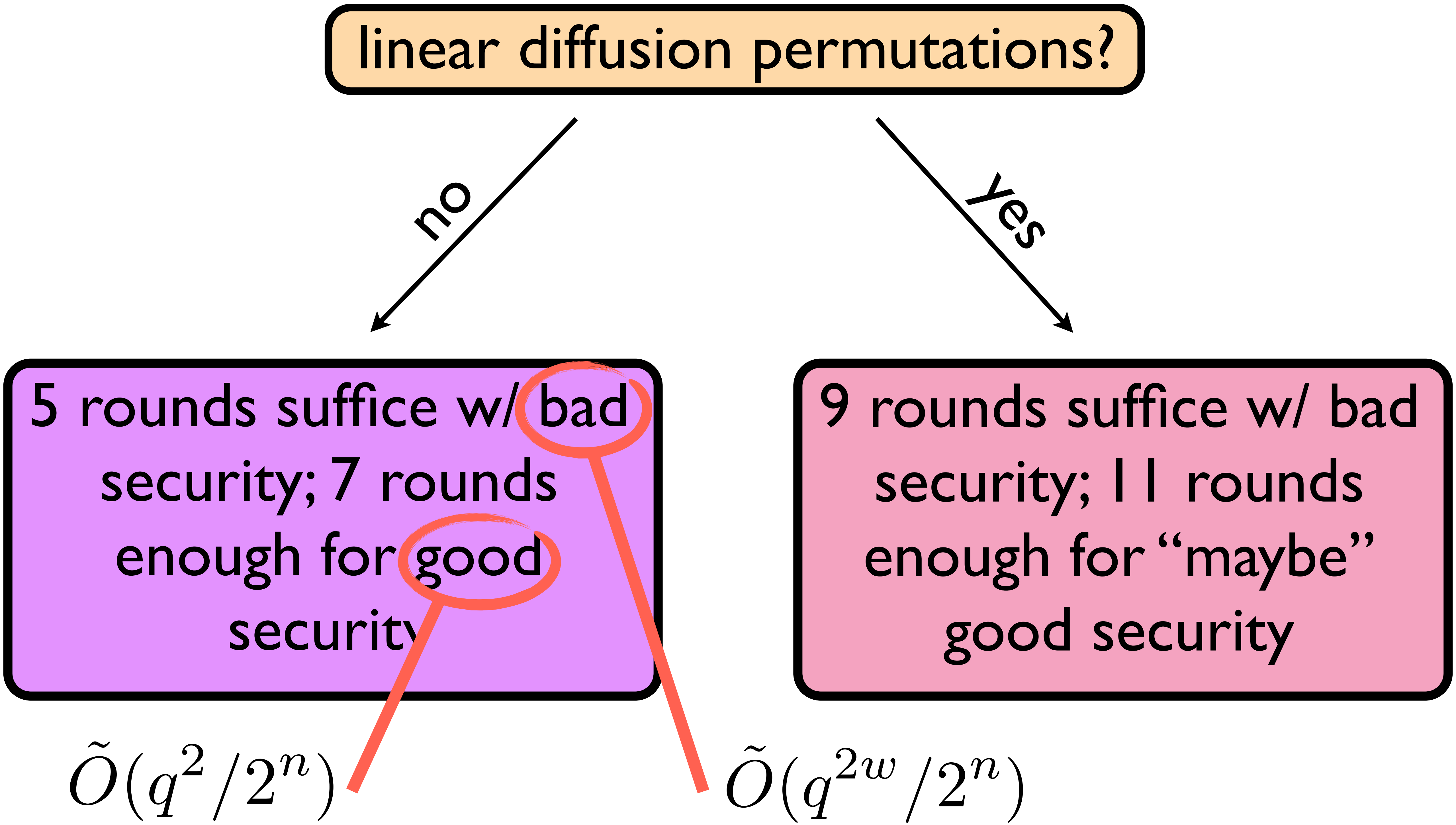
$|\{(\vec{x}, \vec{y}) : \vec{x} \in U_1 \times \cdots \times U_w,\ \vec{y} \in V_1 \times \cdots \times V_w,\ \pi(\vec{x}) = \vec{y}\}|$

conductance(q): maximum of this over all possible choices of $U_1, \ldots, U_w, V_1, \ldots, V_w$ of size q

(synopsis of results)



(synopsis of results)



(synopsis of results)

linear diffusion permutations?

Only one theorem & simulator in paper! (But subject to 3 boolean flags, for a total of eight flavors.)

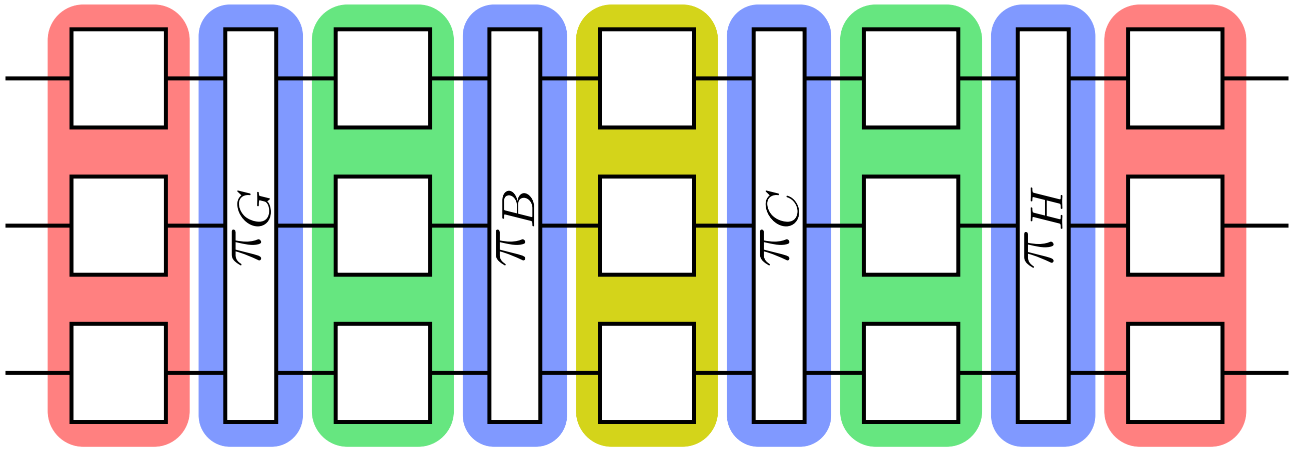
5 rounds suffice w/ bad security; 7 rounds enough for good security

9 rounds suffice w/ bad security; 11 rounds enough for "maybe" good security

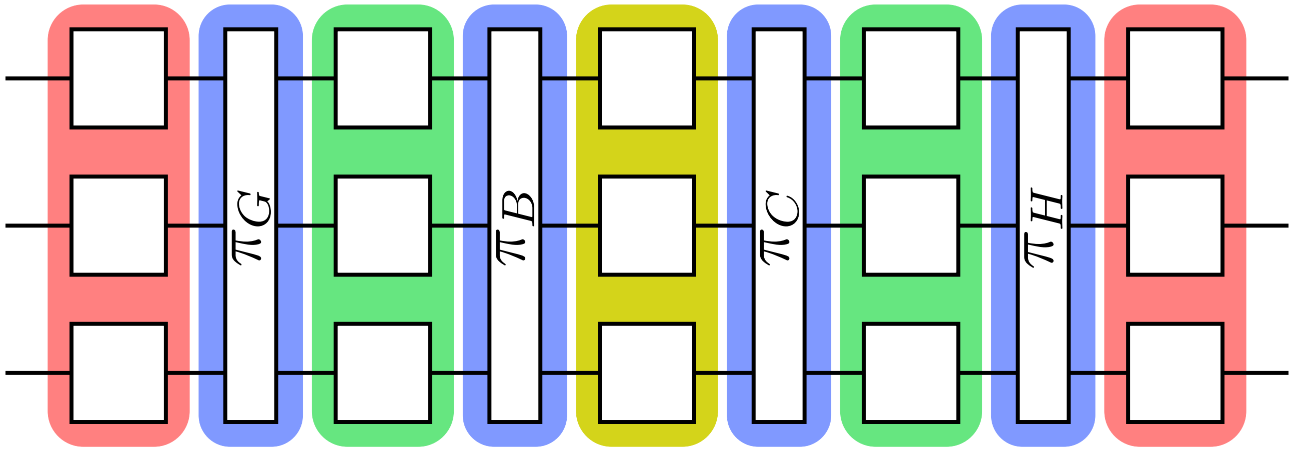
$\tilde{O}(q^2/2^n)$

$\tilde{O}(q^{2w}/2^n)$

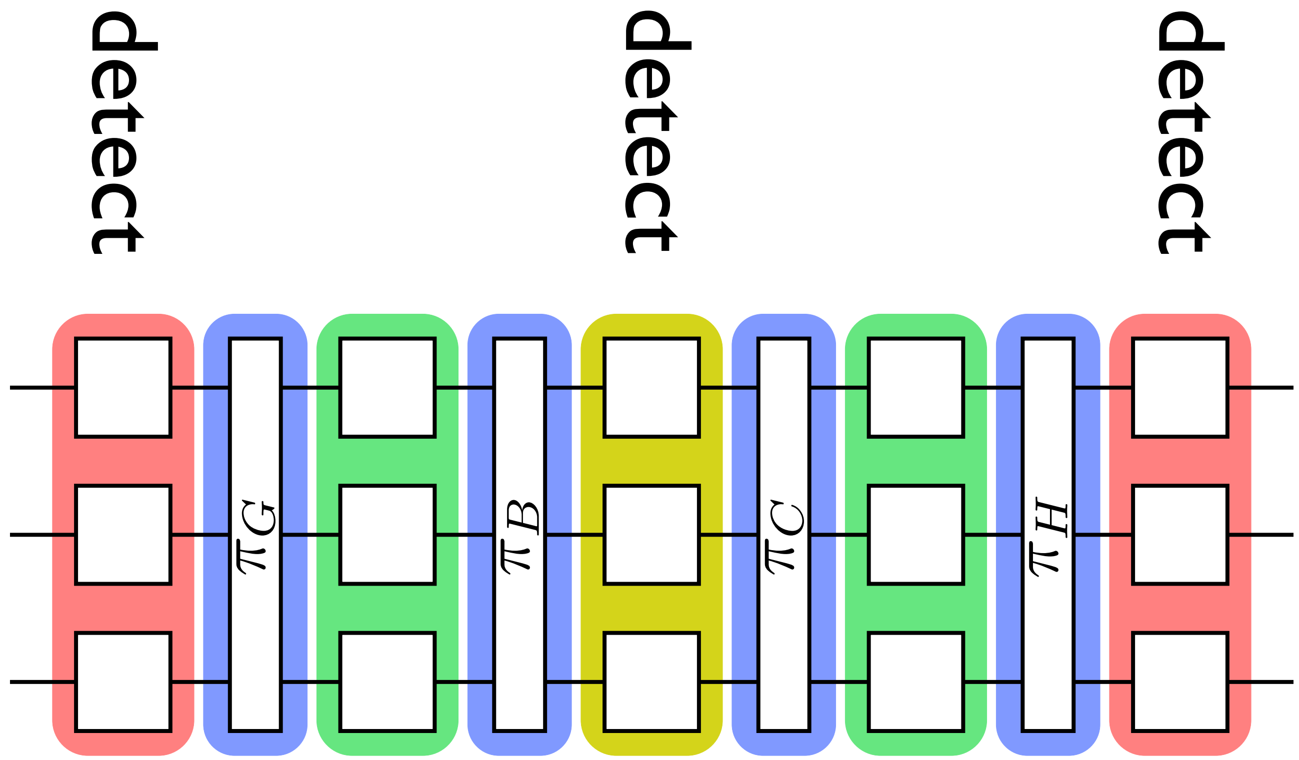
(The 5-round Simulator)



