Stam's Conjecture and Threshold Phenomena in Collision Resistance

John Steinberger[1], Xiaoming Sun, Zhe Yang

ITCS, Tsinghua University

August 21, 2012

[1]Supported by the National Basic Research Program of China Grant 2011CBA00300, 2011CBA00301, the National Natural Science Foundation of China Grant 61033001, 61061130540, 61073174, and by NSF grant 0994380.

The Ideal Primitive Model (IPM)

- Used for proving the security of hash functions and compression functions that are based on some smaller primitive.

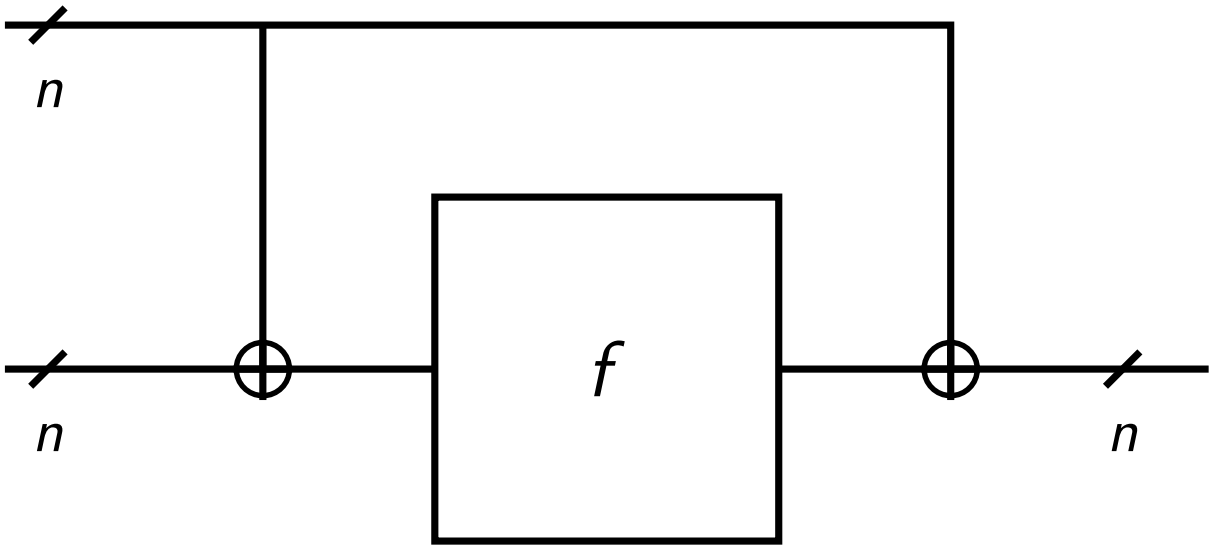
The Ideal Primitive Model (IPM)

- Used for proving the security of hash functions and compression functions that are based on some smaller primitive.
- An information-theoretic adversary is given oracle access to the primitive, which is modeled as "ideal" or "perfectly random".

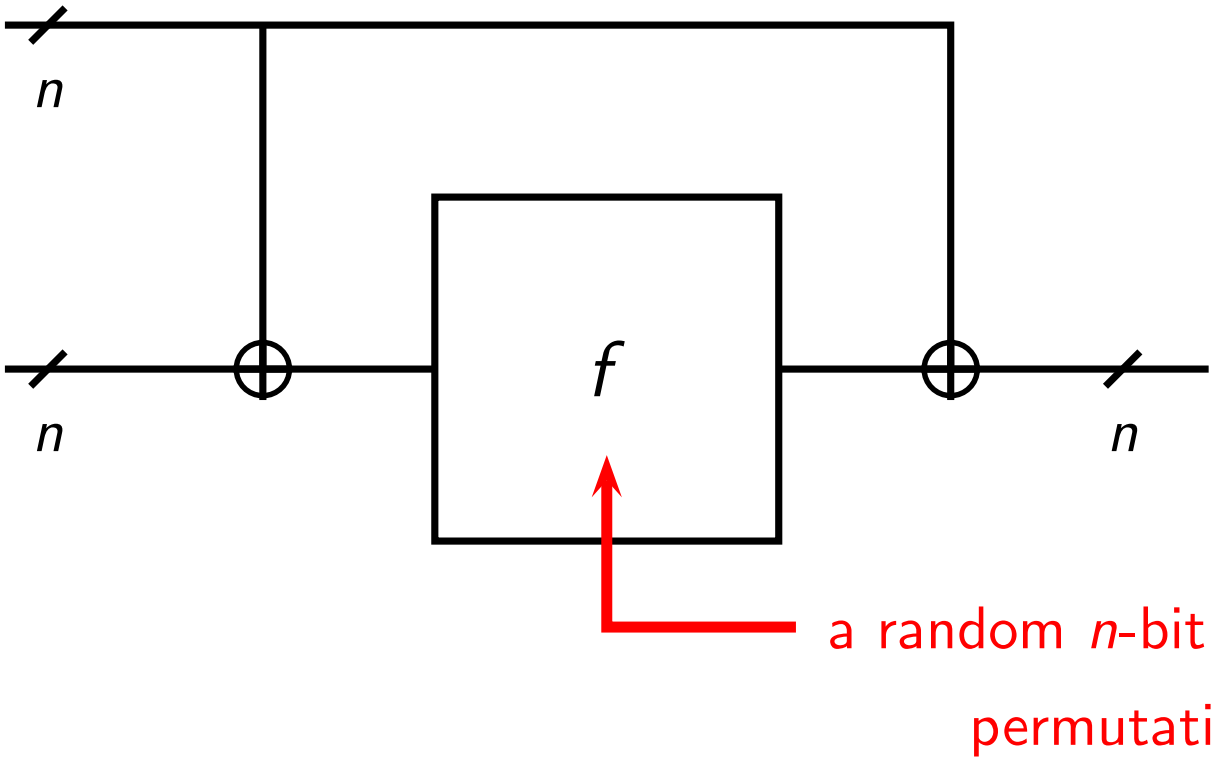
The Ideal Primitive Model (IPM)

- Used for proving the security of hash functions and compression functions that are based on some smaller primitive.
- An information-theoretic adversary is given oracle access to the primitive, which is modeled as "ideal" or "perfectly random".
- The only obstacle to the adversary's success is the randomness of the query responses.

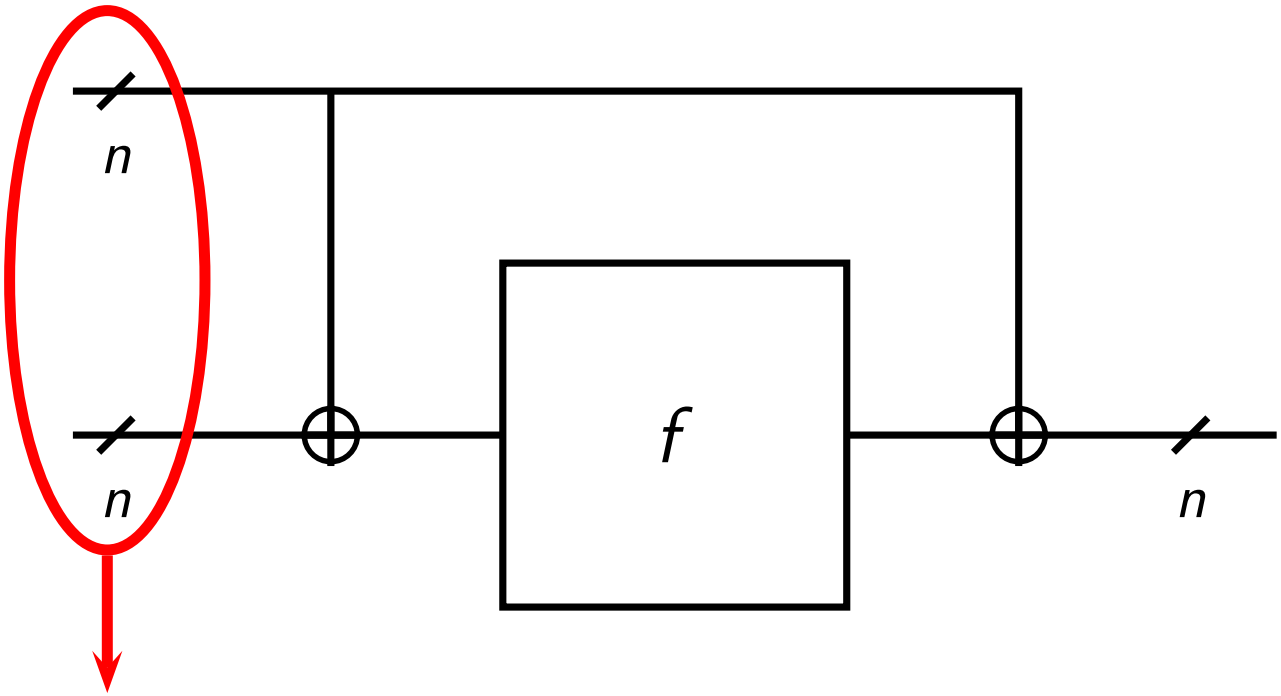
The limits of IPM security



The limits of IPM security

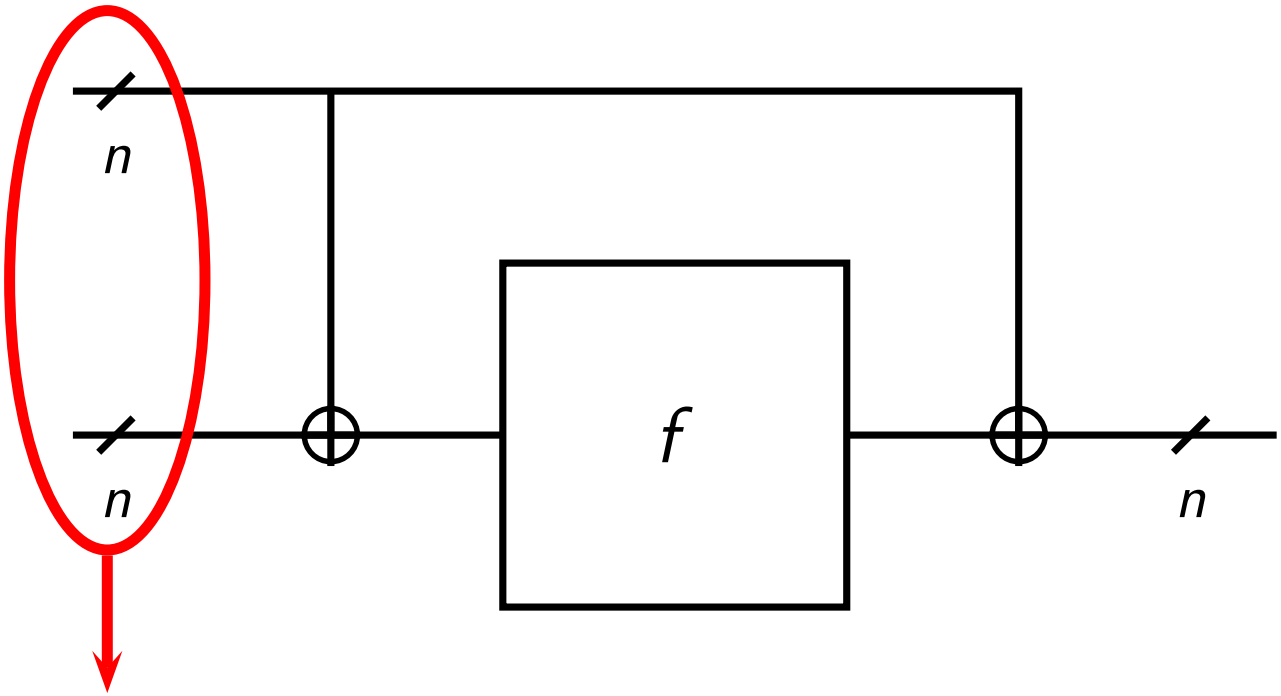


The limits of IPM security



Domain of the compression function is $\{0,1\}^{2n}$

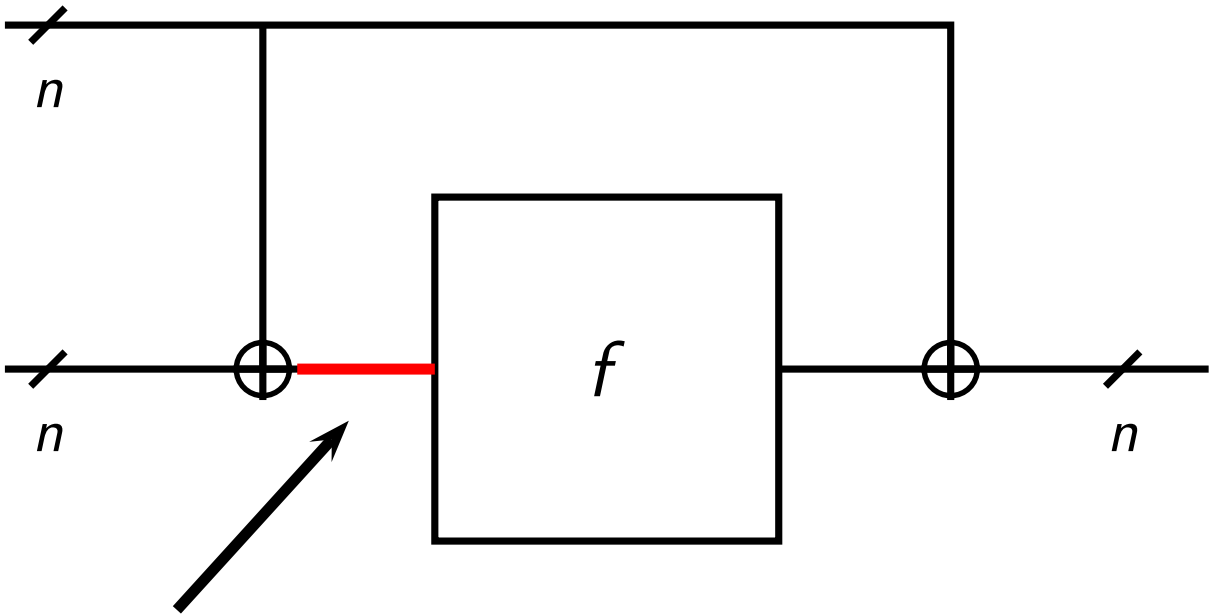
The limits of IPM security



Domain of the compression function is $\{0,1\}^{2n}$

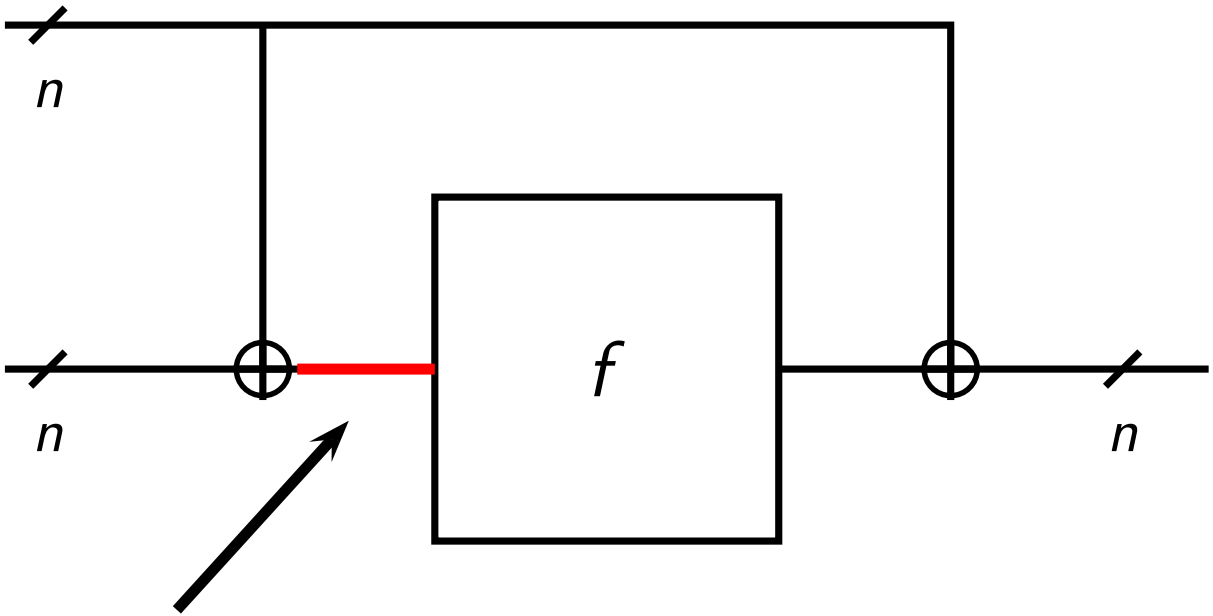
...the size of the domain is 2^{2n}

The limits of IPM security



The adversary learns 2^n inputs by making one query to f...

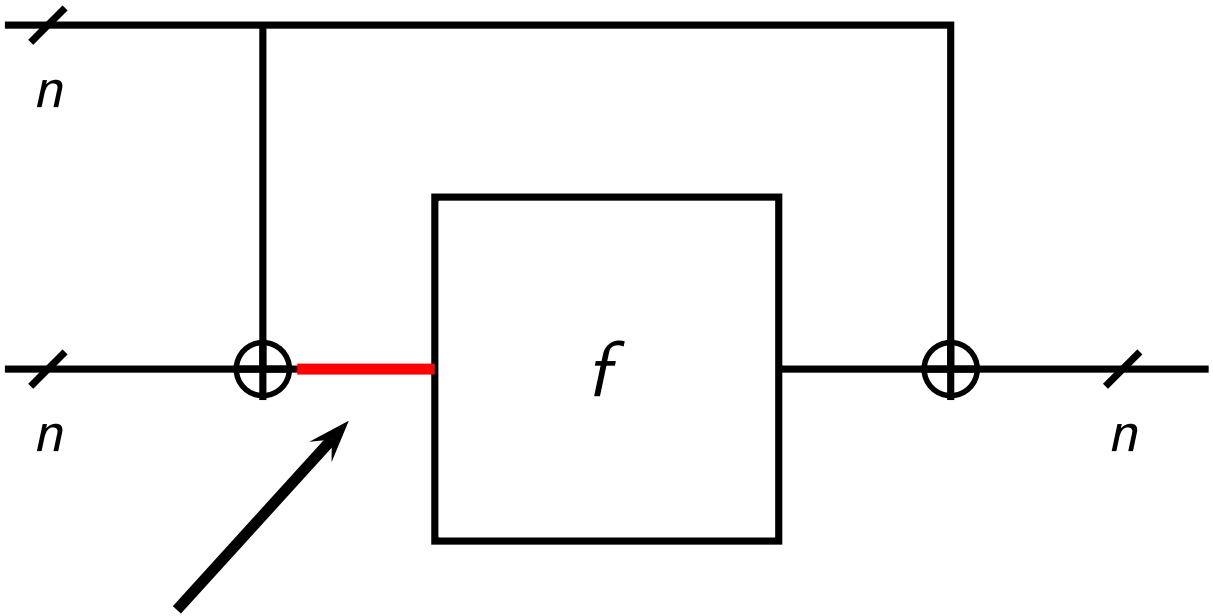
The limits of IPM security



The adversary learns 2^n inputs by making one query to f...

...and learns $2 \cdot 2^n$ inputs by making two queries to f...

The limits of IPM security

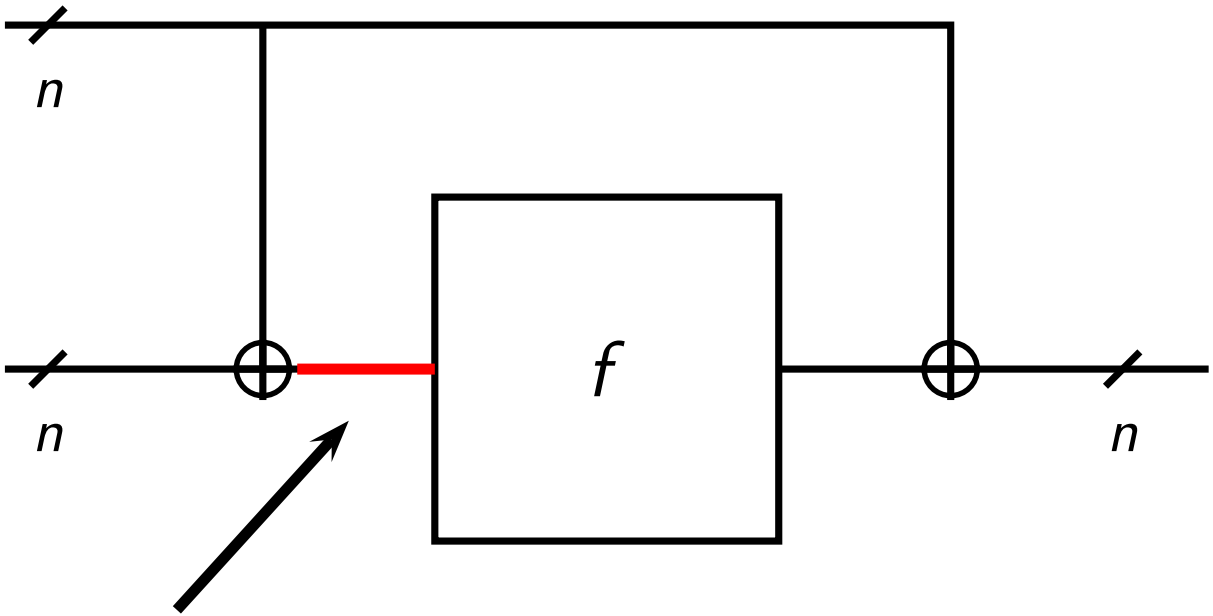


The adversary learns 2^n inputs by making one query to f...

...and learns $2 \cdot 2^n$ inputs by making two queries to f...

...but there are only 2^n outputs total...

