Reducing Solid-State Drive Read Latency by Optimizing Read-Retry

Jisung Park[1], Myungsuk Kim[2], Myoungjun Chun[2],
Lois Orosa[1], Jihong Kim[2], and Onur Mutlu[1]

[1] SAFARI ETH zürich

[2] VERI LUX MEA TAS

ASPLOS 2021 (Session 17: Solid State Drives)

Executive Summary

- **Problem:** Long read latency in modern SSDs due to read-retry
  - Multiple retry steps required to read an erroneous page
  - Read-retry frequently occurs in modern NAND flash memory

- **Goal:** Reduce the latency of each read-retry operation

- **Key Ideas:**
  - **Pipelined Read-Retry (PR^2):** Concurrently perform consecutive retry steps using the CACHE READ command
  - **Adaptive Read-Retry (AR^2):** Reduce read-timing parameters for every retry step by exploiting the reliability margin provided by strong ECC

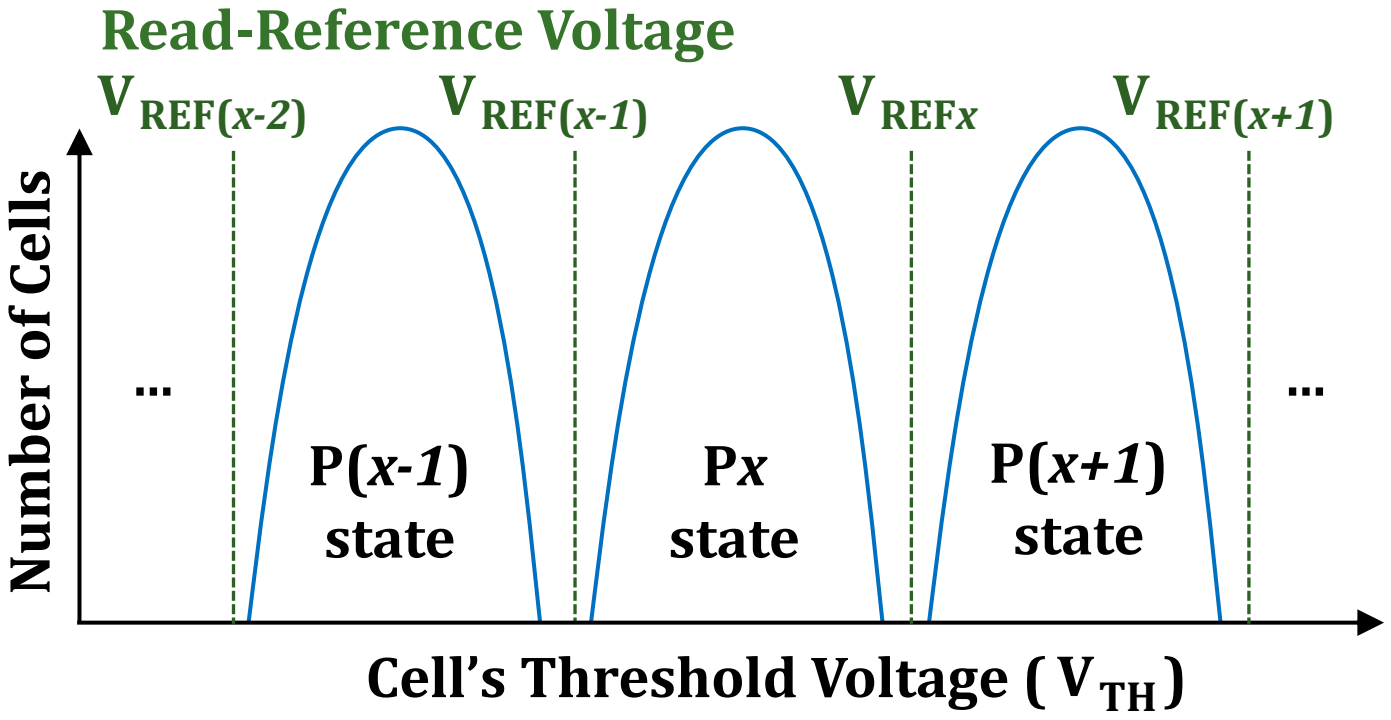
- **Evaluation Results:** Our proposal improves SSD response time by
  - Up to 51% (35% on average) compared to a high-end SSD
  - Up to 32% (17% on average) compared to a state-of-the-art baseline

Talk Outline

- Read-Retry in Modern NAND Flash-Based SSDs
- PR^2: Pipelined Read-Retry
- AR^2: Adaptive Read-Retry
- Evaluation Results