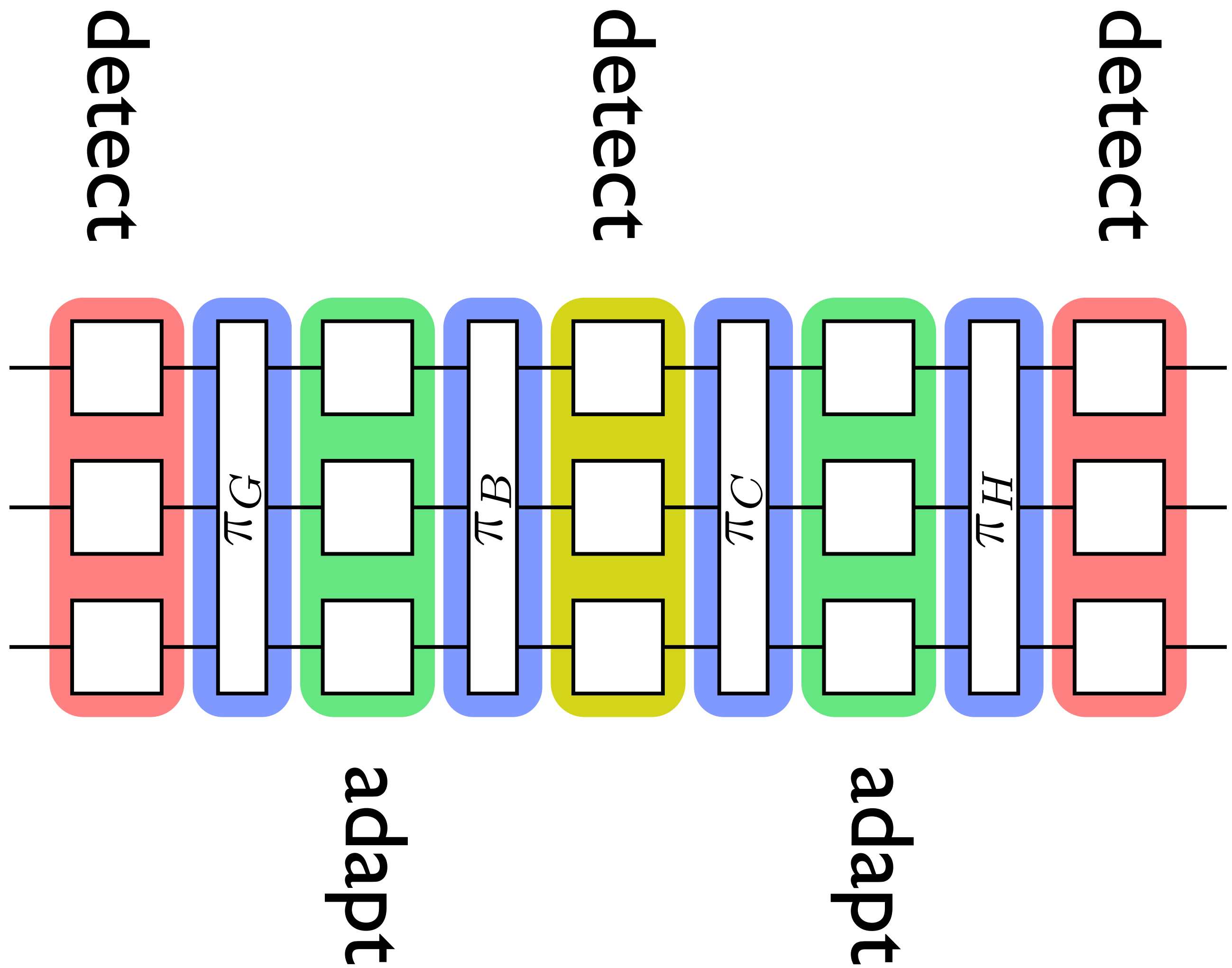
(The 5-round Simulator)

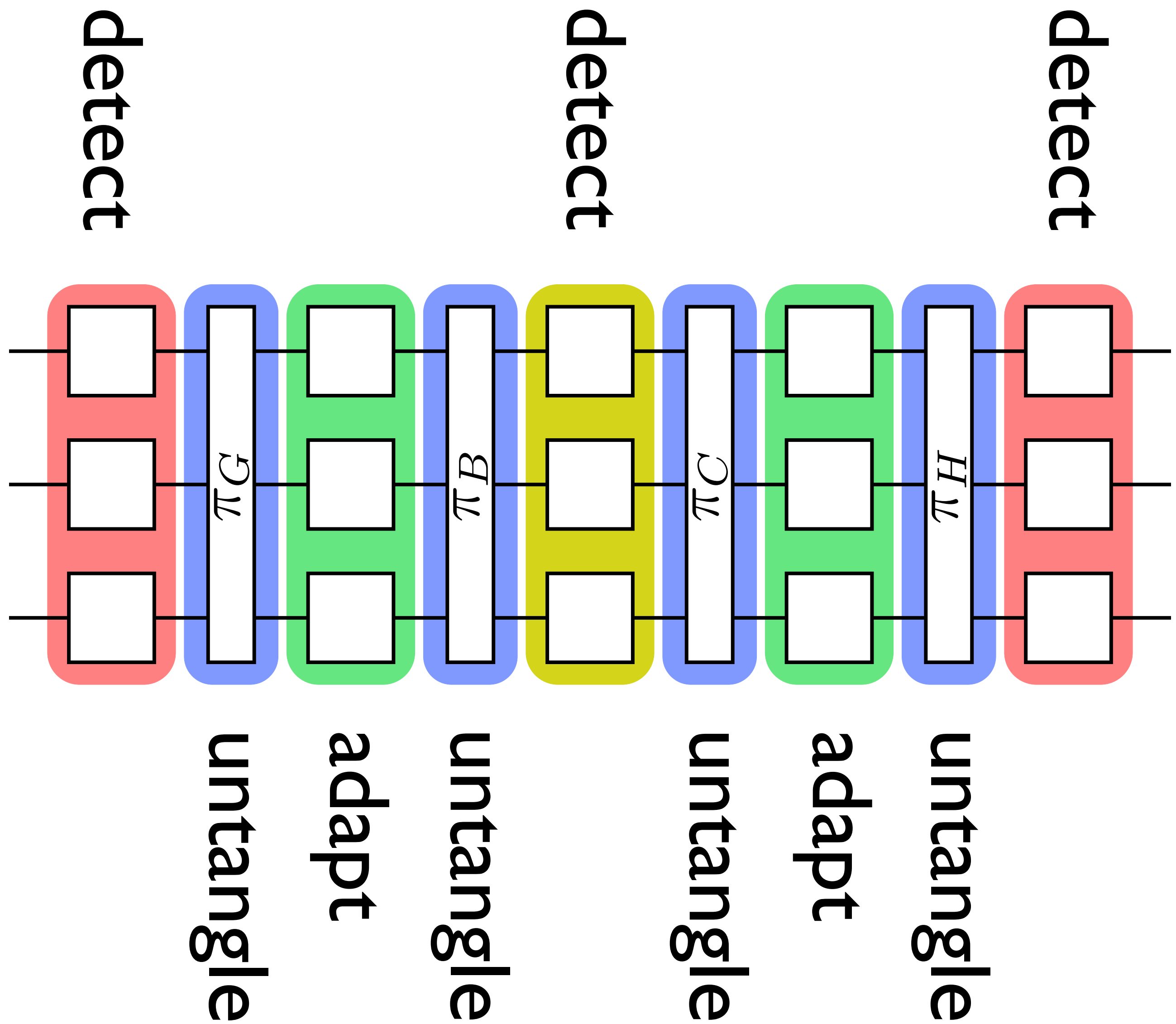


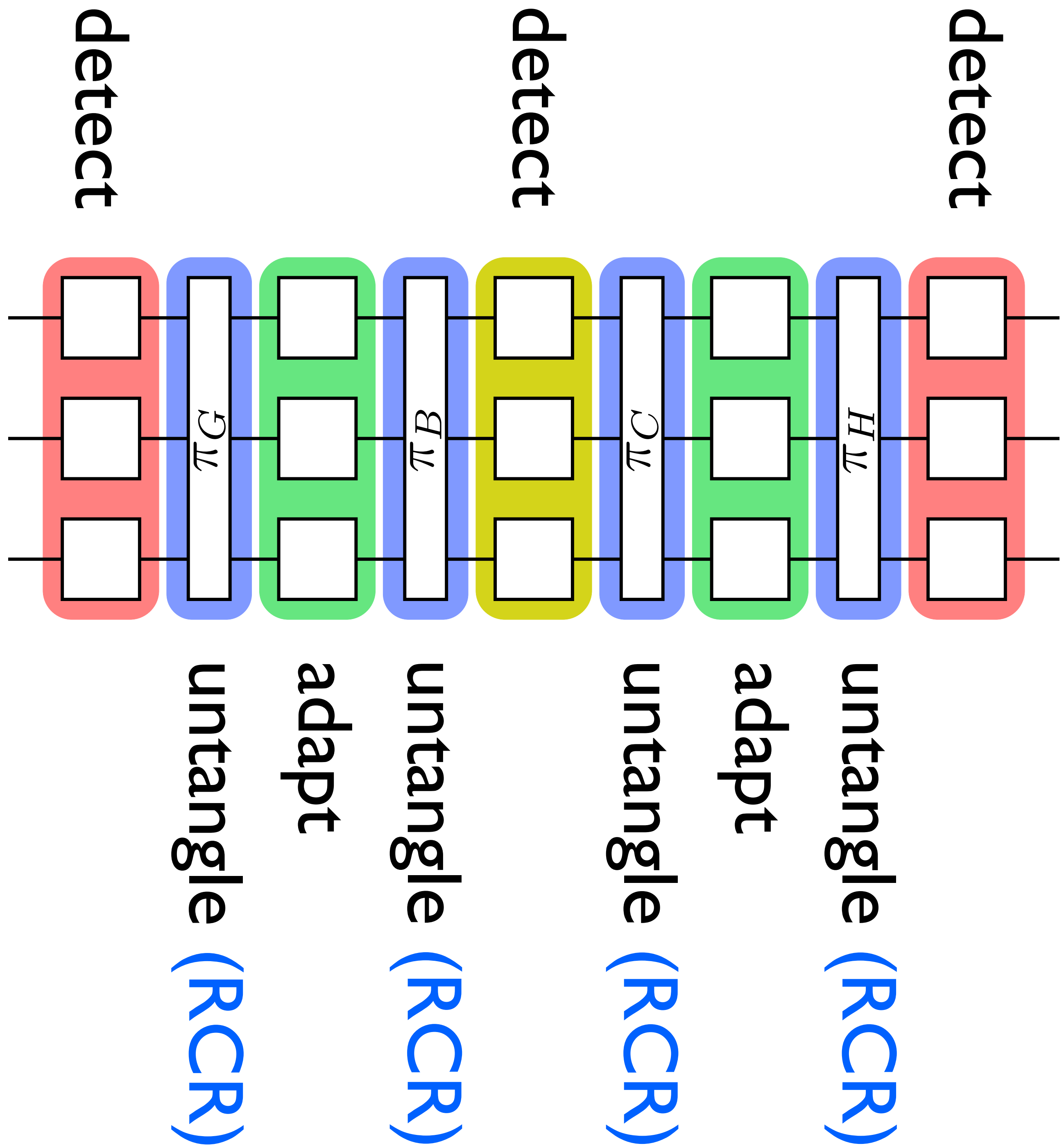
Basic Idea: Path-completion strategy similar to 14-round & 10-round Feistel simulators of HKT11, Seurin09