The limits of IPM security



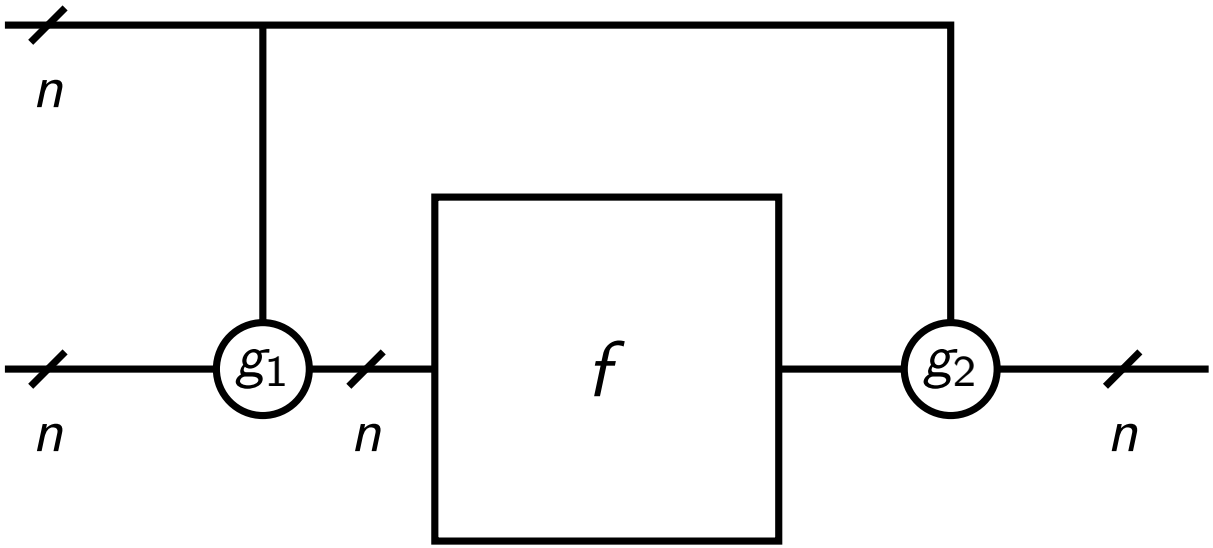
The adversary learns 2^n inputs by making one query to f...

...and learns $2 \cdot 2^n$ inputs by making two queries to f...

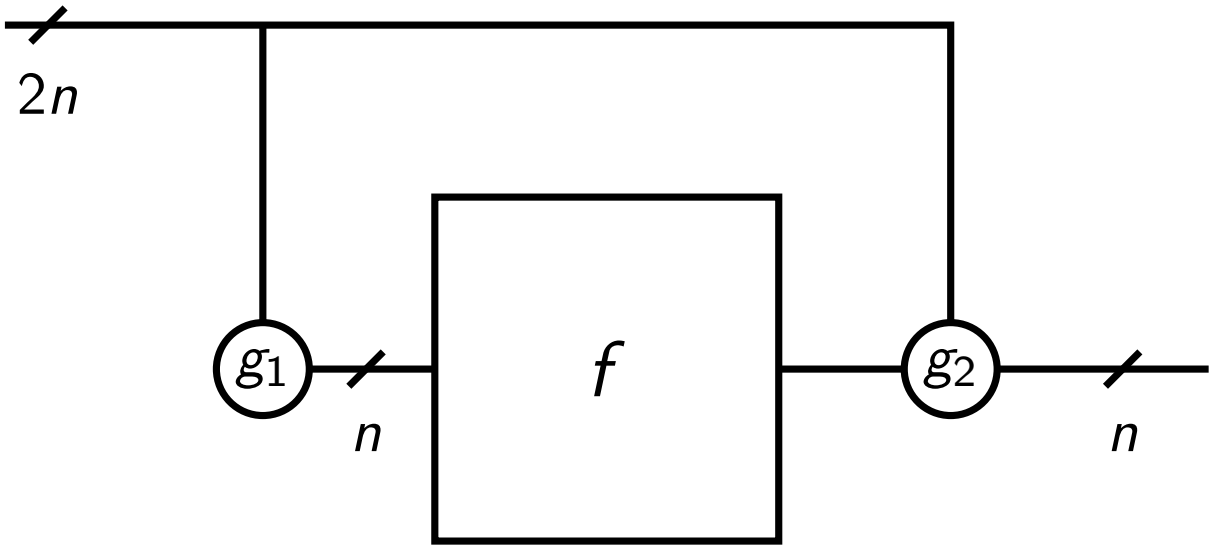
...but there are only 2^n outputs total...

...so the adversary breaks collision security in two queries.

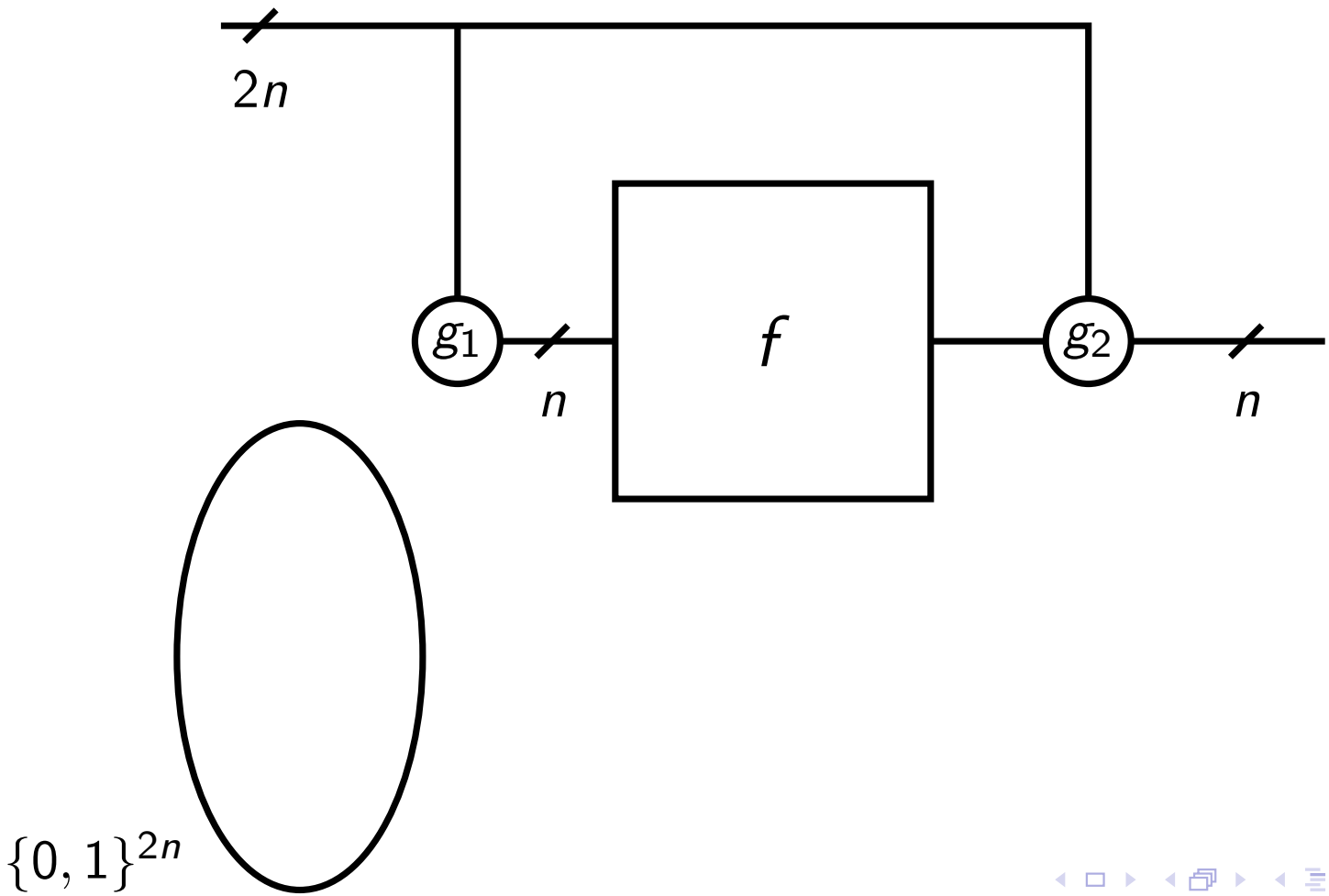
The limits of IPM security



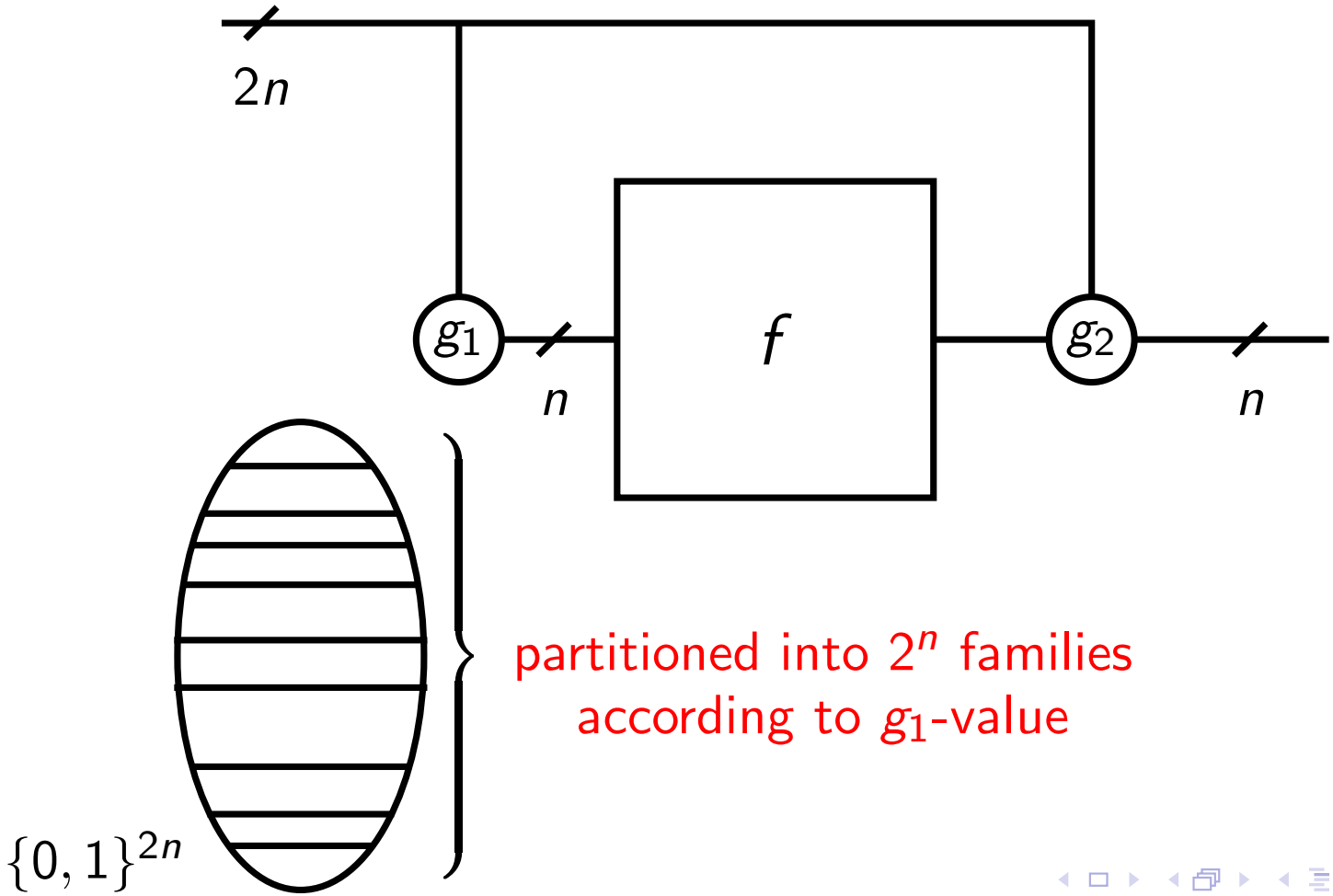
The limits of IPM security



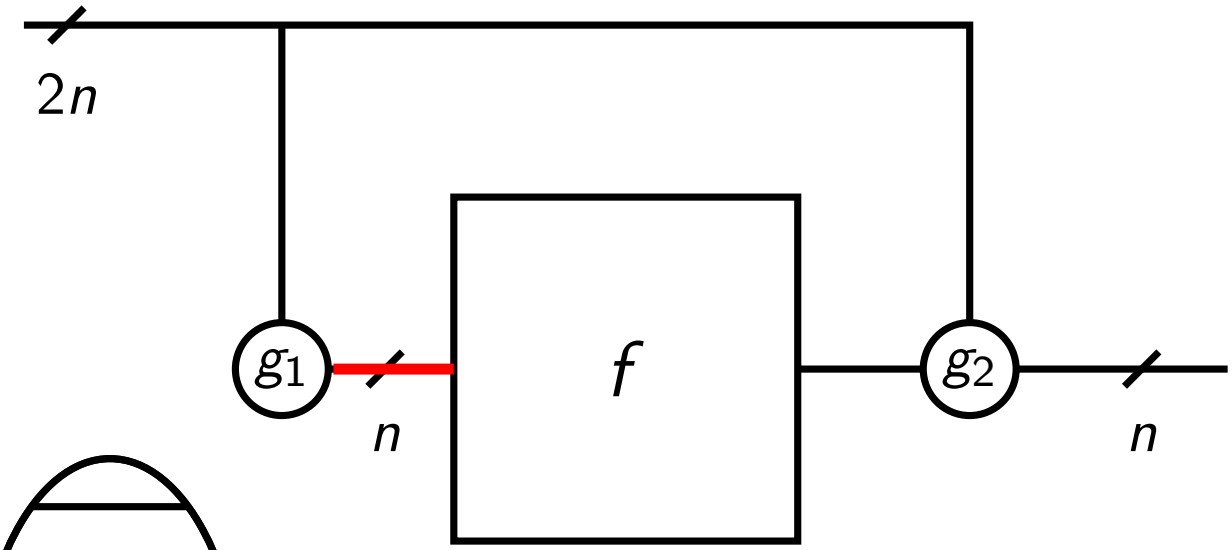
The limits of IPM security

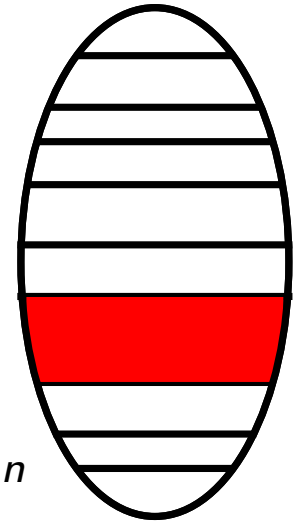


The limits of IPM security



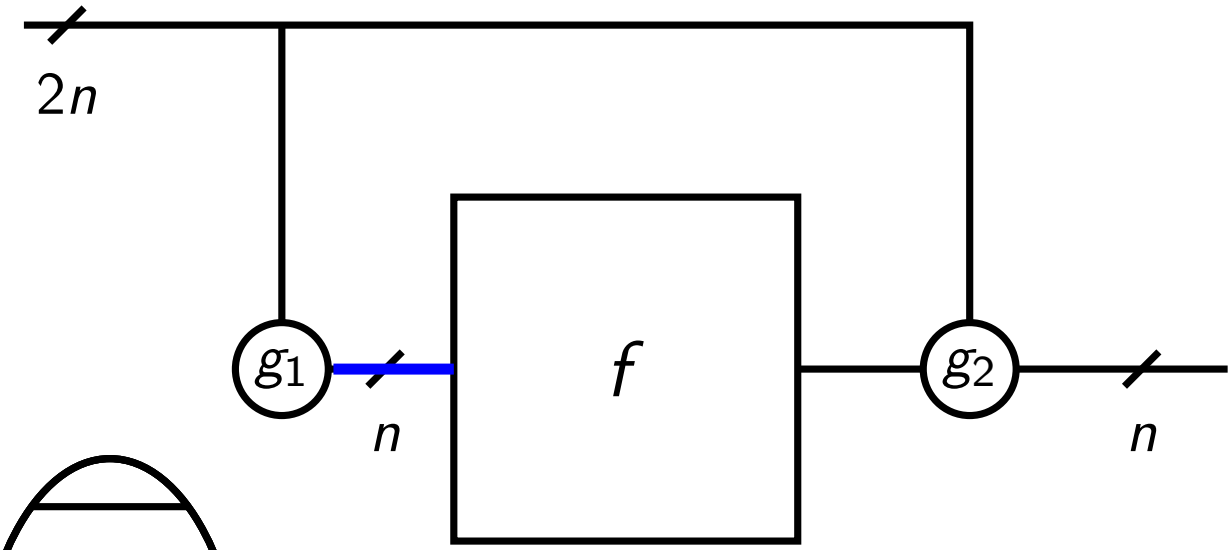
The limits of IPM security

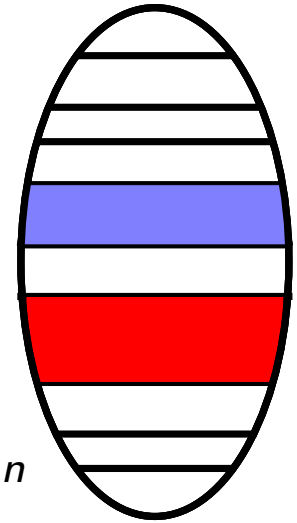




By making 1 greedy query the adversary can learn the output value for $\geq 2^n$ inputs

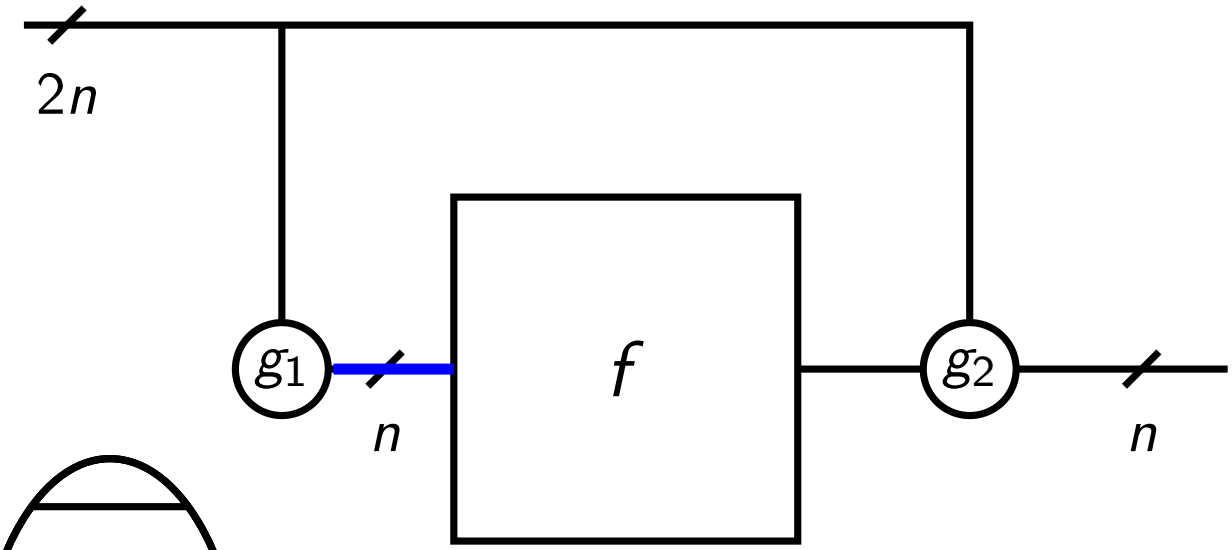
The limits of IPM security

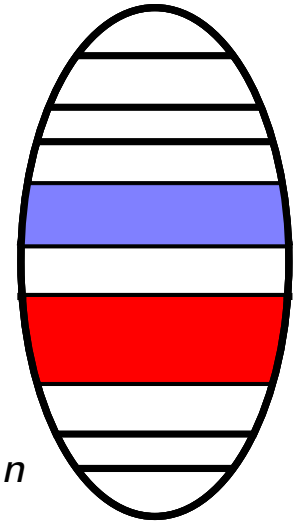




By making two greedy queries the adversary can learn the output value for $\geq 2 \cdot 2^n$ inputs

The limits of IPM security



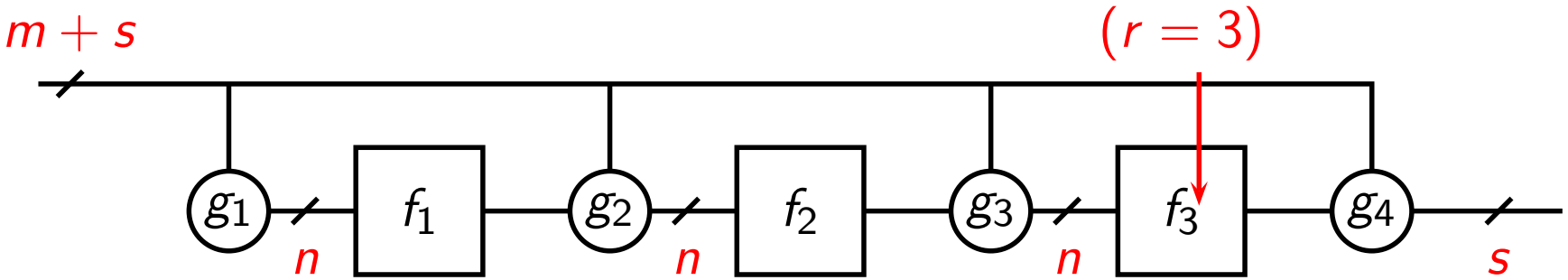


By making two greedy queries the adversary can learn the output value for $\geq 2 \cdot 2^n$ inputs

Collision security $= 2$ queries

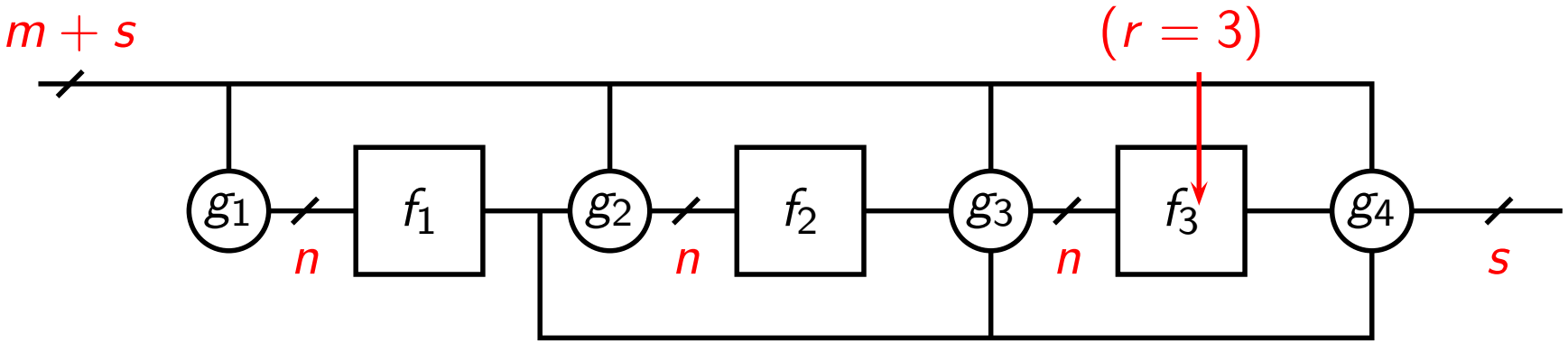
The General Model

An $(m+s)$-bit to s-bit compression function making r calls to a primitive f of n-bit input.