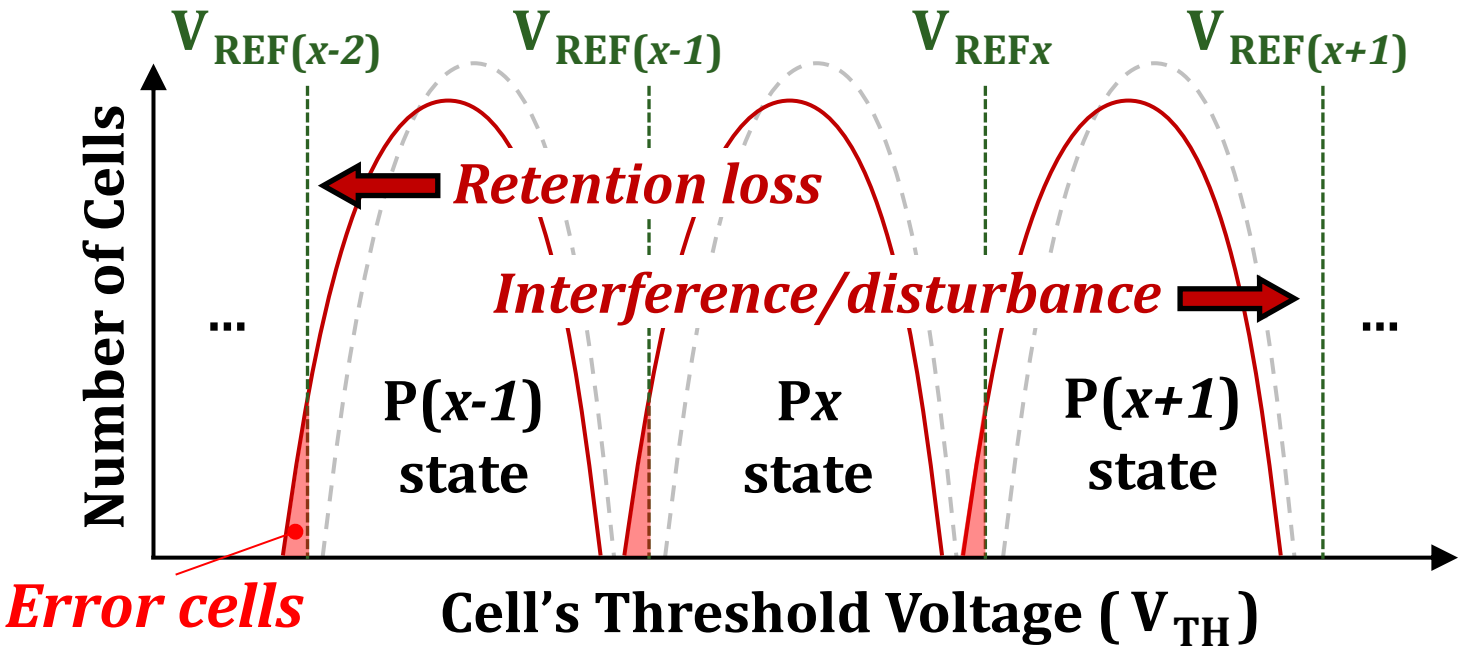
Errors in NAND Flash Memory

- NAND flash memory stores data by using cells' V_{TH} values



Errors in NAND Flash Memory

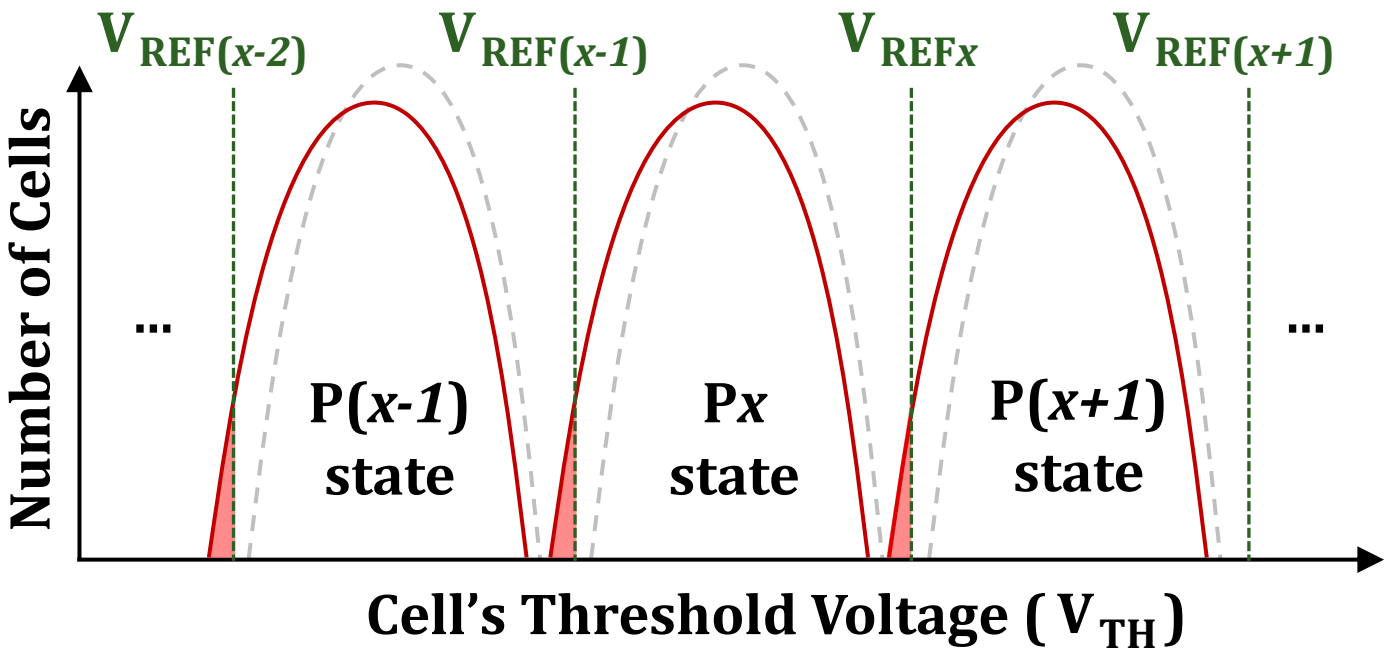
- Various sources shift and widen programmed V_{TH} states
  - Retention loss, program interference, read disturbance, etc.