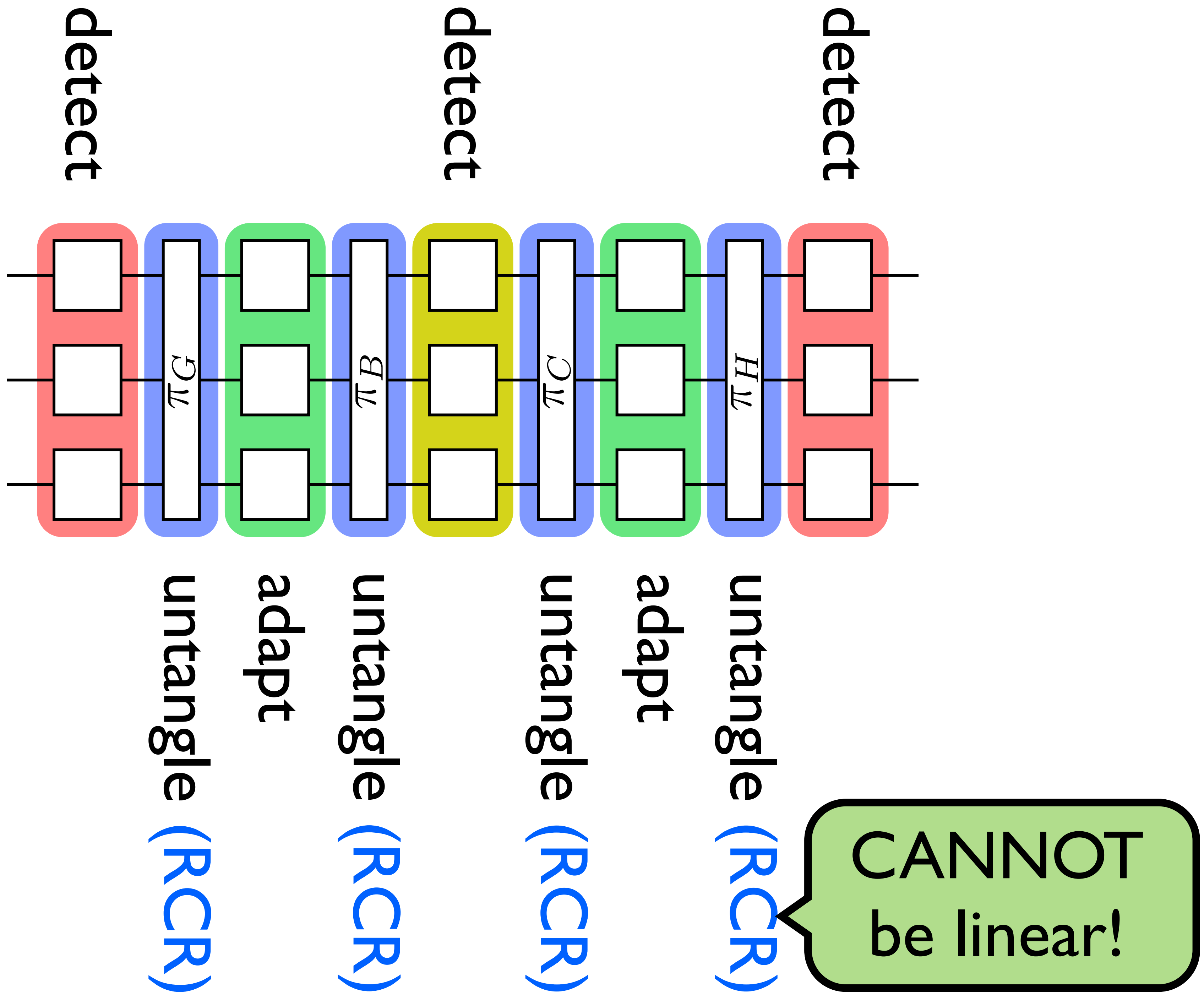


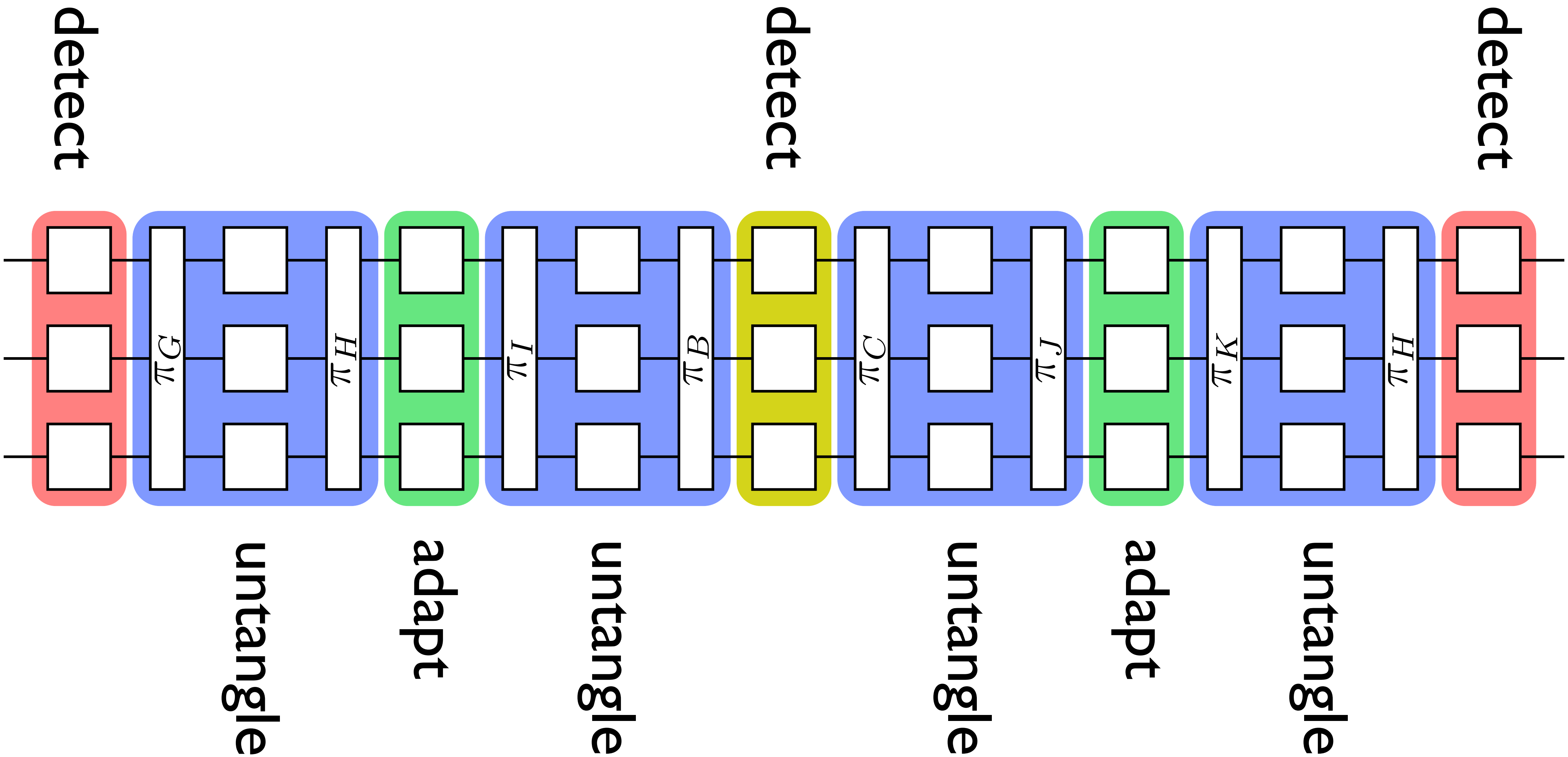
Basic Idea: Path-completion strategy similar to 14-round & 10-round Feistel simulators of HKT11, Seurin09

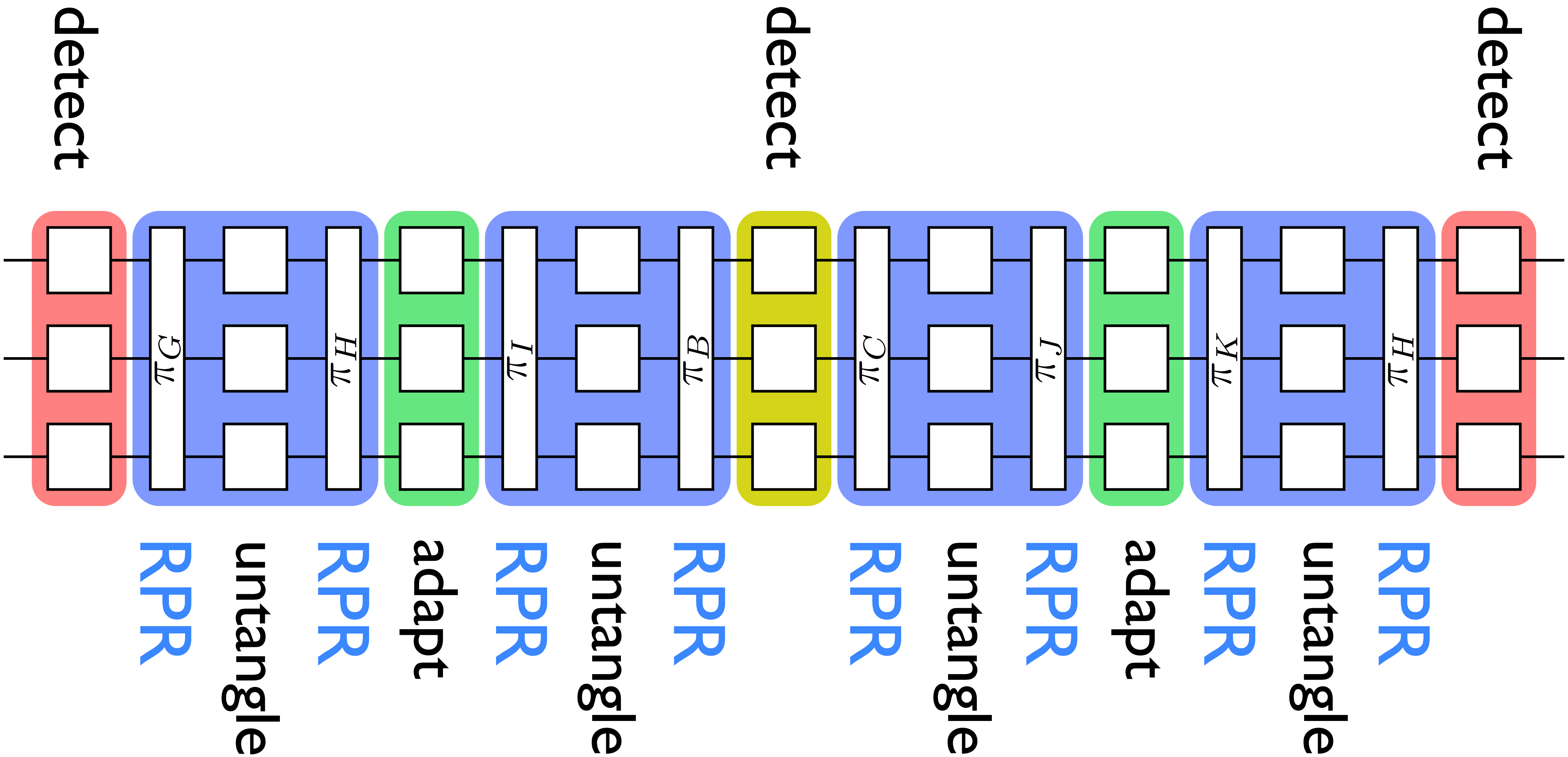

detect
detect
detect
π_G
π_B
π_C
π_H
adapt
adapt

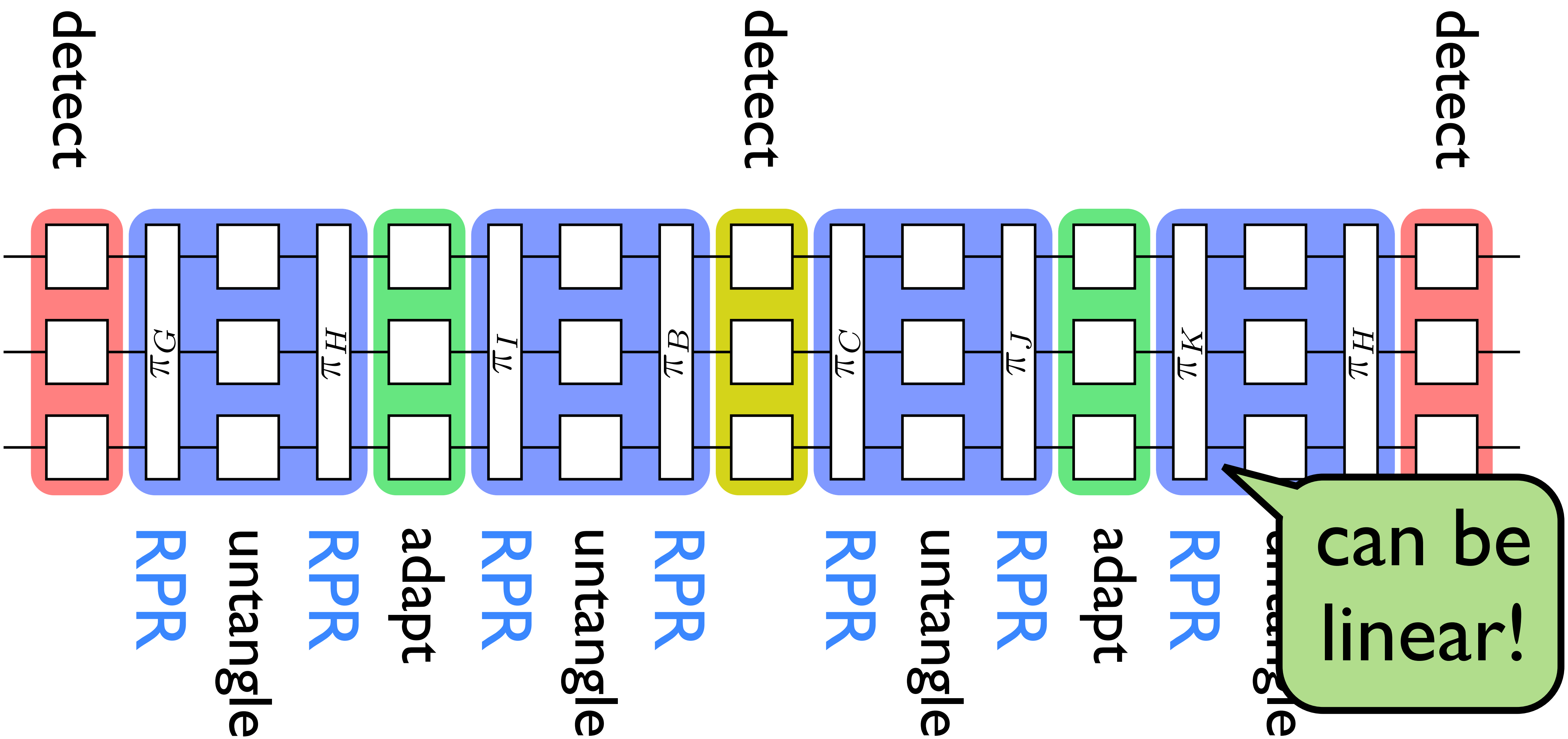

detect
untangle
π_G
adapt
untangle
π_B
detect
untangle
π_C
adapt
untangle
π_H
detect

