Theoretical limitations of multi-layer Transformer

Warning: Theory paper, not meant for practicality.

Lijie Chen
UC Berkeley

Binghui Peng
Stanford

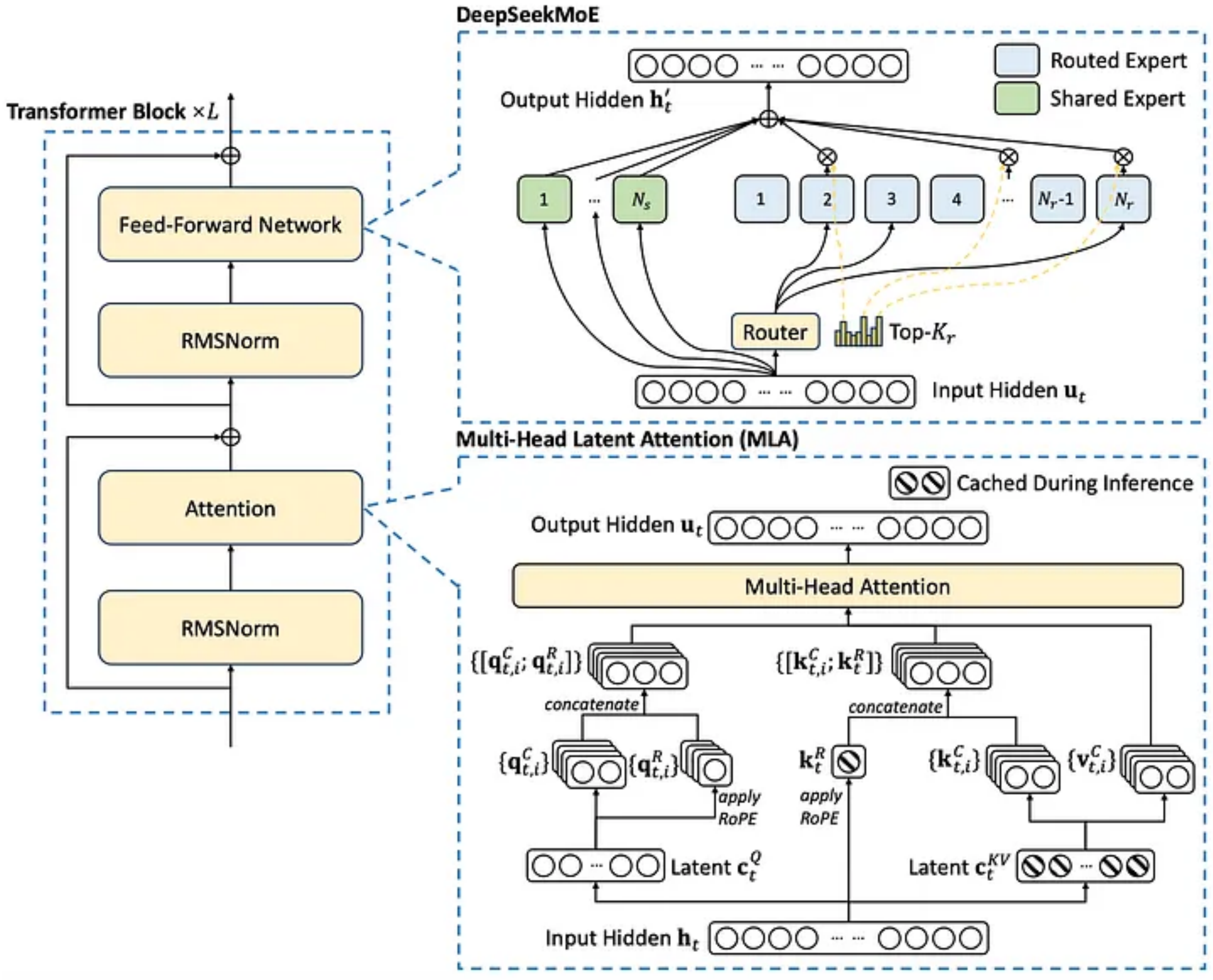
Hongxun Wu
UC Berkeley

Our Problem

- What tasks can **not** be solved by Transformers without CoT?
- Existing Answer:
  - Transformers are constant-depth threshold circuits TC^0. (Merrill et al.'24)
  - It cannot solve inherently sequential tasks.

- In which way is it unsatisfactory (to theoreticians):
  - We do not understand TC^0! (Complexity Theory's Waterloo)
  - Any constant-layer architecture is in TC^0. Didn't say much about transformers.

Transformers



- Many variations!
- MHA / MQA / MLA / … , tokenization, positional encoding, mixture-of-expert, pre-norm/ post-norm/hybrid-norm, activation function, residual connection, gated components…

Communication Game

$$\xrightarrow{\Pi_1}$$

$$\xleftarrow{\Pi_2}$$

$$\vdots$$

$$\xrightarrow{\Pi_r}$$

$$\xleftarrow{f(a,b)}$$

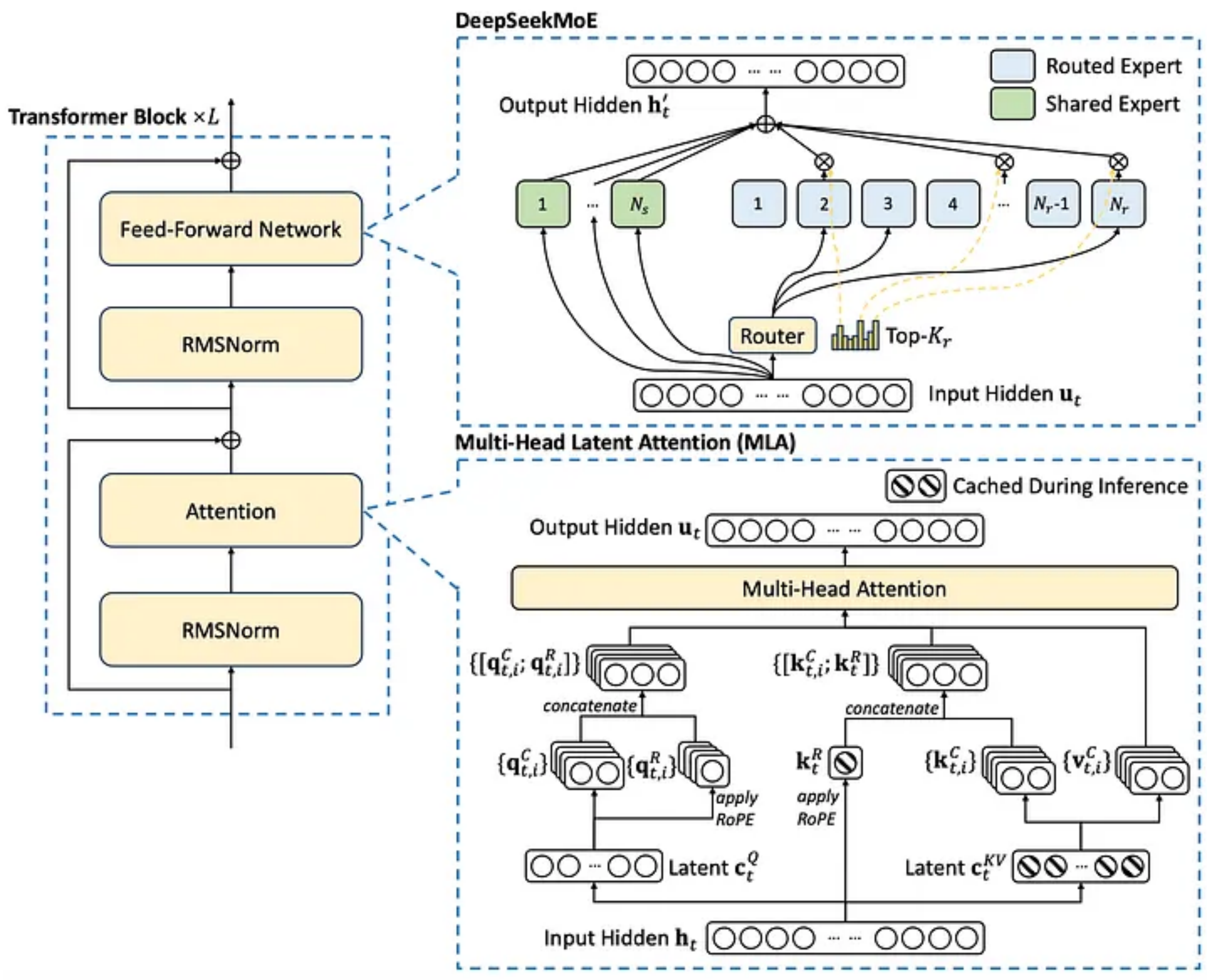
$a \in \{0,1\}^n$

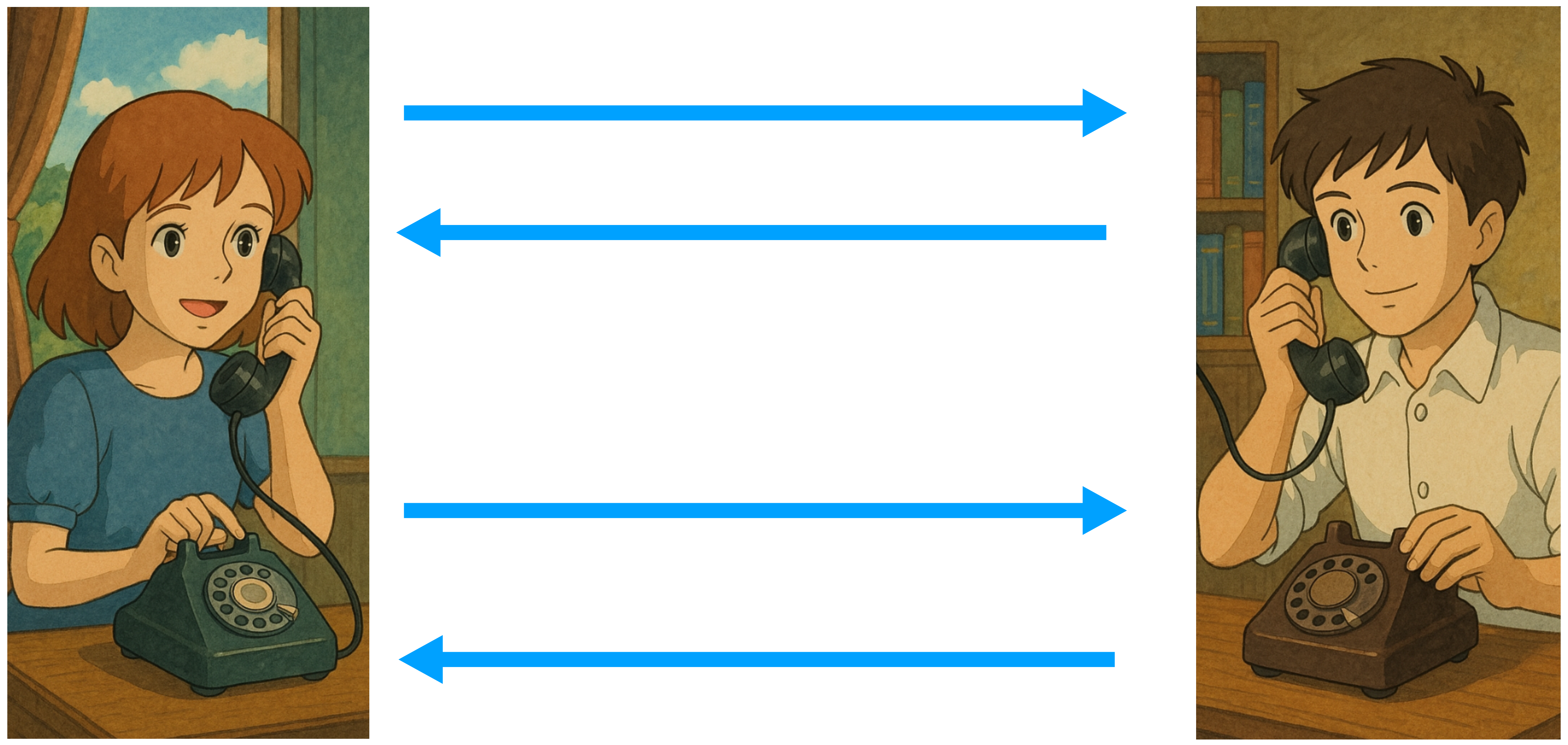
minimize $|\Pi_1| + |\Pi_2| + \cdots + |\Pi_r|$