# of error cells > ECC correction capability
→ Uncorrectable errors in stored data

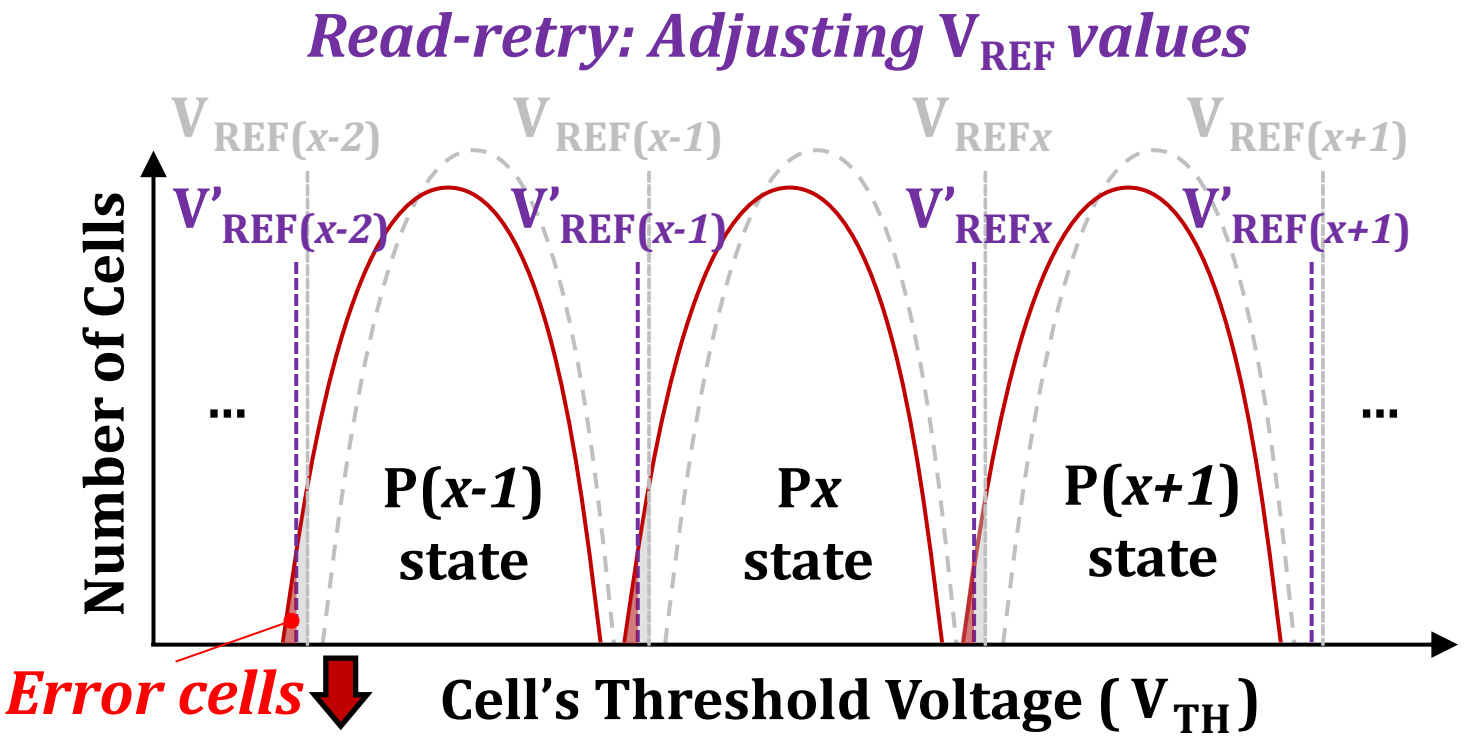
Read-Retry Operation

- Reads the page again with adjusted V_{REF} values



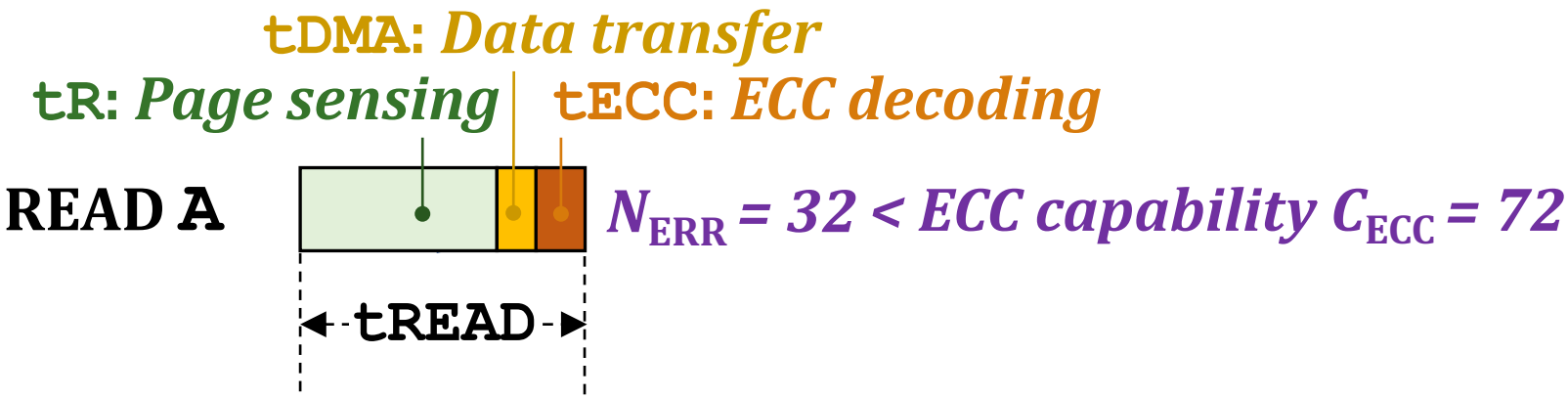
Read-Retry Operation

- Reads the page again with adjusted V_{REF} values

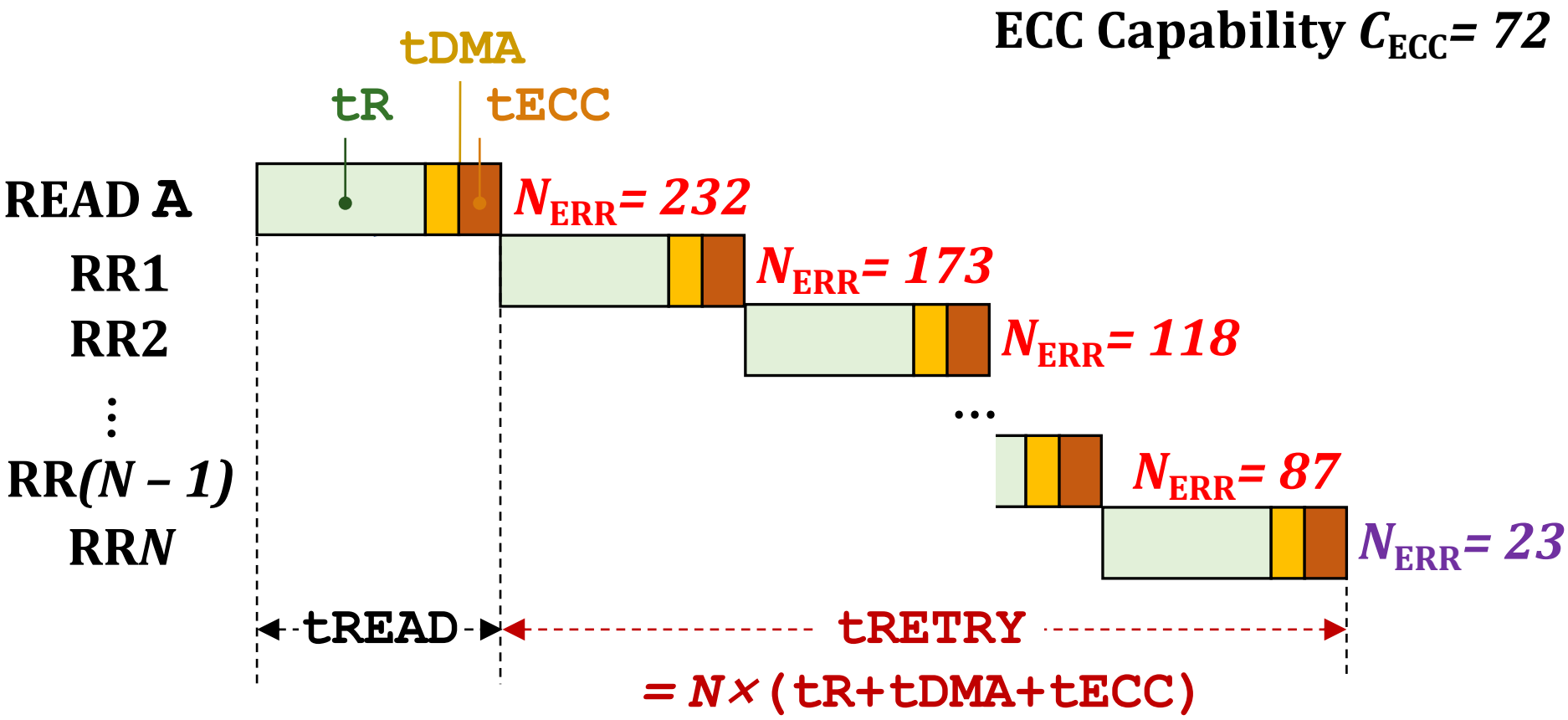


Read using properly-adjusted V_{REF} values