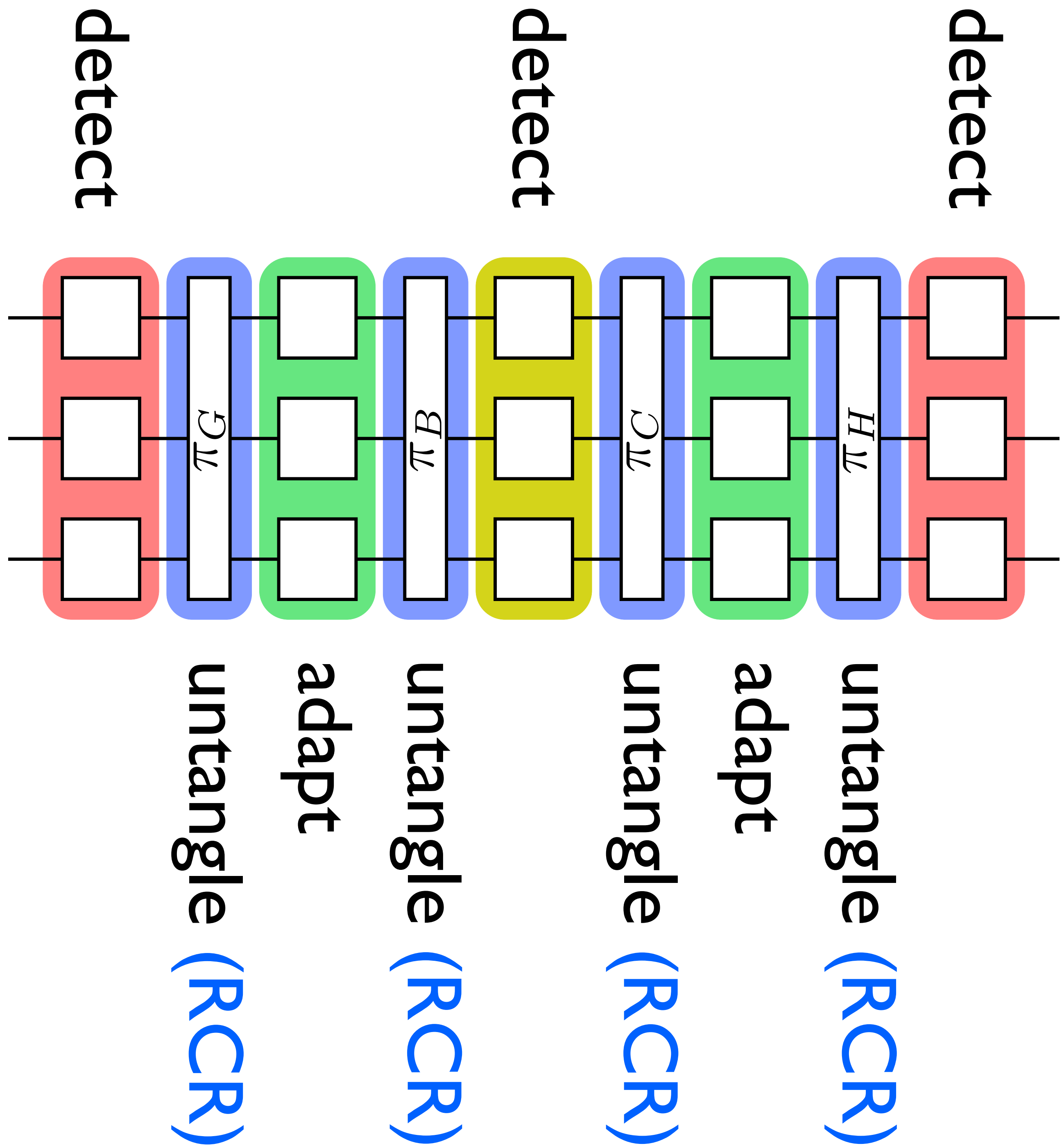

detect
detect
detect
π_G
π_B
π_C
π_H
untangle (RCR)
adapt
untangle (RCR)
untangle (RCR)
adapt
untangle (RCR)

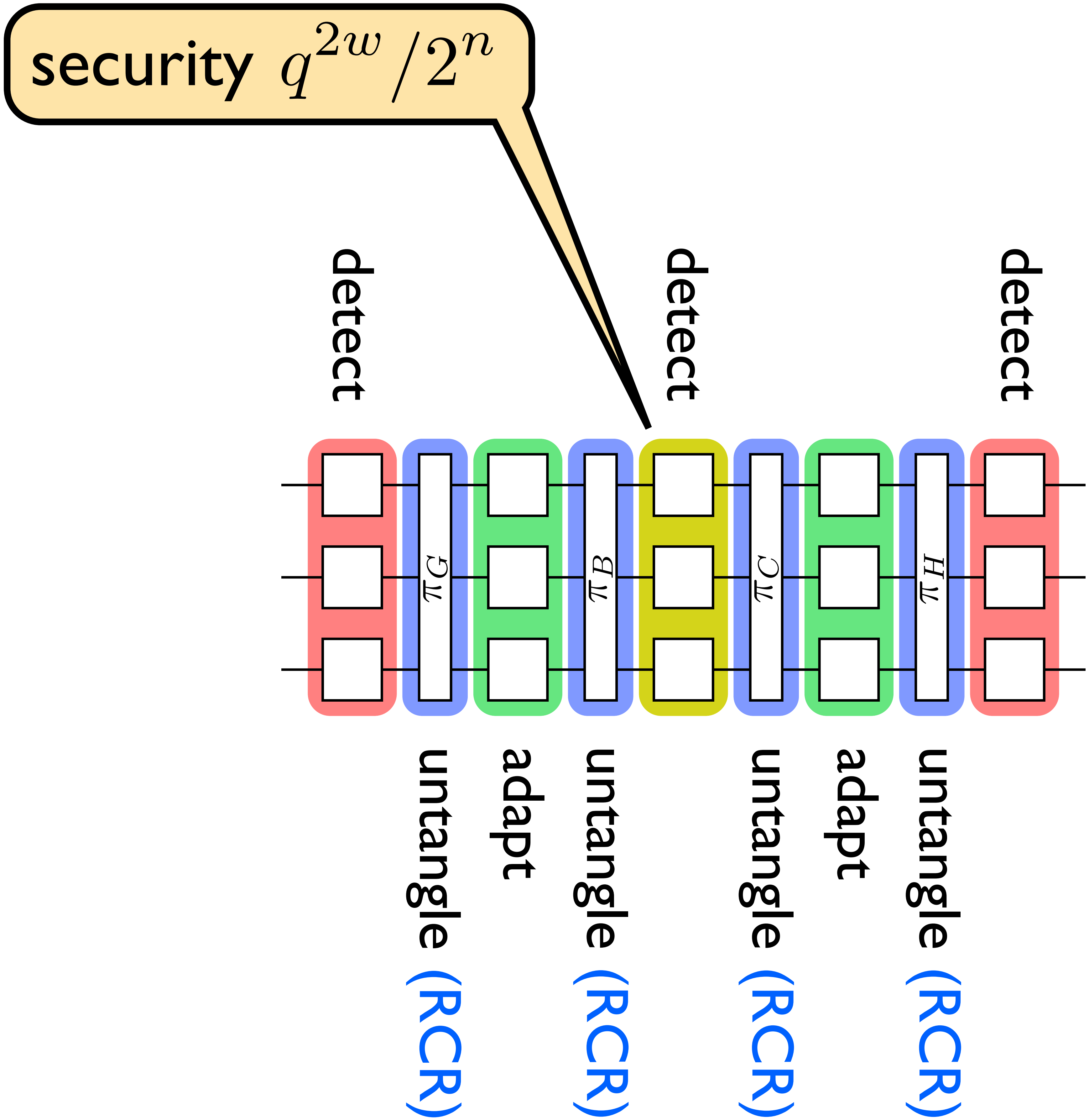

detect
detect
detect
π_G
π_B
π_C
π_H
untangle (RCR)
adapt
untangle (RCR)
untangle (RCR)
adapt
untangle (RCR)
CANNOT be linear!

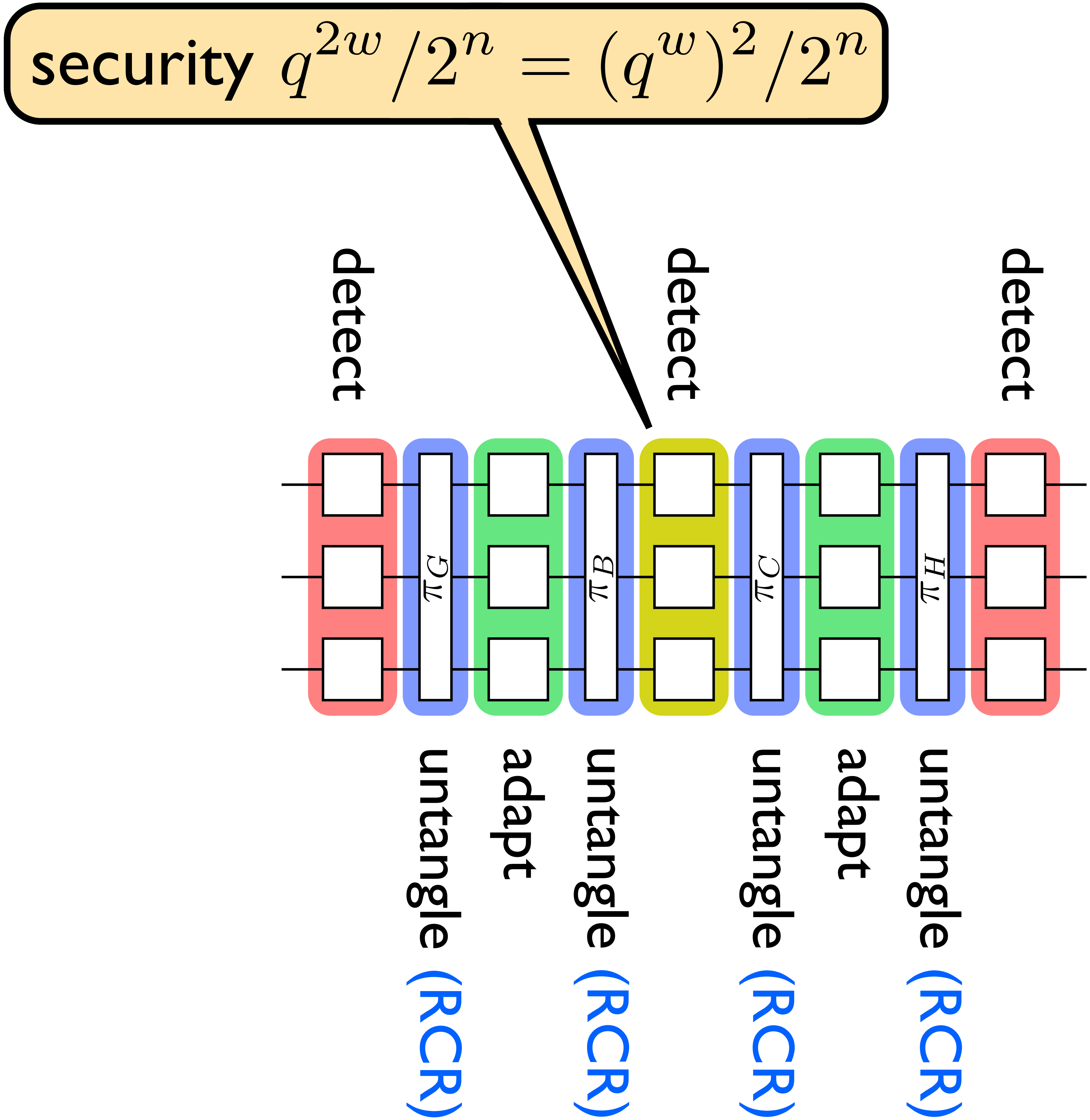

detect
π_G
π_H
untangle
adapt
π_I
π_B
untangle
detect
π_C
π_J
untangle
adapt
π_K
π_H
untangle
detect