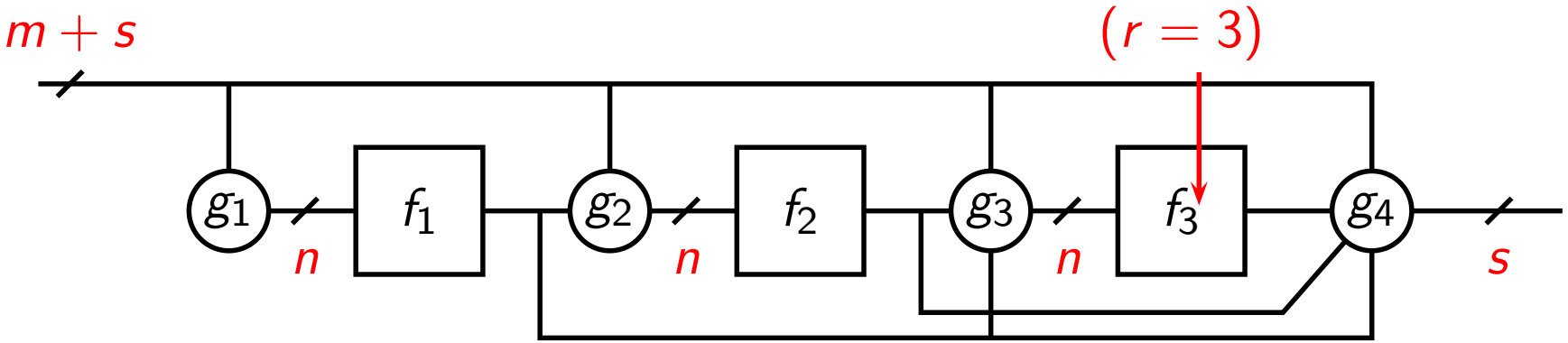
The General Model

An $(m+s)$-bit to s-bit compression function making r calls to a primitive f of n-bit input.



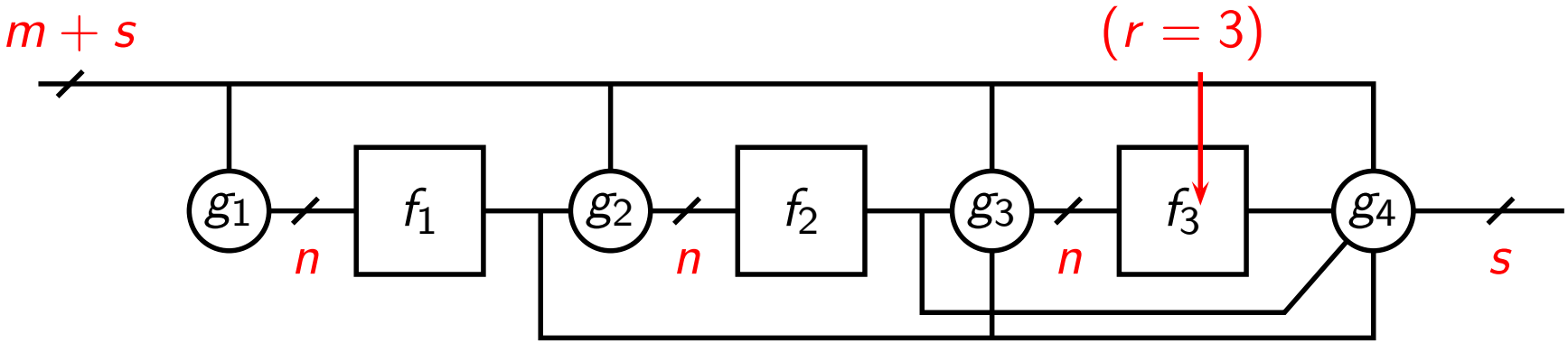
The General Model

An $(m+s)$-bit to s-bit compression function making r calls to a primitive f of n-bit input.



The General Model

An $(m+s)$-bit to s-bit compression function making r calls to a primitive f of n-bit input.



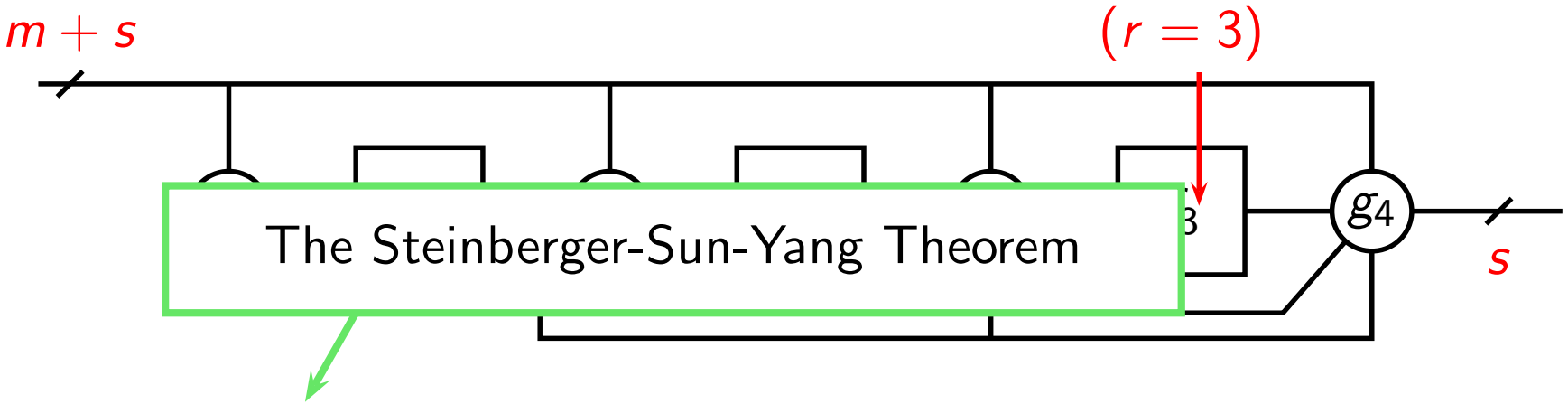
Stam's conjecture: The collision security of such a compression function is upper bounded by

$$r \cdot \min(2^{(nr-m)/(r+1)}, 2^{s/2})$$

(up to a possible constant factor).

The General Model

An $(m+s)$-bit to s-bit compression function making r calls to a primitive f of n-bit input.

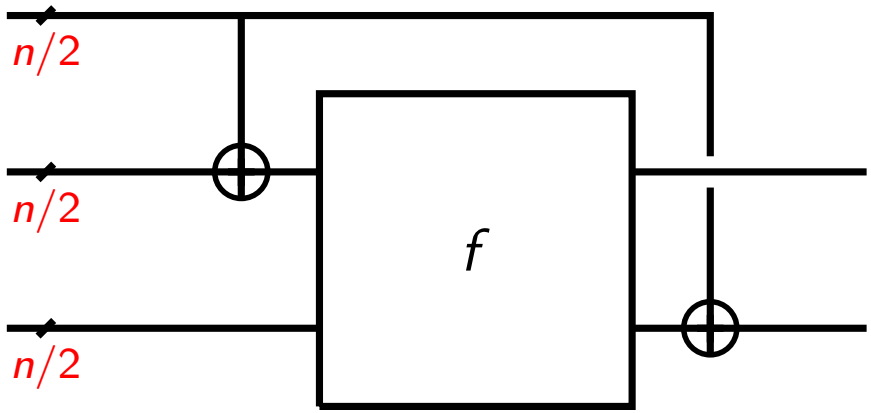


~~**Stam's conjecture:**~~ The collision security of such a compression function is upper bounded by

$$r \cdot \min(2^{(nr-m)/(r+1)}, 2^{s/2})$$

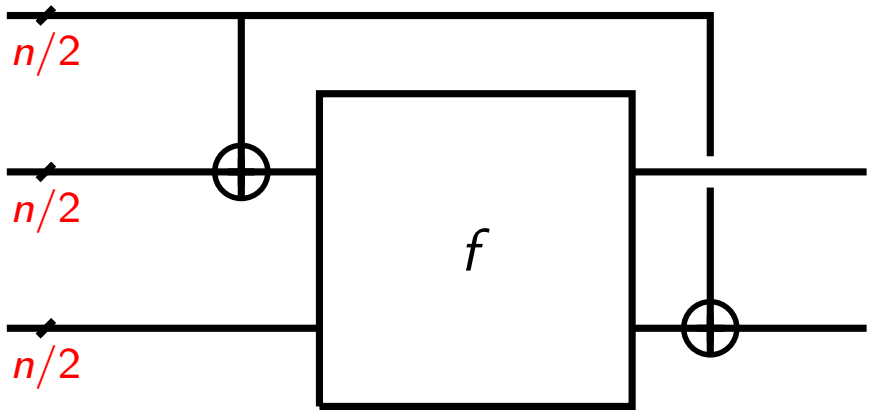
(up to a possible constant factor).

Example: the JH compression function



$s = n$ $\quad$ $m = n/2$ $\quad$ $r = 1$

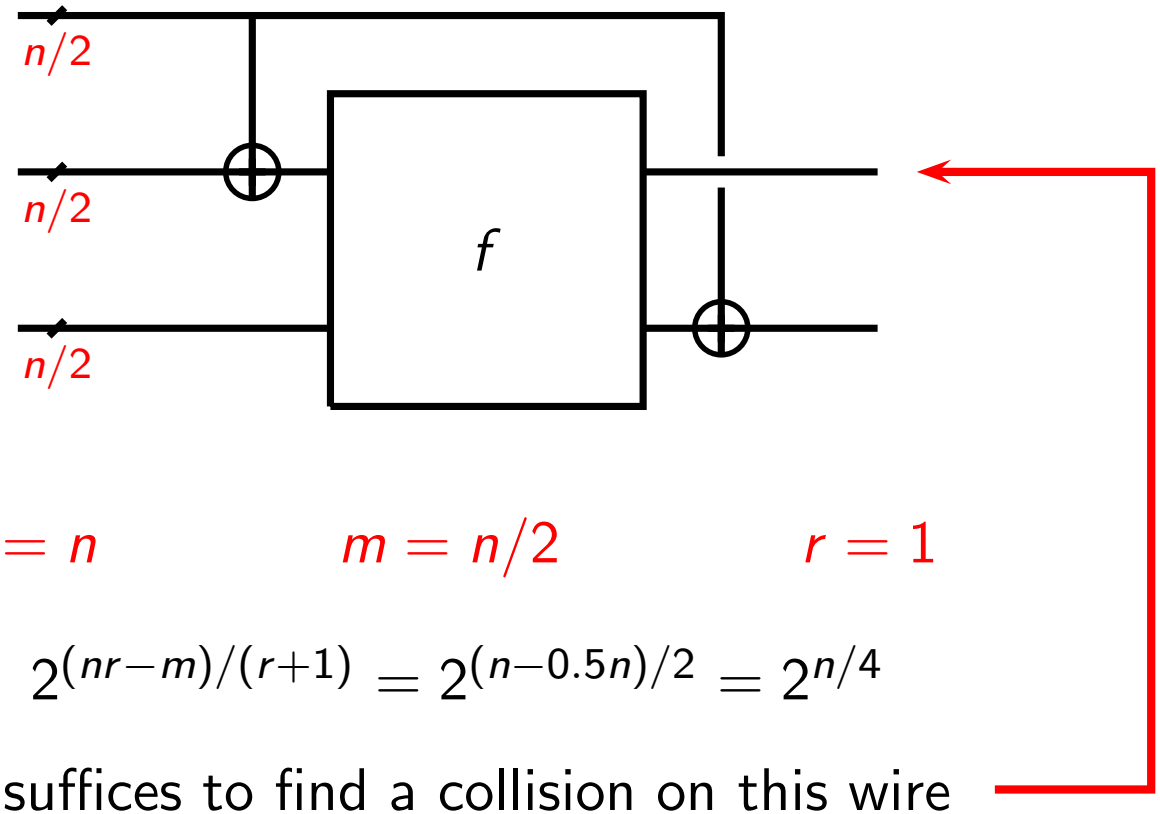
Example: the JH compression function



$s = n$ $\quad$ $m = n/2$ $\quad$ $r = 1$

$$2^{(nr-m)/(r+1)} = 2^{(n-0.5n)/2} = 2^{n/4}$$

Example: the JH compression function



$s = n$ $\quad$ $m = n/2$ $\quad$ $r = 1$

$$2^{(nr-m)/(r+1)} = 2^{(n-0.5n)/2} = 2^{n/4}$$

It suffices to find a collision on this wire

(Main Theorem)

With

$$q = O(1)r2^{(nr-m)/(r+1)}$$

queries one can obtain at least

$$2^{2(s/2-(nr-m)/(r+1))}$$

collisions with high probability.

(Main Theorem)

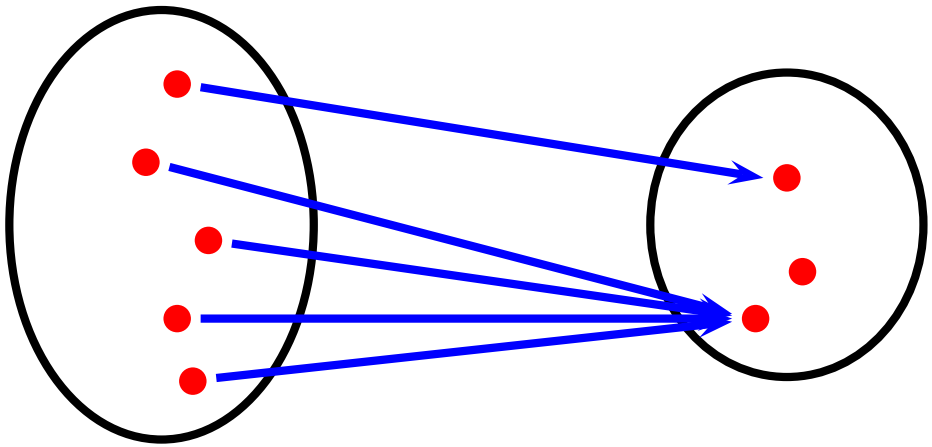
With

$$q = O(1)r2^{(nr-m)/(r+1)}$$

queries one can obtain at least

$$2^{2(s/2-(nr-m)/(r+1))}$$

collisions with high probability.



$= 4$ "collisions"

(Main Theorem)

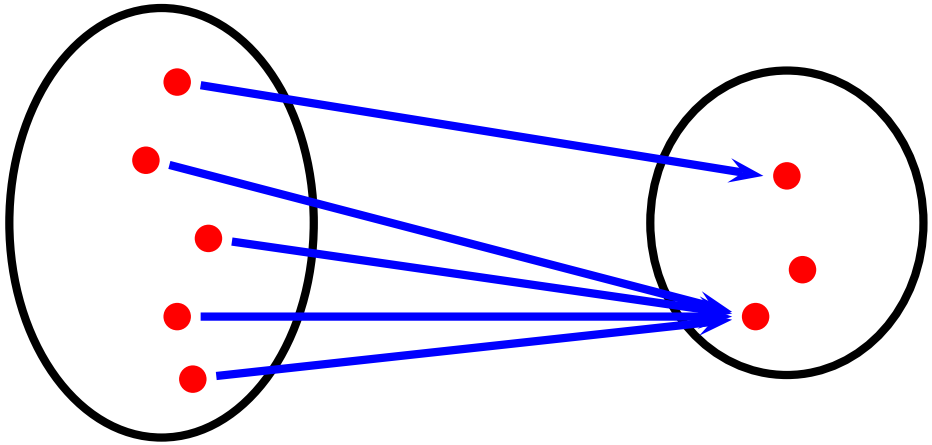
With

$$q = O(1)r2^{(nr-m)/(r+1)}$$

queries one can obtain at least

$$2^{2(s/2-(nr-m)/(r+1))}$$

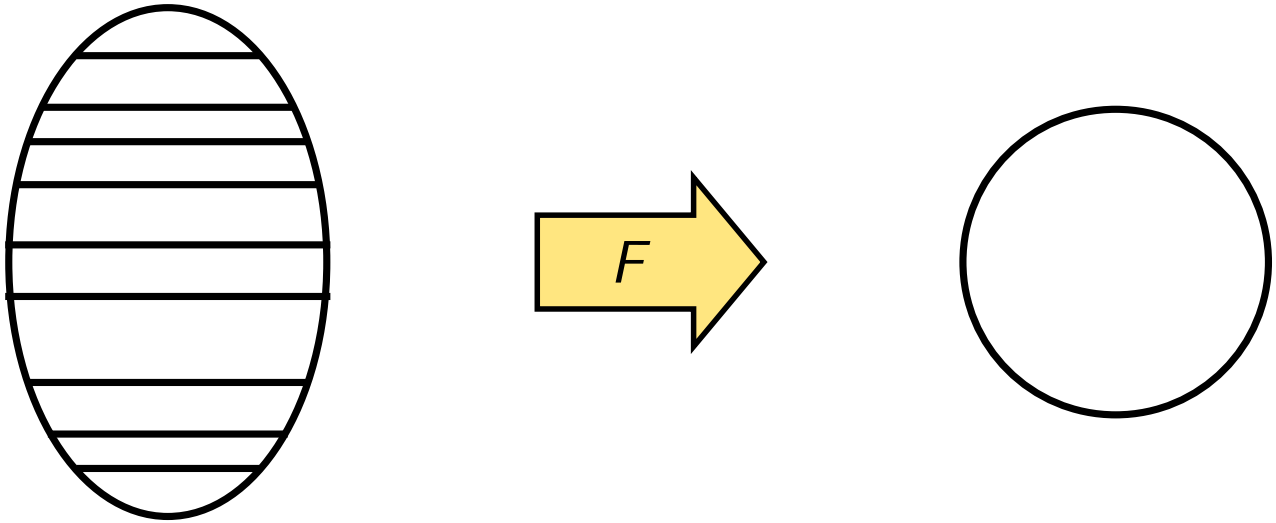
collisions with high probability.



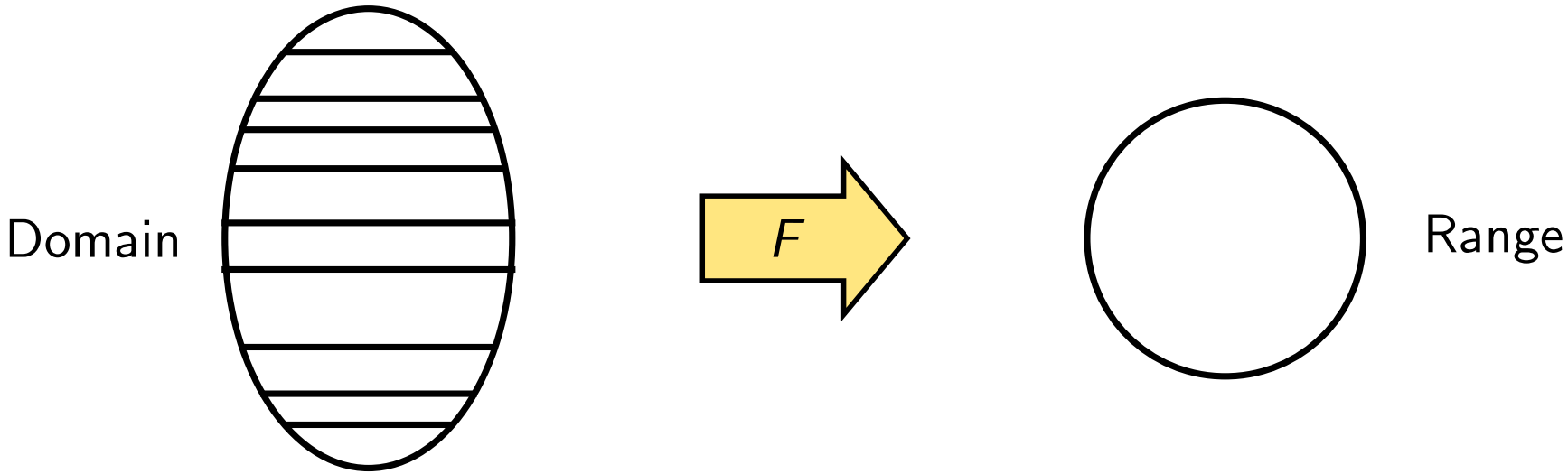
$= 4$ "collisions"

