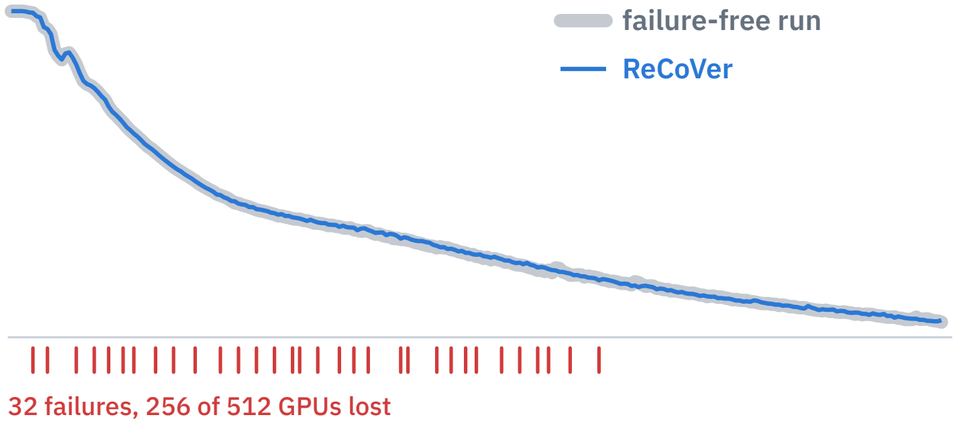
NeurIPS 2026

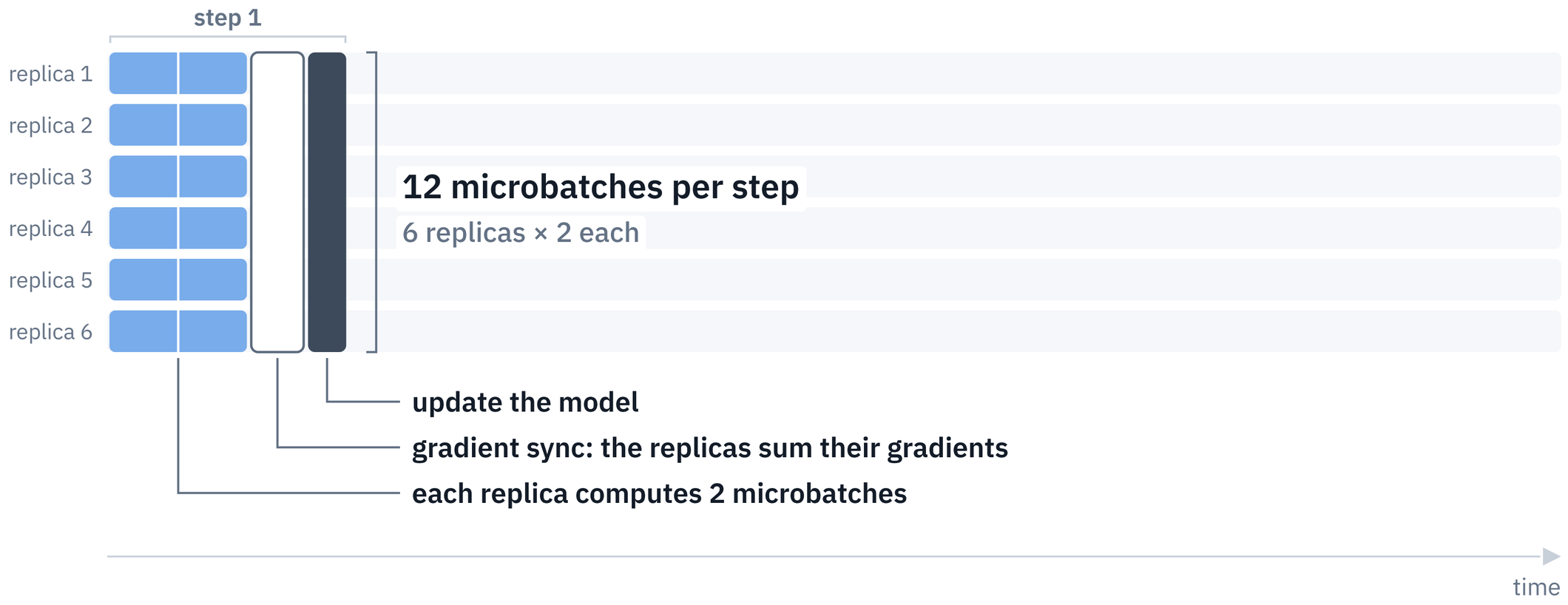
ReCoVer: Resilient LLM Pre-Training System via Fault-Tolerant Collective and Versatile Workload

Ziyue Liu[1], Zhengyang Wang[1], Ruijie Zhang[1], Avinash Maurya[2], Hui Zhou[2], Paul Hovland[2], Sheng Di[2], Franck Cappello[2], Bogdan Nicolae[2], Zheng Zhang[1]