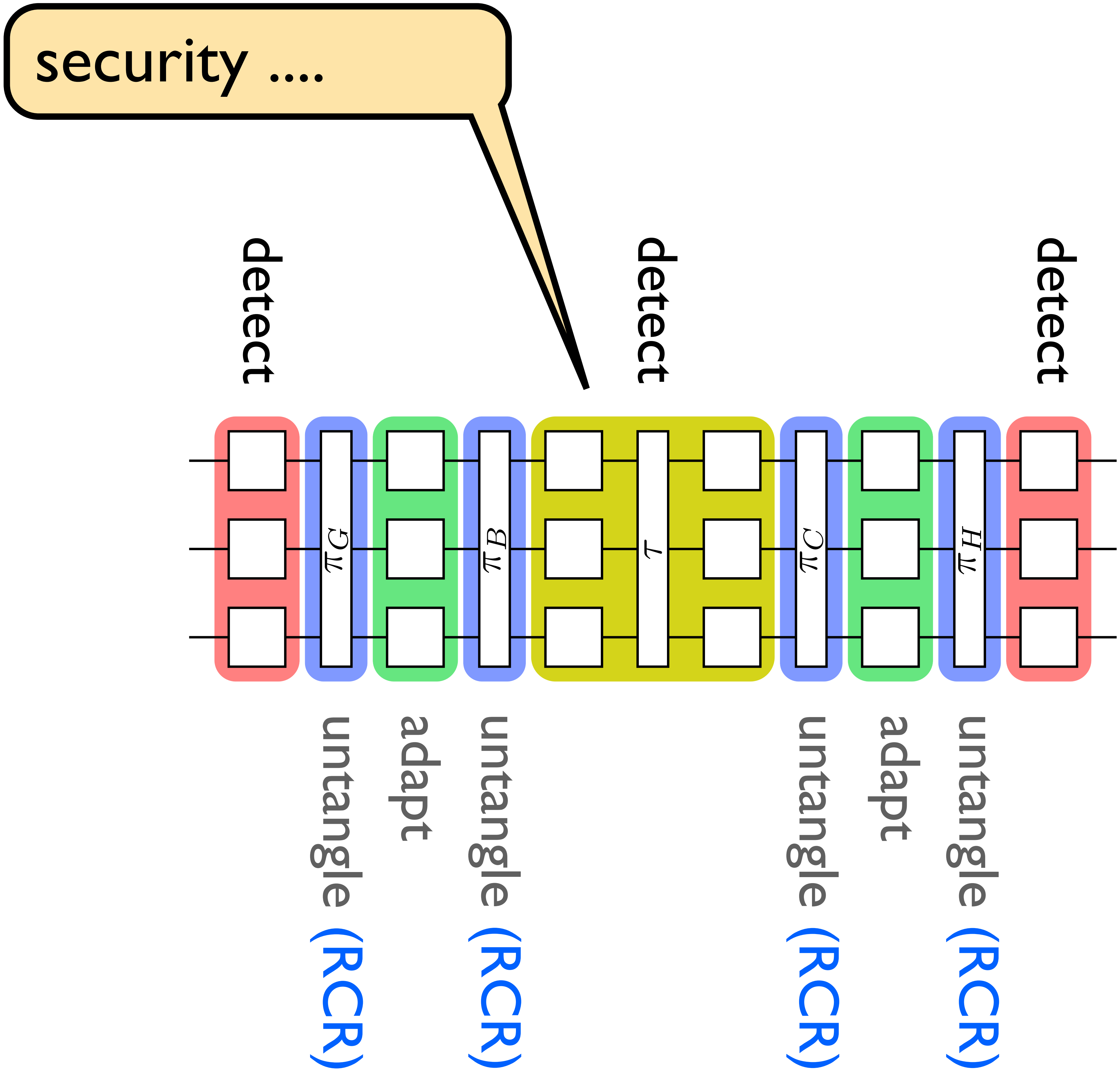

detect
detect
detect
π_G
π_H
π_I
π_B
π_C
π_J
π_K
π_H
RPR
untangle
RPR
adapt
RPR
untangle
RPR
RPR
untangle
RPR
adapt
RPR
untangle
RPR


detect
detect
detect
π_G
π_H
π_I
π_B
π_C
π_J
π_K
π_H
RPR
untangle
RPR
adapt
RPR
untangle
RPR
RPR
untangle
RPR
adapt
RPR
can be linear!

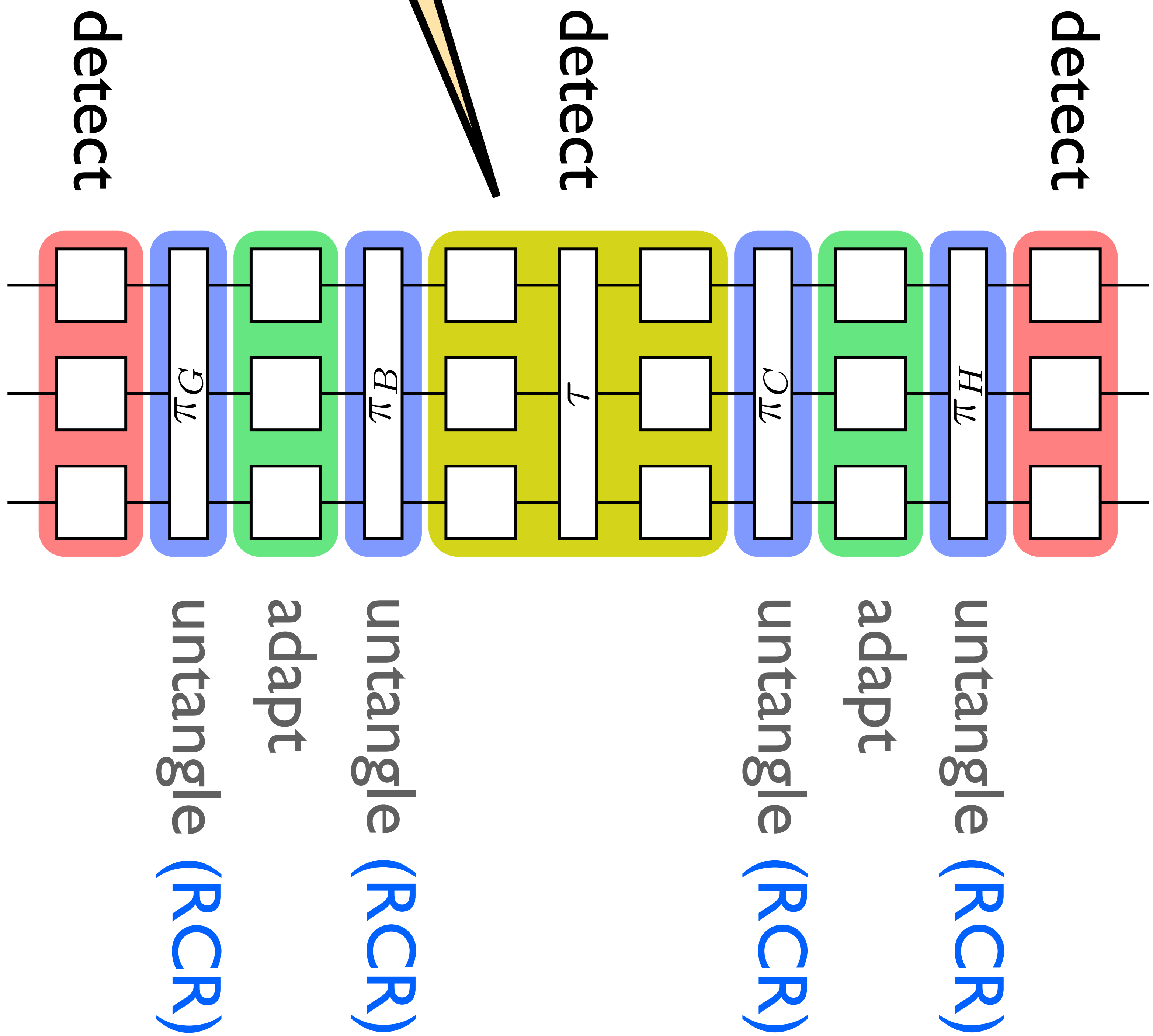

detect
detect
detect
π_G
π_B
π_C
π_H
untangle (RCR)
adapt
untangle (RCR)
untangle (RCR)
adapt
untangle (RCR)


security $q^{2w}/2^n$
detect
detect
detect
π_G
π_B
π_C
π_H
untangle (RCR)
adapt
untangle (RCR)
untangle (RCR)
adapt
untangle (RCR)

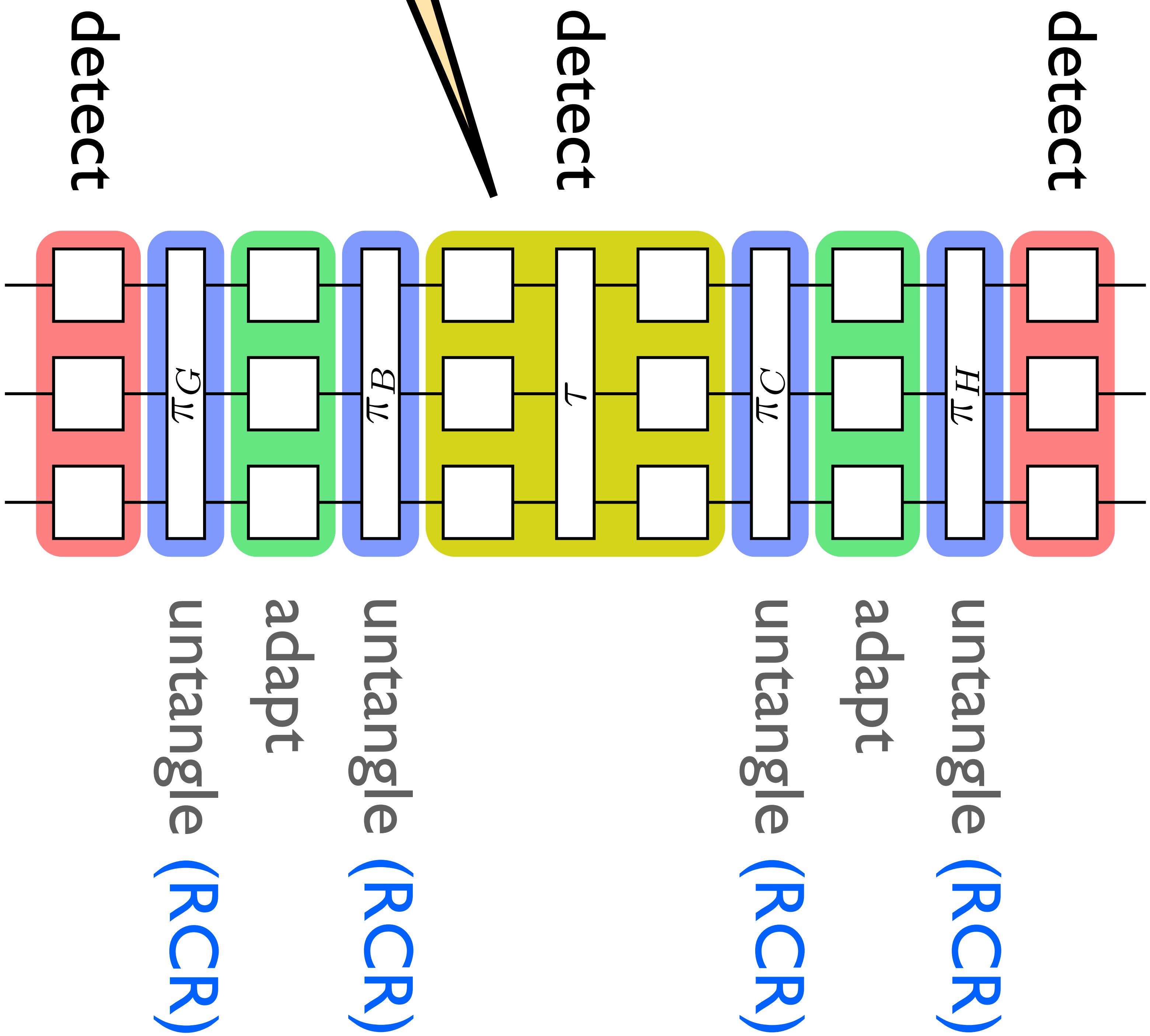

security $q^{2w}/2^n = (q^w)^2/2^n$
detect
detect
detect
π_G
π_B
π_C
π_H
untangle (RCR)
adapt
untangle (RCR)
untangle (RCR)
adapt
untangle (RCR)


security ....
detect
detect
detect
πG
πB
τ
πC
πH
untangle (RCR)
adapt
untangle (RCR)
untangle (RCR)
adapt
untangle (RCR)