$\geq |Domain| - |Range|$

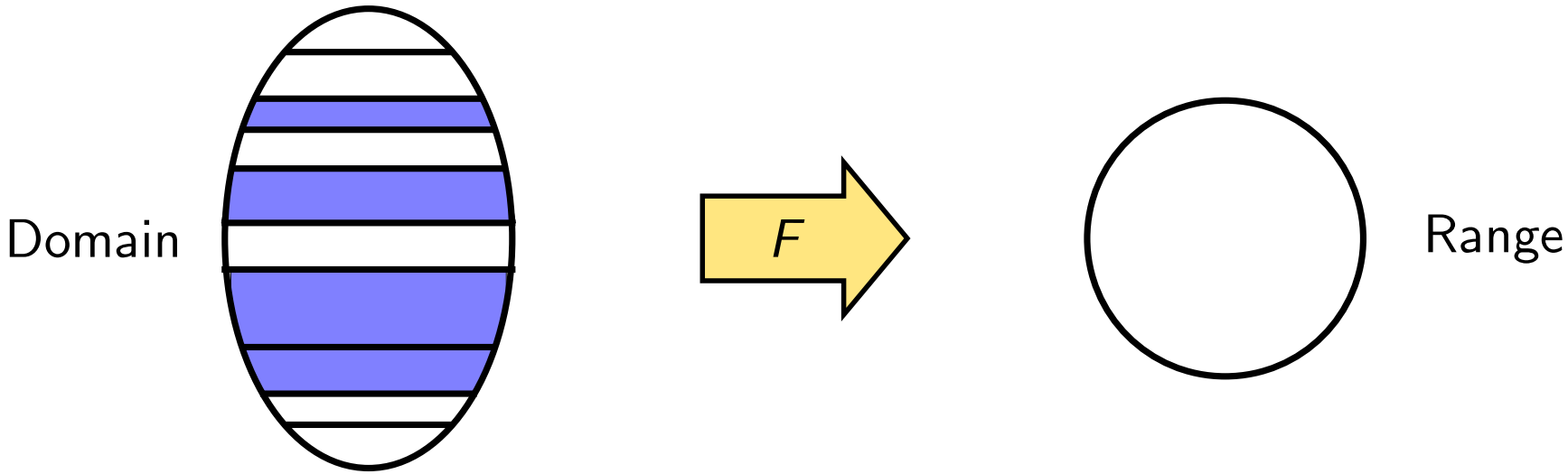
Key Lemma



Key Lemma

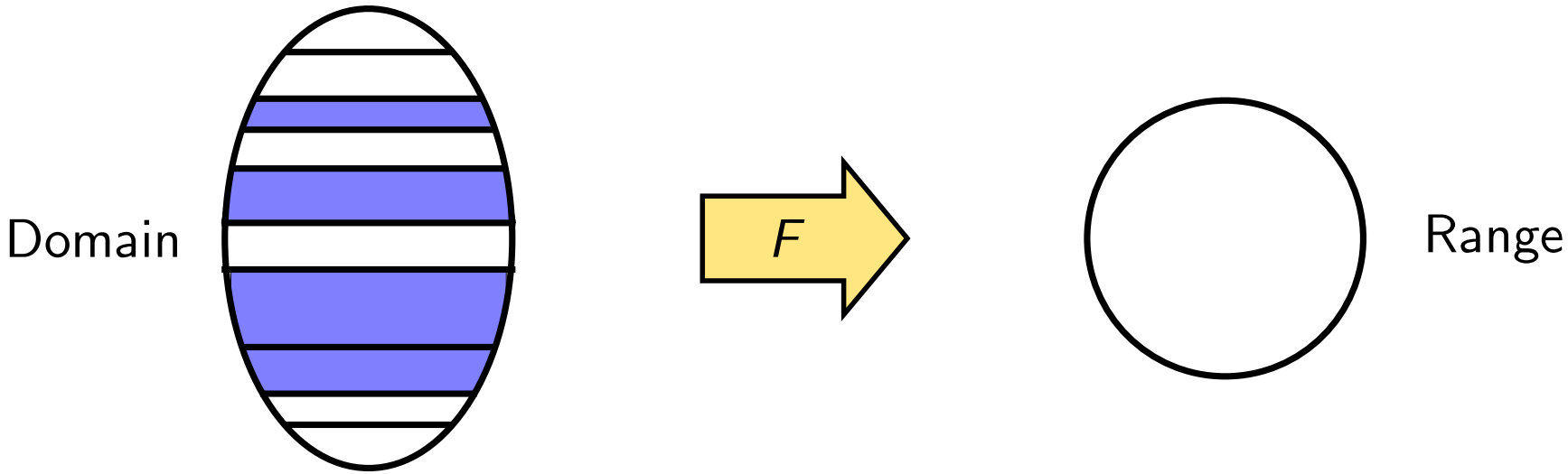


Key Lemma



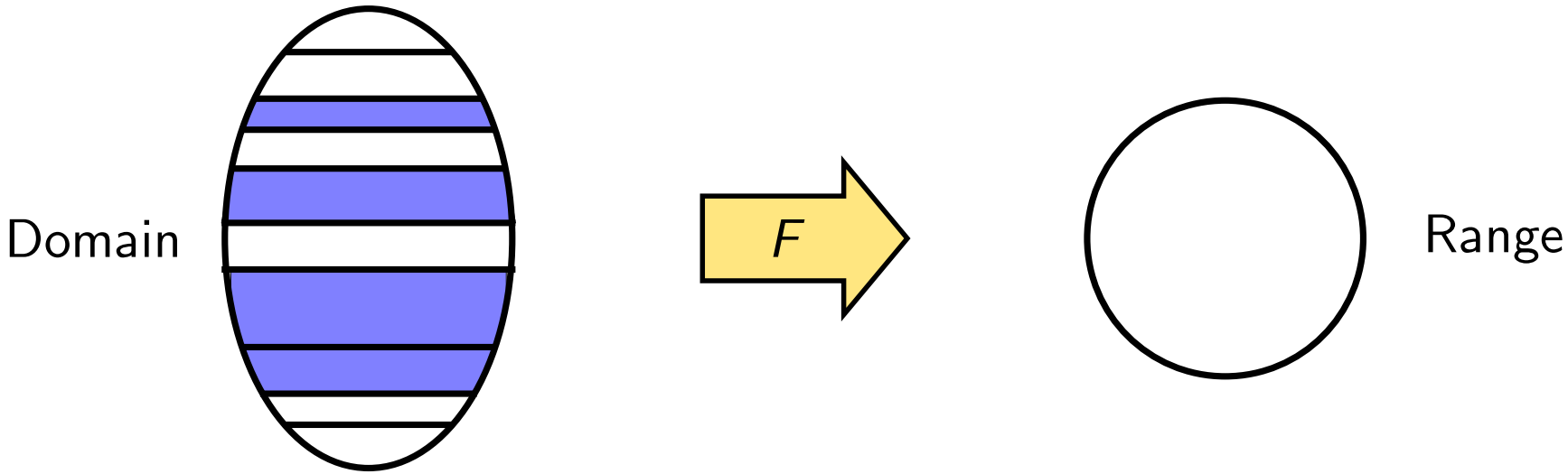
Basic idea: That if we restrict the domain by selecting a certain number of layers randomly, the number of leftover colliding inputs is close to its expectation with high probability.

Key Lemma



C: Original number of colliding inputs in F

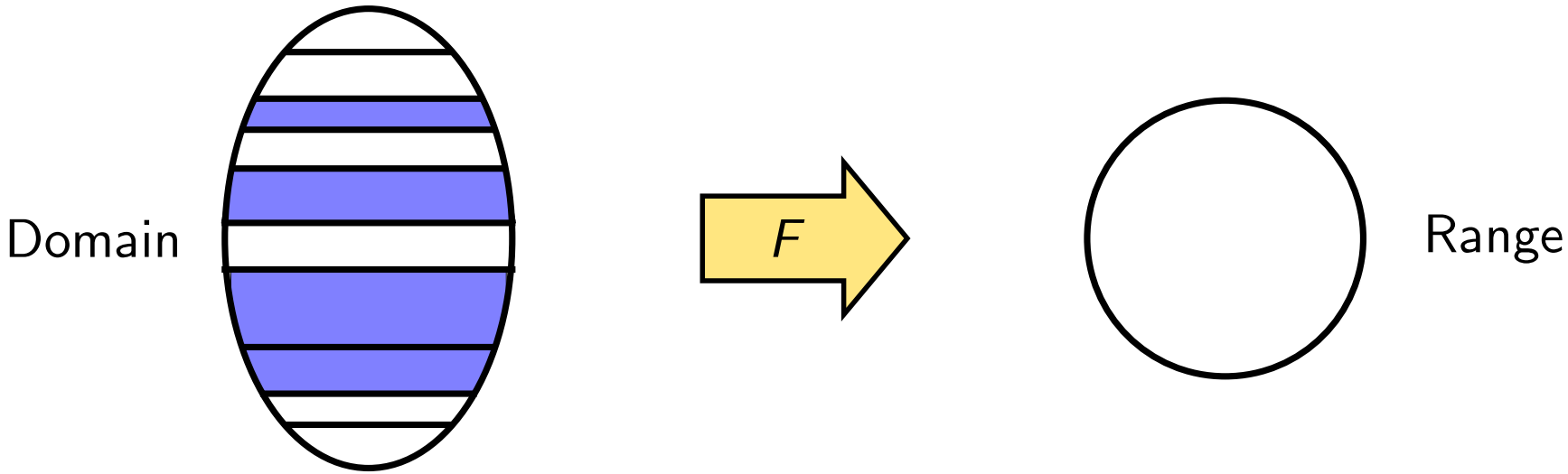
Key Lemma



C: Original number of colliding inputs in F

k: Number of layers in egg

Key Lemma

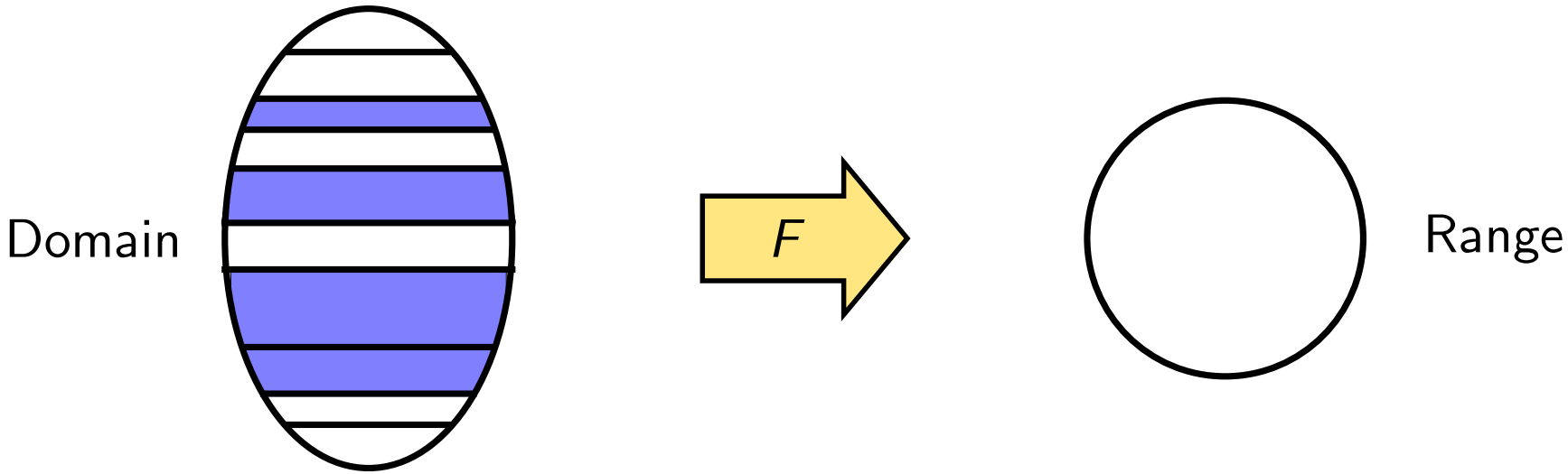


C: Original number of colliding inputs in F

k: Number of layers in egg

q: Number of layers to be randomly selected

Key Lemma



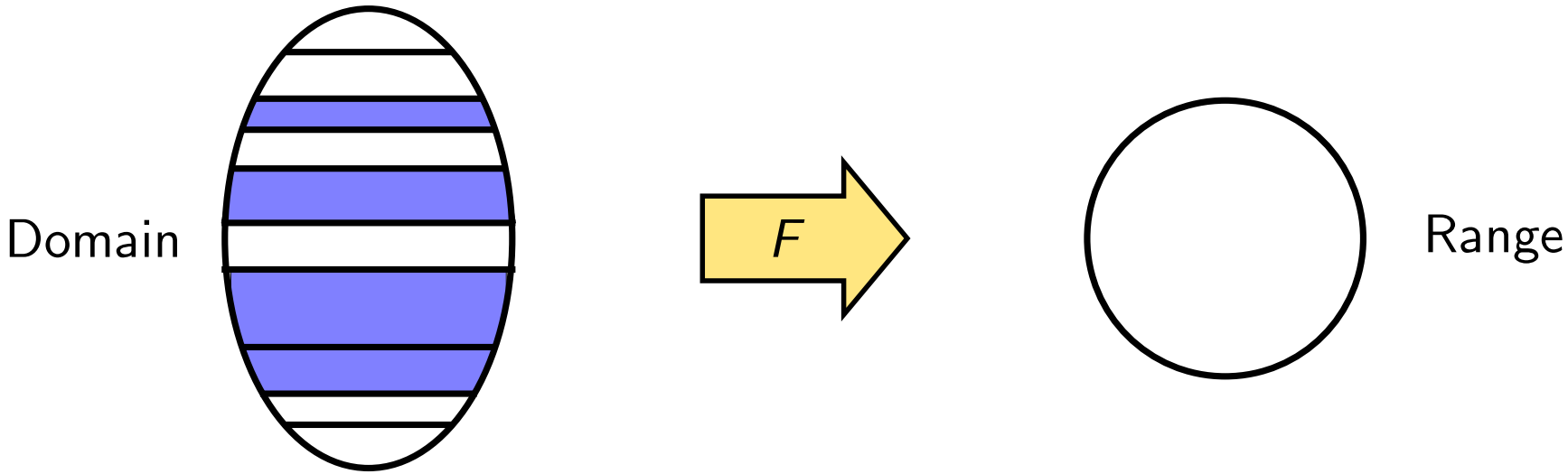
C: Original number of colliding inputs in F

k: Number of layers in egg

q: Number of layers to be randomly selected

$p = q/k$: The probability of a layer being selected

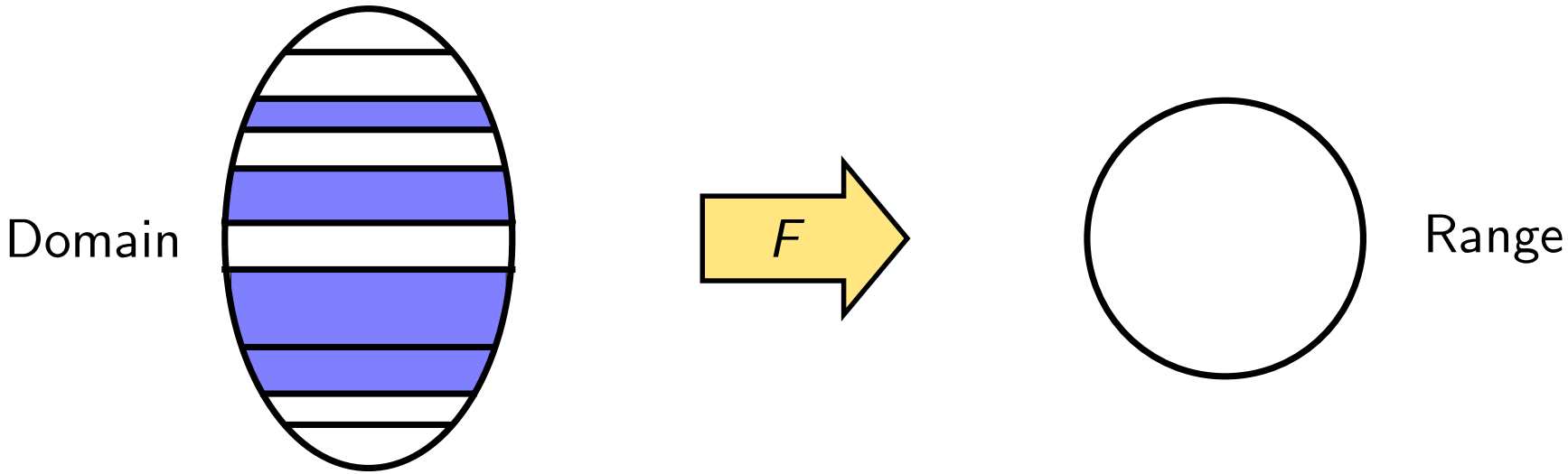
Key Lemma



C: Original number of colliding inputs in F

p: The probability of a layer being selected

Key Lemma

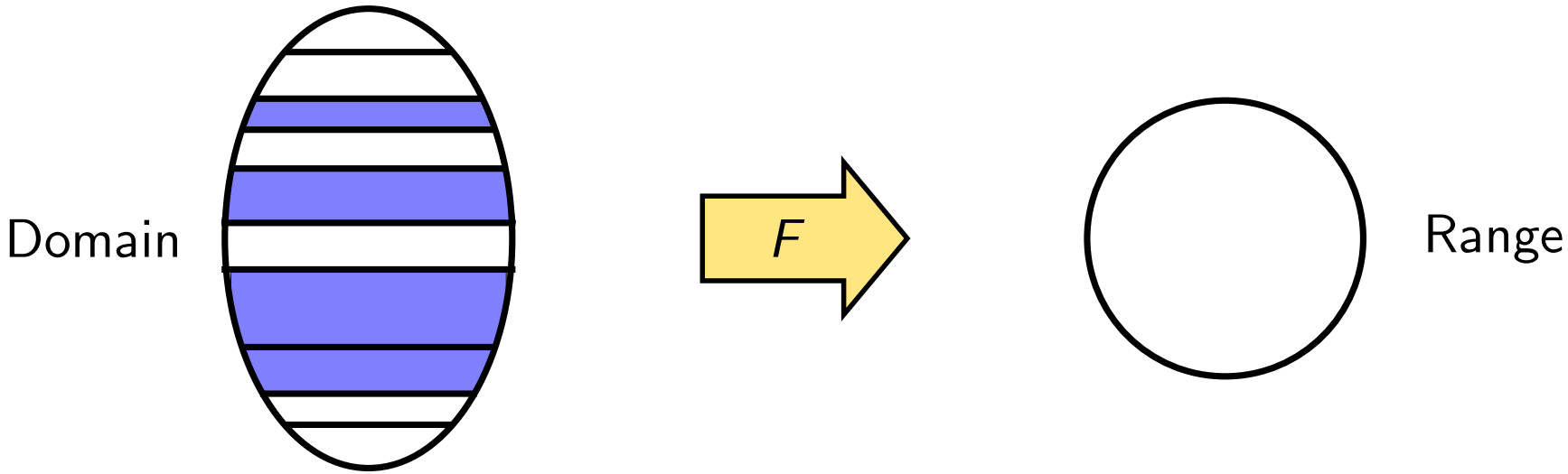


C: Original number of colliding inputs in F

p: The probability of a layer being selected

Back-of-the-envelope expected number of leftover collisions: $p^2 C$

Key Lemma



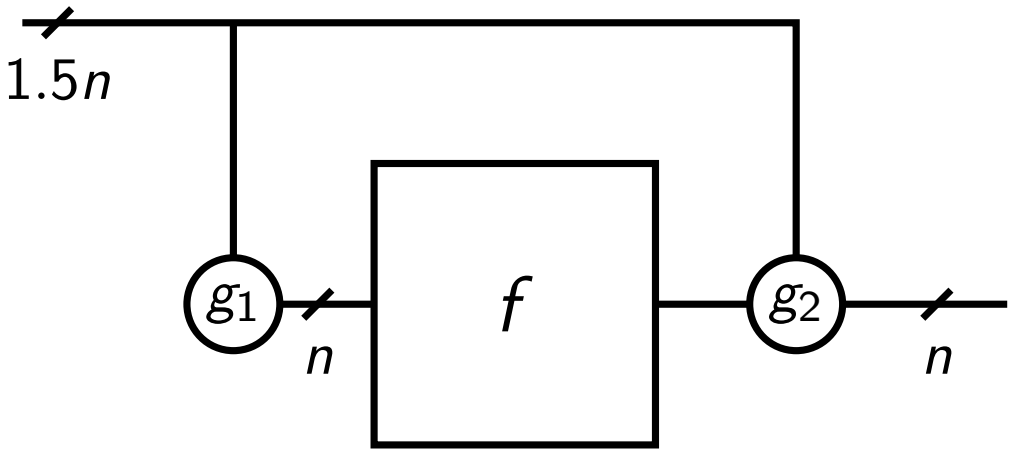
C: Original number of colliding inputs in F

p: The probability of a layer being selected

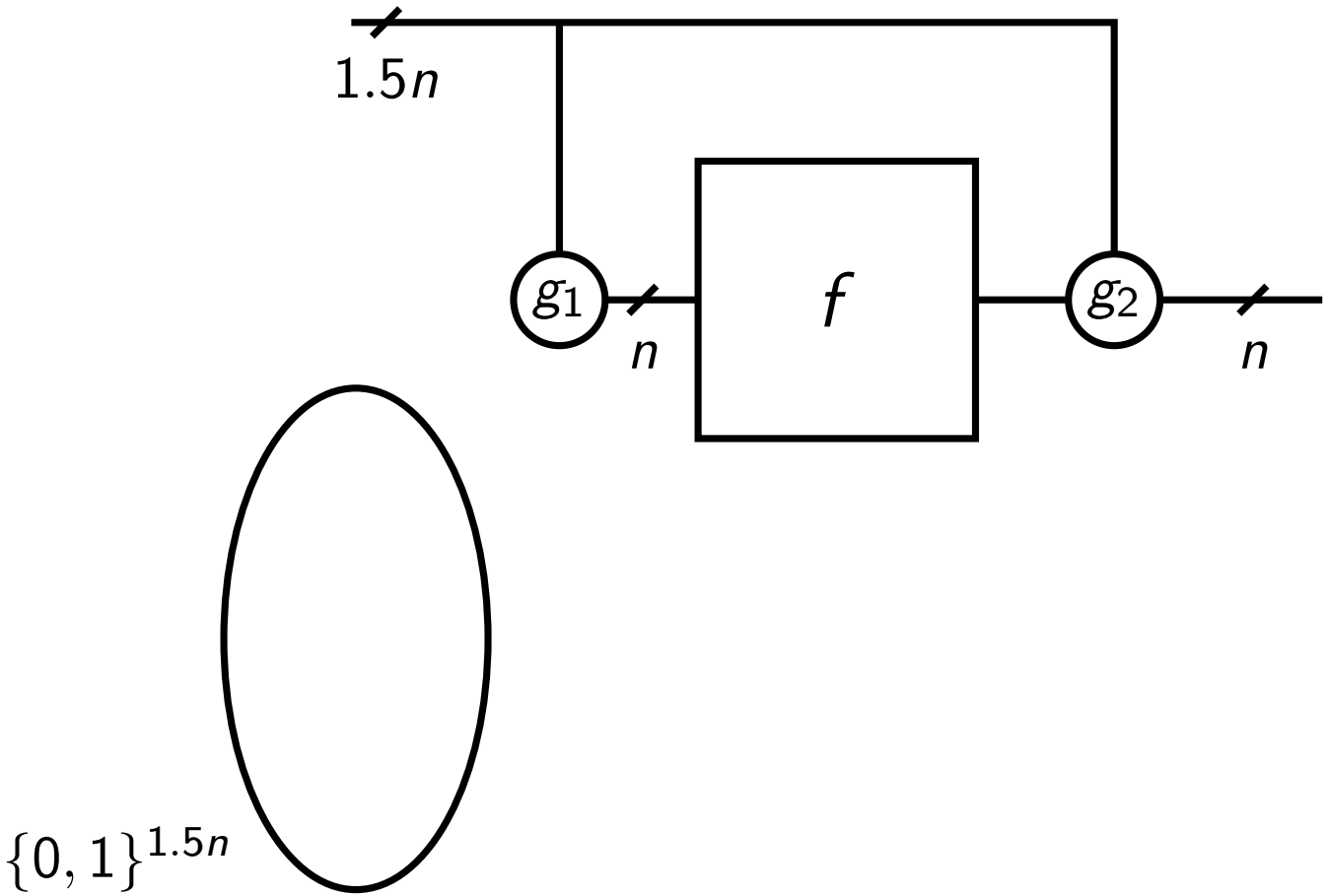
Back-of-the-envelope expected number of leftover collisions: p^2C

Lemma says: Leftover number of colliding inputs is not much less than p^2C as long as p^2C is greater than the largest size of a layer.