[1]University of California, Santa Barbara [2]Argonne National Laboratory

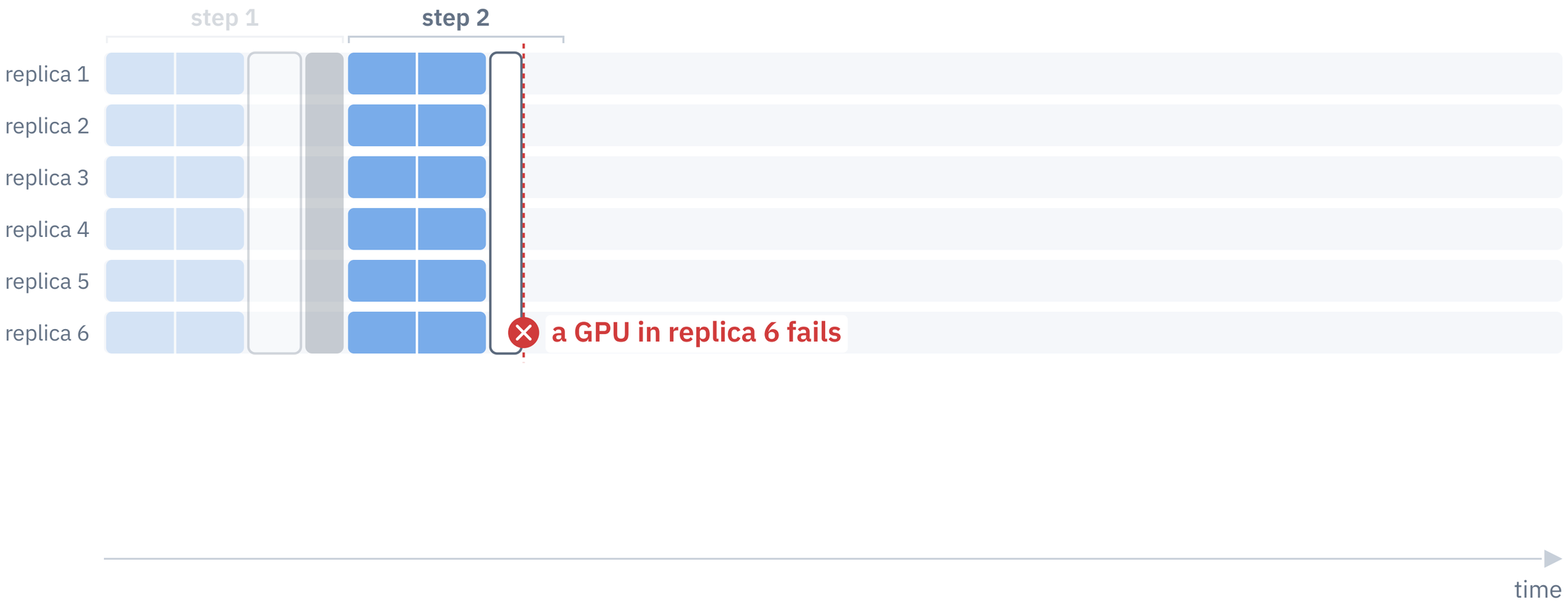


Synchronous training: 12 microbatches per step



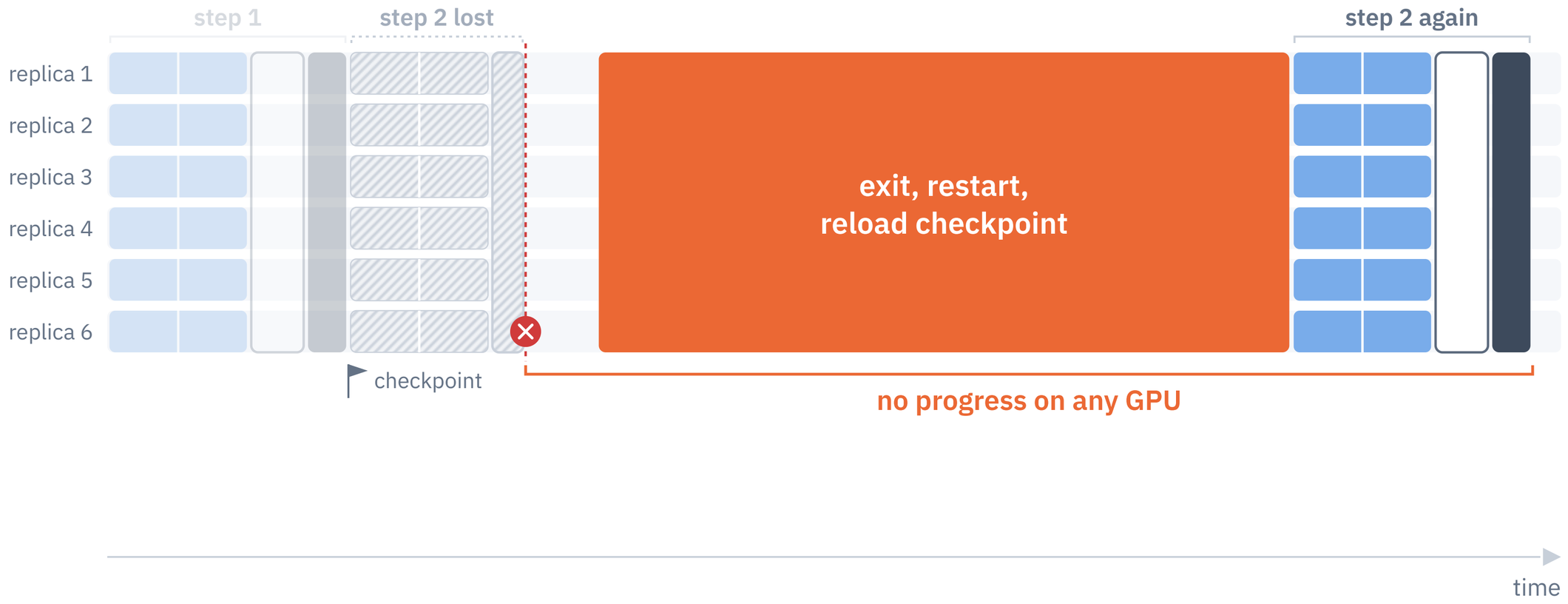
THE PROBLEM

A GPU fails in the middle of the gradient sync



Checkpoint-restart: every GPU stops, reloads, and redoes the step