→ Decreases # of raw bit errors

Read-Retry: Performance Overhead



Read-Retry: Performance Overhead



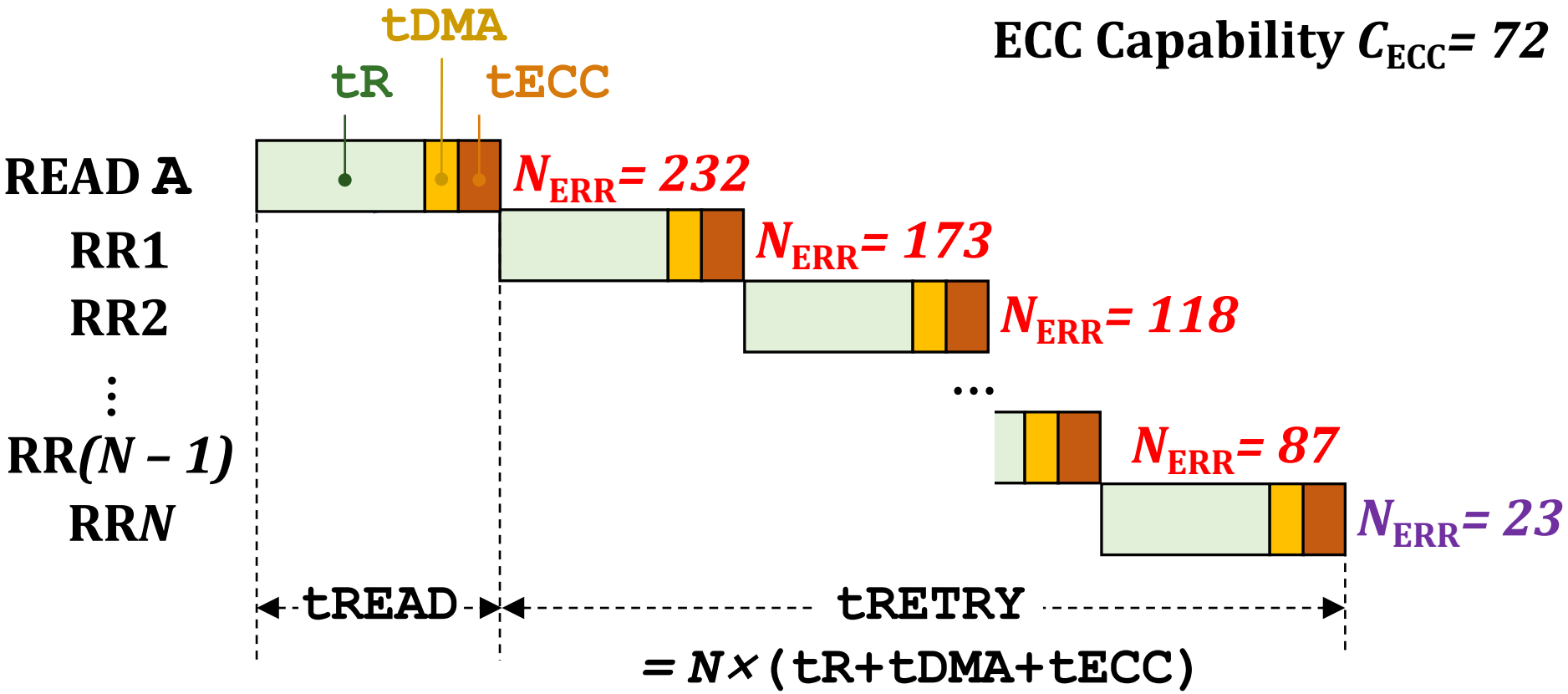
Read-retry increases the read latency almost linearly with the number of retry steps

Talk Outline

- Read-Retry in Modern NAND Flash-Based SSDs
- PR^2: Pipelined Read-Retry
- AR^2: Adaptive Read-Retry
- Evaluation Results

