Distributed Deep Learning in Open Collaborations

Michael Diskin*, Alexey Bukhtiyarov*, Max Ryabinin*,
Lucile Saulnier, Quentin Lhoest, Anton Sinitsin, Dmitry Popov,
Dmitry Pyrkin, Maxim Kashirin, Alexander Borzunov,
Albert Villanova del Moral, Denis Mazur, Ilia Kobelev, Yacine Jernite,
Thomas Wolf, Gennady Pekhimenko

NATIONAL RESEARCH UNIVERSITY

What is this work about?

- Distributed training over the Internet **is possible**!

What is this work about?

- Distributed training over the Internet **is possible**!
- We develop a practical solution for decentralized DL

What is this work about?

- Distributed training over the Internet **is possible**!
- We develop a practical solution for decentralized DL
- Propose an algorithm that adapts to the infrastructure

What is this work about?

- Distributed training over the Internet **is possible**!
- We develop a practical solution for decentralized DL
- Propose an algorithm that adapts to the infrastructure
- Report the first large-scale collaborative training run

Background

- Transfer learning is bread and butter of modern DL

Background

- Transfer learning is bread and butter of modern DL
- Pretraining requires lots of resources and time

Background

- Transfer learning is bread and butter of modern DL
- Pretraining requires lots of resources and time
- Doing this from scratch is often infeasible without $$$

Background

- Transfer learning is bread and butter of modern DL
- Pretraining requires lots of resources and time
- Doing this from scratch is often infeasible without $$$
- *What if we could train neural networks together?*

Challenges of collaborative DL…

Peers can join and leave at random

Challenges of collaborative DL...

Peers can join and leave at random

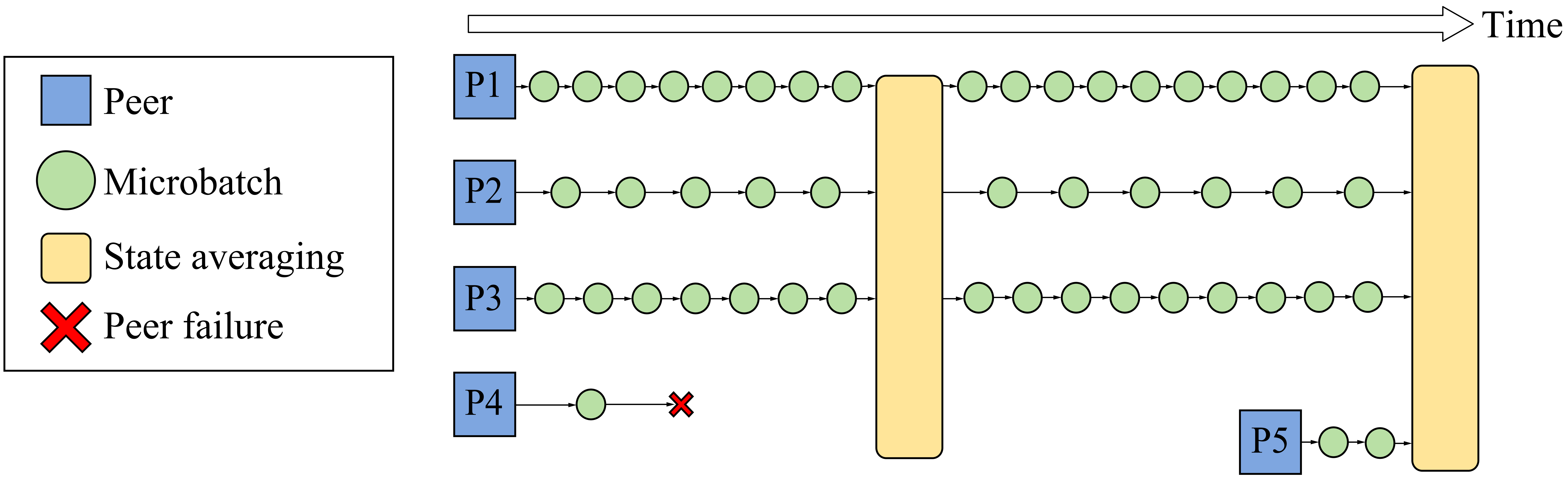
Heterogeneous hardware and network

…and how DeDLOC resolves them

Peers can join and leave at random	Gradient accumulation over entire collaboration
Heterogeneous hardware and network	Averaging strategy that adapts to the participants