Checkpoint-restart



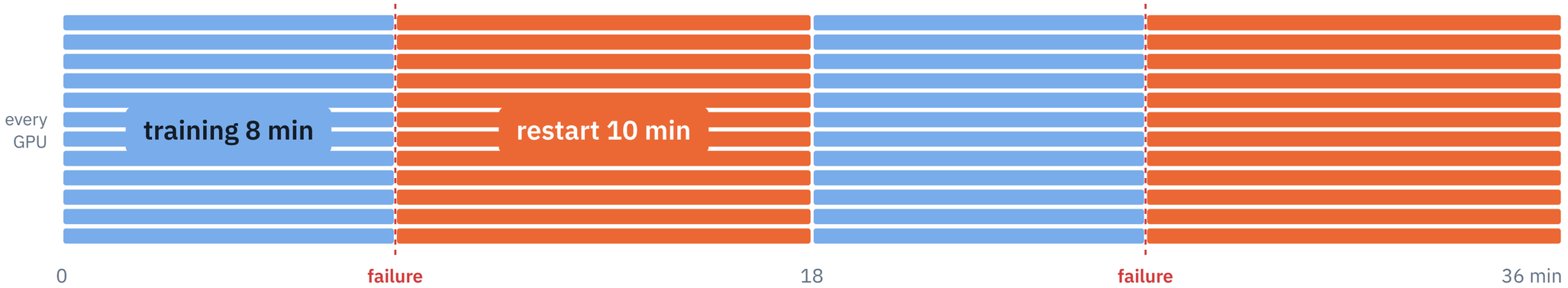
THE PROBLEM

At extreme scale, restarts take more than half of the GPU time

Checkpoint-restart

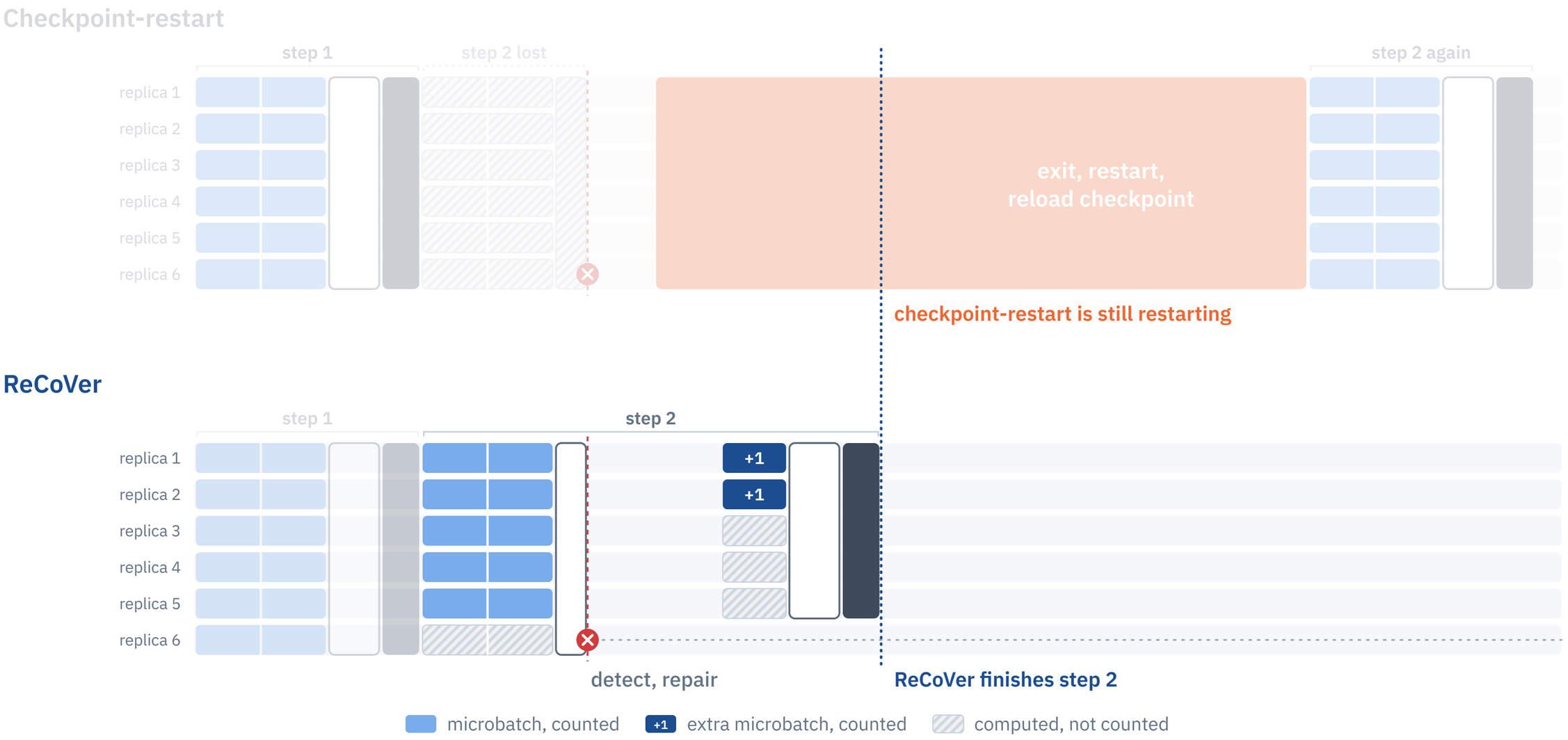
At 100,000 GPUs a failure about every 18 min, each restart about 10 min