$b \in \{0,1\}^n$

Theme

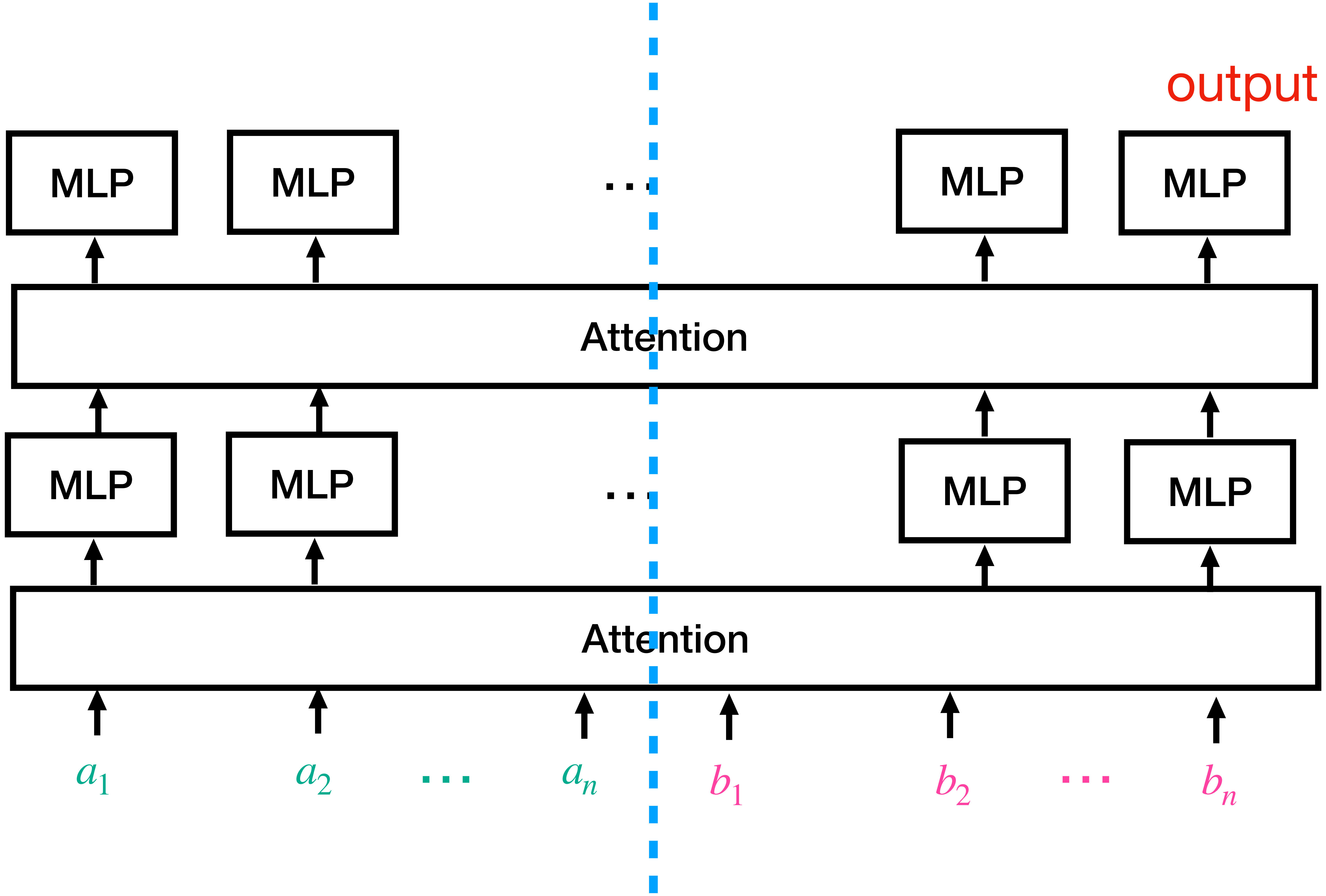
- Understand the limitations of a transformer by understand its inside communicaition!





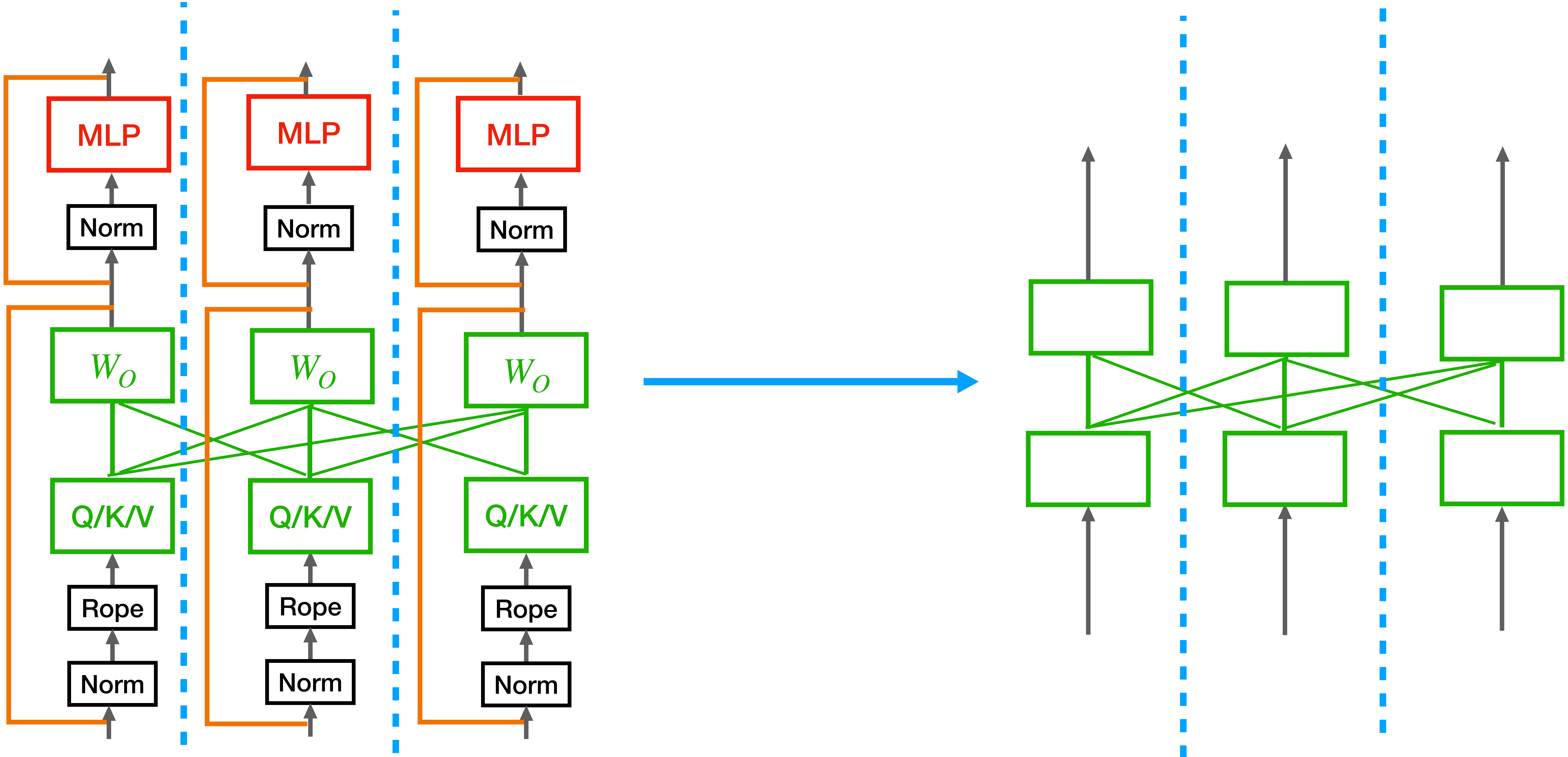
Theme

- Understand the limitations of a transformer by understand its inside communicaition!



Theme

- When we focus on communicaition, many architectural choice goes away.