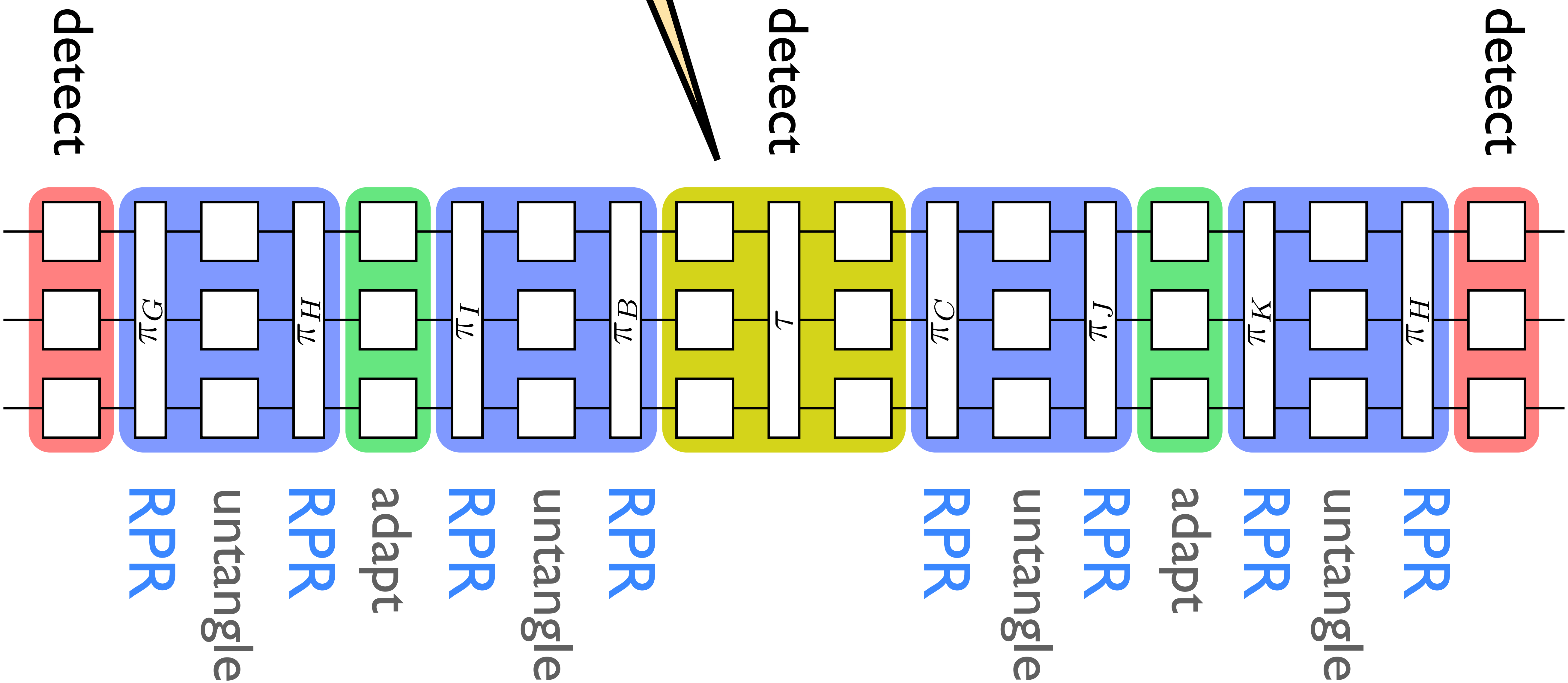
security $\mathrm{cond}_\tau(q)^2/2^n$

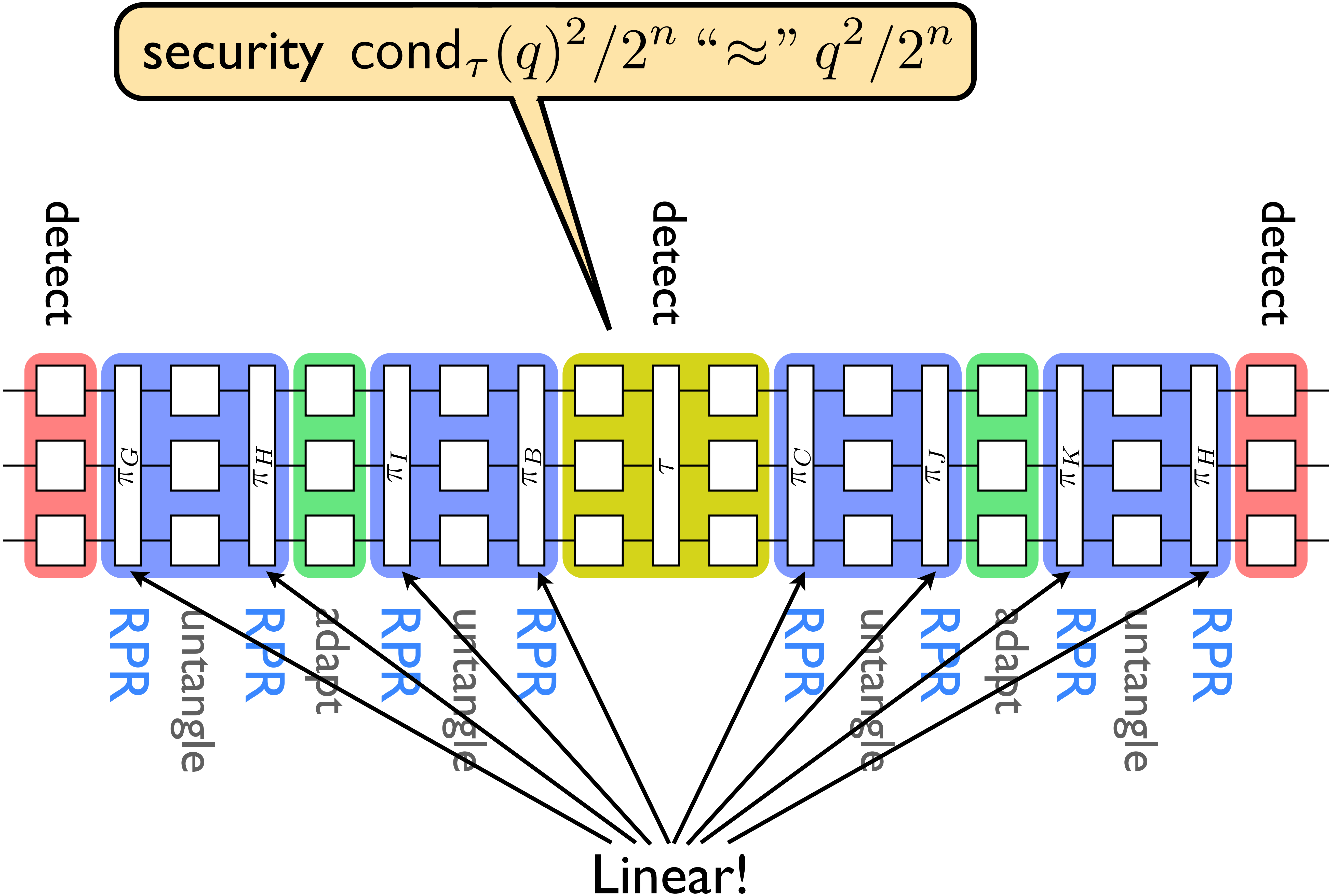


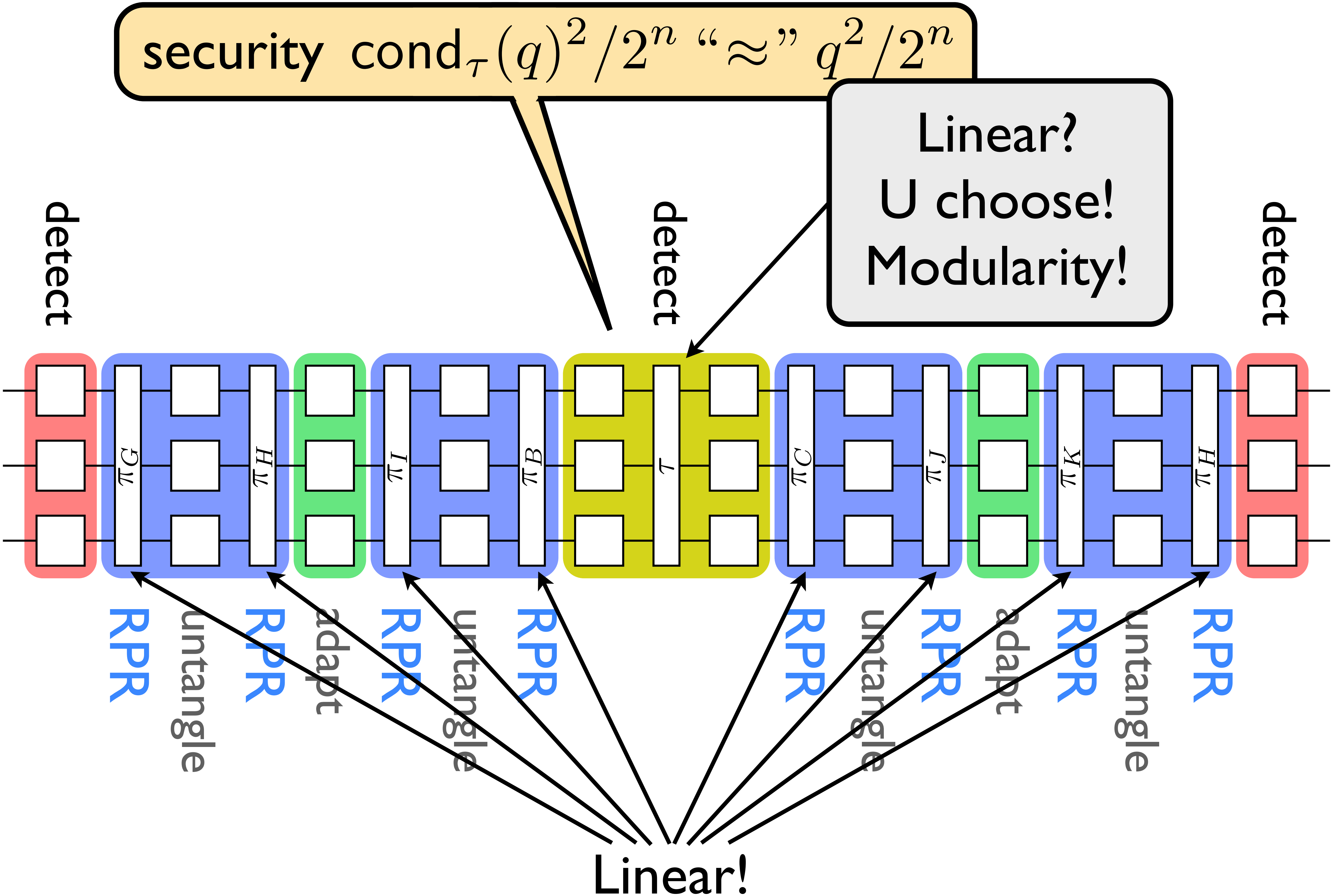
security $\text{cond}_\tau(q)^2/2^n$ "$\approx$" $q^2/2^n$