DeDLOC: core concepts

- Train on very large batches
- Accumulate gradients over all peers
- If somebody disconnects, others will compensate for that



Adaptive averaging

- Each peer has different network speed and compute performance

Adaptive averaging

- Each peer has different network speed and compute performance
- Formulate throughput optimization as an LP problem

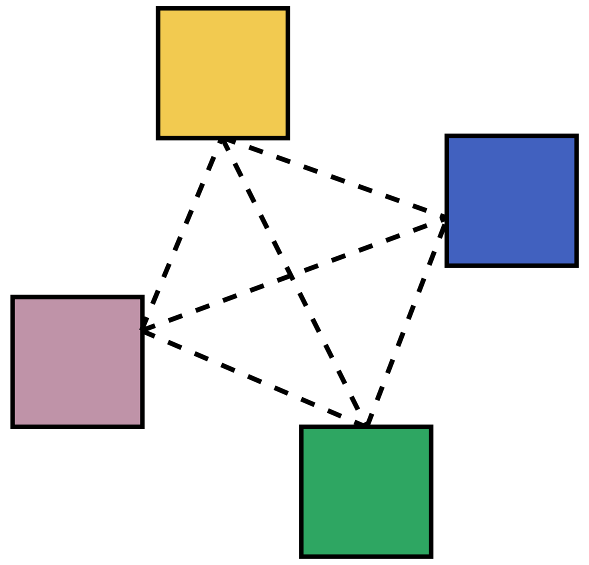
Adaptive averaging

- Each peer has different network speed and compute performance
- Formulate throughput optimization as an LP problem
- In special cases, we recover well-known algorithms!