> 50% of GPU time lost to restarts



Failure rate and restart time at 100k GPUs: Salpekar et al., 2026

ReCoVer: the survivors recover inside the step

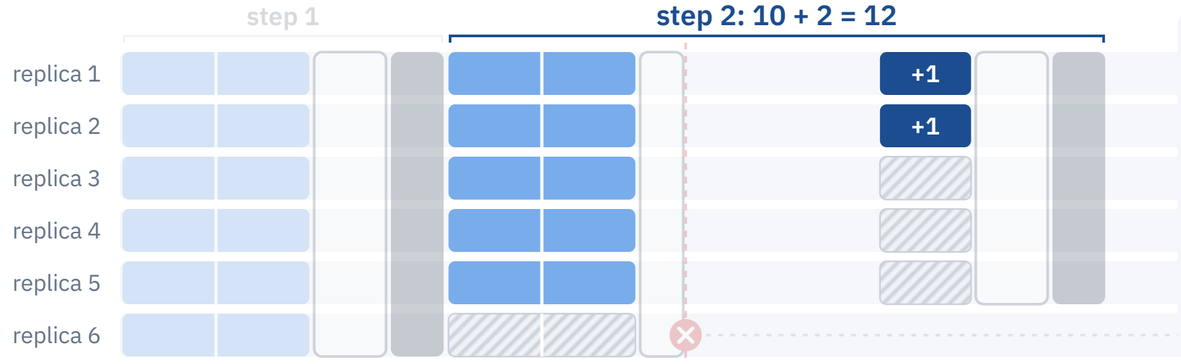


THE IDEA

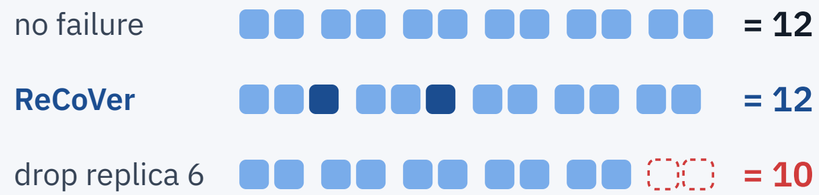
Batch invariance: every step still has 12 microbatches

Checkpoint-restart

ReCoVer



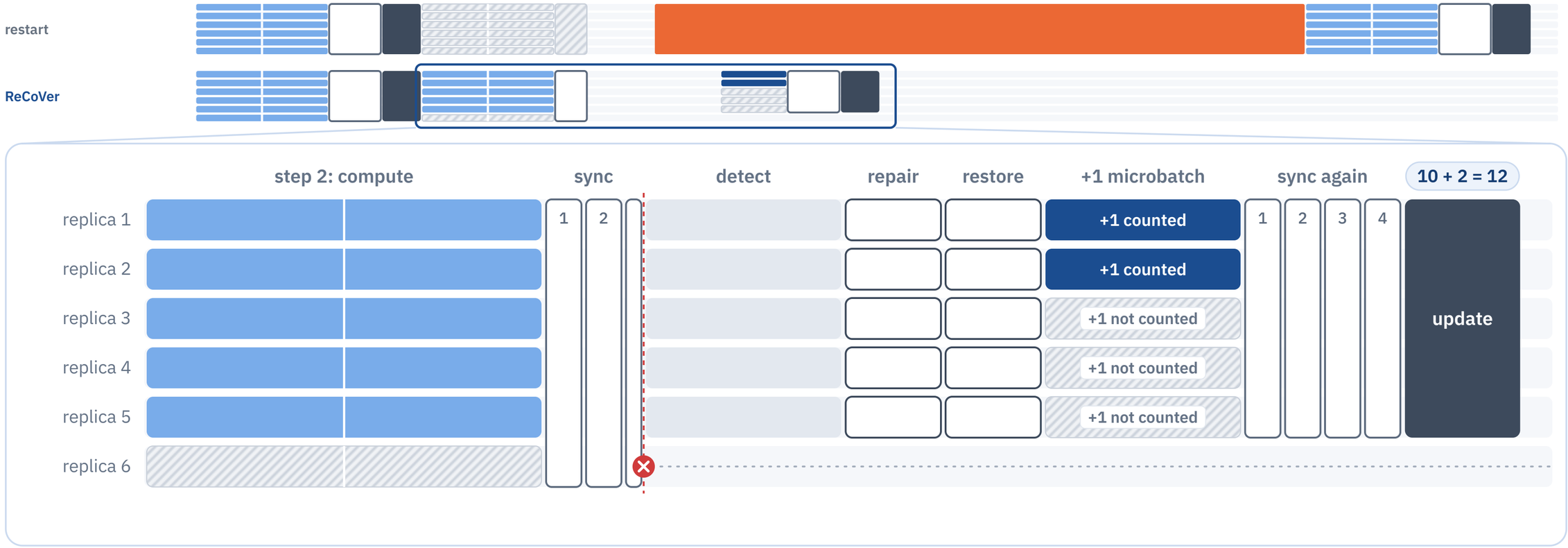
MICROBATCHES IN STEP 2



microbatch, counted +1 extra microbatch, counted computed, not counted

HOW RECOVER WORKS

Three layers make this recovery work



BOTTOM LAYER

Fault-tolerant collectives

Every survivor learns of the failure, the group shrinks, and all agree on who is left. Built on ULFM MPI.

MIDDLE LAYER

In-step recovery

Gradient parts already summed with replica 6 are restored and summed again. Finished work is kept.