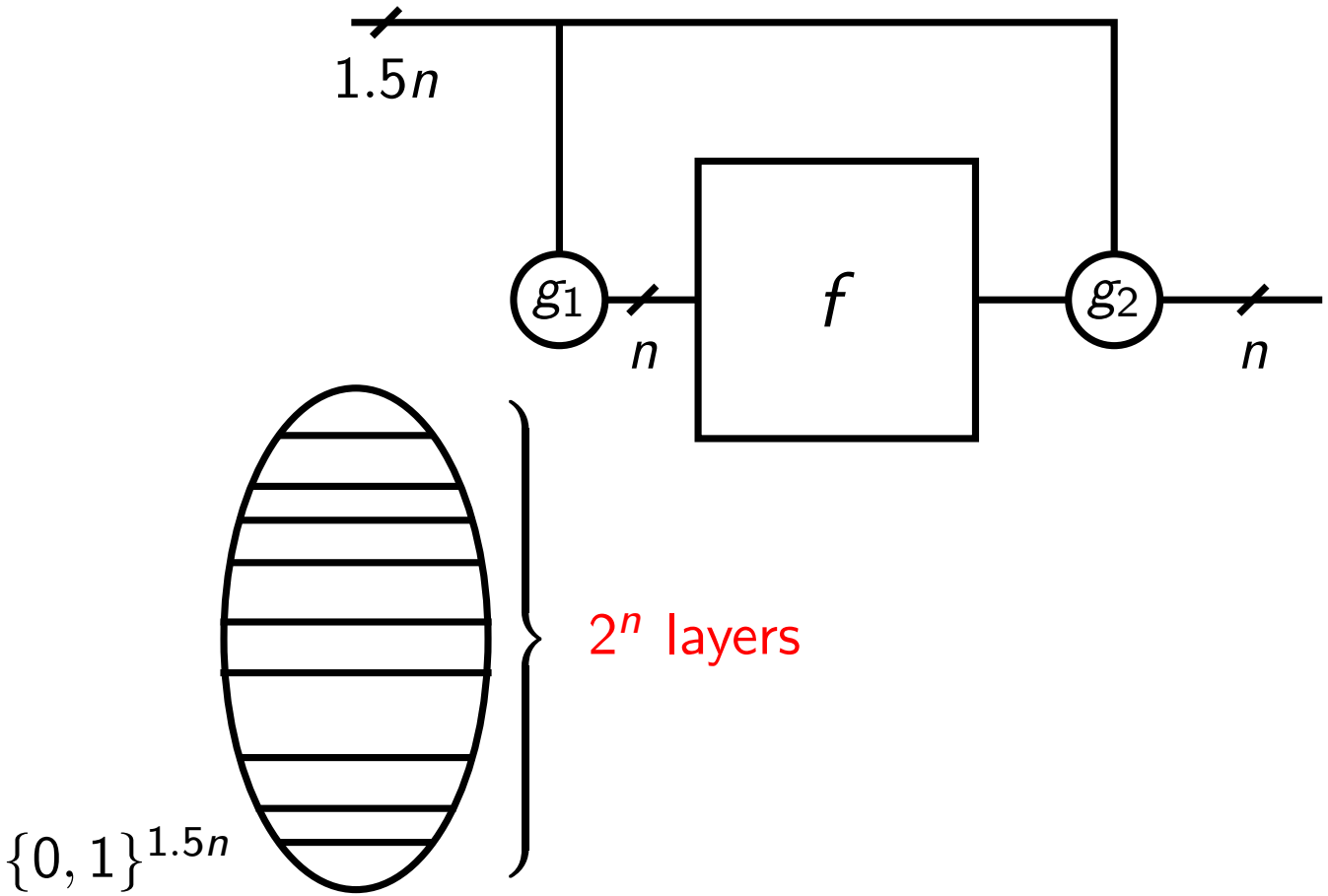
Proof sketch (r = 1)



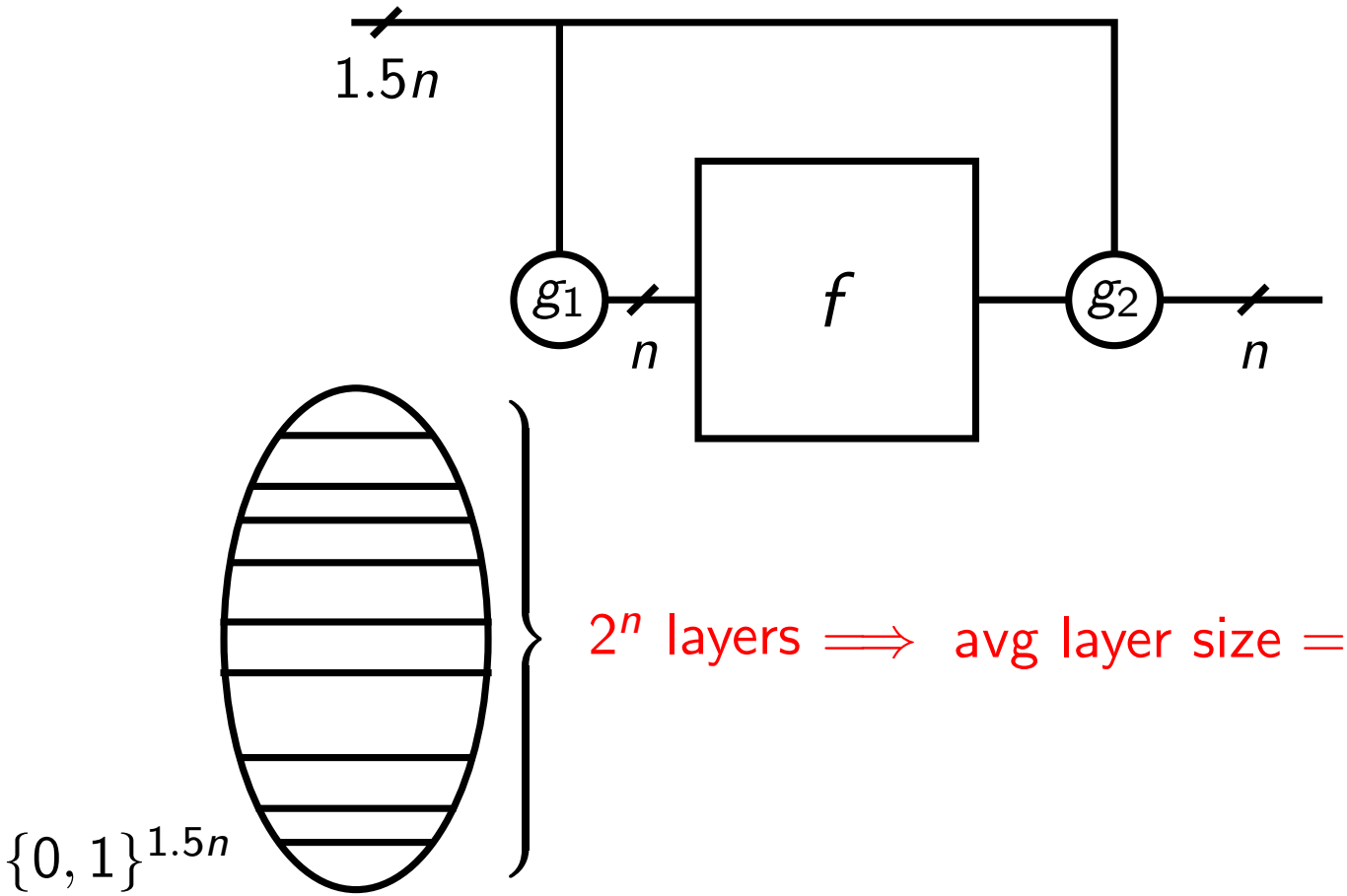
Proof sketch (r = 1)



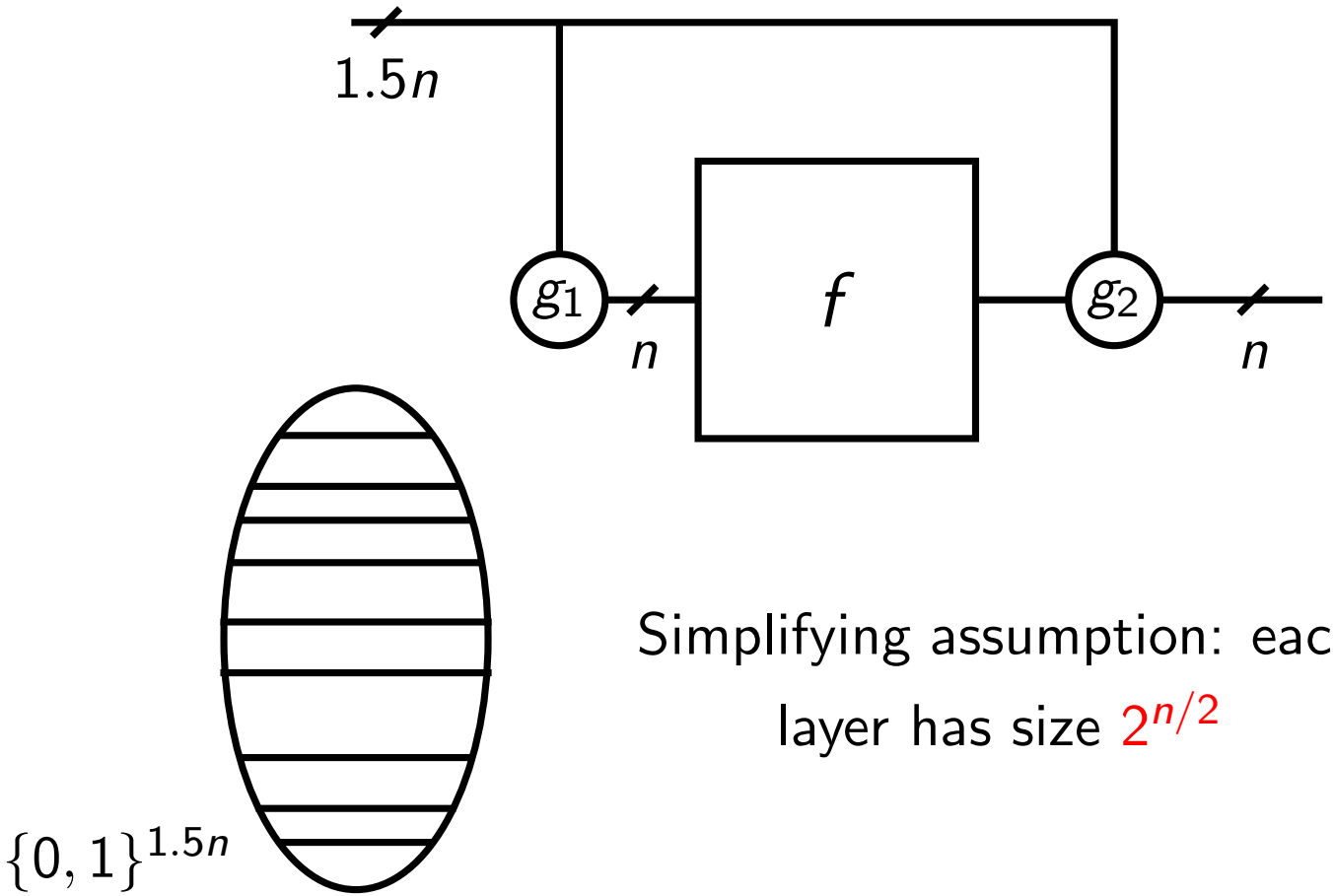
Proof sketch (r = 1)



Proof sketch (r = 1)

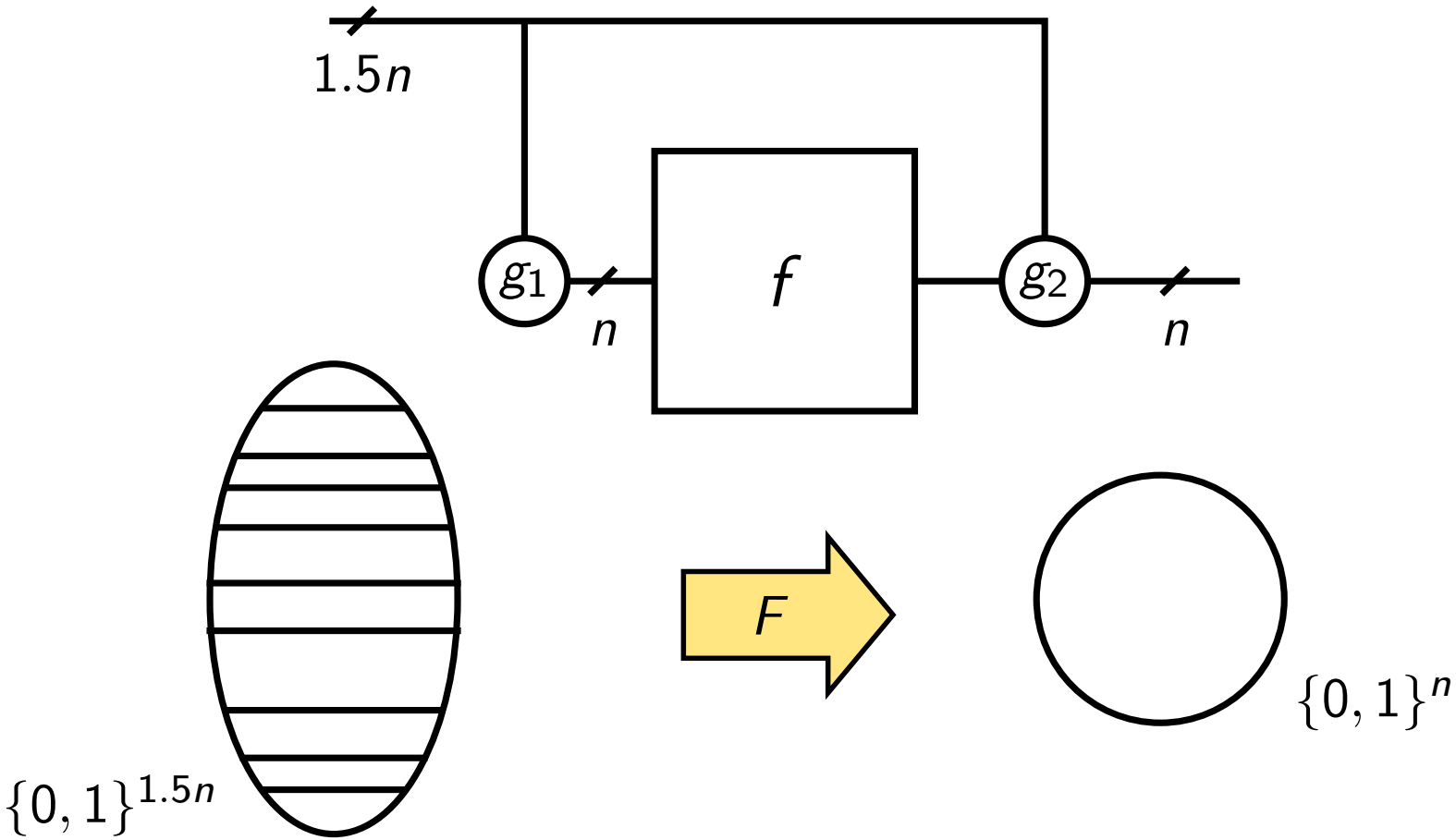


Proof sketch (r = 1)

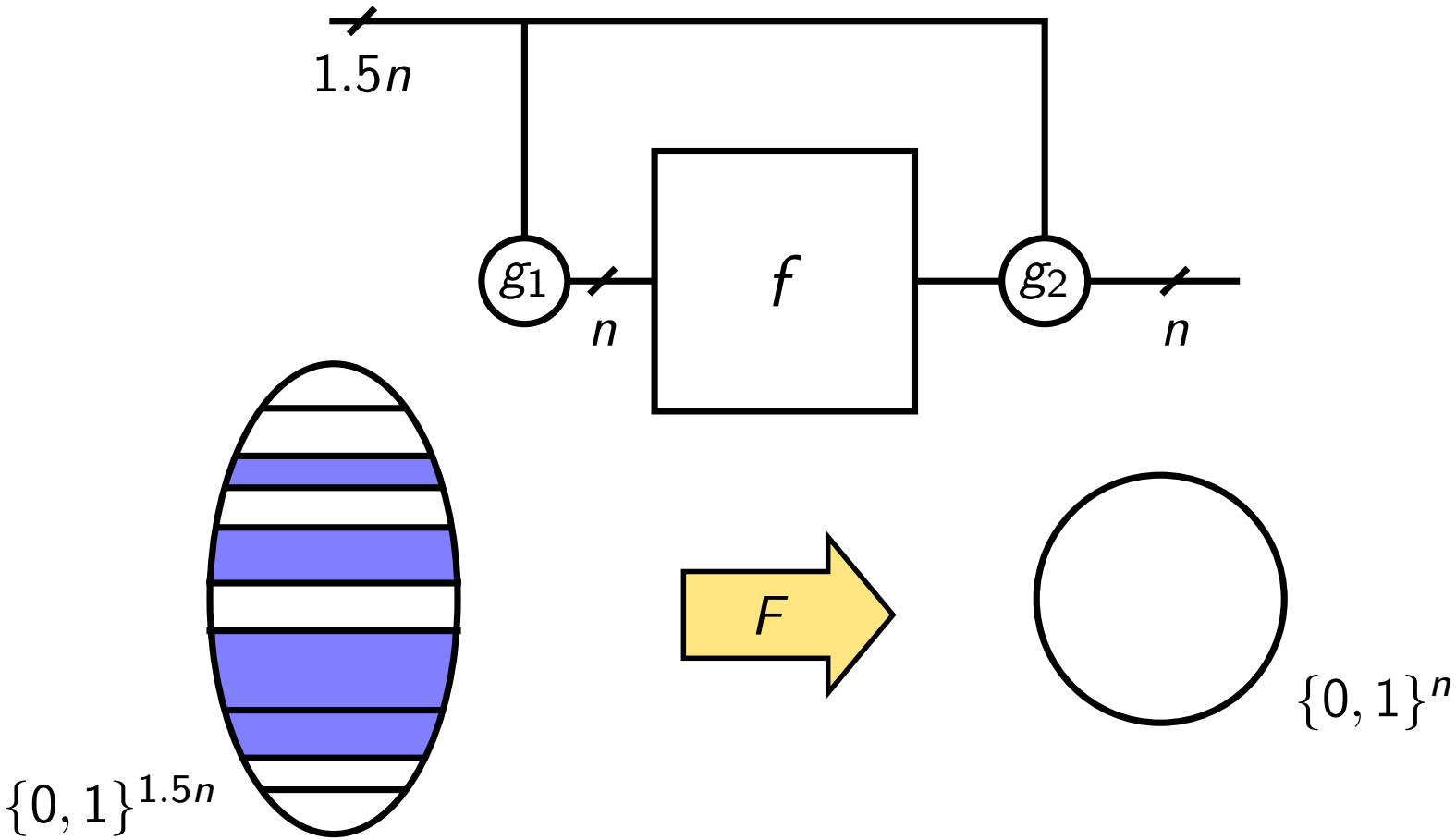


Simplifying assumption: each layer has size $2^{n/2}$

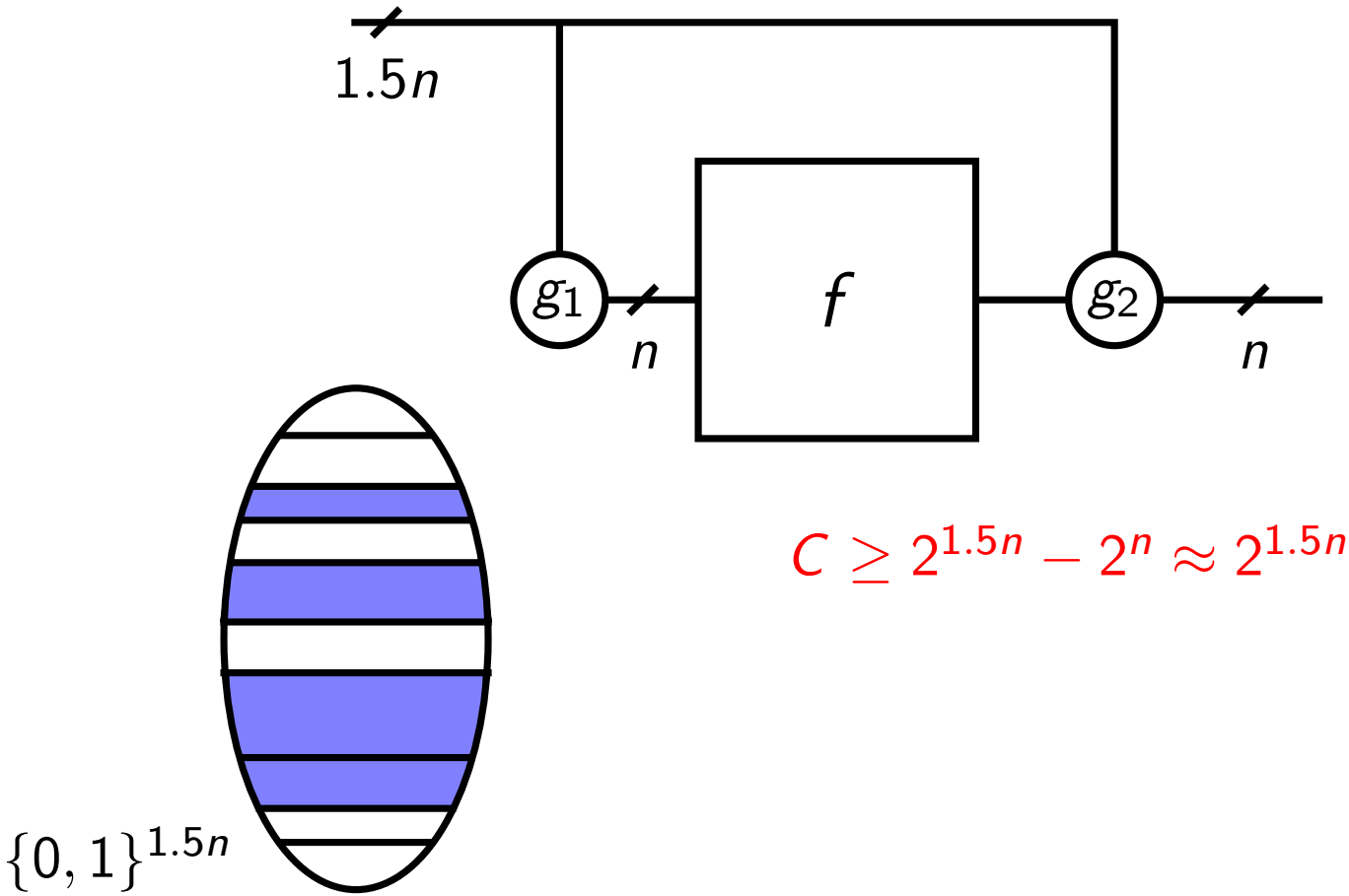
Proof sketch (r = 1)