Adaptive averaging

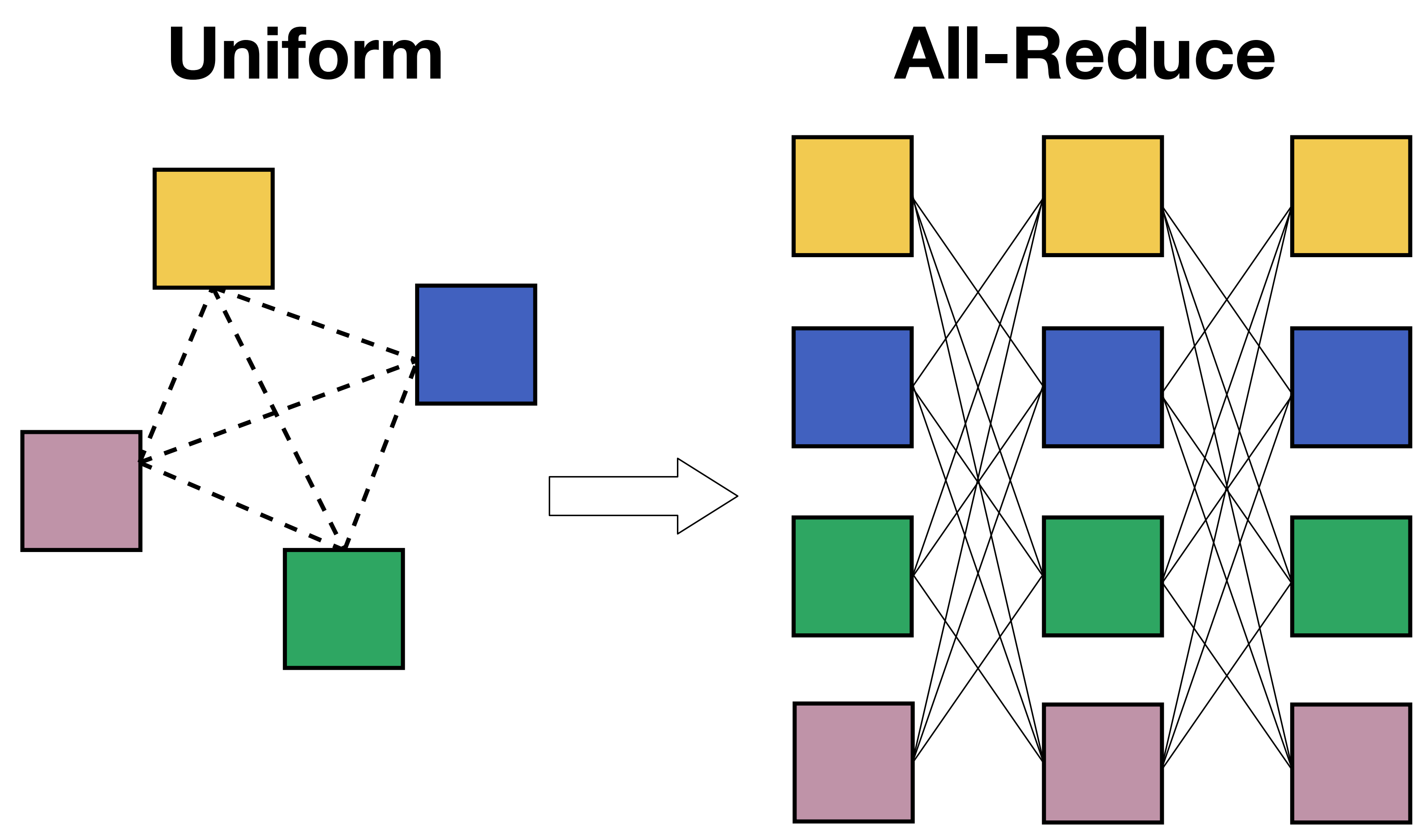
- Each peer has different network speed and compute performance
- Formulate throughput optimization as an LP problem
- In special cases, we recover well-known algorithms!