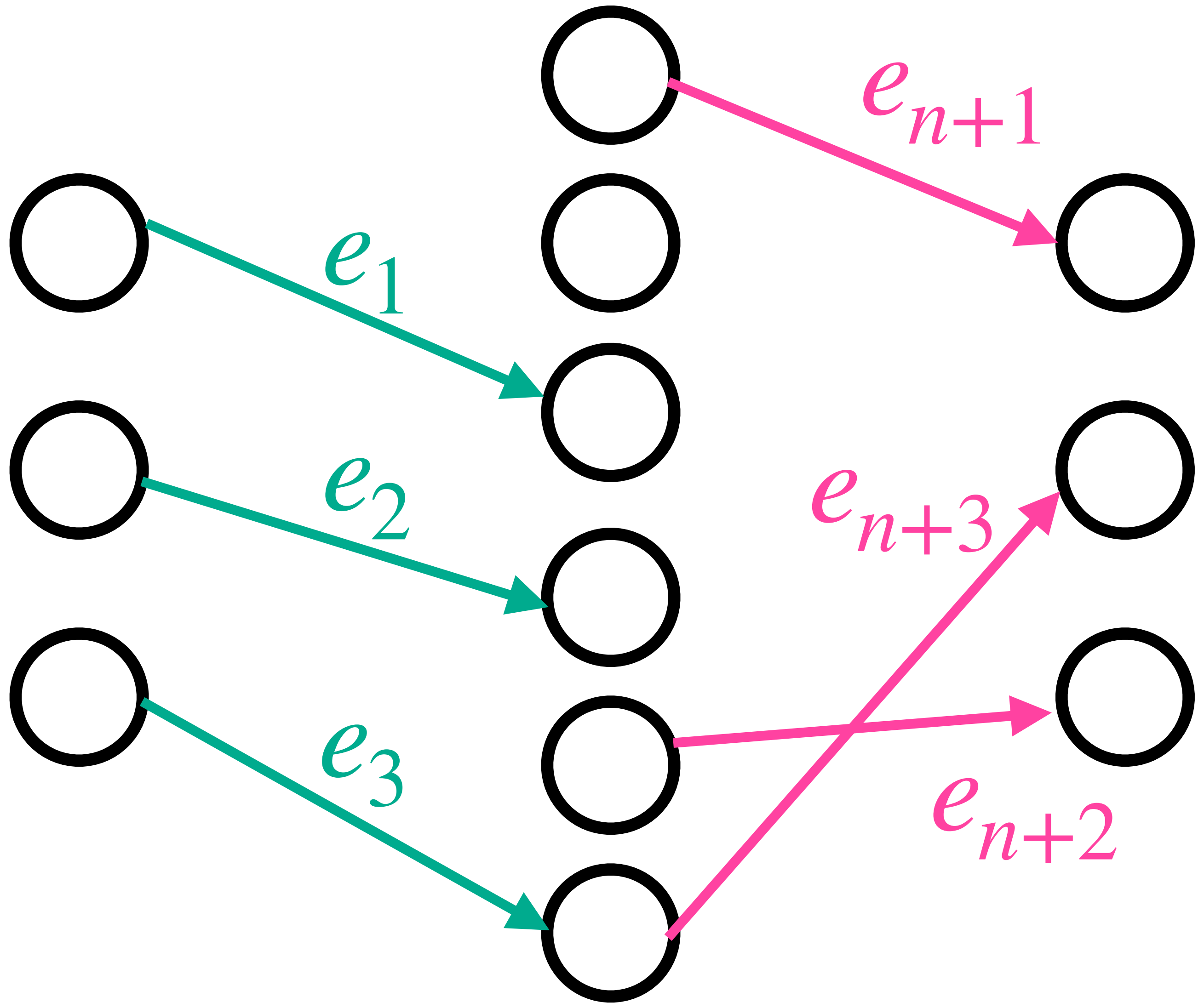
Warm up: One Layer - Example 1

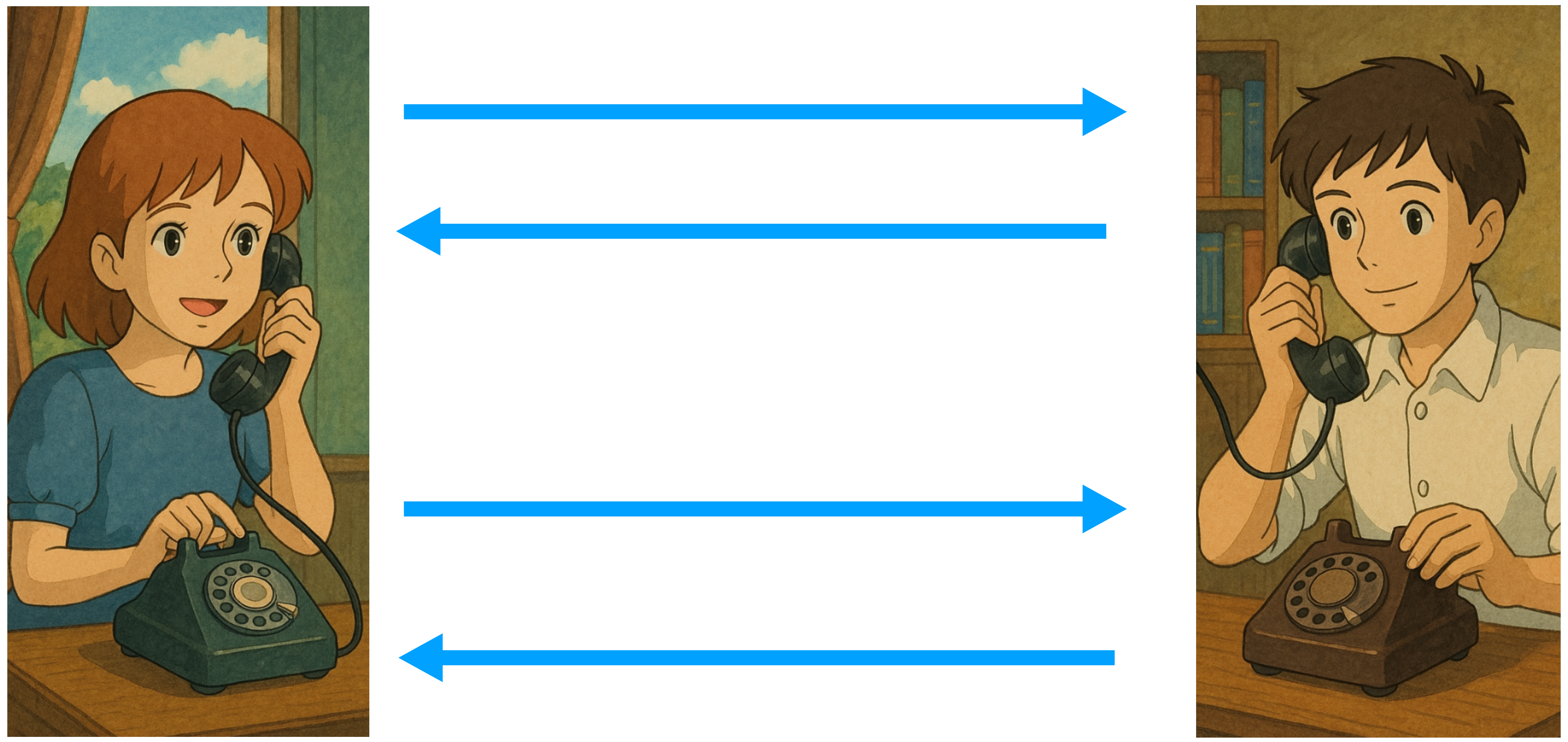
n: prompt length
d: hidden dimension
H=1: attention head
p: precision (bits)

Theorem (Mixture of Parrot by Jelassi et al.'25)

A single-layer transformer cannot find a length-2 path in a graph unless $dp = \Omega(n)$.

Problem (Length-2 Path)





e_1 e_2 $\cdots$ e_n

e_{n+1} e_{n+2} $\cdots$ e_{2n}

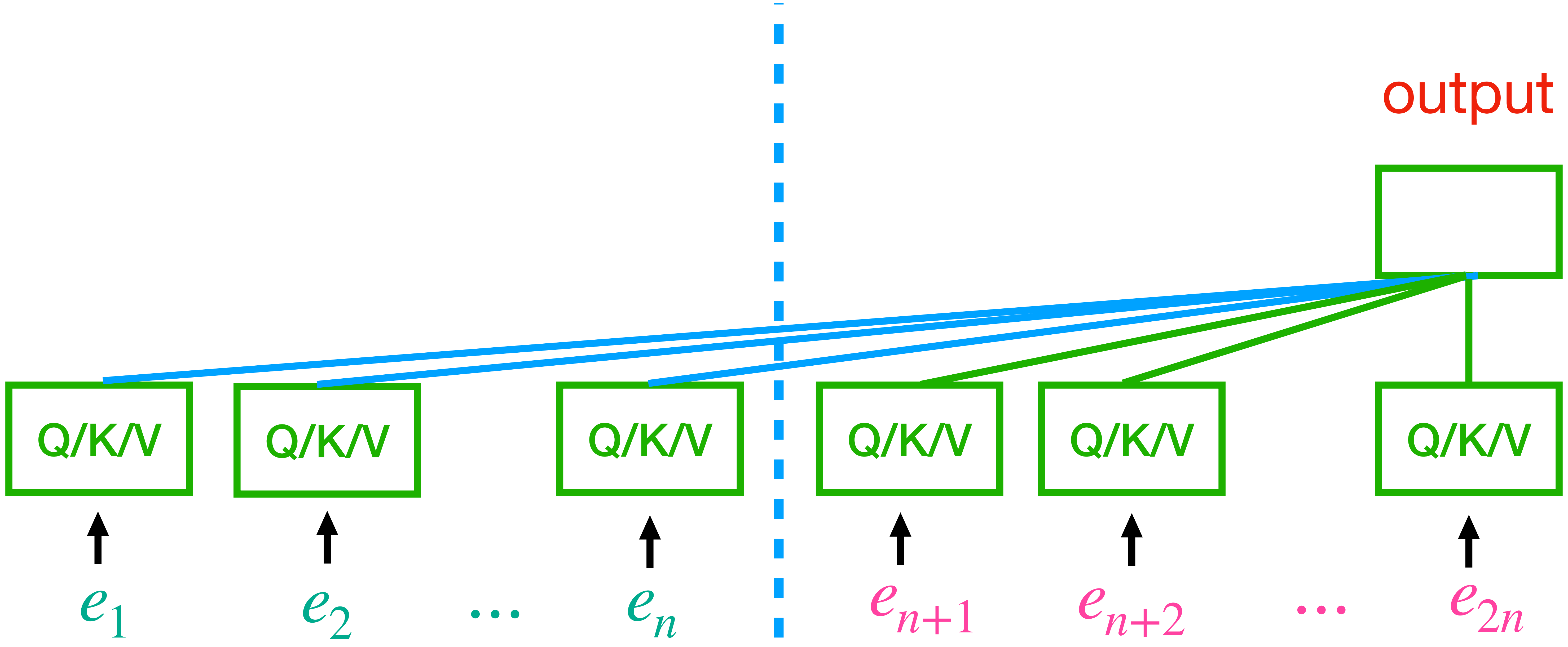
Theorem: To decide whether there is a Length-2 Path, they must exchange at least $\Omega(n)$ bits of information!

Warm up: One Layer - Example 1

n: prompt length
d: hidden dimension
H=1: attention head
p: precision (bits)

Theorem (Mixture of Parrot by Jelassi et al.'25)

A single-layer transformer cannot find a length-2 path in a graph unless $dp = \Omega(n)$.



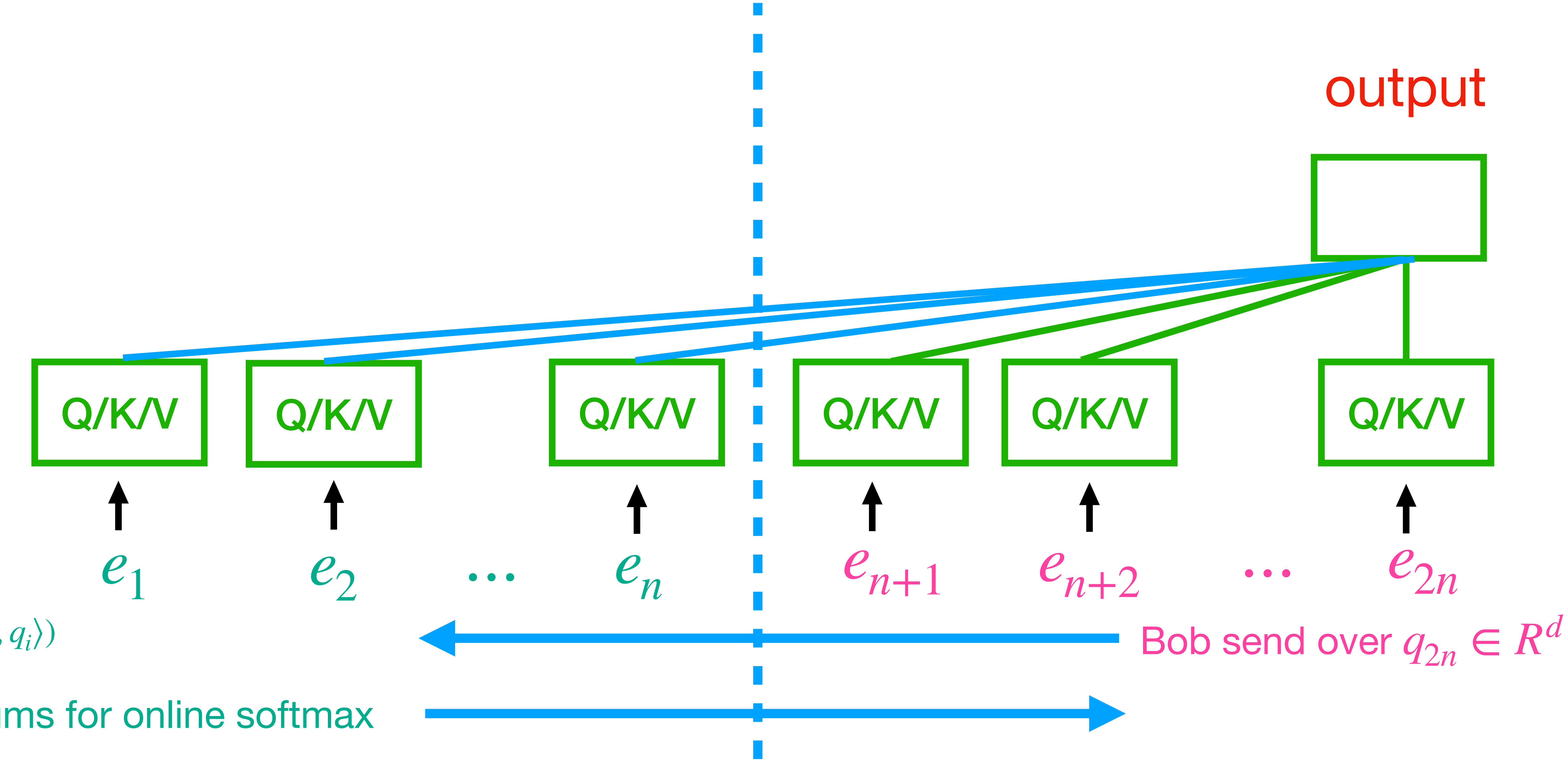
Key: For 1 layer, only the last queries matter!