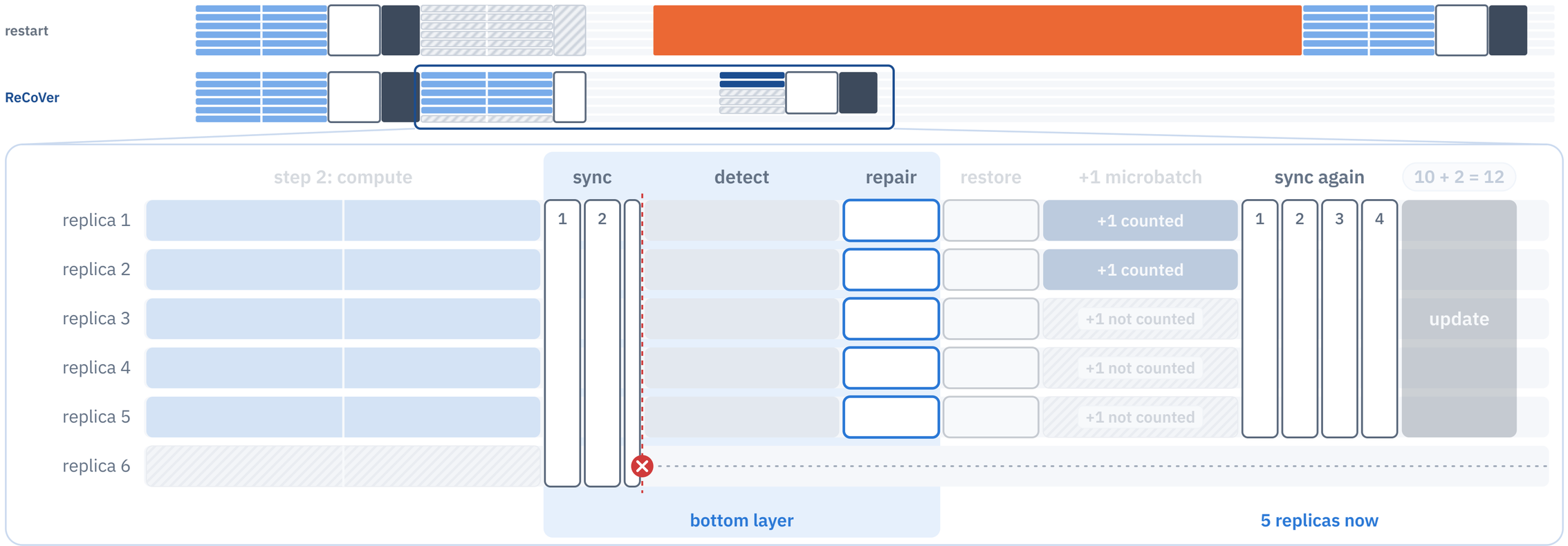
TOP LAYER

Versatile workload

Decides how many microbatches each survivor counts, so the step has exactly 12.

HOW RECOVER WORKS

Bottom layer: the sync survives the failure



BOTTOM LAYER

Fault-tolerant collectives

Every survivor learns of the failure, the group shrinks, and all agree on who is left. Built on ULFM MPI.

MIDDLE LAYER

In-step recovery

Gradient parts already summed with replica 6 are restored and summed again. Finished work is kept.

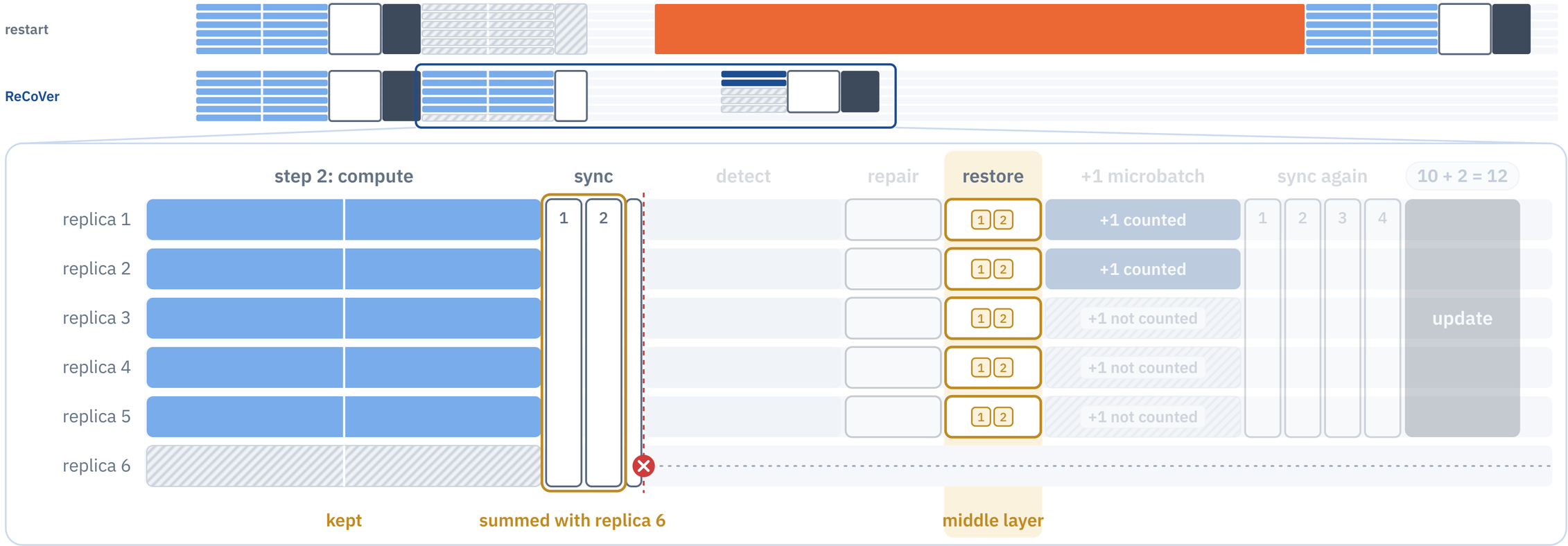
TOP LAYER

Versatile workload

Decides how many microbatches each survivor counts, so the step has exactly 12.

HOW RECOVER WORKS

Middle layer: work already done is kept



BOTTOM LAYER

Fault-tolerant collectives

Every survivor learns of the failure, the group shrinks, and all agree on who is left. Built on ULFM MPI.

MIDDLE LAYER

In-step recovery

Gradient parts already summed with replica 6 are restored and summed again. Finished work is kept.