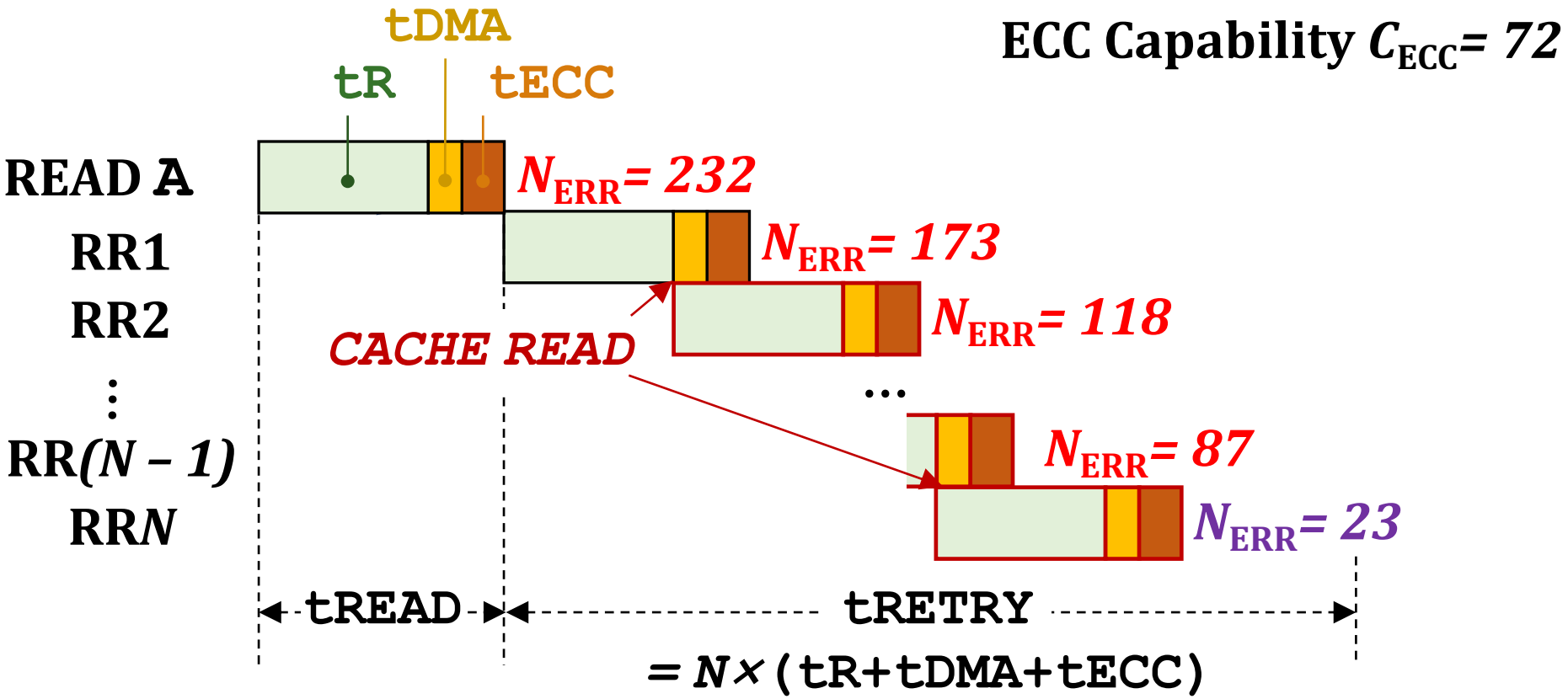
PR²: Pipelined Read-Retry

- **Key idea:** Concurrently perform consecutive retry steps



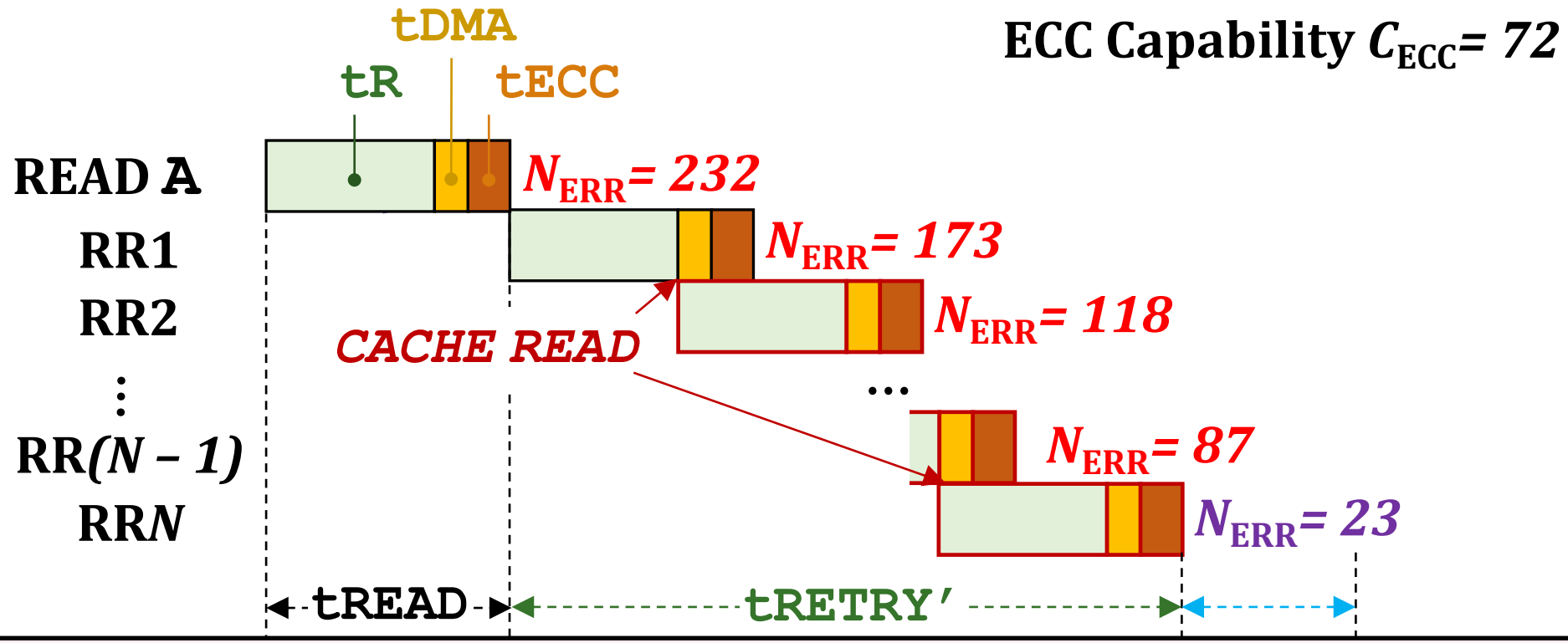
PR²: Pipelined Read-Retry

- **Key idea:** Concurrently perform consecutive retry steps



PR²: Pipelined Read-Retry

- **Key idea:** Concurrently perform consecutive retry steps