Warm up: One Layer - Example 1

n: prompt length
d: hidden dimension
H=1: attention head
p: precision (bits)

Theorem (Mixture of Parrot by Jelassi et al.’25)

A single-layer transformer cannot find a length-2 path in a graph unless $dp = \Omega(n)$.



$\sum_{i=1}^{n} \exp(\langle k_i, q_i \rangle) v_i, \quad \sum_{i=1}^{n} \exp(\langle k_i, q_i \rangle)$

Alice send back partial sums for online softmax

Warm up: One Layer - Example 2

n: prompt length
d: hidden dimension
H=1: attention head
p: precision (bits)

Theorem (Sanford et al.'24)

A single-layer transformer cannot solve the induction head task unless $dp = \Omega(n)$.

Problem (Induction-head)

a b c d c a * b d a b *

Find the character before the last occurrence of the same token.

Problem (Indexing)

s = [a, b, c, d, c, a, b, d, a, b]

s[i] = ?

Alice has s while Bob has i

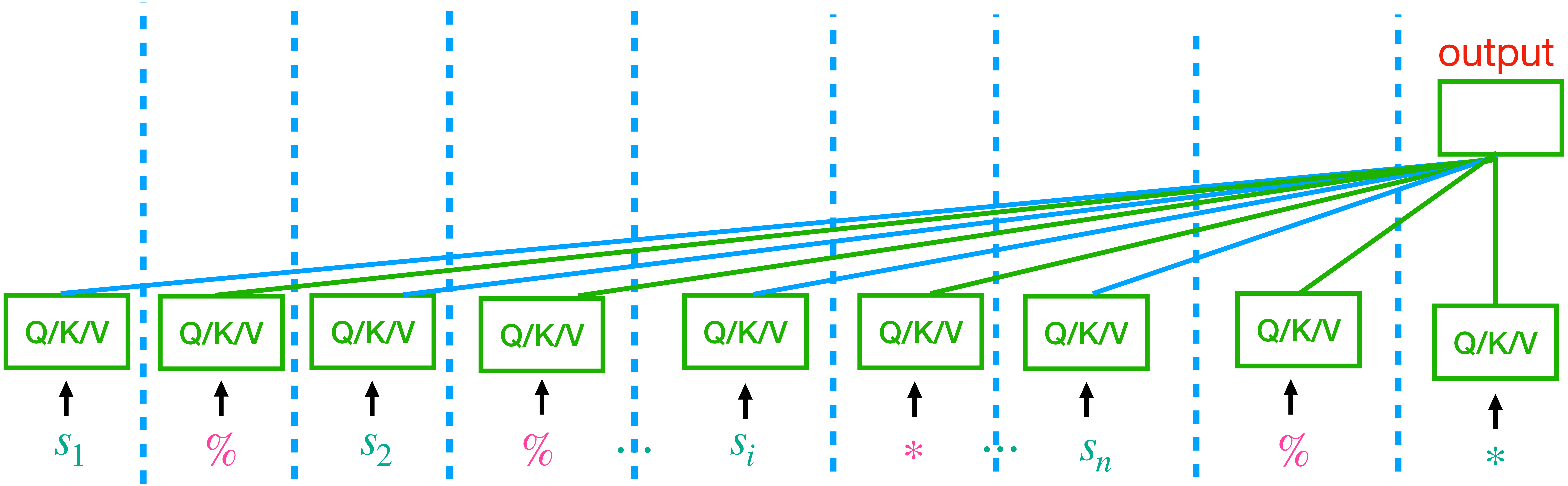
Theorem: One-way (Alice ➡️ Bob) communication complexity of indexing = $\Omega(n)$

Warm up: One Layer - Example 2

n: prompt length
d: hidden dimension
H=1: attention head
p: precision (bits)

Theorem (Sanford et al.'24)

A single-layer transformer cannot solve the induction head task unless $dp = \Omega(n)$.

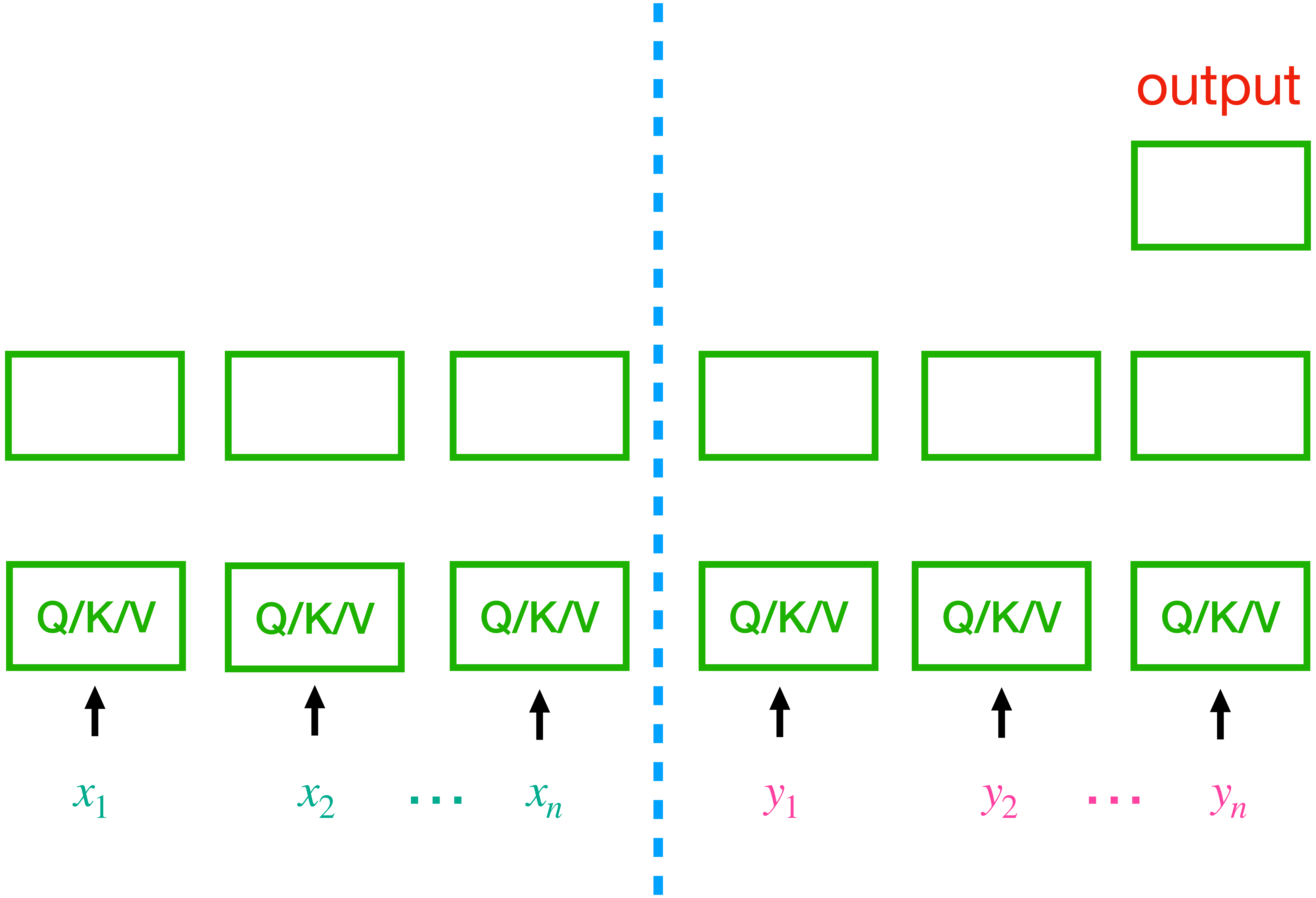


Alice has all the odd positions, while Bob has all the even positions.

Why did they stop at 1-layer?

- Constant-layer encoder can simulate constant-depth threshold circuits TC^0.
- We do not understand TC^0 or even $\mathsf{THR} \circ \mathsf{THR}$! (Complexity Theory's Waterloo)
- But why not even 2-layer decoders?

Why did they stop at 1-layer?

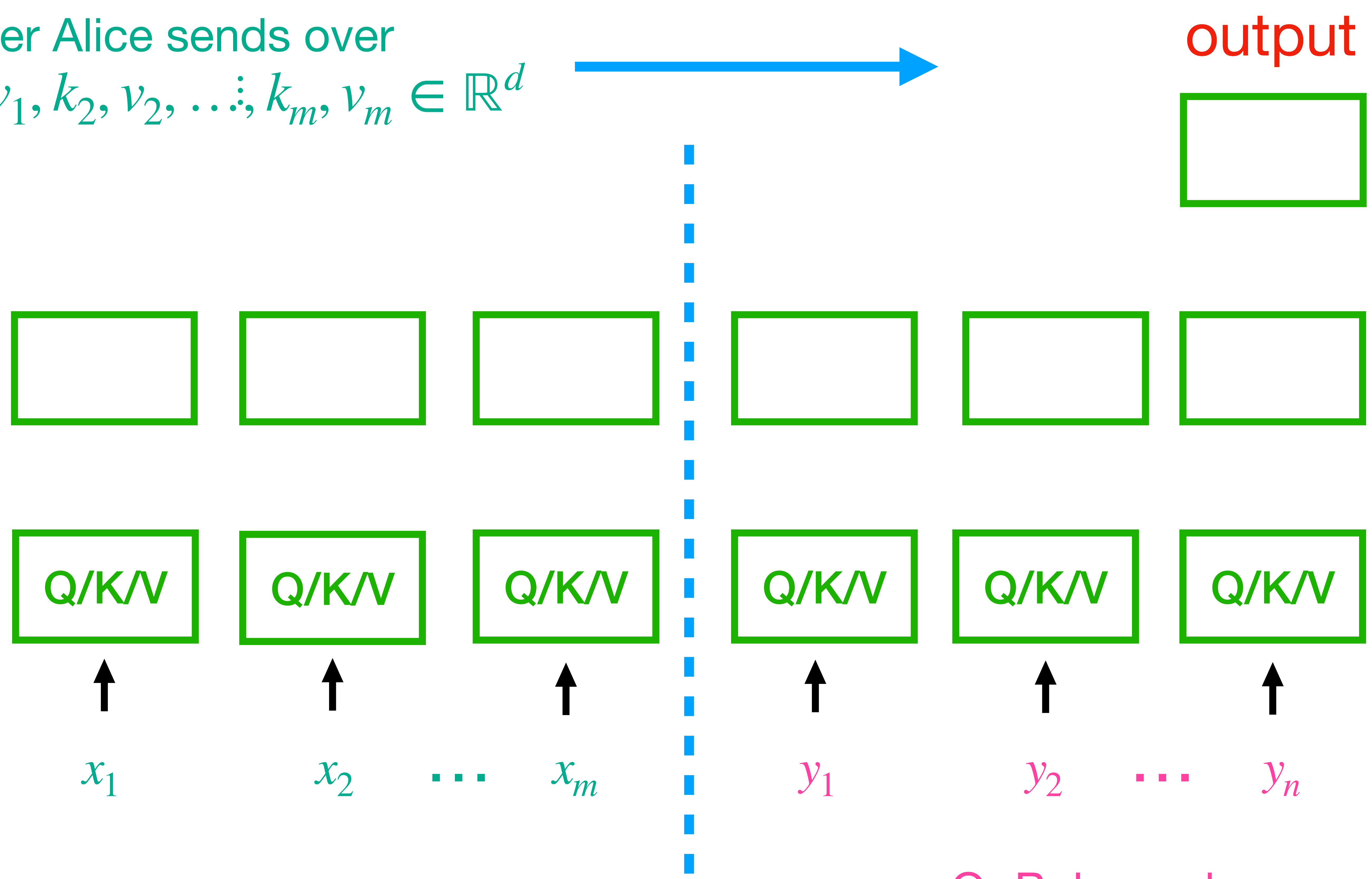


Now we need to first compute every hidden state on the second level.