Proof sketch (r = 1)

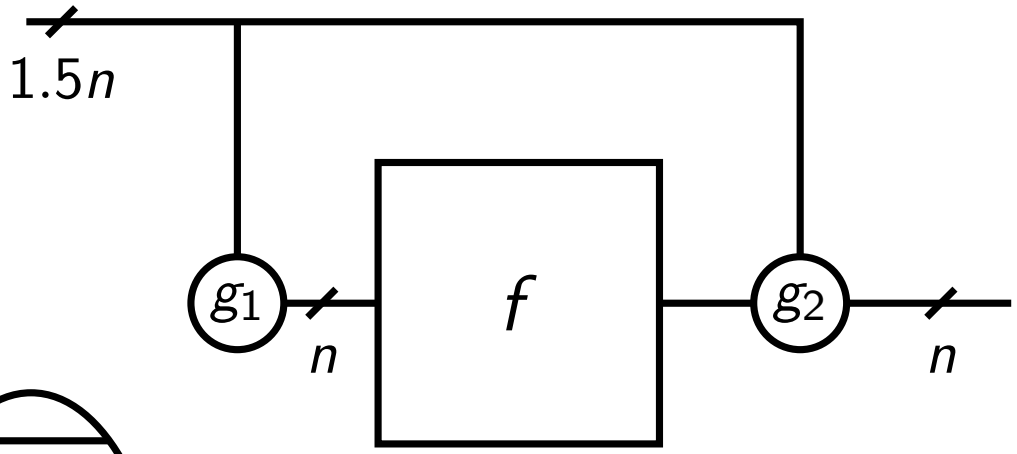


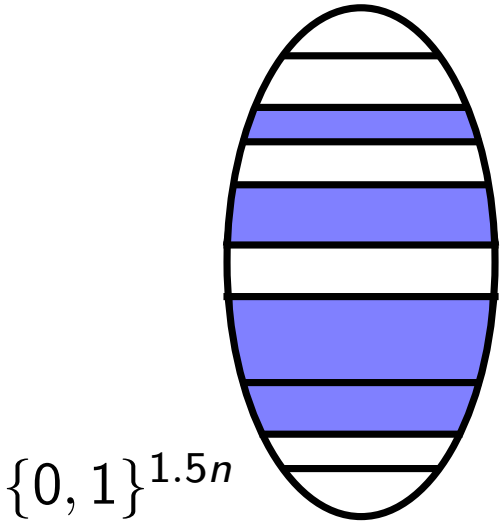
Proof sketch (r = 1)



$$C \geq 2^{1.5n} - 2^n \approx 2^{1.5n}$$

Proof sketch (r = 1)

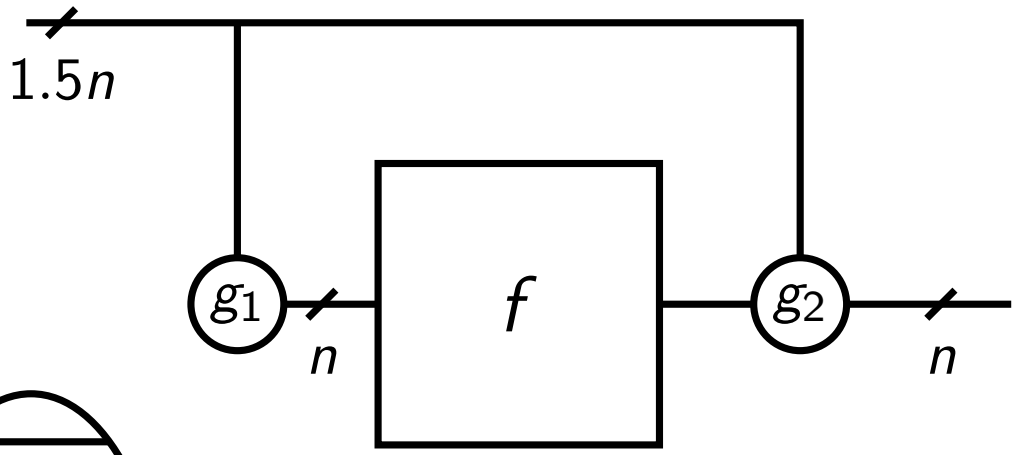


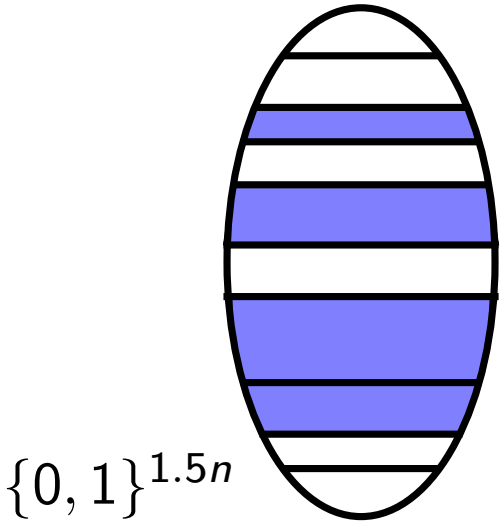


$$C \geq 2^{1.5n} - 2^n \approx 2^{1.5n}$$

$$k = 2^n,\ q = 2^{n/4},\ p = q/k = 2^{-3n/4}$$

Proof sketch (r = 1)



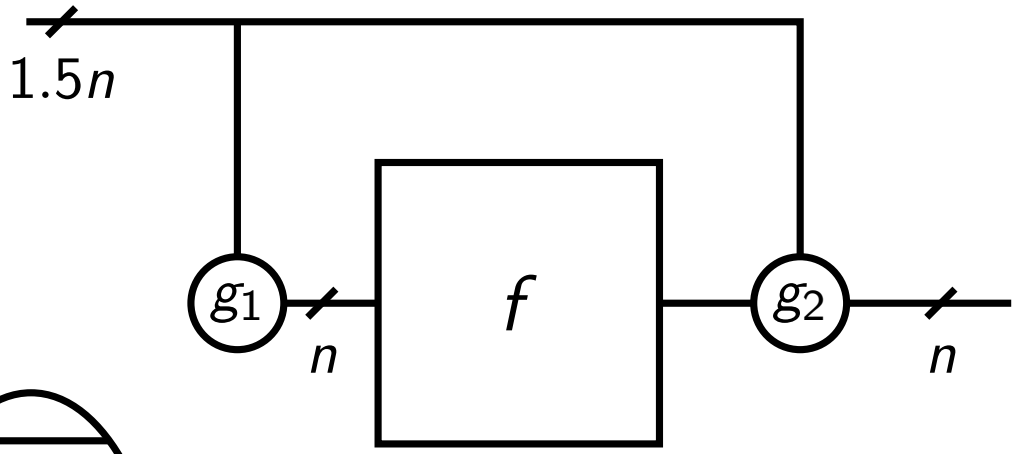


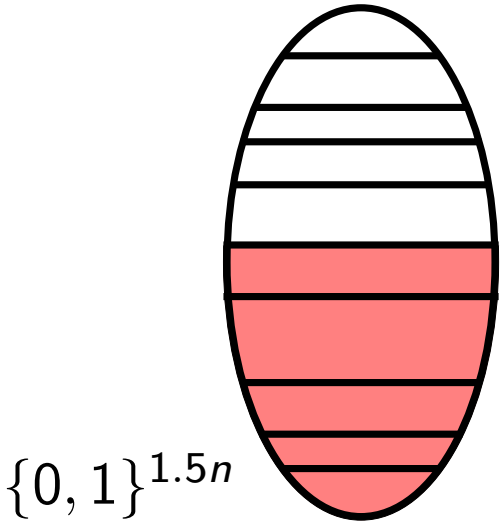
$$C \geq 2^{1.5n} - 2^n \approx 2^{1.5n}$$

$$k = 2^n,\ q = 2^{n/4},\ p = q/k = 2^{-3n/4}$$

$$p^2 C = 2^0 \ll \text{max layer size}$$

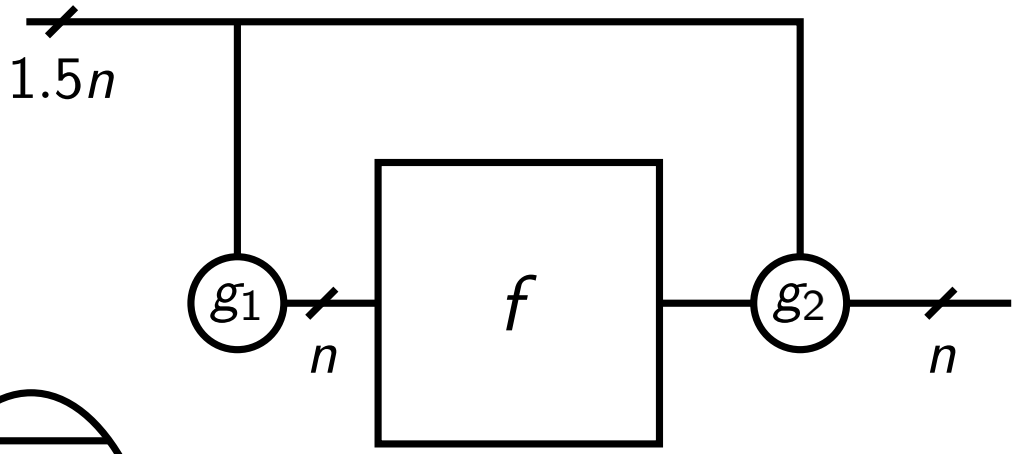
Proof sketch (r = 1)

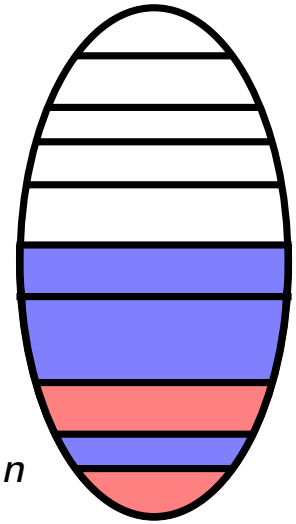




Idea: Pre-select a "ground set" of $2^{n/2+1}$ layers

Proof sketch (r = 1)

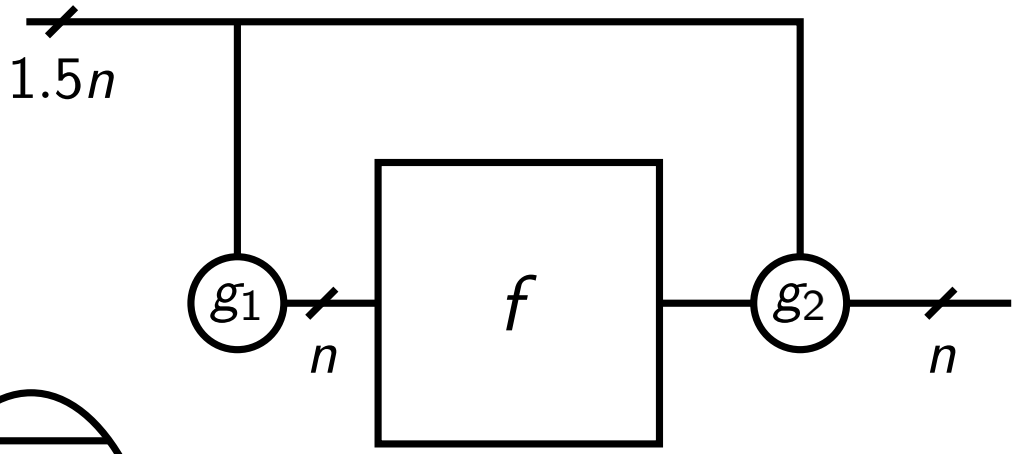


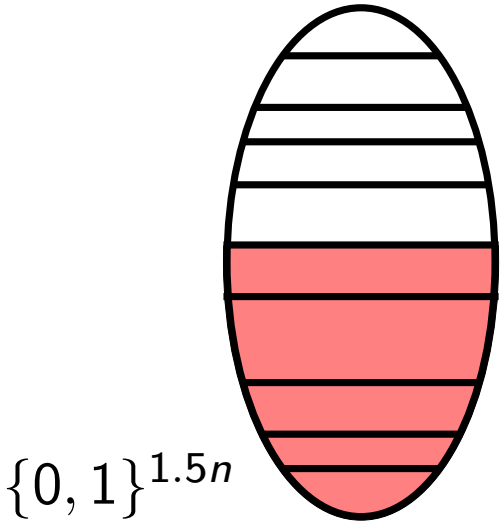


Idea: Pre-select a "ground set" of $2^{n/2+1}$ layers

...then select q layers among those...

Proof sketch (r = 1)

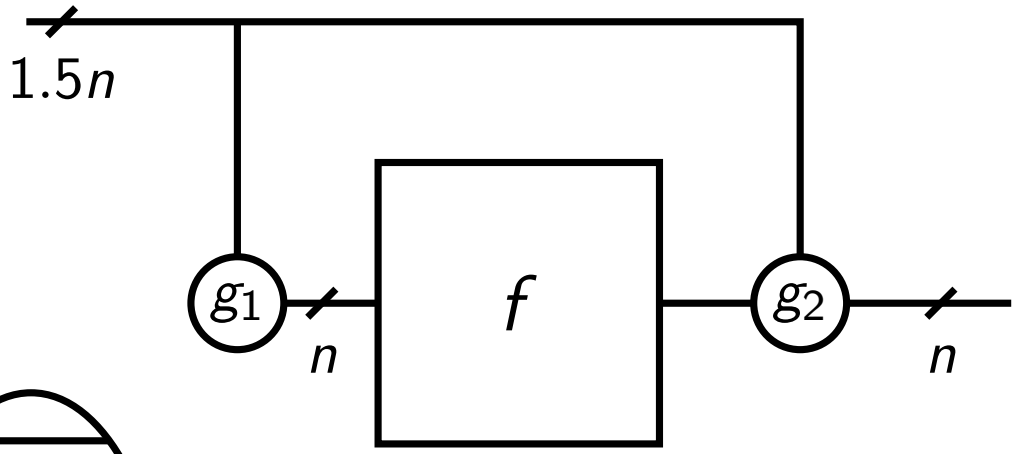


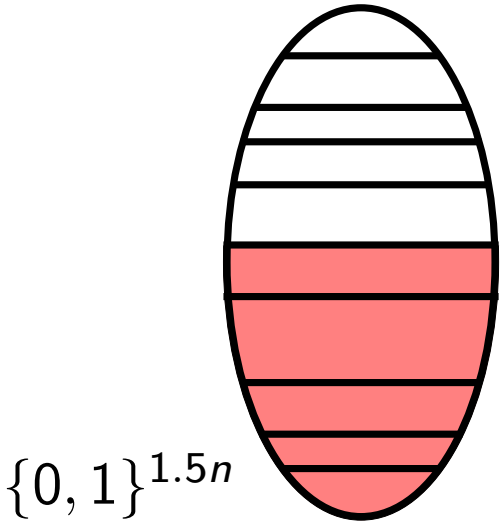


Idea: Pre-select a "ground set" of $2^{n/2+1}$ layers

Have $|Domain| = 2^{n/2+1} \cdot 2^{n/2} = 2^{n+1}$

Proof sketch (r = 1)



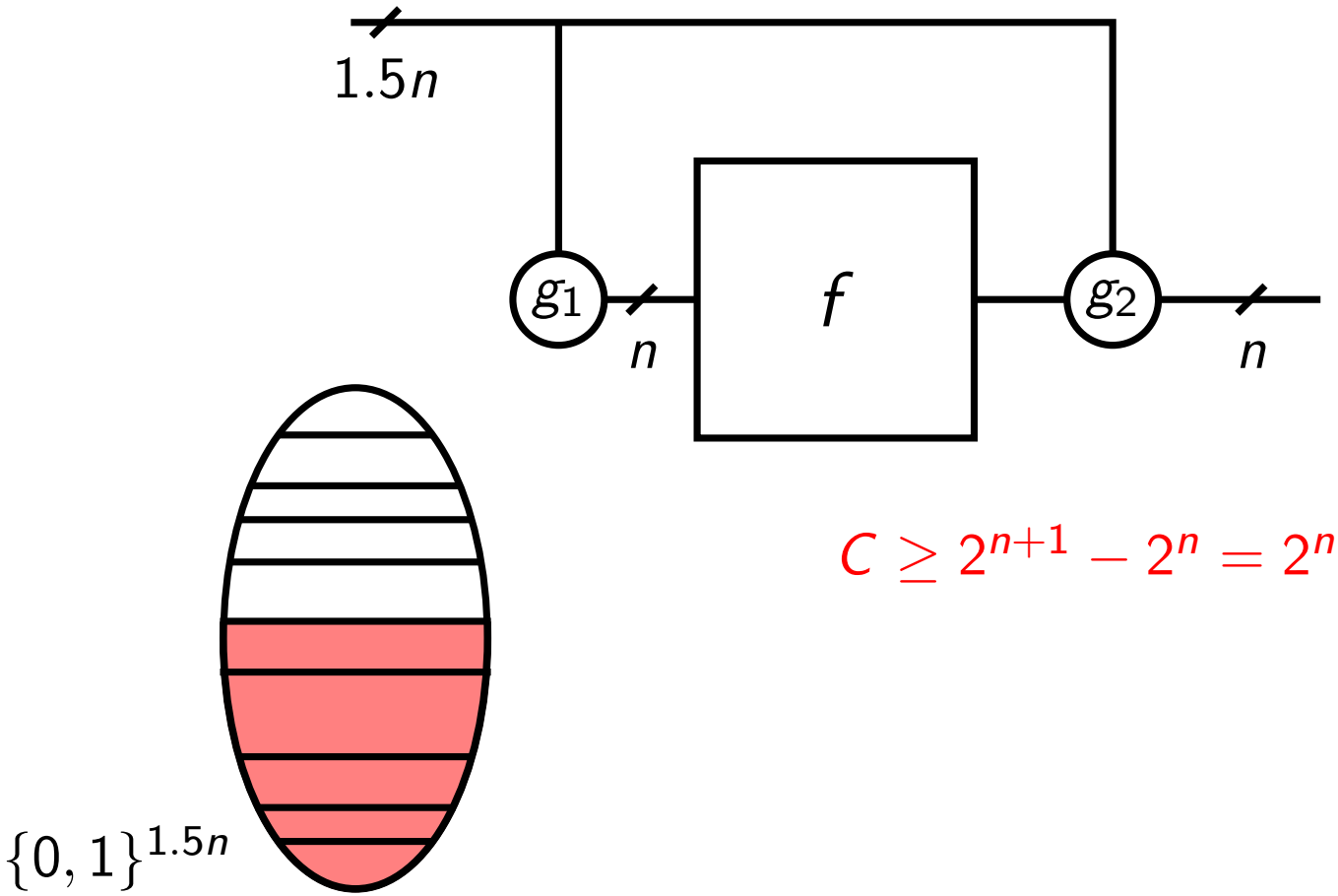


Idea: Pre-select a "ground set" of $2^{n/2+1}$ layers

Have $|Domain| = 2^{n/2+1} \cdot 2^{n/2} = 2^{n+1}$

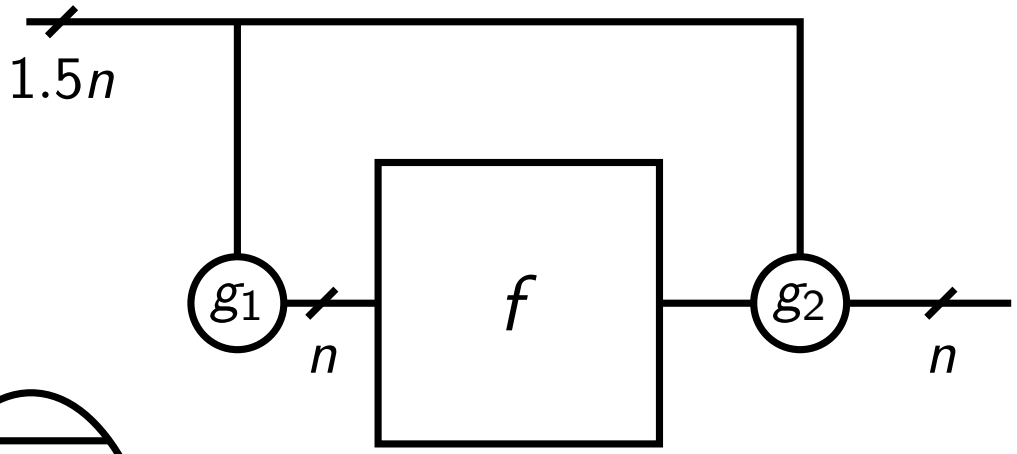
...and $|Range| = 2^n$, so still have 2^n collisions

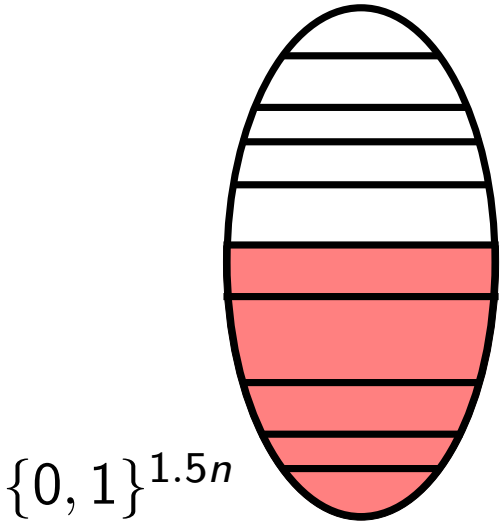
Proof sketch (r = 1)



$$C \geq 2^{n+1} - 2^n = 2^n$$

Proof sketch (r = 1)