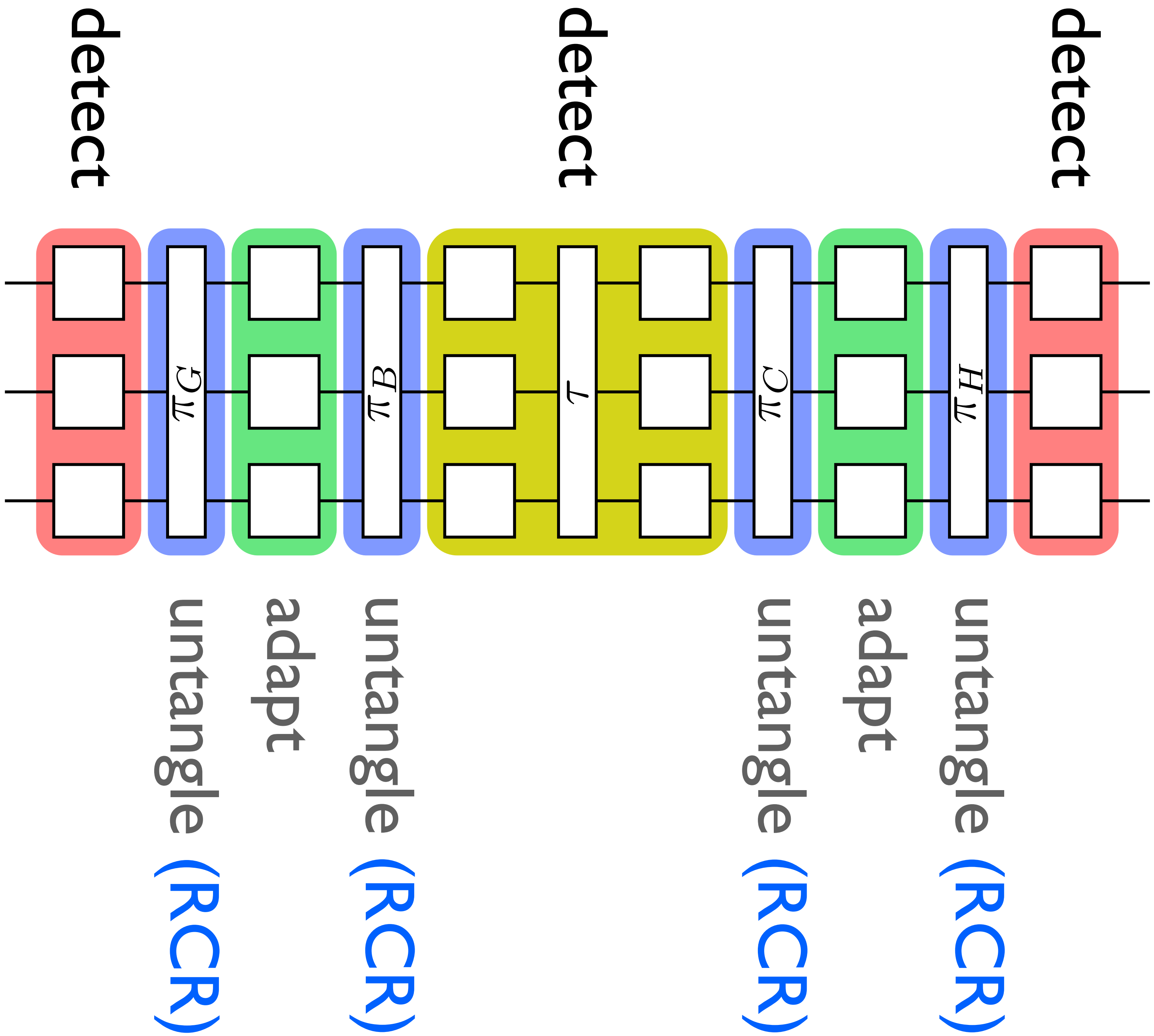


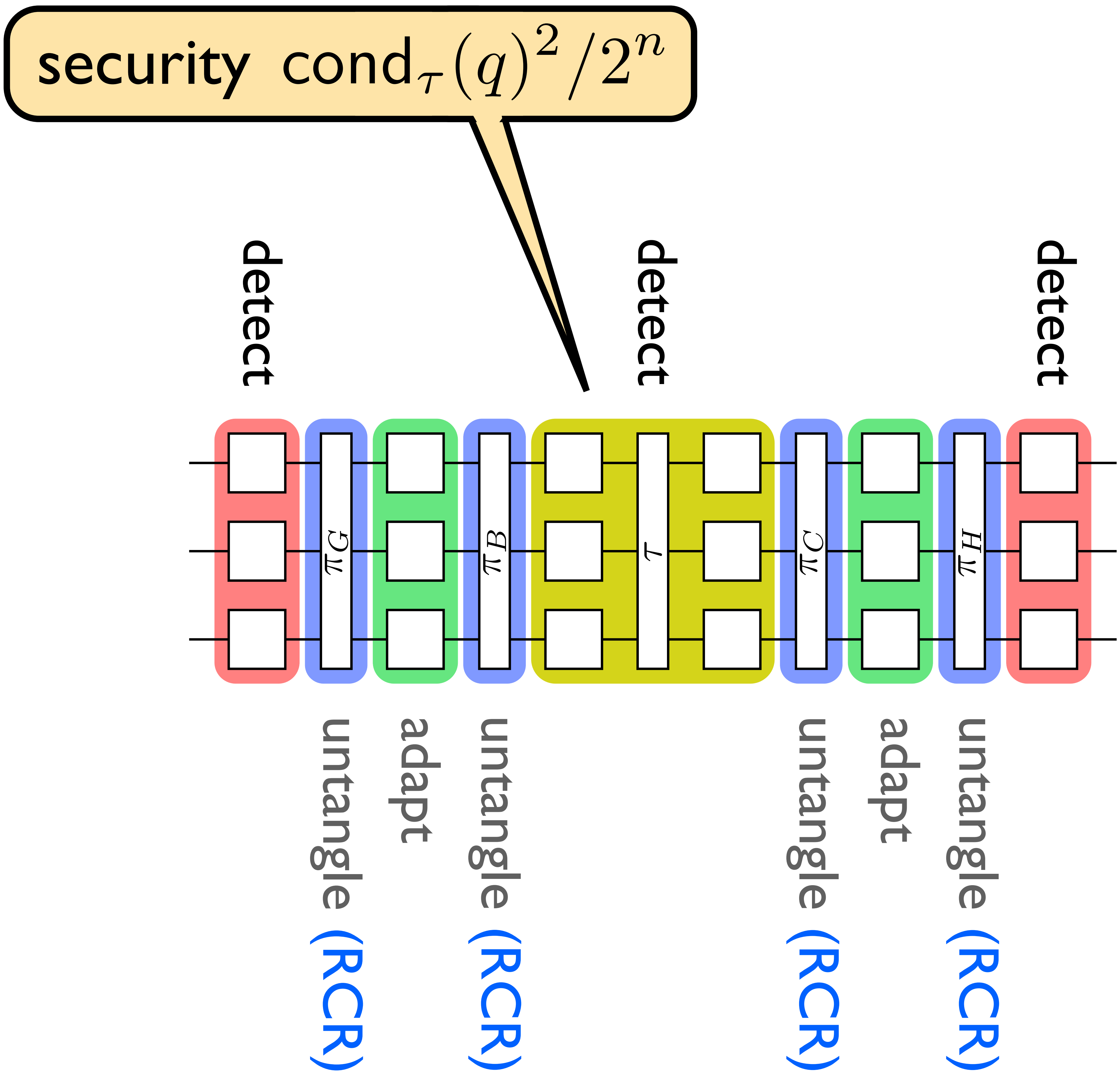
security $\text{cond}_\tau(q)^2/2^n$ "$\approx$" $q^2/2^n$