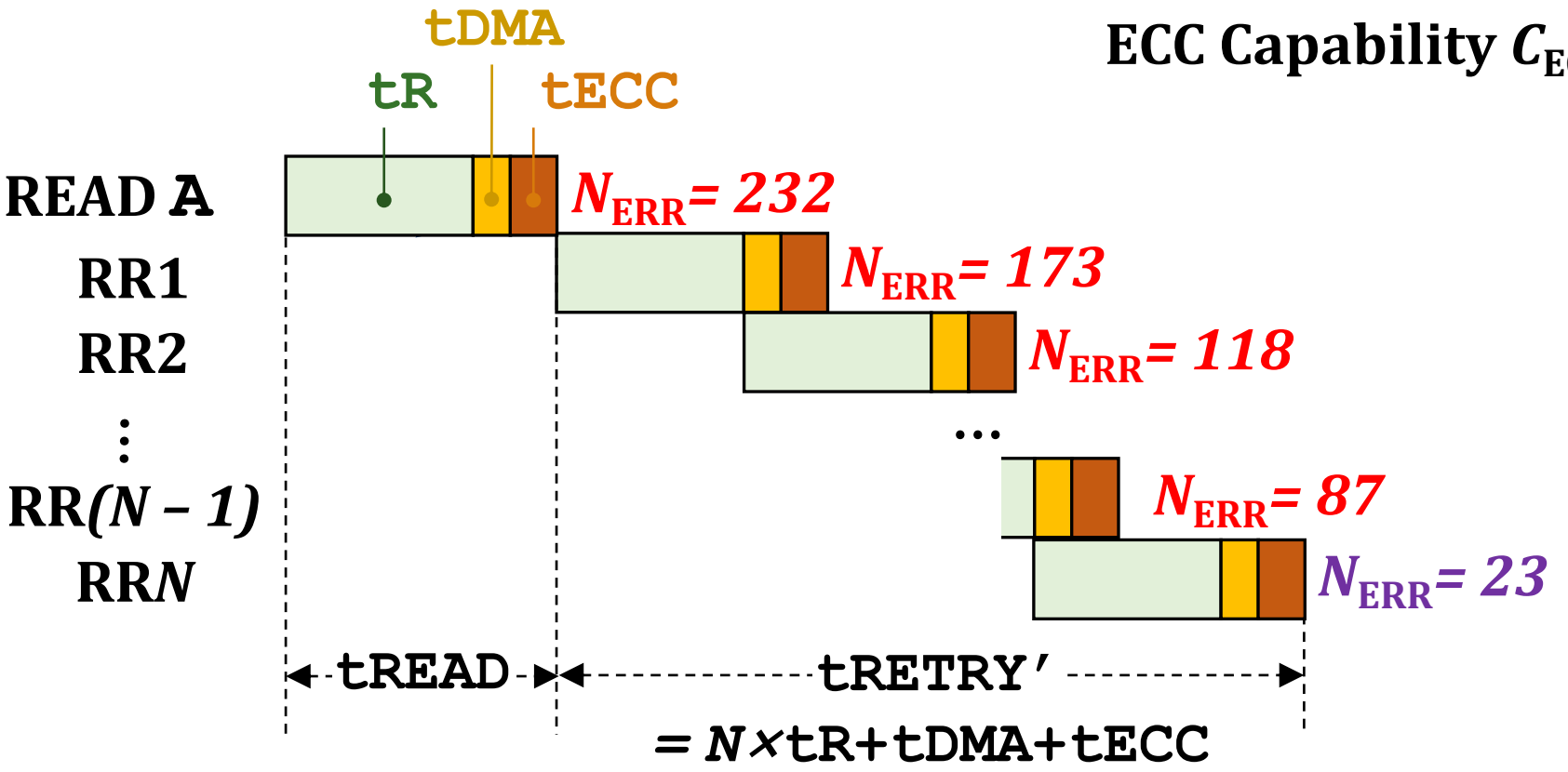


PR²: Removes tDMA & tECC (~30% of each retry step) from the critical path

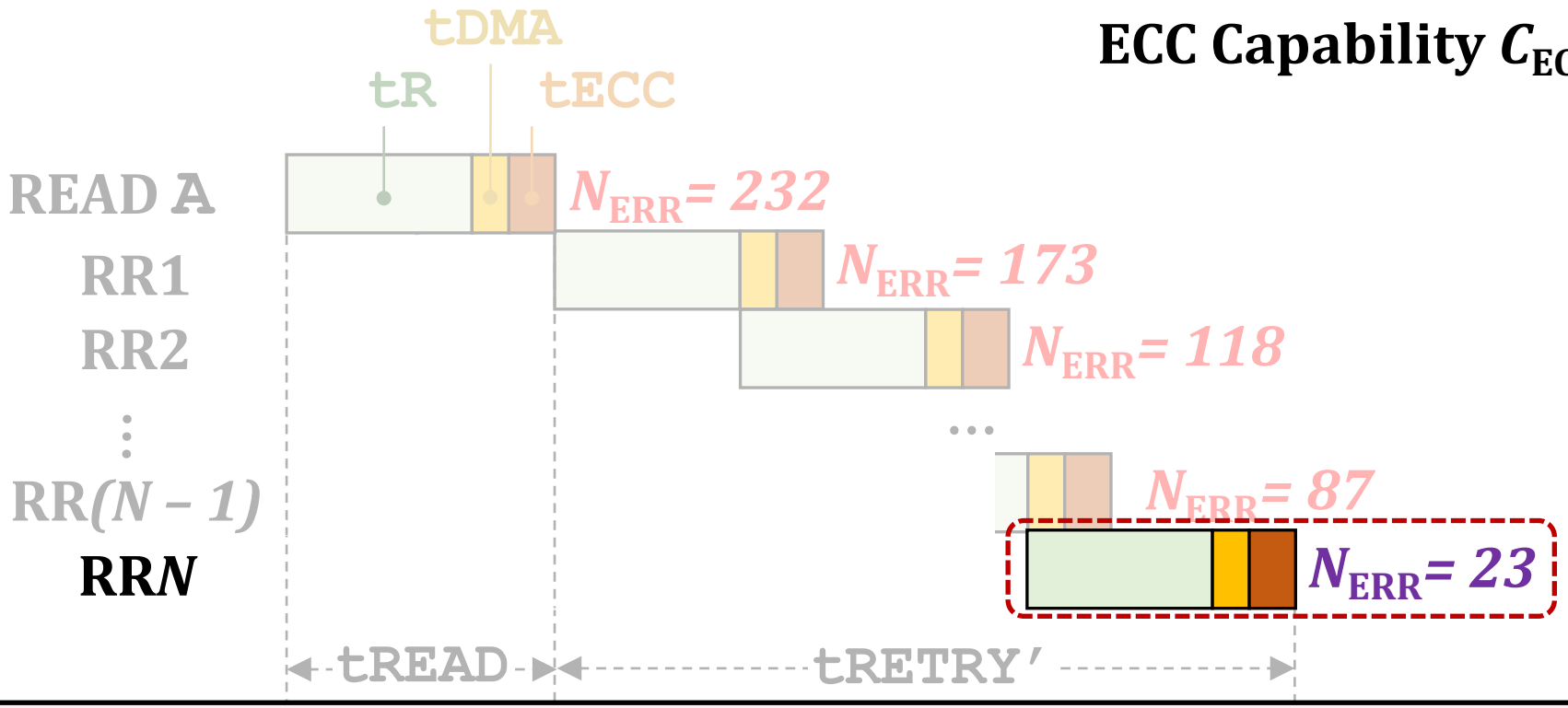
Talk Outline

- Read-Retry in Modern NAND Flash-Based SSDs
- PR^2: Pipelined Read-Retry
- AR^2: Adaptive Read-Retry
- Evaluation Results