Uniform



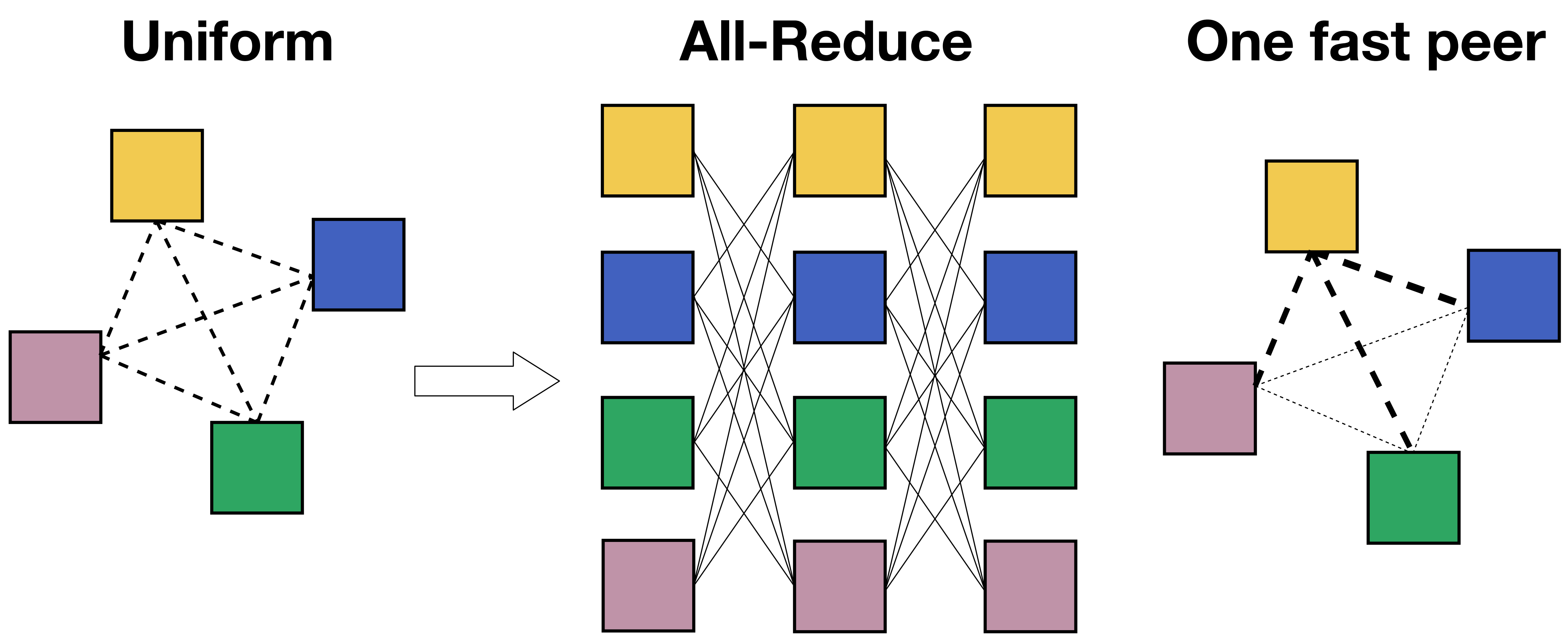
Adaptive averaging

- Each peer has different network speed and compute performance
- Formulate throughput optimization as an LP problem
- In special cases, we recover well-known algorithms!



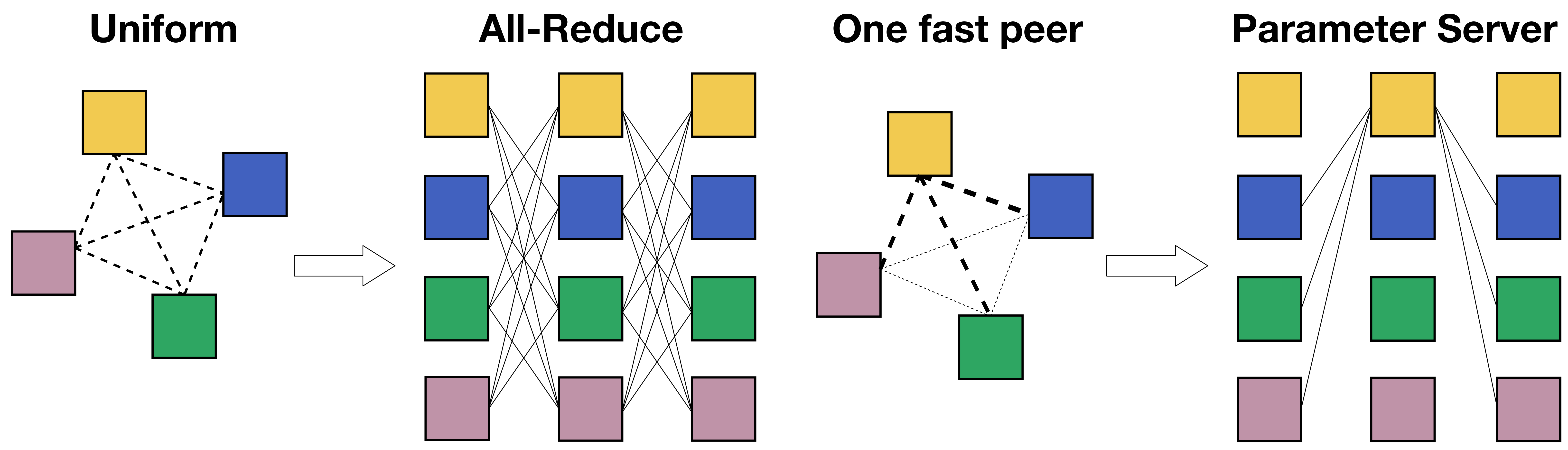
Adaptive averaging

- Each peer has different network speed and compute performance
- Formulate throughput optimization as an LP problem
- In special cases, we recover well-known algorithms!