=> Every query matters now!

Why did they stop at 1-layer?

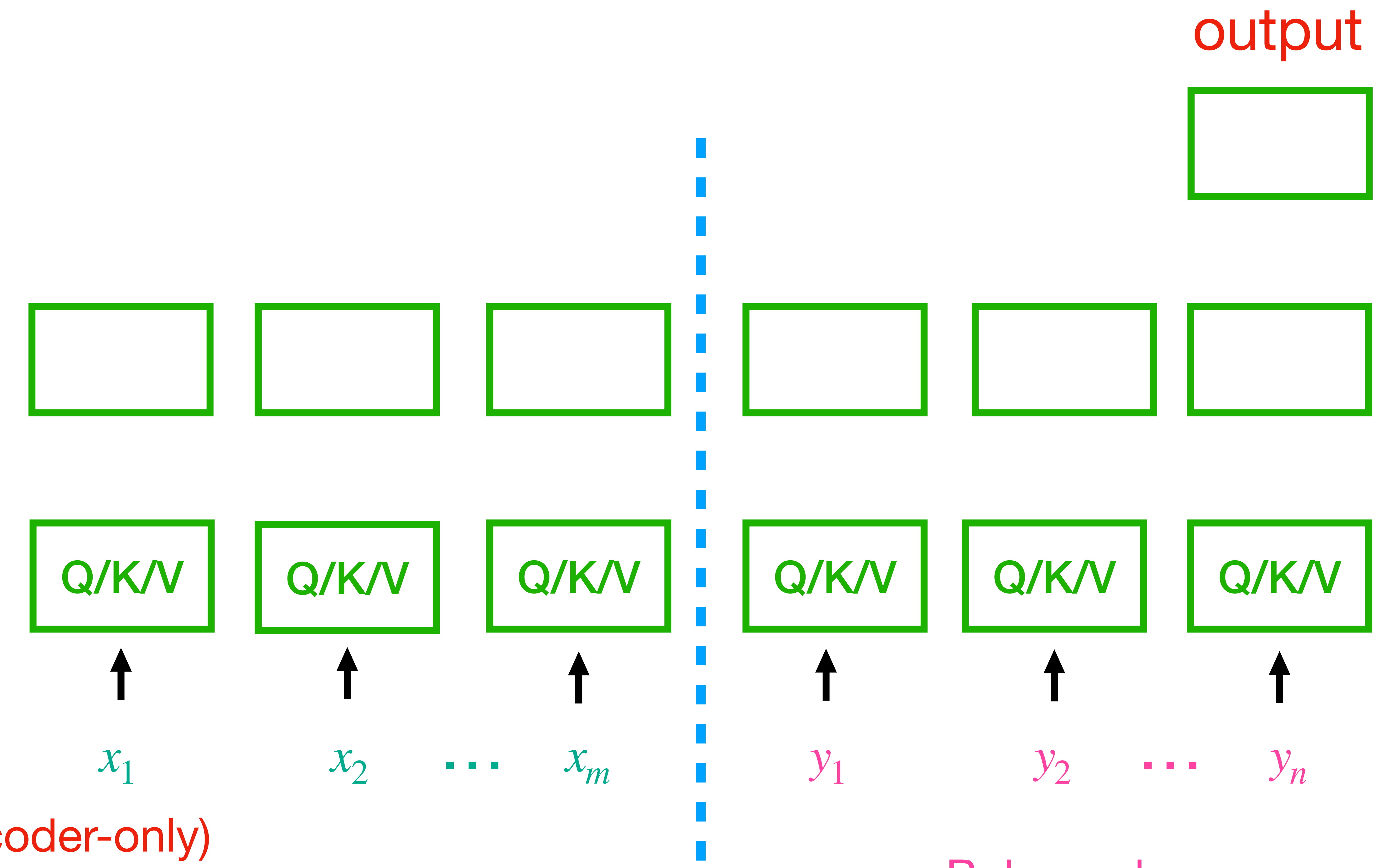
Either Alice sends over
$k_1, v_1, k_2, v_2, \ldots, k_m, v_m \in \mathbb{R}^d$



Or Bob sends over
$q_{m+1}, q_{m+2}, \ldots, q_{m+n} \in \mathbb{R}^d$

They have to send $\Omega(dm)$ / $\Omega(dn)$ information,
which is more than enough to reveal their input.

Autoregressive (“forgetful”) communication



Bob sends over
$q_{m+1}, q_{m+2}, \ldots, q_{m+n} \in \mathbb{R}^d$

Our Simple Observation (decoder-only)

Communication game fails to capture:

After Bob sends his queries, and Alice replies with her partial sums, Alice “forgets” these queries!

Autoregressive ("forgetful") communication

- N players, where the player $i \in [N]$ receives z_i as input.
- L epochs of communication
- For $\ell = 1,2,\ldots,L$, the ℓ-th epoch proceeds as follows
  - For each player $i \in [N]$, player i sends a message $\Gamma_{i,j}^{(\ell)}$ to previous player $j \in [1 : i-1]$
    - The player j, based on its own information state $X_j^{(\ell-1)}$, and the message $\Gamma_{i,j}^{(\ell)}$, replies with a message $\Pi_{j,i}^{(\ell)}$ to player i
  - The player i updates its information state as

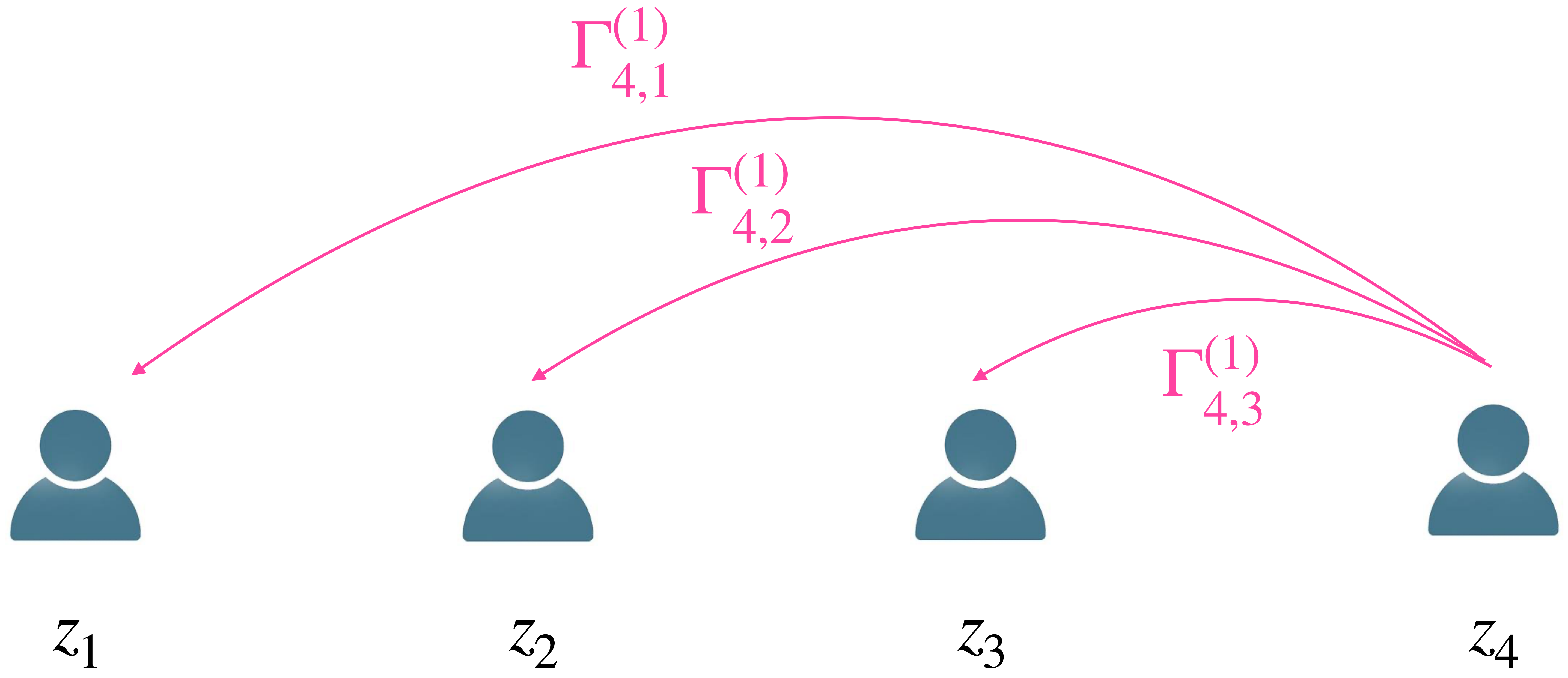
Inductively defined, initially $X_j^{(0)} := z_j$

$$X_i^{(\ell)} := X_i^{(\ell-1)} \cup \bigcup_{j<i} \Pi_{j,i}^{(\ell)}$$

Example

Number of players $N = 4$
Epochs of communication $L = 1$

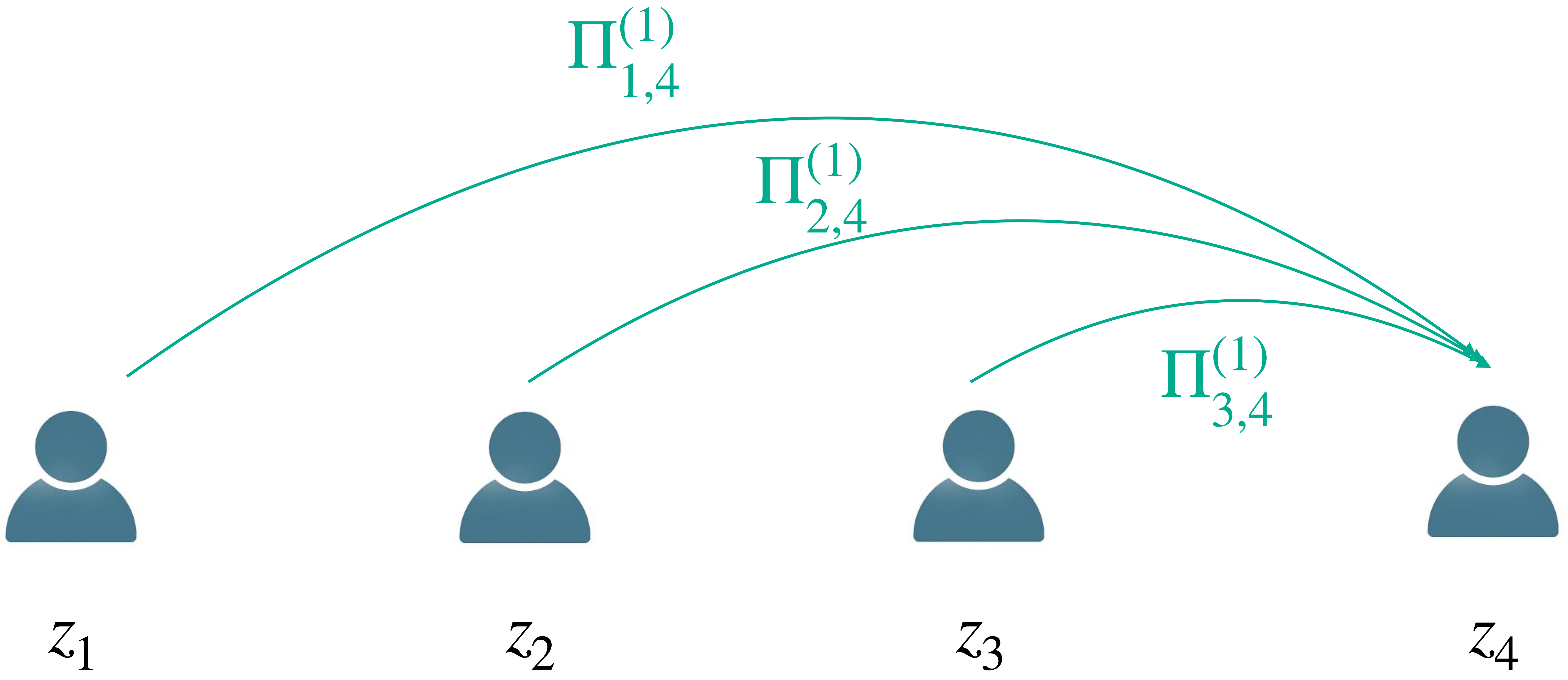
$\Gamma^{(1)}_{4,1}, \Gamma^{(1)}_{4,2}, \Gamma^{(1)}_{4,3}$ depends only on $X^{(0)}_1 := z_1$