AR²: Adaptive Read-Retry



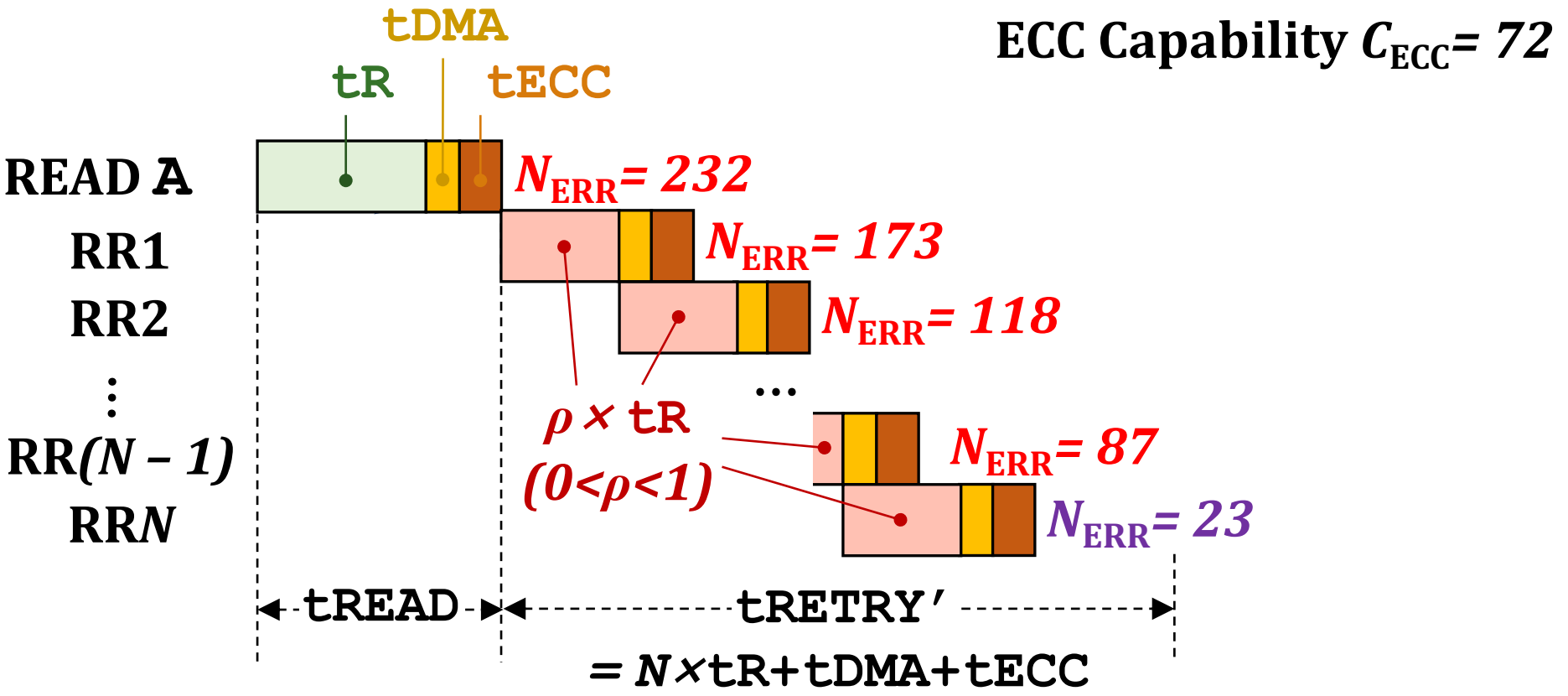
AR²: Adaptive Read-Retry



Observation: A positive ECC margin
in the final retry step when read-retry succeeds