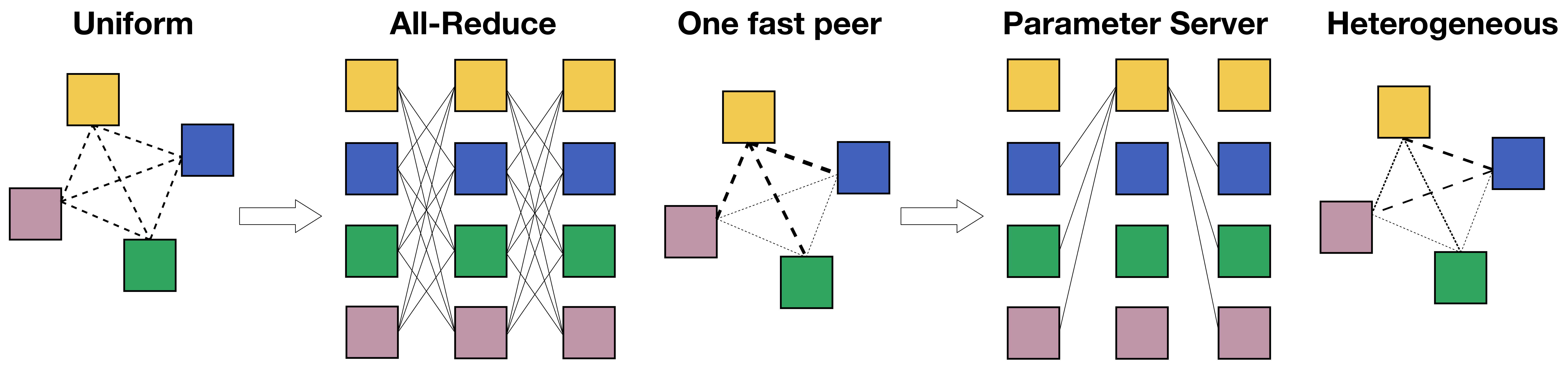
Adaptive averaging

- Each peer has different network speed and compute performance
- Formulate throughput optimization as an LP problem
- In special cases, we recover well-known algorithms!



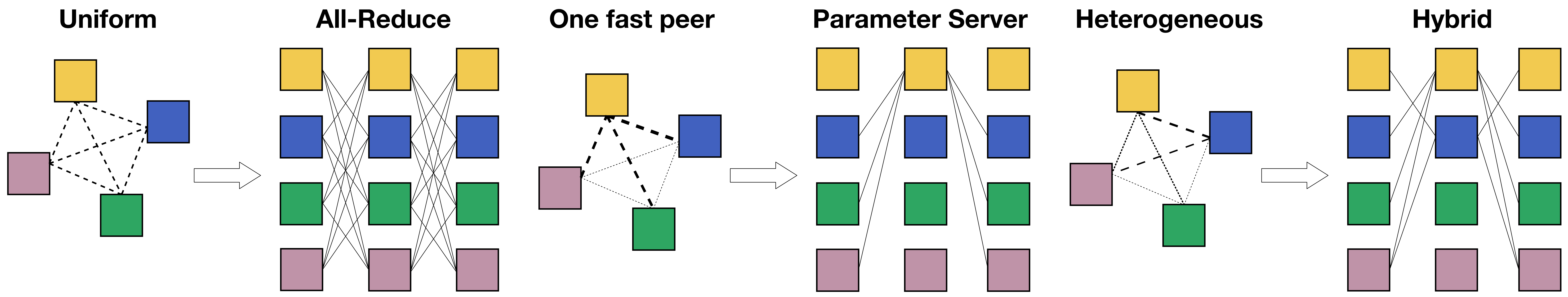
Adaptive averaging

- Each peer has different network speed and compute performance
- Formulate throughput optimization as an LP problem
- In special cases, we recover well-known algorithms!