Example

Number of players $N = 4$
Epochs of communication $L = 1$

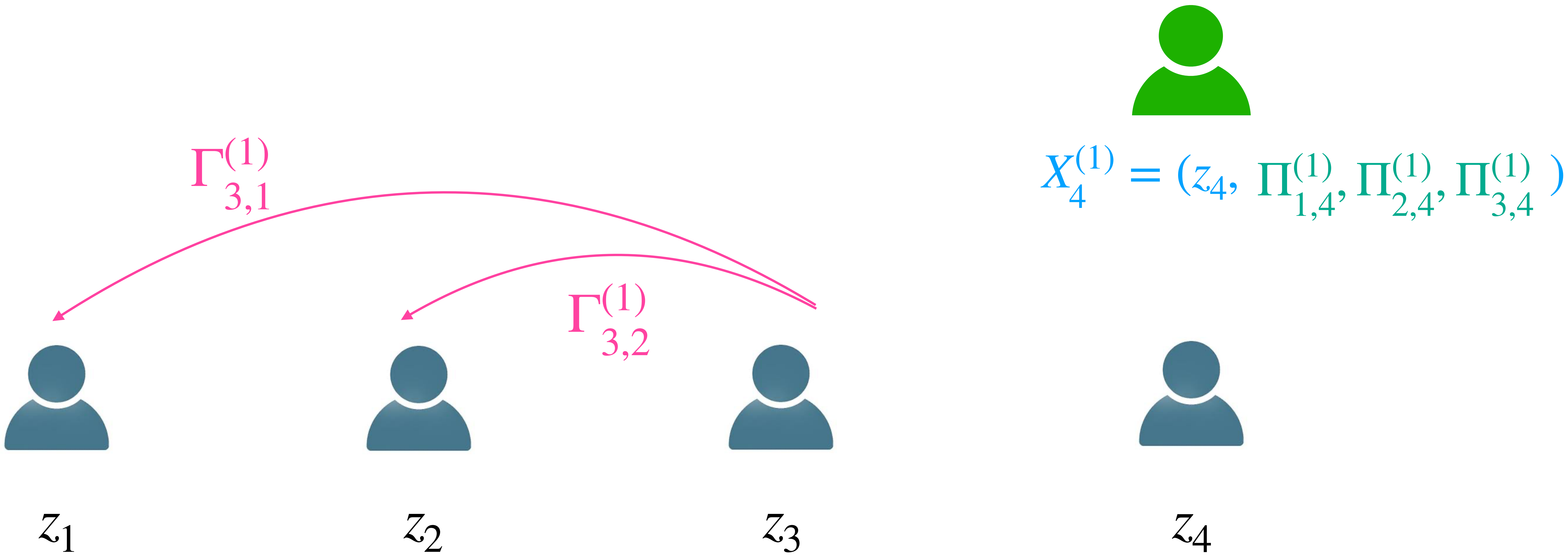
$\Pi^{(1)}_{1,4}$ depends on $X^{(0)}_{1} := z_1$ and $\Gamma^{(1)}_{4,1}$



Example

Number of players $N = 4$
Epochs of communication $L = 1$

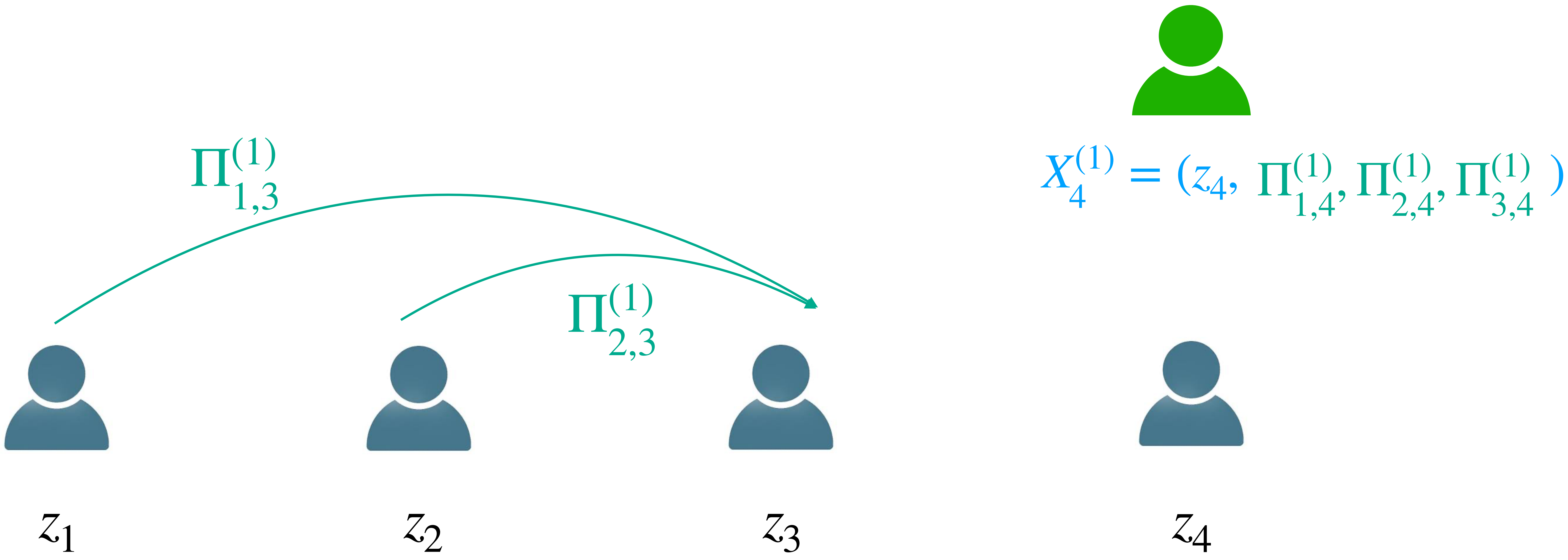
$\Gamma^{(1)}_{3,1}, \Gamma^{(1)}_{3,2},$ depends only on $X^{(0)}_3 := z_3$