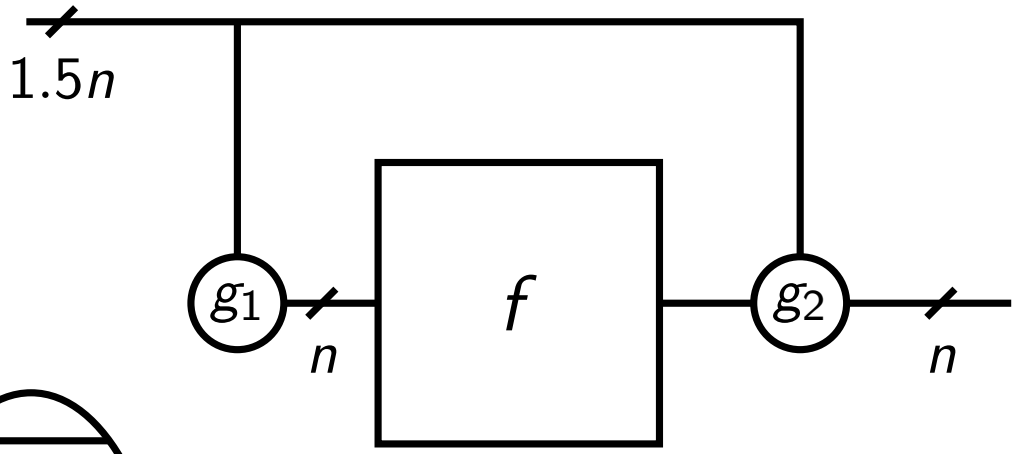


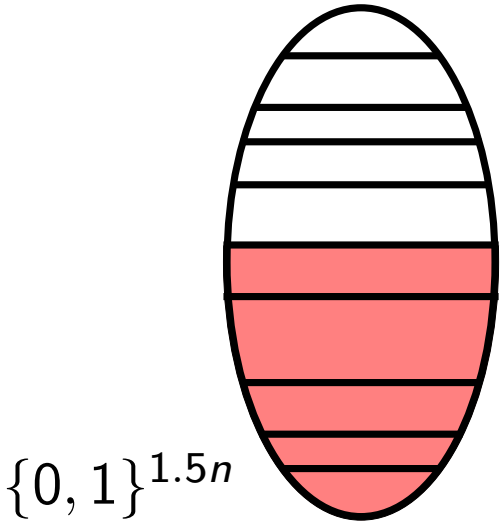


$$C \geq 2^{n+1} - 2^n = 2^n$$

$$k = 2^{n/2+1},\ q = 2^{n/4},\ p = q/k \approx 2^{-n/4}$$

Proof sketch (r = 1)



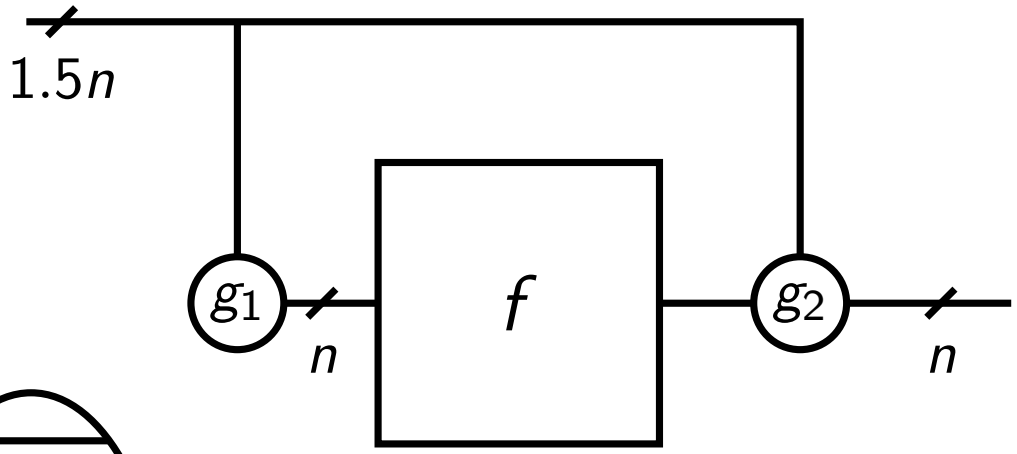


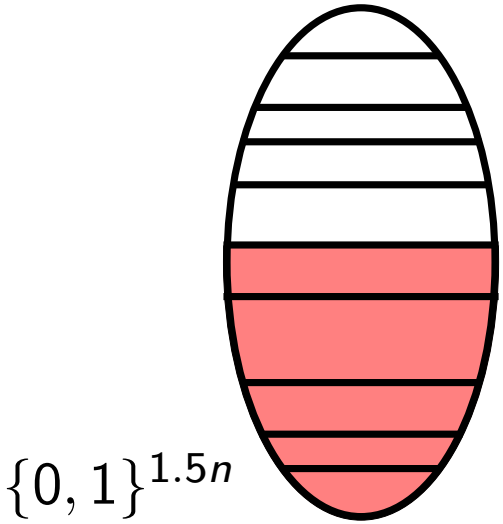
$$C \geq 2^{n+1} - 2^n = 2^n$$

$$k = 2^{n/2+1},\ q = 2^{n/4},\ p = q/k \approx 2^{-n/4}$$

$$p^2 C = 2^{n/2} \approx \text{max layer size}$$

Proof sketch (r = 1)



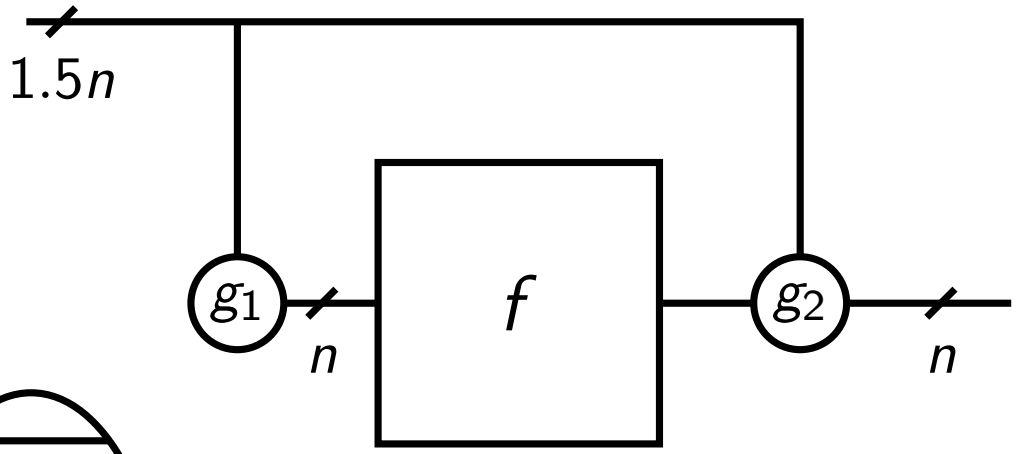


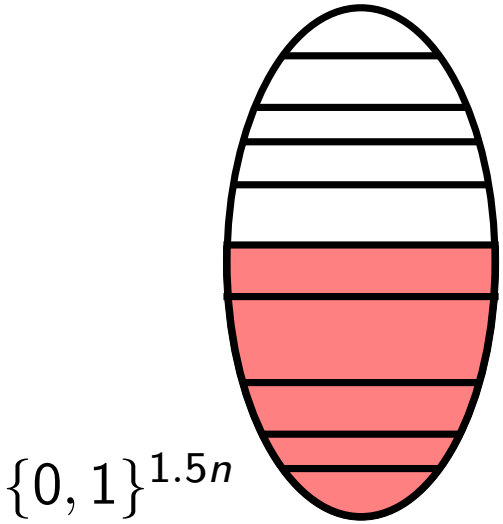
$$C \geq 2^{n+1} - 2^n = 2^n$$

$$k = 2^{n/2+1},\ q = 2^{n/4},\ p = q/k \approx 2^{-n/4}$$

$$p^2 C = 2^{n/2} \approx \text{max layer size}$$

Proof sketch (r = 1)

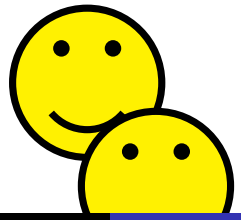




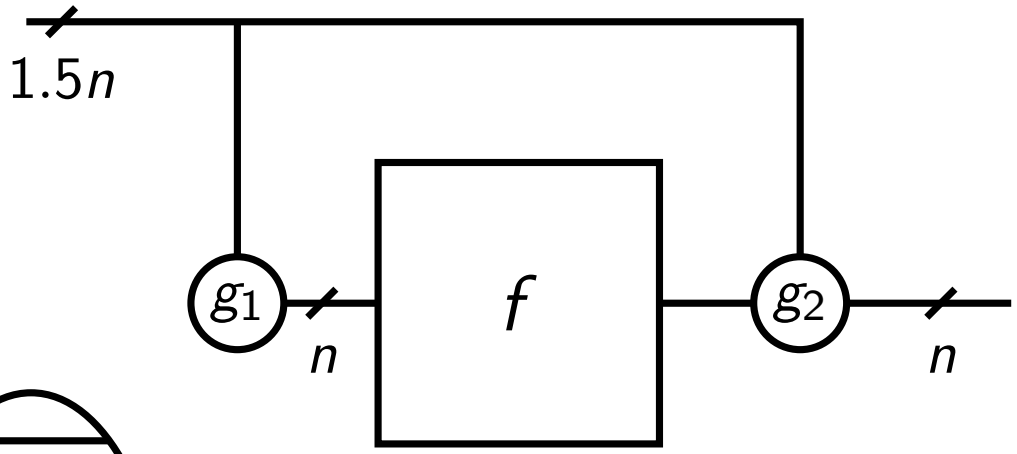
$$C \geq 2^{n+1} - 2^n = 2^n$$

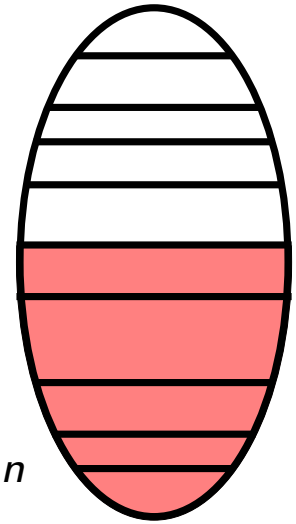
$$k = 2^{n/2+1},\ q = 2^{n/4},\ p = q/k \approx 2^{-n/4}$$

$$p^2 C = 2^{n/2} \approx \text{max layer size}$$



Proof sketch (r = 1)

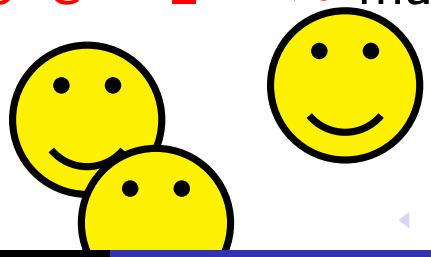




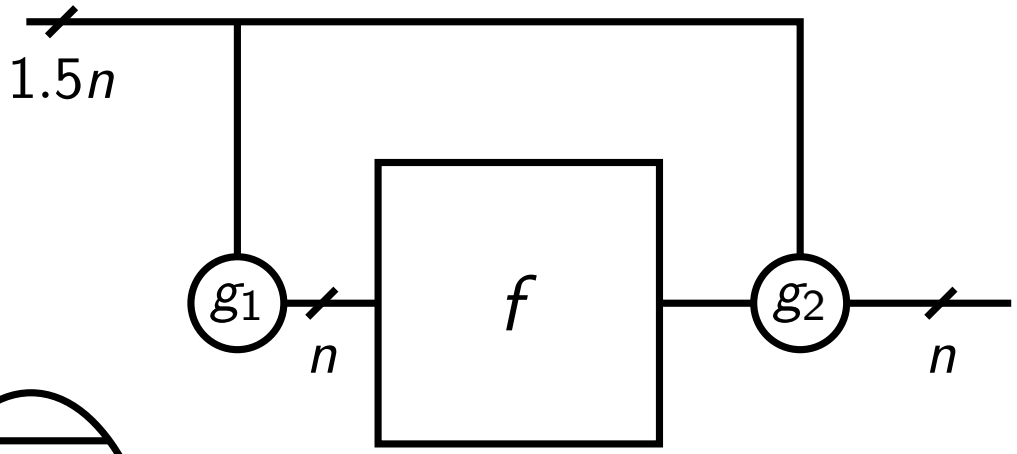
$$C \geq 2^{n+1} - 2^n = 2^n$$

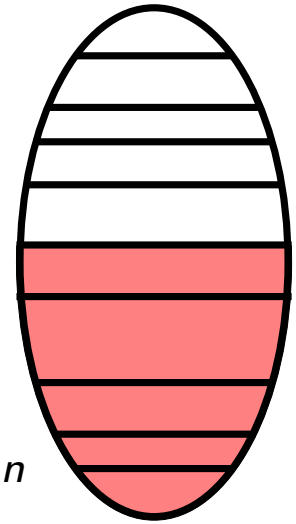
$$k = 2^{n/2+1},\ q = 2^{n/4},\ p = q/k \approx 2^{-n/4}$$

$$p^2 C = 2^{n/2} \approx \text{max layer size}$$



Proof sketch (r = 1)



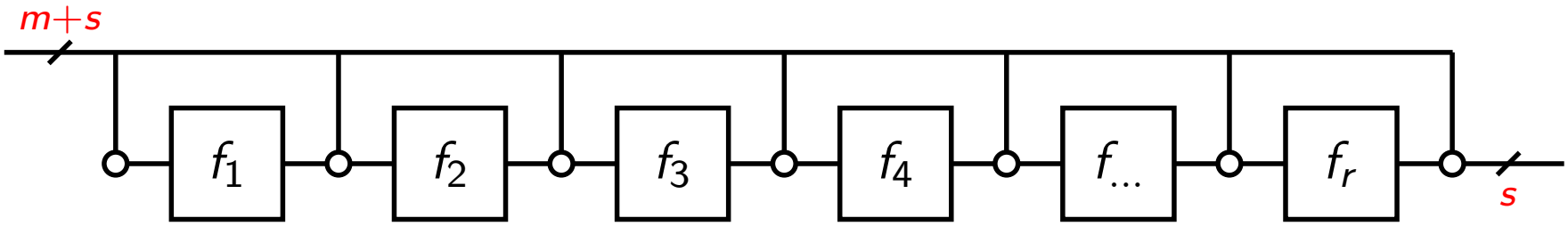


$$C \geq 2^{n+1} - 2^n = 2^n$$

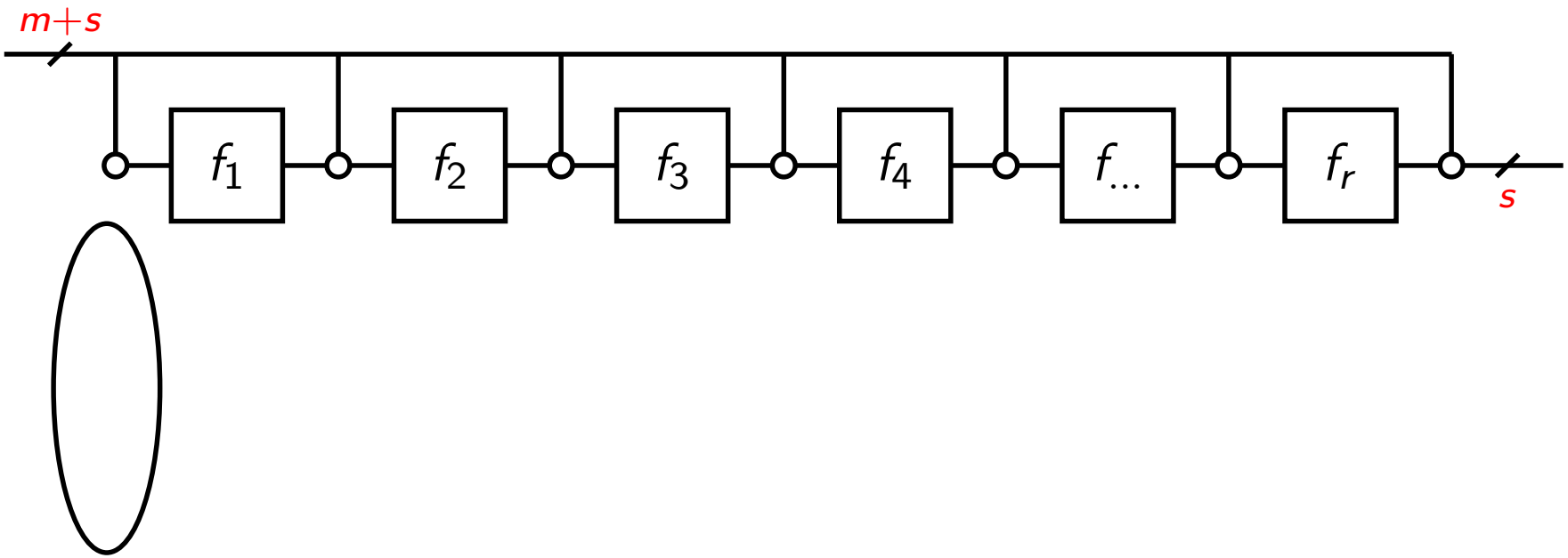
$$k = 2^{n/2+1},\ q = 2^{n/4},\ p = q/k \approx 2^{-n/4}$$