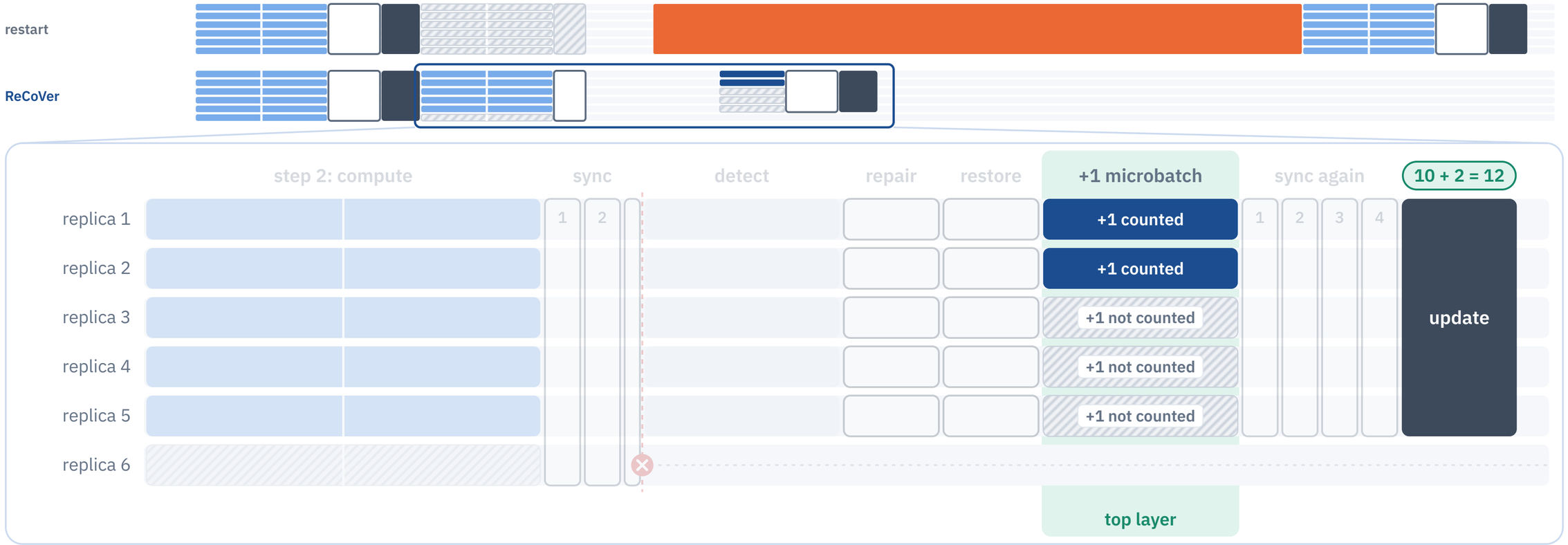
TOP LAYER

Versatile workload

Decides how many microbatches each survivor counts, so the step has exactly 12.

HOW RECOVER WORKS

Top layer: the step still has 12 microbatches



BOTTOM LAYER

Fault-tolerant collectives

Every survivor learns of the failure, the group shrinks, and all agree on who is left. Built on ULFM MPI.

MIDDLE LAYER

In-step recovery

Gradient parts already summed with replica 6 are restored and summed again. Finished work is kept.

TOP LAYER

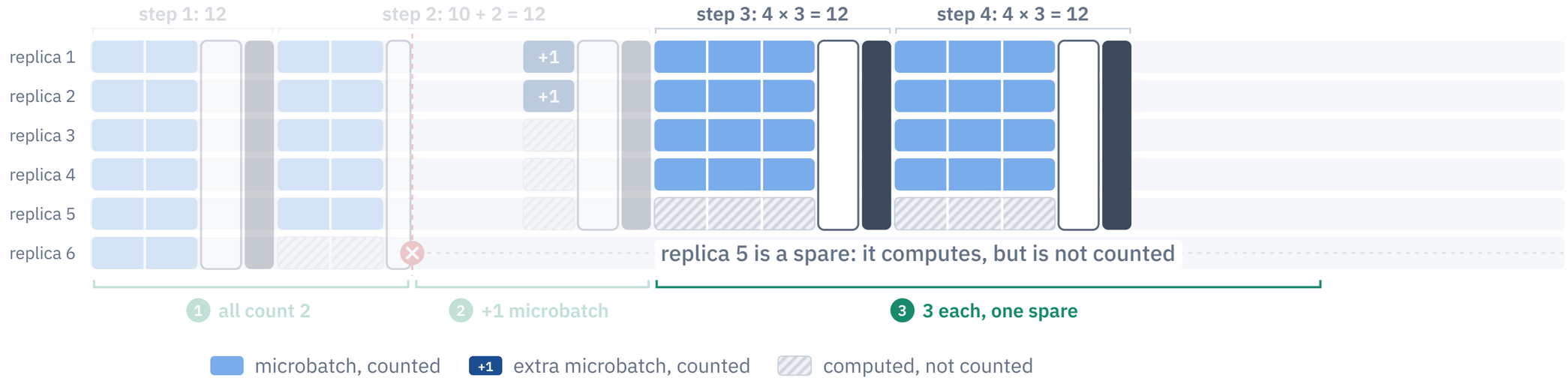
Versatile workload

Decides how many microbatches each survivor counts, so the step has exactly 12.