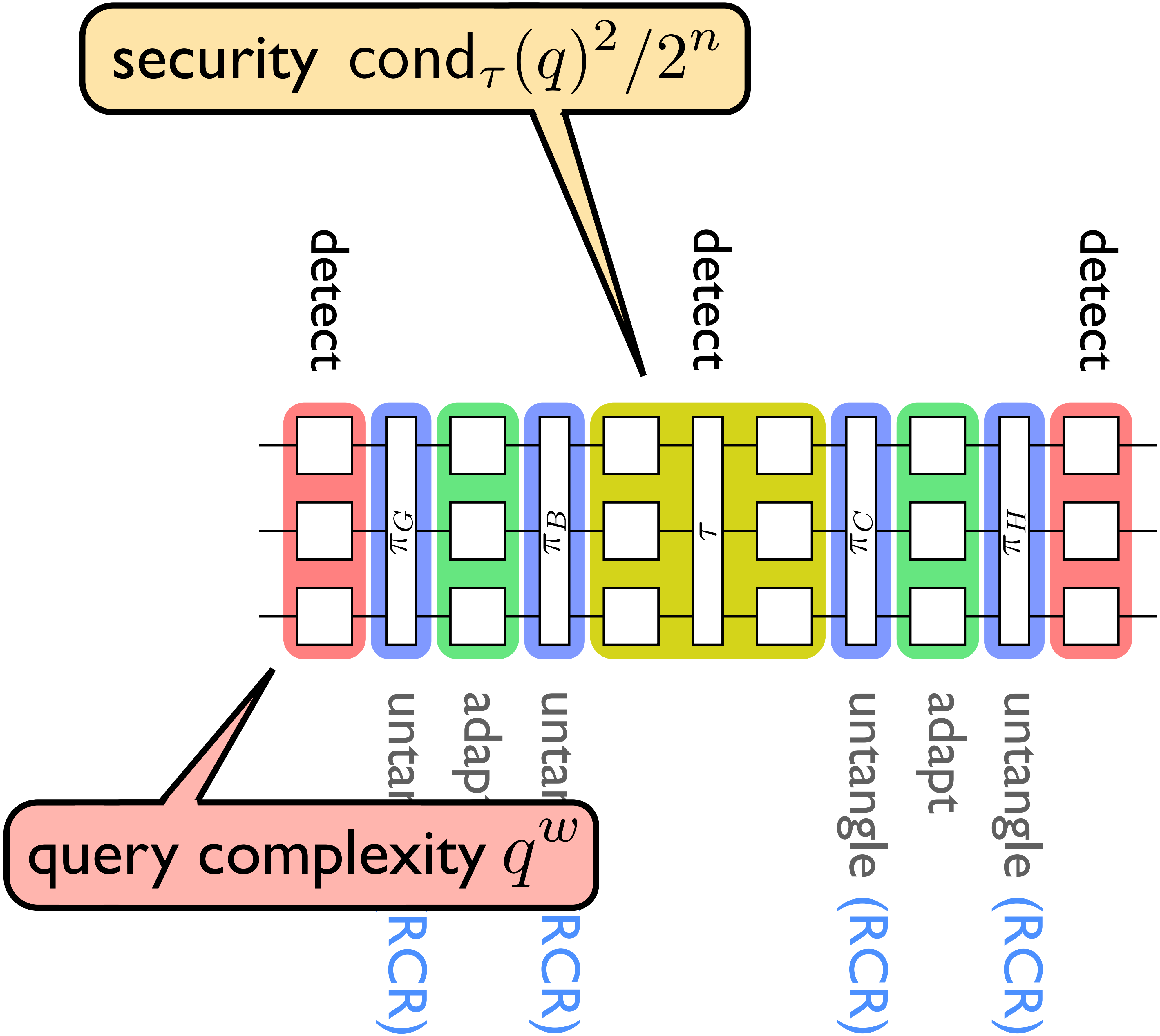


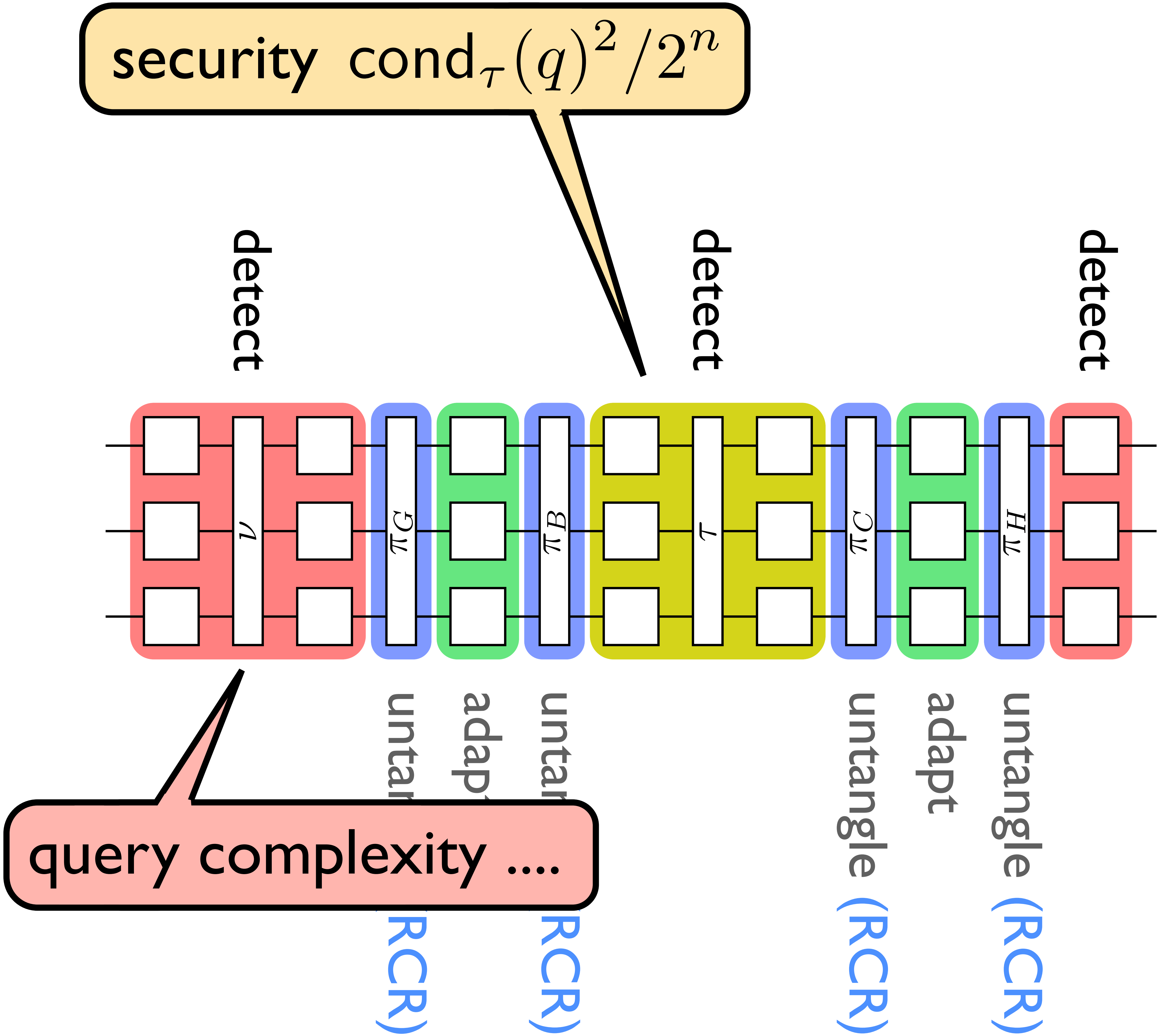

security $\text{cond}_\tau(q)^2/2^n$ "≈" $q^2/2^n$
detect
detect
detect
π_G
π_H
π_I
π_B
τ
π_C
π_J
π_K
π_H
RPR
untangle
RPR
adapt
RPR
untangle
RPR
RPR
untangle
RPR
adapt
RPR
untangle
RPR
Linear!

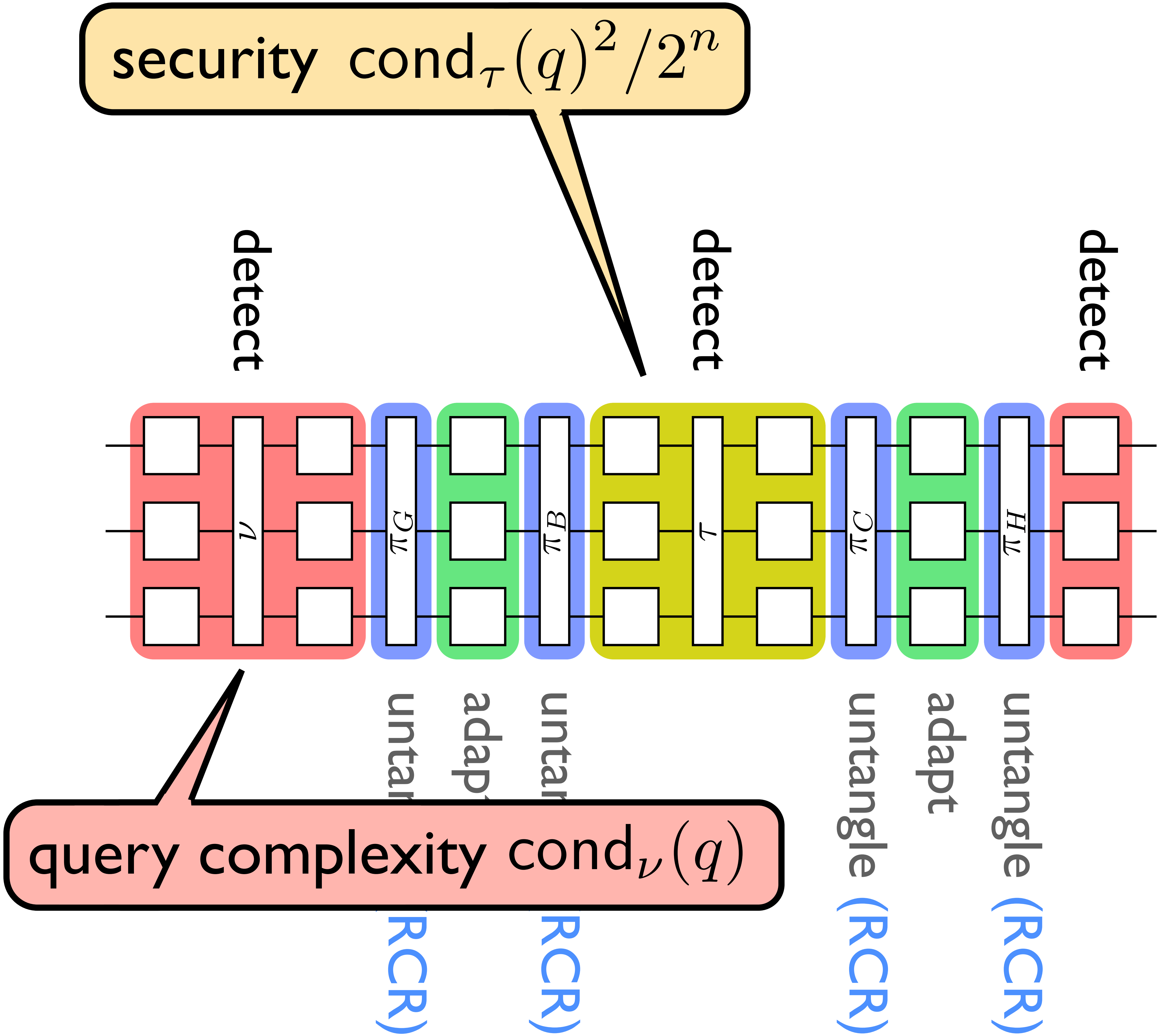

security $\text{cond}_\tau(q)^2/2^n$ "≈" $q^2/2^n$
Linear?
U choose!
Modularity!
detect
detect
detect
π_G
π_H
π_I
π_B
τ
π_C
π_J
π_K
π_H
RPR
untangle
RPR
adapt
RPR
untangle
RPR
RPR
untangle
RPR
adapt
RPR
untangle
RPR
Linear!


detect
detect
detect
πG
πB
τ
πC
πH
untangle (RCR)
adapt
untangle (RCR)
untangle (RCR)
adapt
untangle (RCR)


security $\mathrm{cond}_\tau(q)^2/2^n$
detect
detect
detect
π_G
π_B
τ
π_C
π_H
untangle (RCR)
adapt
untangle (RCR)
untangle (RCR)
adapt
untangle (RCR)


security $\text{cond}_\tau(q)^2/2^n$
detect
detect
detect
π_G
π_B
τ
π_C
π_H
untangle (RCR)
adapt
untangle (RCR)
query complexity q^w

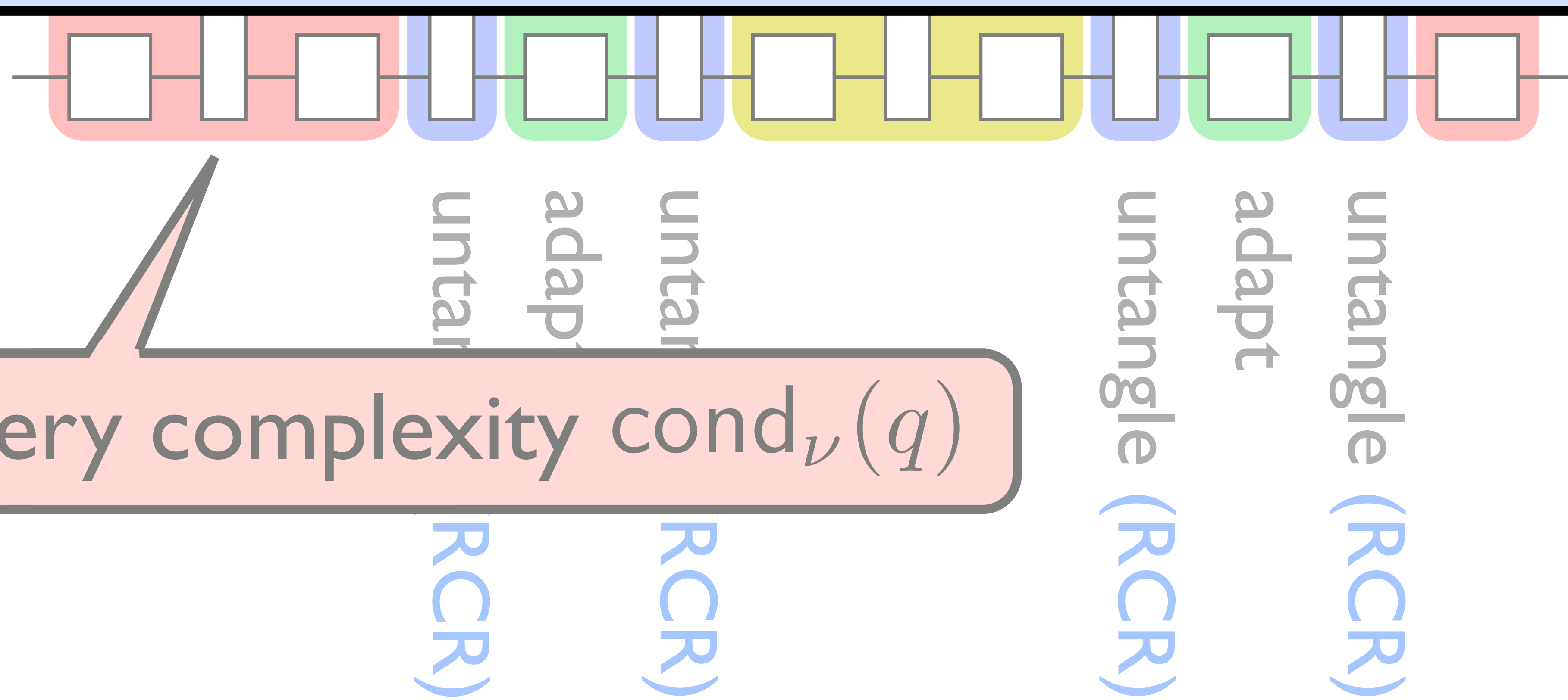

security cond$_\tau(q)^2/2^n$
query complexity ....
detect
detect
detect
ν
π_G
π_B
τ
π_C
π_H
untangle (RCR)
adapt
untangle (RCR)


security $\mathsf{cond}_\tau(q)^2/2^n$
detect
detect
detect
ν
π_G
π_B
τ
π_C
π_H
untangle (RCR)
adapt
untangle (RCR)
query complexity $\mathsf{cond}_\nu(q)$

security $\text{cond}_{\tau}(q)^2/2^n$

Altogether, the three boolean flags control...

- Security (XtraMiddleRnd)
- Query Complexity (XtraOuterRnd)
- Linearity of Untangle zones (XtraUntangleRnds)



query complexity $\text{cond}_{\nu}(q)$

Domain Extension: Our Work vs Previous

	RP $\to$ RO $\to$ RP via 8-round Feistel	CD length 5 (explicit)	CD length 7 (existential)
SECURITY	$q^8/2^n$	$q^4/2^n$	$q^2/2^n$
NUM CALLS TO RP	16	10	14
QUERY COMPLEXITY	q^4	q^4	q
SIM COMPLEXITY	q^4	q^4	q^2

$$(w = 2)$$