$$p^2 C = 2^{n/2} \approx \text{max layer size}$$

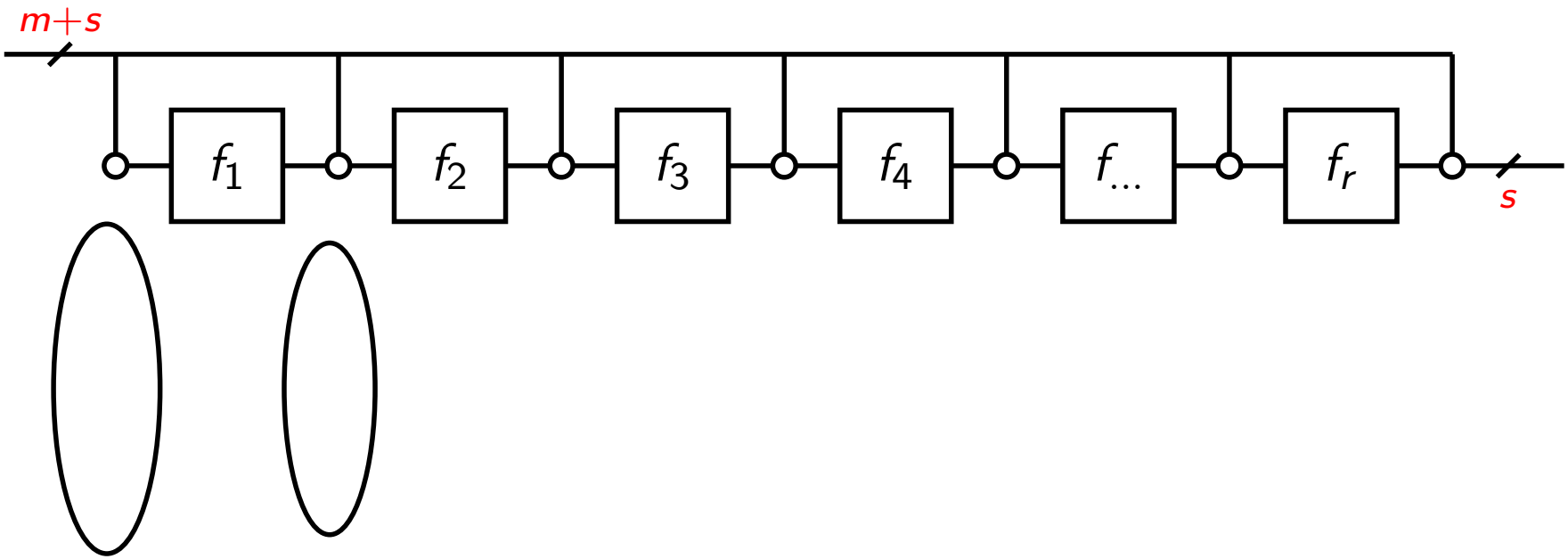
General Case



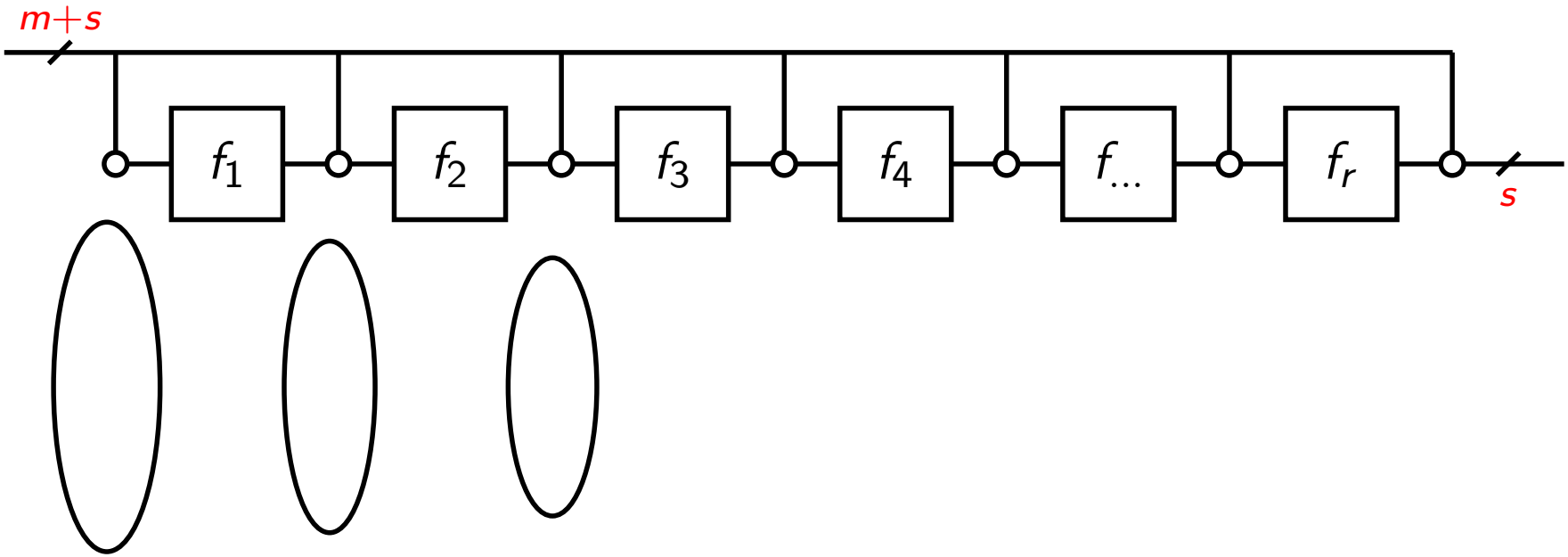
General Case



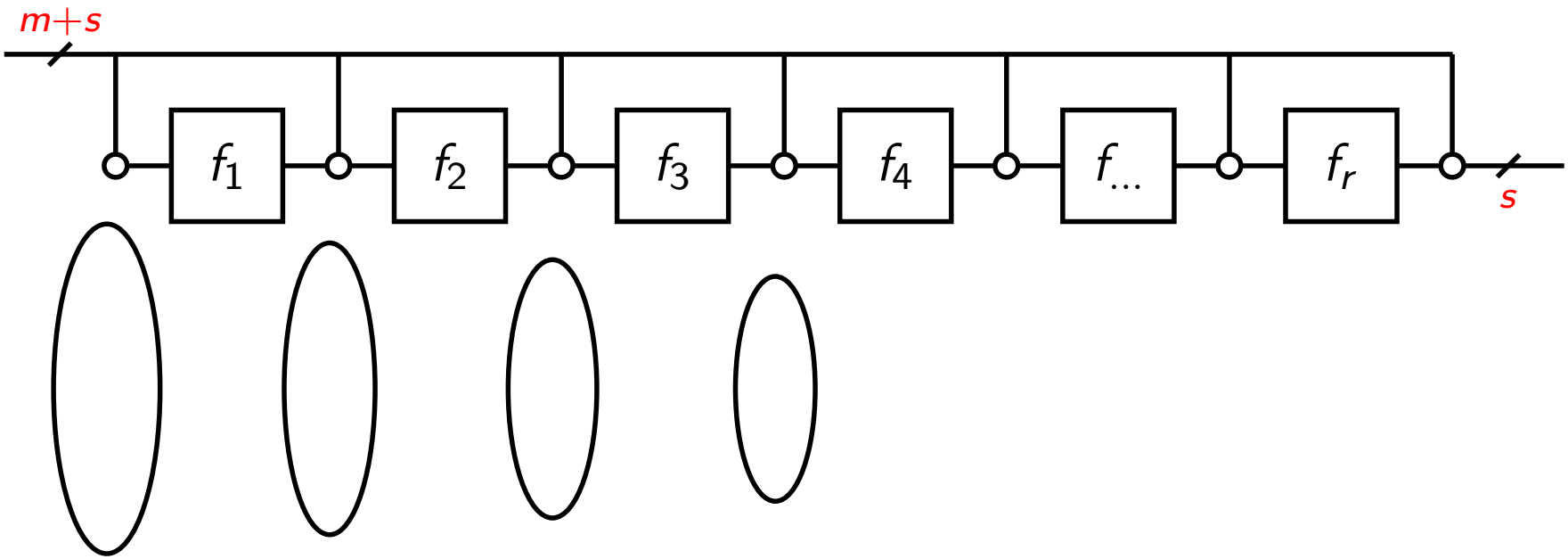
General Case



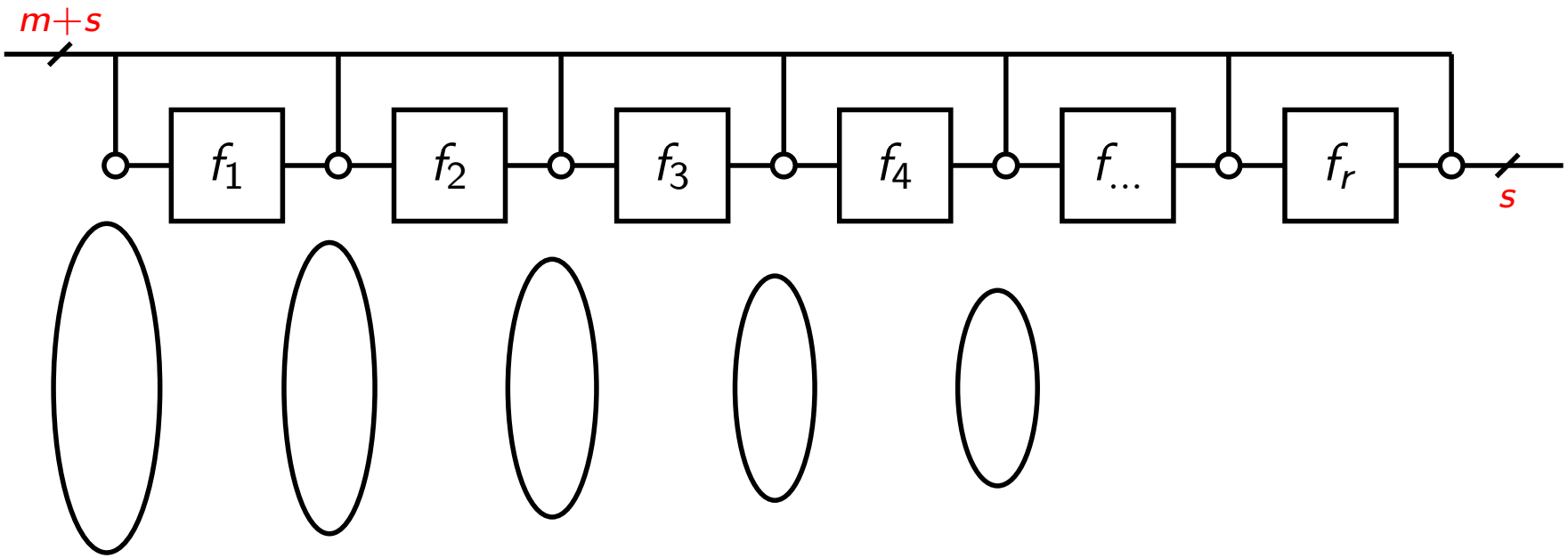
General Case



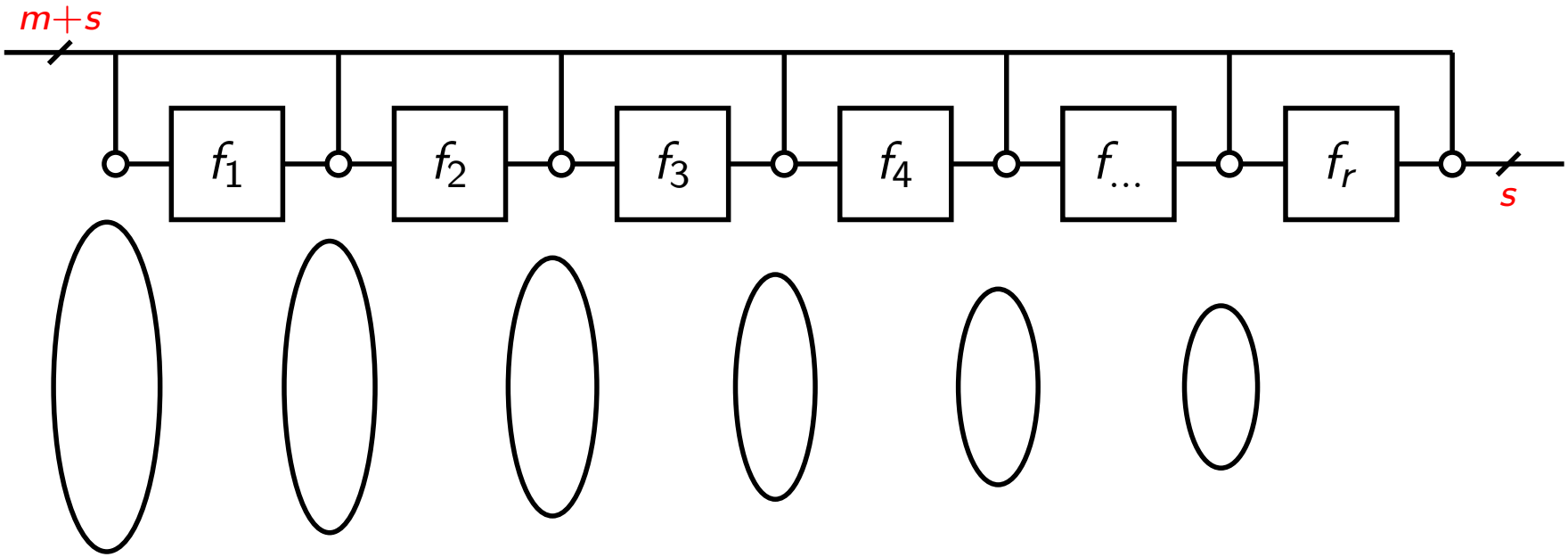
General Case



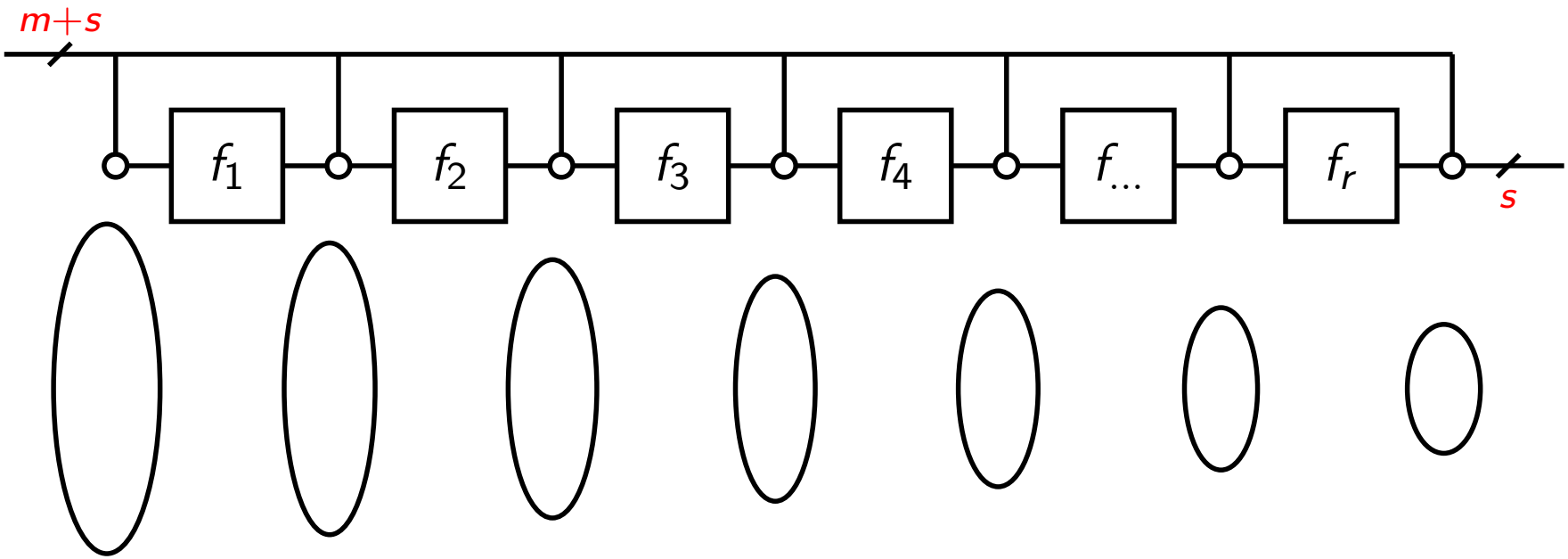
General Case



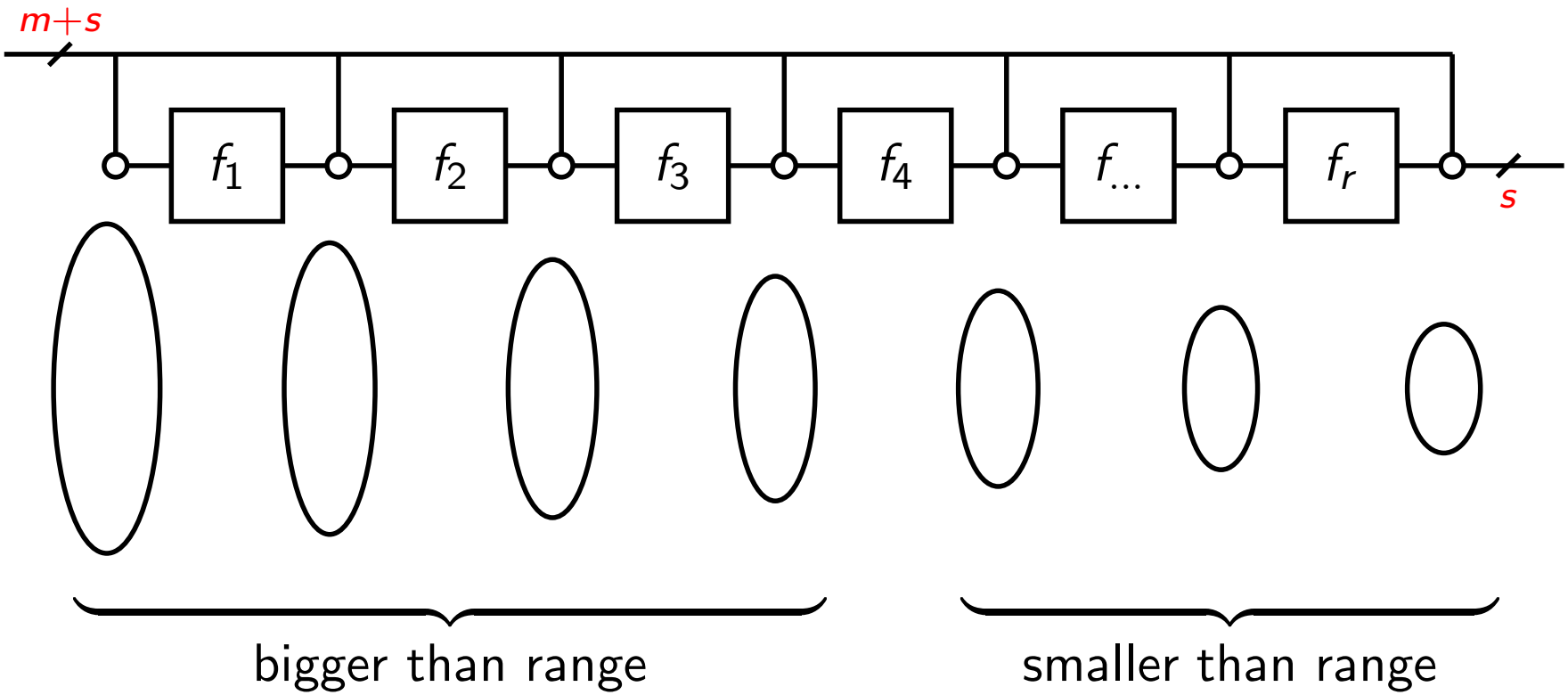
General Case



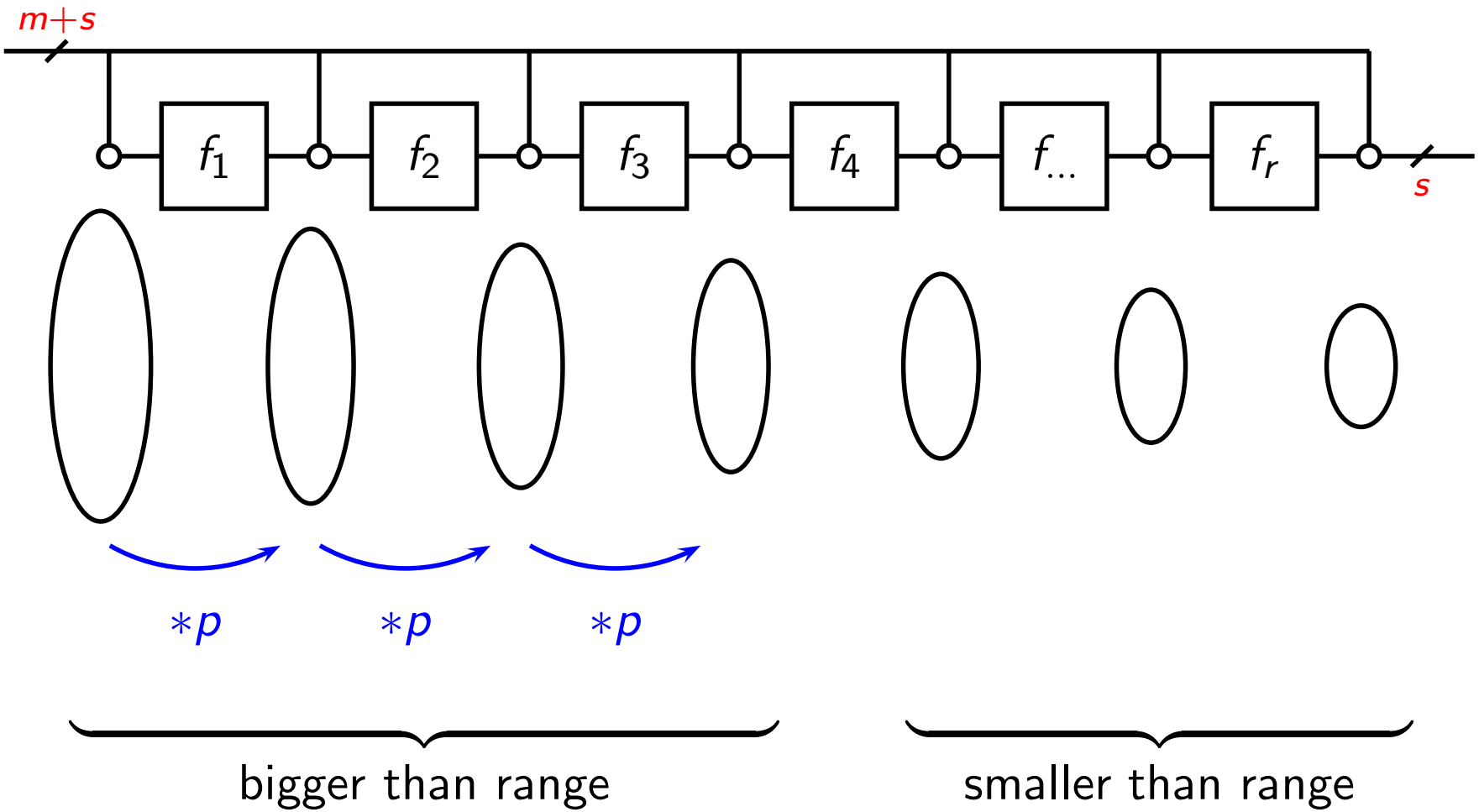
General Case



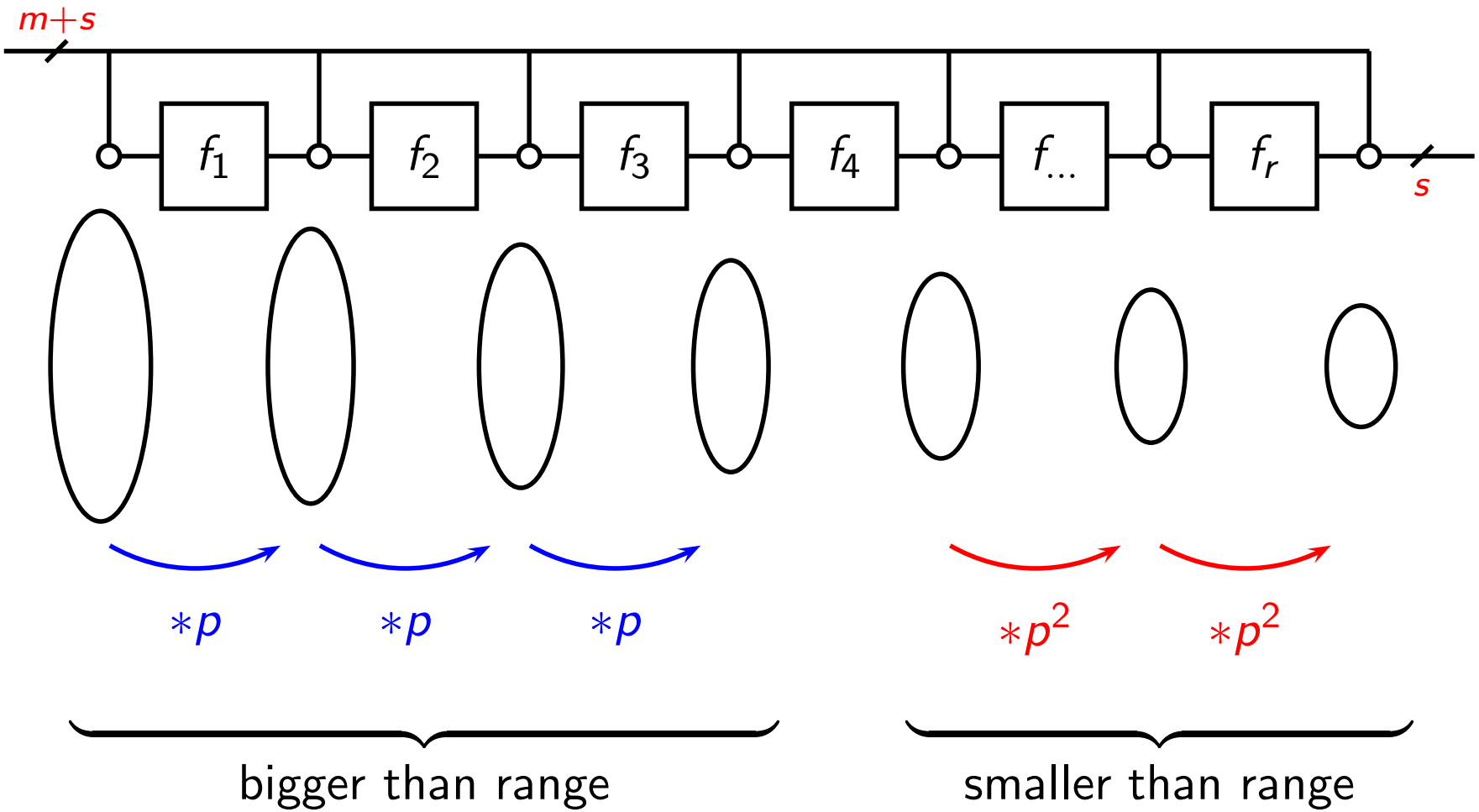
General Case



General Case



General Case



General Case