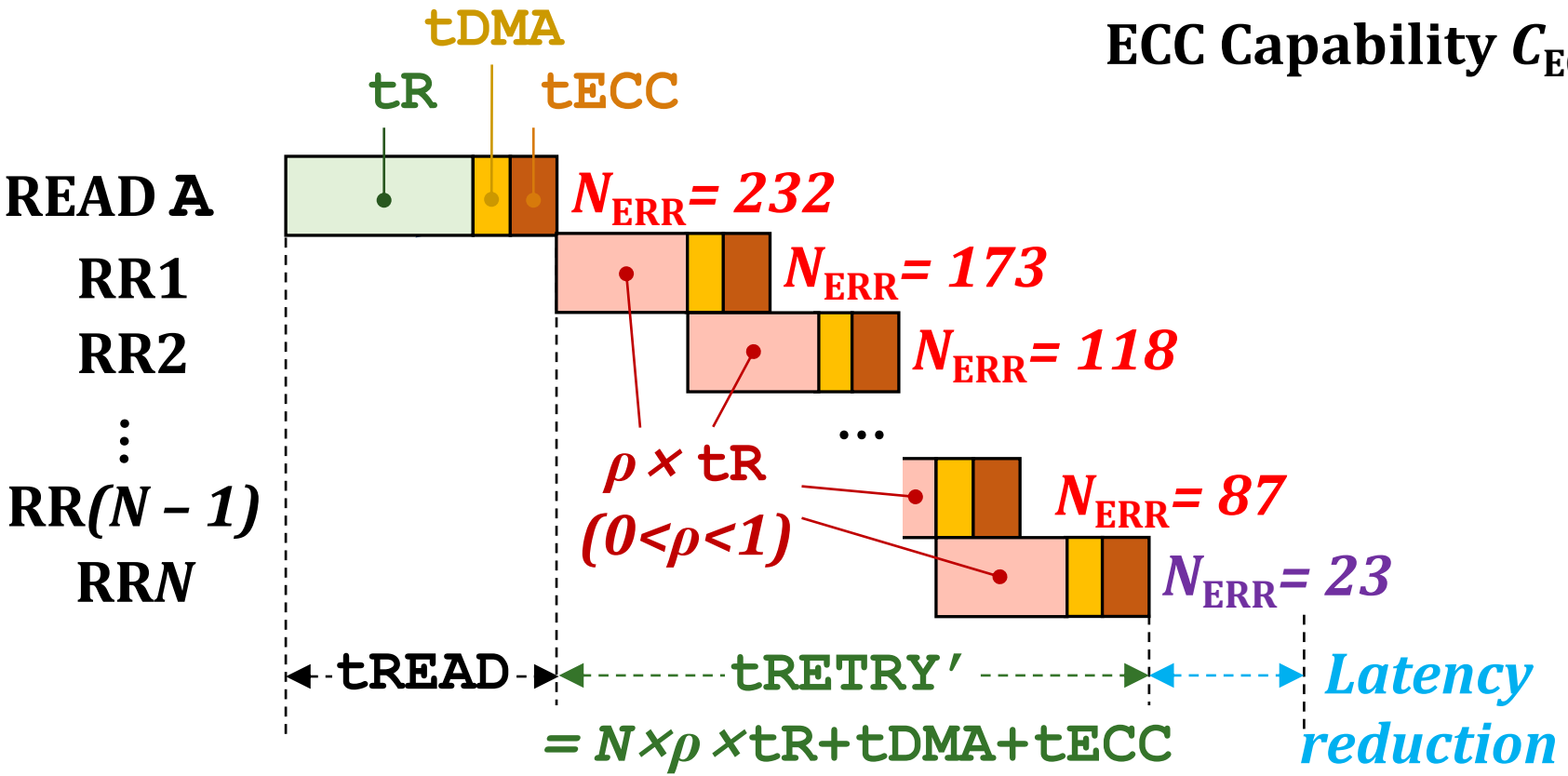
AR²: Adaptive Read-Retry

- **Key idea:** Reduce read-timing parameters for every retry step



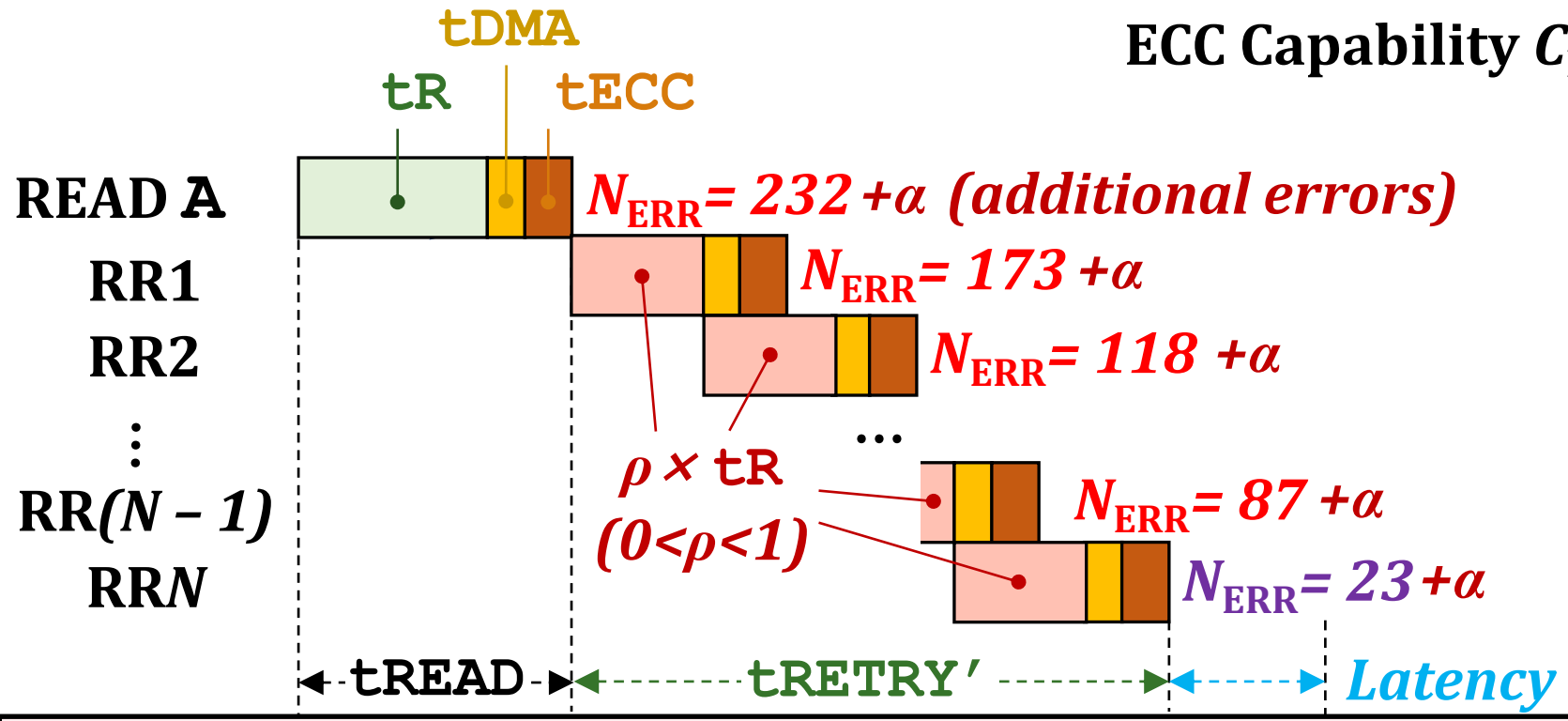
AR²: Adaptive Read-Retry

- **Key idea:** Reduce read-timing parameters for every retry step



AR²: Adaptive Read-Retry

- **Key idea:** Reduce read-timing parameters for every retry step



Needs to ensure that
# of additional errors < ECC margin

Validation with Real 3D NAND Flash Chips

- 160 real 48-layer Triple-Level Cell (TLC) NAND flash chips
- **Observation 1:** A large ECC margin in the final retry step even under worst-case operating conditions
  - Max. 40 errors per KiB under 1-year retention @ 2K P/E cycles
- **Observation 2:** A large reliability margin incorporated in read-timing parameters
  - 25% `tR` reduction → Max. 23 additional errors

AR^2 can easily work in state-of-the-art NAND flash chips w/ at least 25% `tR` reduction

Talk Outline

- Read-Retry in Modern NAND Flash-Based SSDs
- PR^2: Pipelined Read-Retry
- AR^2: Adaptive Read-Retry
- Evaluation Results

Evaluation Results

- Simulation using MQSim [Tavakkol, FAST18] and 12 real workloads

- Our proposal improves SSD response time by
  - Up to 51% (35% on average) compared to a high-end SSD w/o read-retry mitigation

  - Up to 32% (17% on average) compared to a state-of-the-art read-retry mitigation technique

- Many more detailed studies in the paper

Reducing Solid-State Drive Read Latency by Optimizing Read-Retry

Jisung Park[1], Myungsuk Kim[2], Myoungjun Chun[2],
Lois Orosa[1], Jihong Kim[2], and Onur Mutlu[1]

[1] SAFARI ETH zürich

[2] VERI LUX MEA TAS

ASPLOS 2021 (Session 17: Solid State Drives)