Adaptive averaging

- Each peer has different network speed and compute performance
- Formulate throughput optimization as an LP problem
- In special cases, we recover well-known algorithms!



Training under NAT

- Household PCs often don’t have dedicated IP addresses

Training under NAT

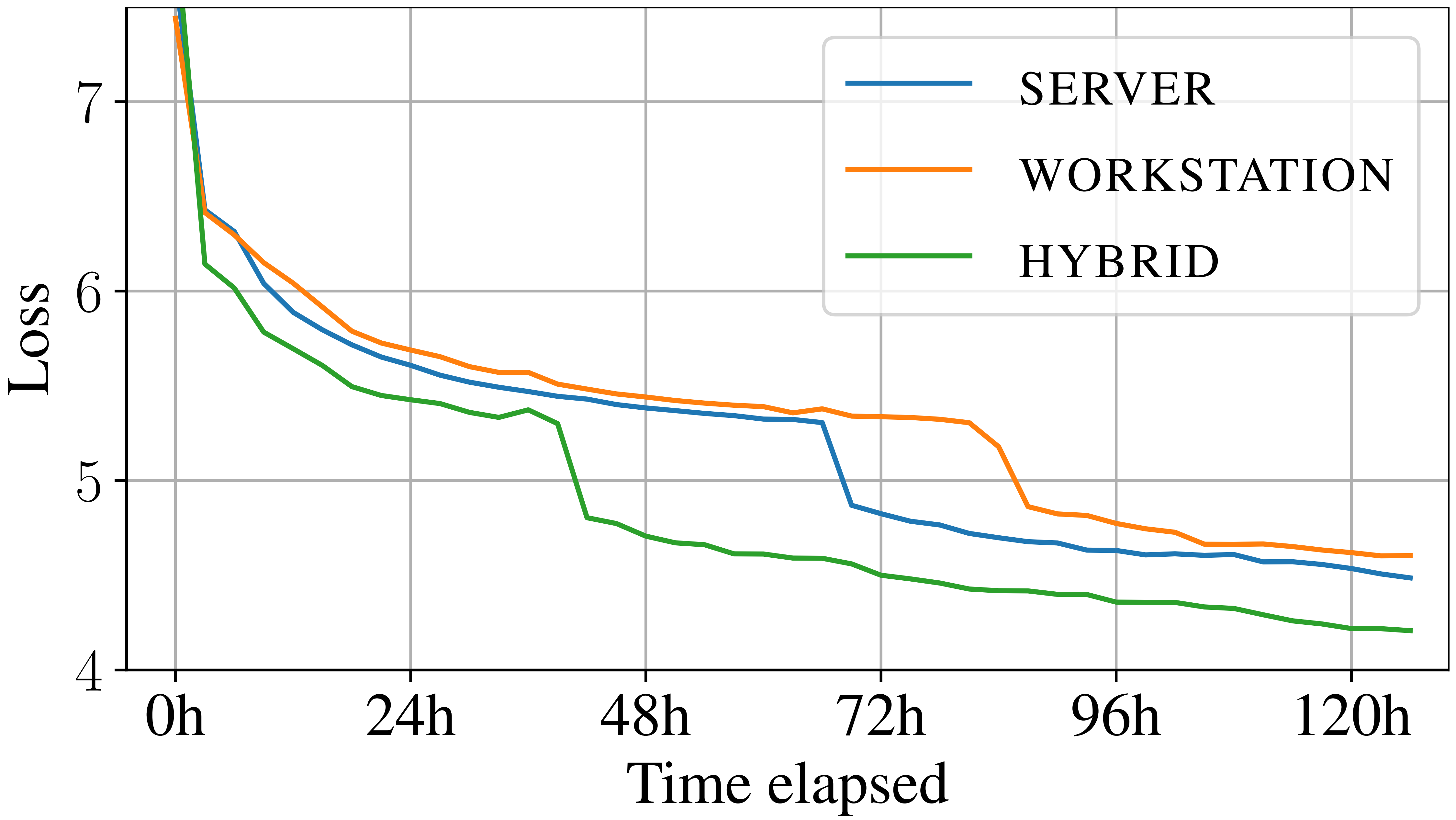
- Household PCs often don’t have dedicated IP addresses
- NAT makes it harder to establish peer-to-peer connections

Training under NAT

- Household PCs often don’t have dedicated IP addresses
- NAT makes it harder to establish peer-to-peer connections
- We employ NAT traversal techniques to resolve those issues!