Example

Number of players $N = 4$

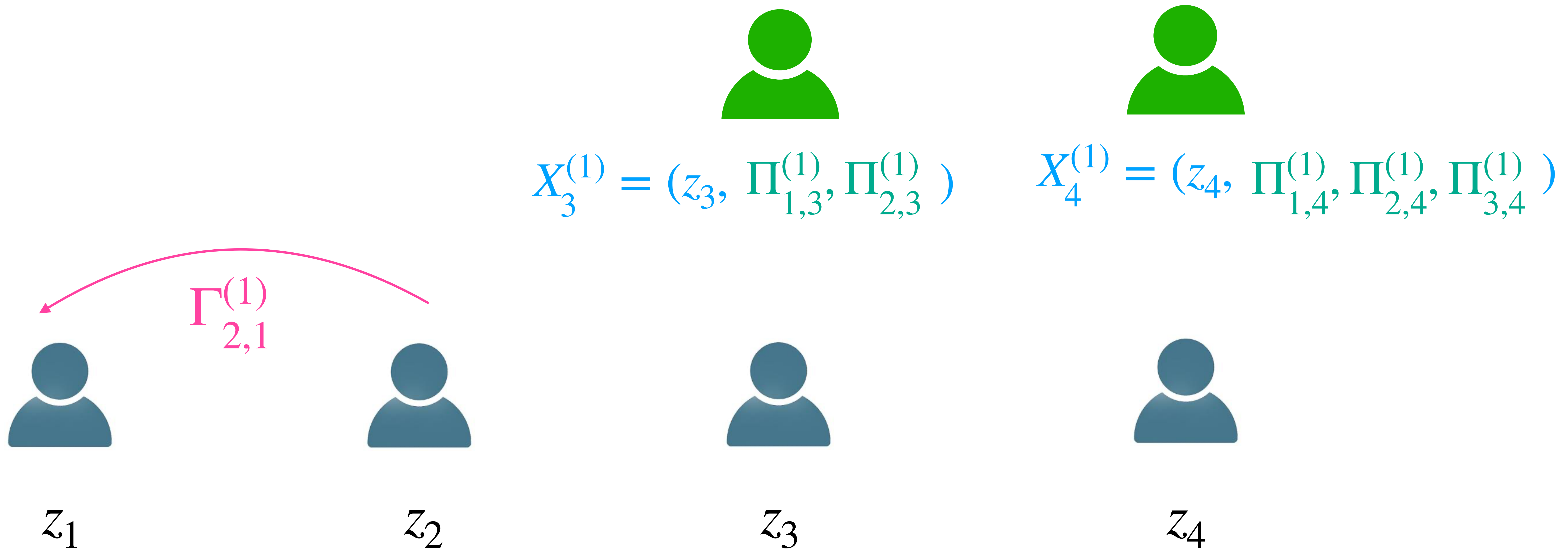
Epochs of communication $L = 1$



Example

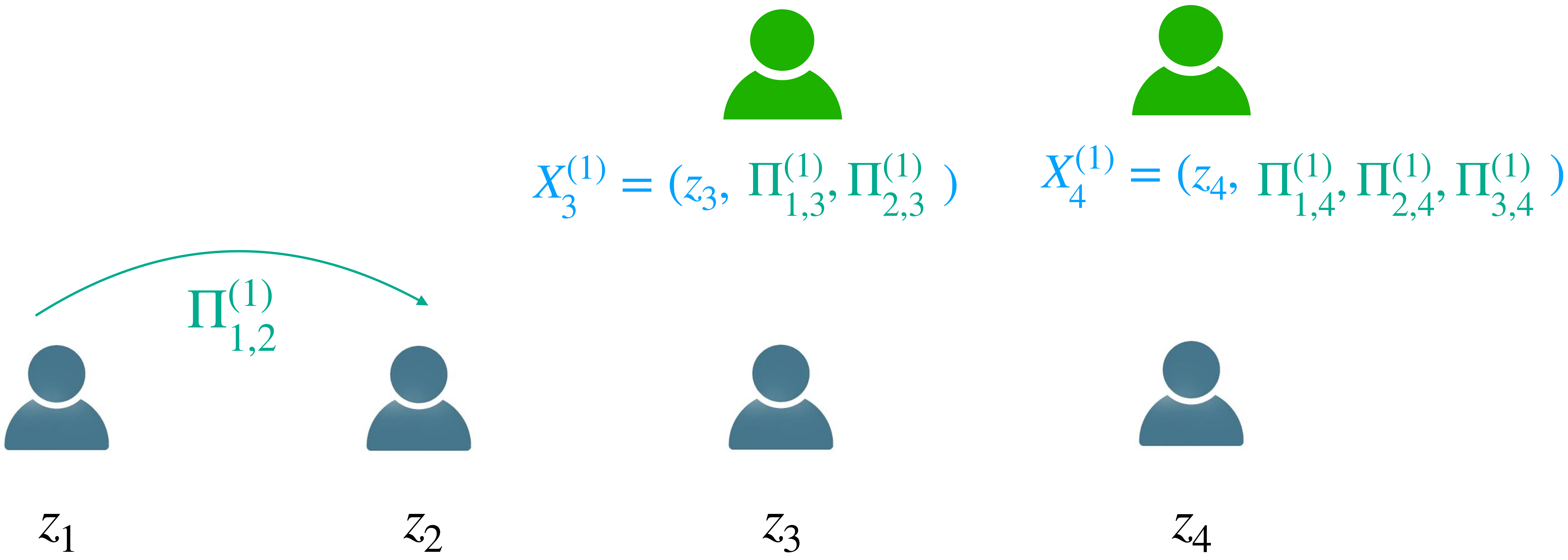
Number of players $N = 4$

Epochs of communication $L = 1$



Example

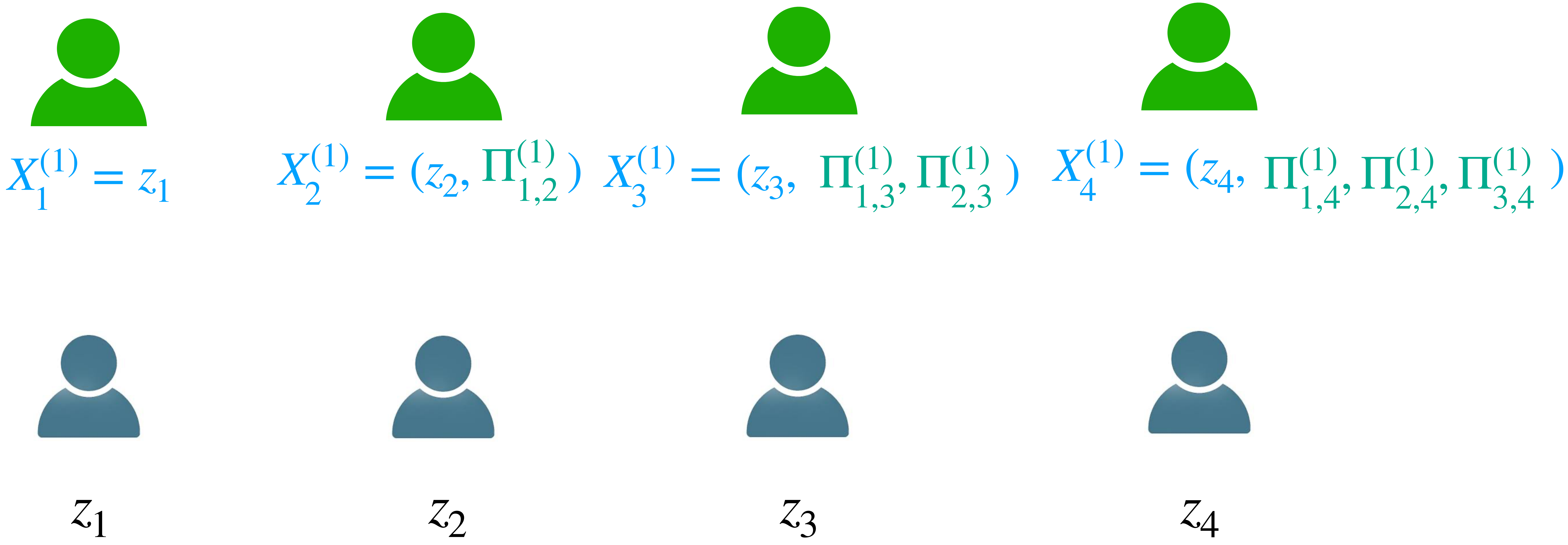
Number of players $N = 4$
Epochs of communication $L = 1$



Example

Number of players $N = 4$

Epochs of communication $L = 1$



Our Results

n: prompt length
d: hidden dimension
H: attention head
p: precision

- **Theorem.** An L-layer decoder-only Transformer could not solve L-sequential function composition unless $Hdp \geq n^{2^{-4L}}$

L-step composition requires $\Omega(L)$ layers of Transformer (asymptotically)

Composition

- L-Sequential function composition: Given L functions $f_1, \ldots, f_L$ and a query $w = (w_1, \ldots, w_L)$, compute

$$i_1 = f_1(w_1), i_2 = f_2(w_2, i_1), \ldots, i_L = f_L(w_L, i_{L-1})$$

  and output i_L

- The input prompt explicitly describes the functions in the order of $f_L, f_{L-1}, \ldots, f_1$, followed by the query w.

Example

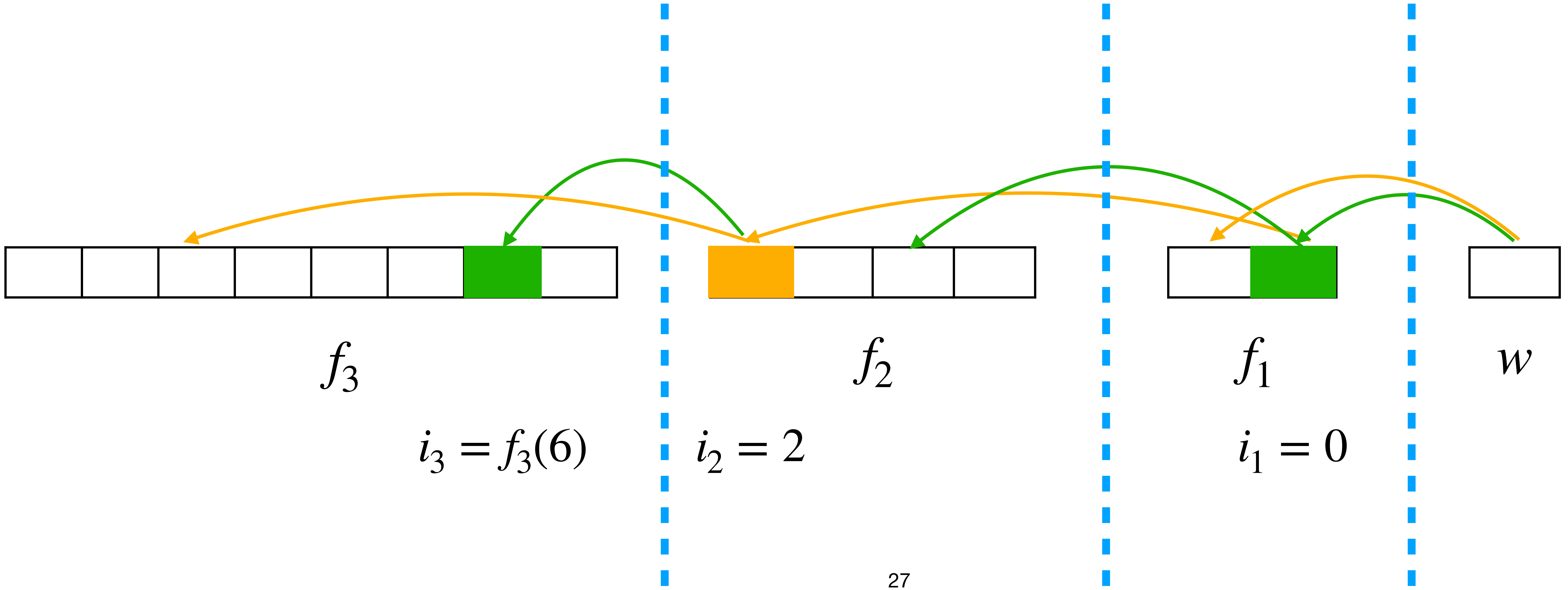
- $L = 3, w_1, w_2, w_3 \in \{0,1\}$
- $i_1 = f_1(w_1), i_2 = f_2(w_2, i_1), i_3 = f_3(w_3, i_2)$
- $f_1 : [2] \to [2], f_2 : [4] \to [4], f_3 : [8] \to [8]$

$w = w_1 w_2 w_3 = 101$

$w_1 = 1$

$w_2 = 0$

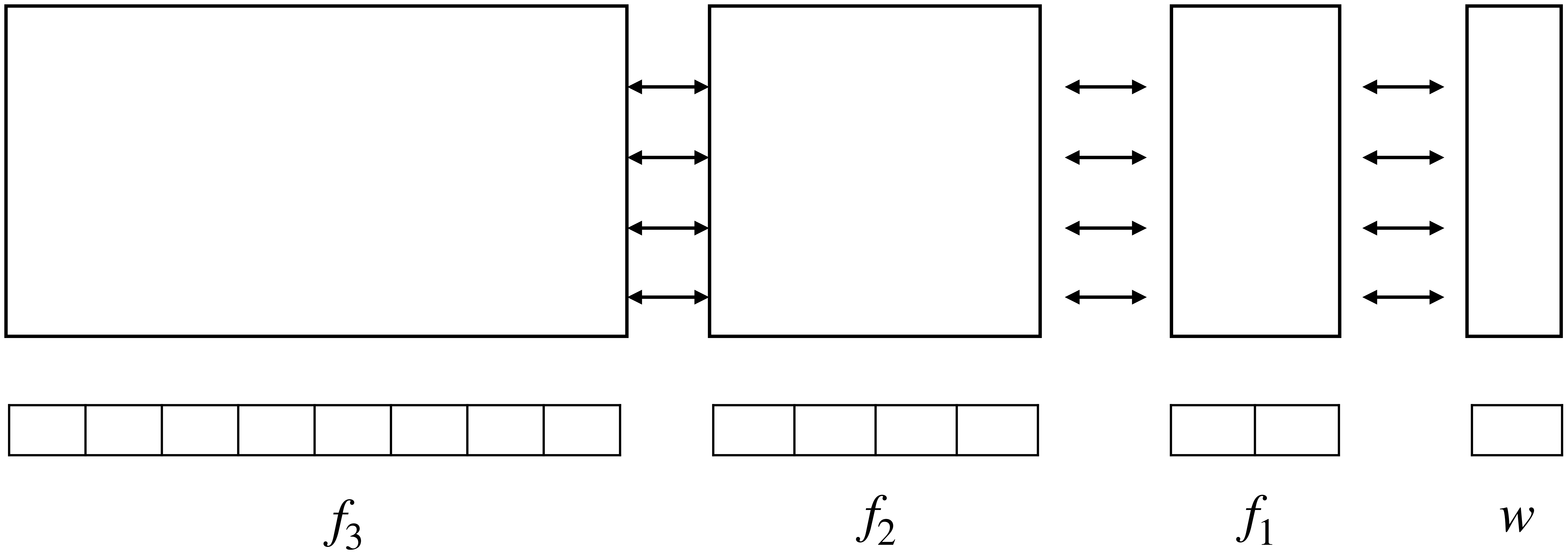
$w_3 = 1$



Intuition

- When important information is presented at the end of sequence, autoregressive Transformer fails to retrieve information efficiently

  Massive computation are “wasted” at the beginning of the sequence



Remarks on the proof

- How we actually prove the lower bound
  - Indistinguishable decomposition: Find $R_{\leq \ell}$ and $Z_{>\ell}$, such that for every inputs $z_{>\ell} \in Z_{>\ell}$ of players $> \ell$, all assignments in $R_{\leq \ell}$ of players $[1 : \ell]$ are indistinguishable after ℓ epochs (i.e., they lead to the same transcripts)
  - We additional need the indistinguishable decomposition we find to have large coverage on i_ℓ