The policy plans ahead: 4 replicas × 3, plus a spare

Checkpoint-restart

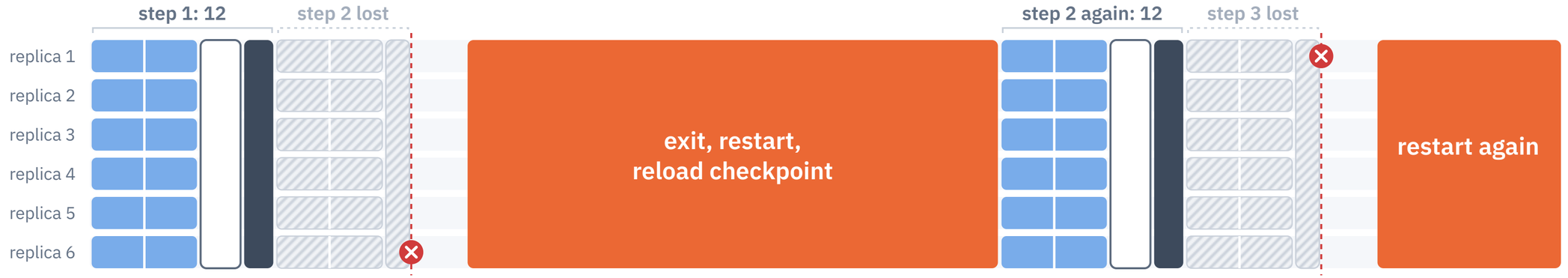
ReCoVer



The next failure: the spare is promoted, no extra work

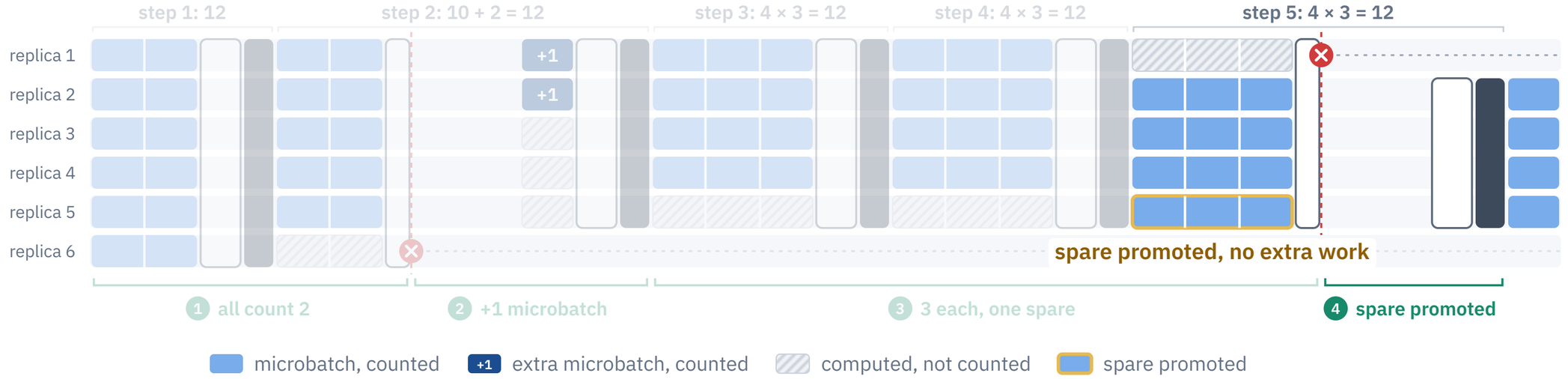
Checkpoint-restart

2 steps done



ReCoVer

5 steps done



RESULTS

Losing 256 of 512 GPUs leaves the loss curve unchanged

7B model, 512 A100 GPUs in 64 replicas of 8 · one replica lost during the gradient sync about every 5 steps, 32 times



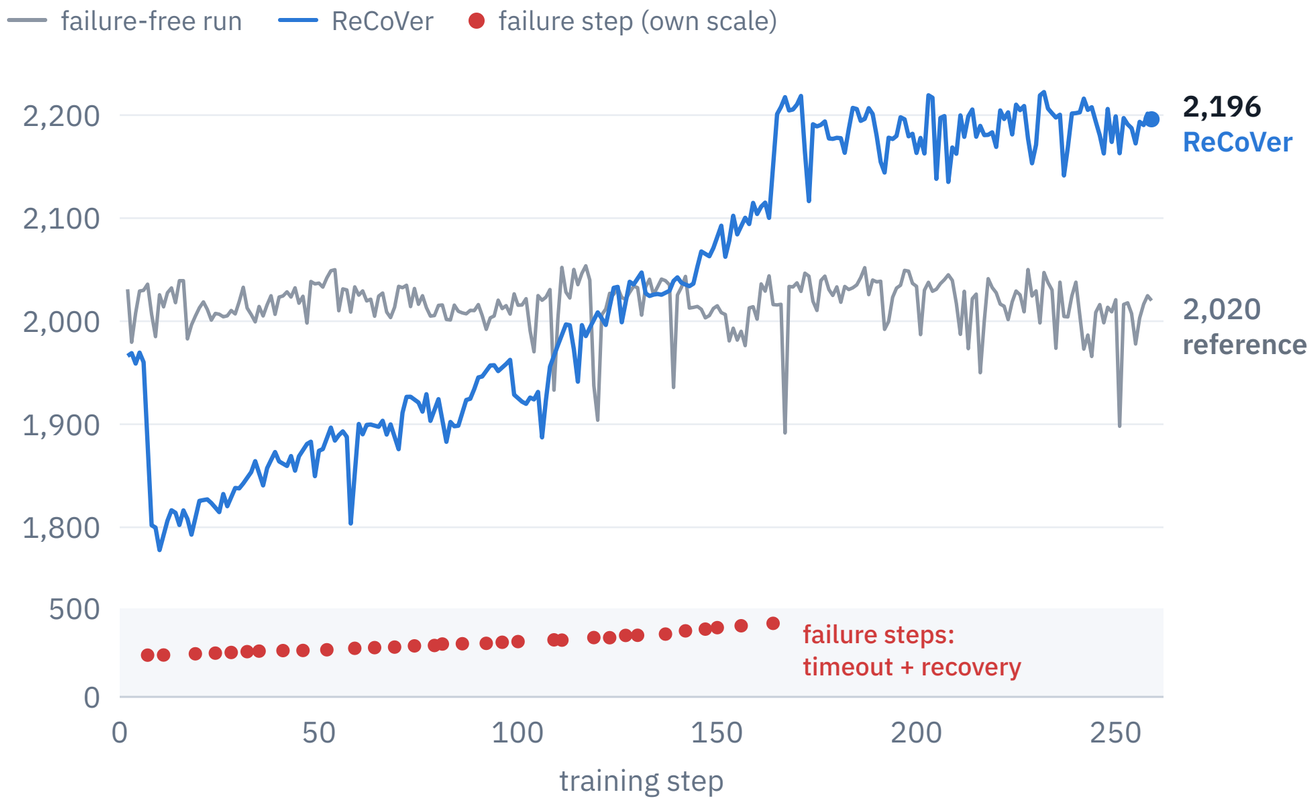
Validation loss stays within **0.024** of the failure-free run.