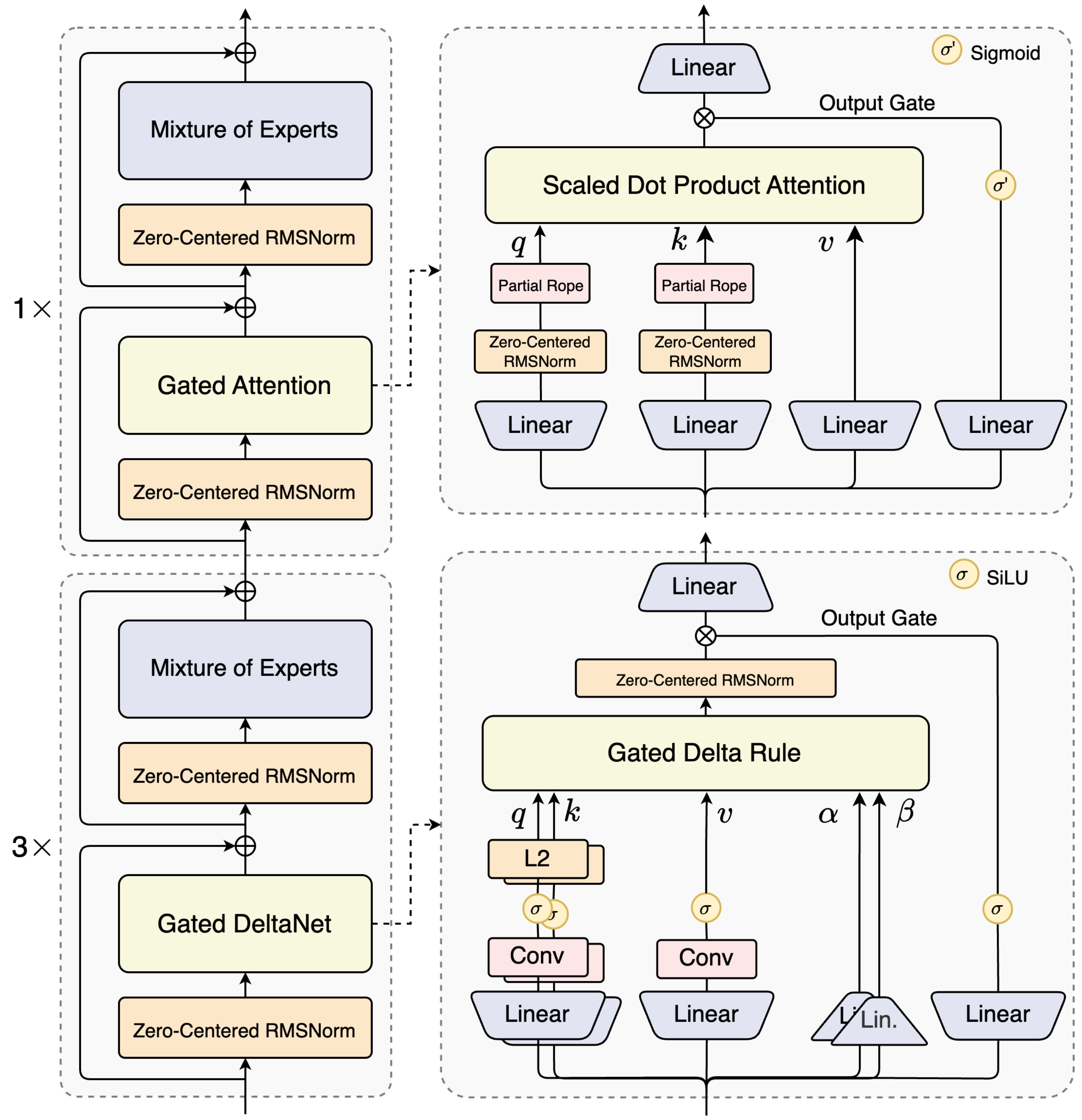
Take away

- The first unconditional lower bound of multi-layer (decoder-only) Transformer

- Key concept idea: Autoregressive communication
  - To study computation model of Transformer, it is crucial to study new communication model (this leads to stronger and unconditional lower bound)

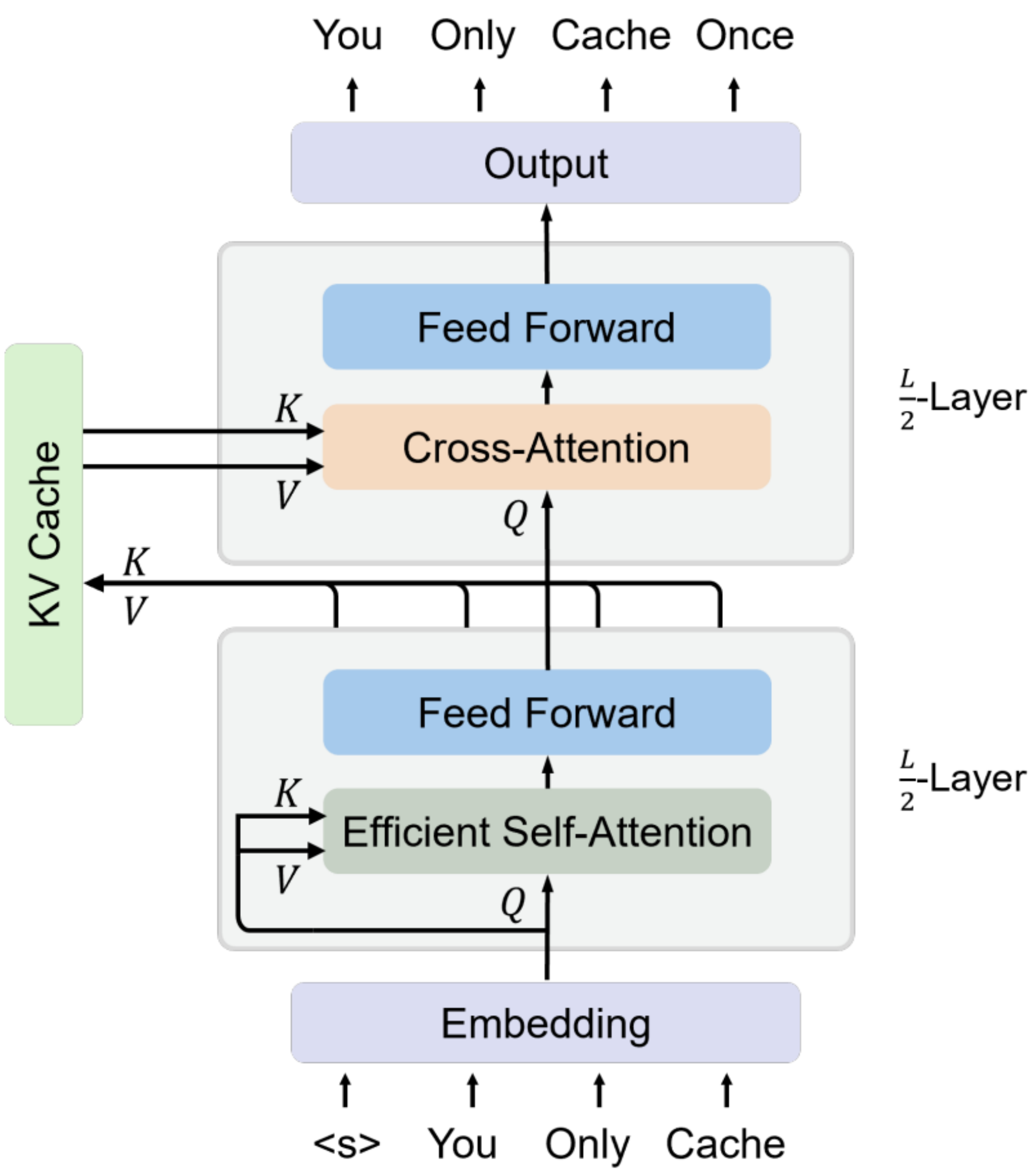
- The communication perspective offers yet another way of looking at architectures.

Discussions - Hybrid models



Qwen3-Next

Discussions - KV-cache sharing



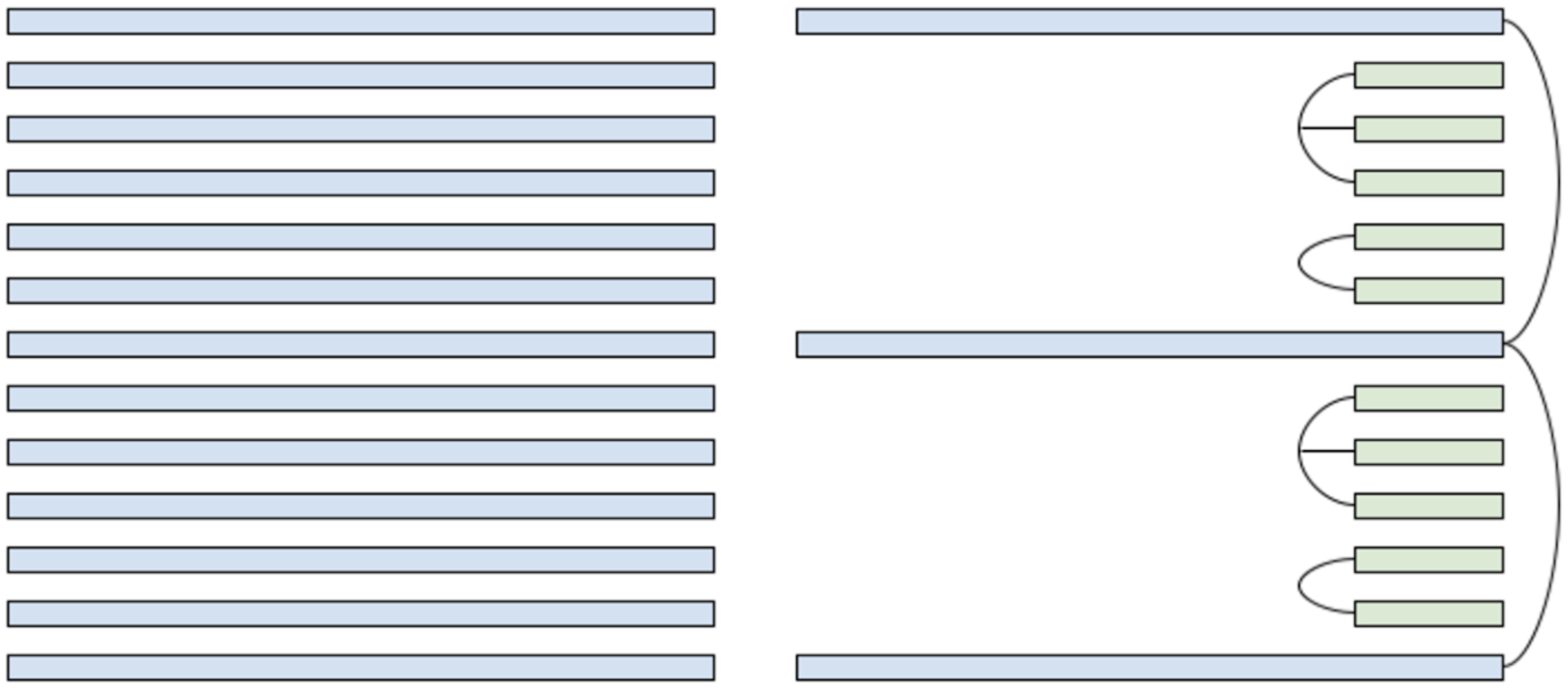


Figure 1. Left: Standard transformer design where every attention is global attention. Right: The attention design in our production model. Blue boxes indicate global attention, green boxes indicate local attention, and curves indicate KV-sharing. For global attention layers, we share KV across multiple non-adjacent layers. This illustration depicts only a subset of the layers in the full model.

YOCO [Sun et al.]

Character.ai blog

Thanks!