RESULTS

Throughput starts lower, then ends higher than without failures

Effective throughput

tokens per second per surviving GPU



ONE STEP OF ONE REPLICA, SCHEMATIC

Failure-free reference

64 replicas × 128 microbatches



ReCoVer, before failures

64 replicas × 128 microbatches



slower sync: MPI, no overlap

RESULTS

Throughput starts lower, then ends higher than without failures

Effective throughput

tokens per second per surviving GPU



ONE STEP OF ONE REPLICA, SCHEMATIC

Failure-free reference

64 replicas × 128 microbatches



ReCoVer, before failures

64 replicas × 128 microbatches



slower sync: MPI, no overlap

ReCoVer, after 32 failures

32 replicas × 256 microbatches



the same sync, spread over twice the work

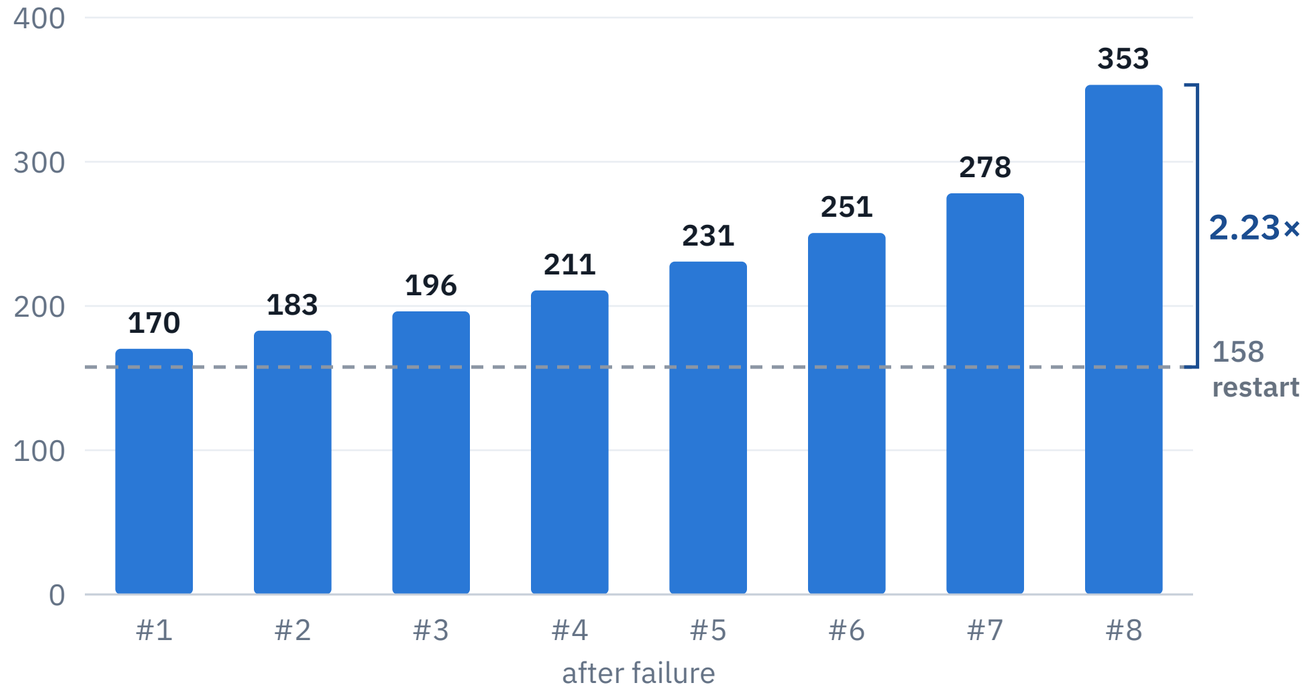
RESULTS

Against checkpoint-restart: 2.23× the effective throughput

7B model on 128 GPUs · 8 failures in a row · checkpoint-restart reloads its last checkpoint after each one

Effective throughput after each failure

tokens per second per surviving GPU

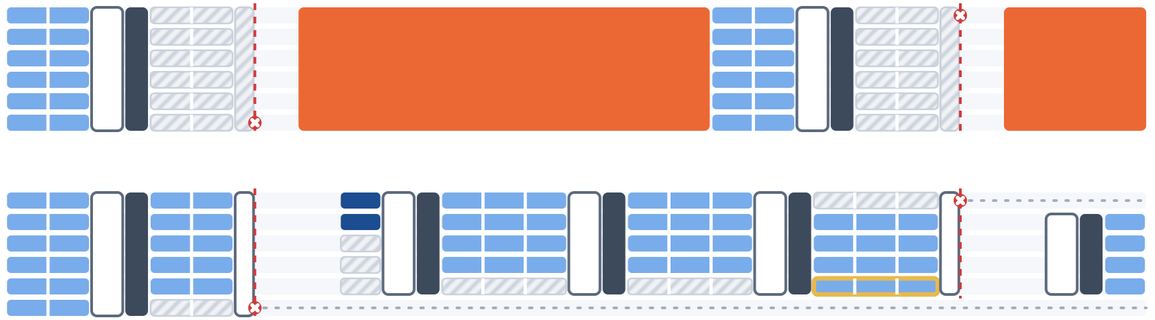


+74.9%

more tokens processed for the same 234 GPU-hours

Thank you!

ReCoVer keeps pre-training running through failures, and every step still has the **same number of microbatches**.



PROJECT PAGE

alvin-zyl.github.io/recover

PAPER arXiv 2605.11215

CONTACT ziyueliu@ucsb.edu