Experiments: adaptivity

- Pretrain ResNet-50 SwAV in different environments
- Succesfully utilize low-performance peers even together with others

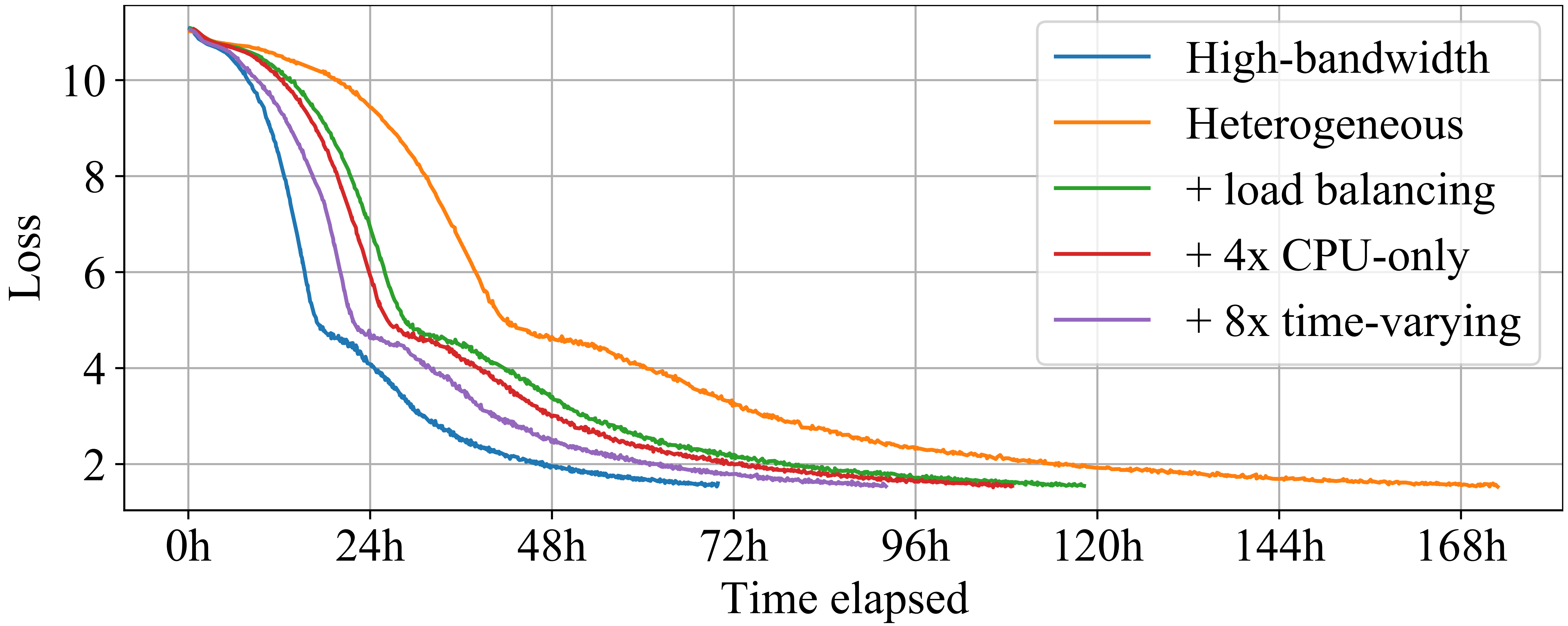


Server: 8 workers with 1xV100, 1Gb/s network

Workstation: 16 workers with 1x1080Ti, 200Mb/s network

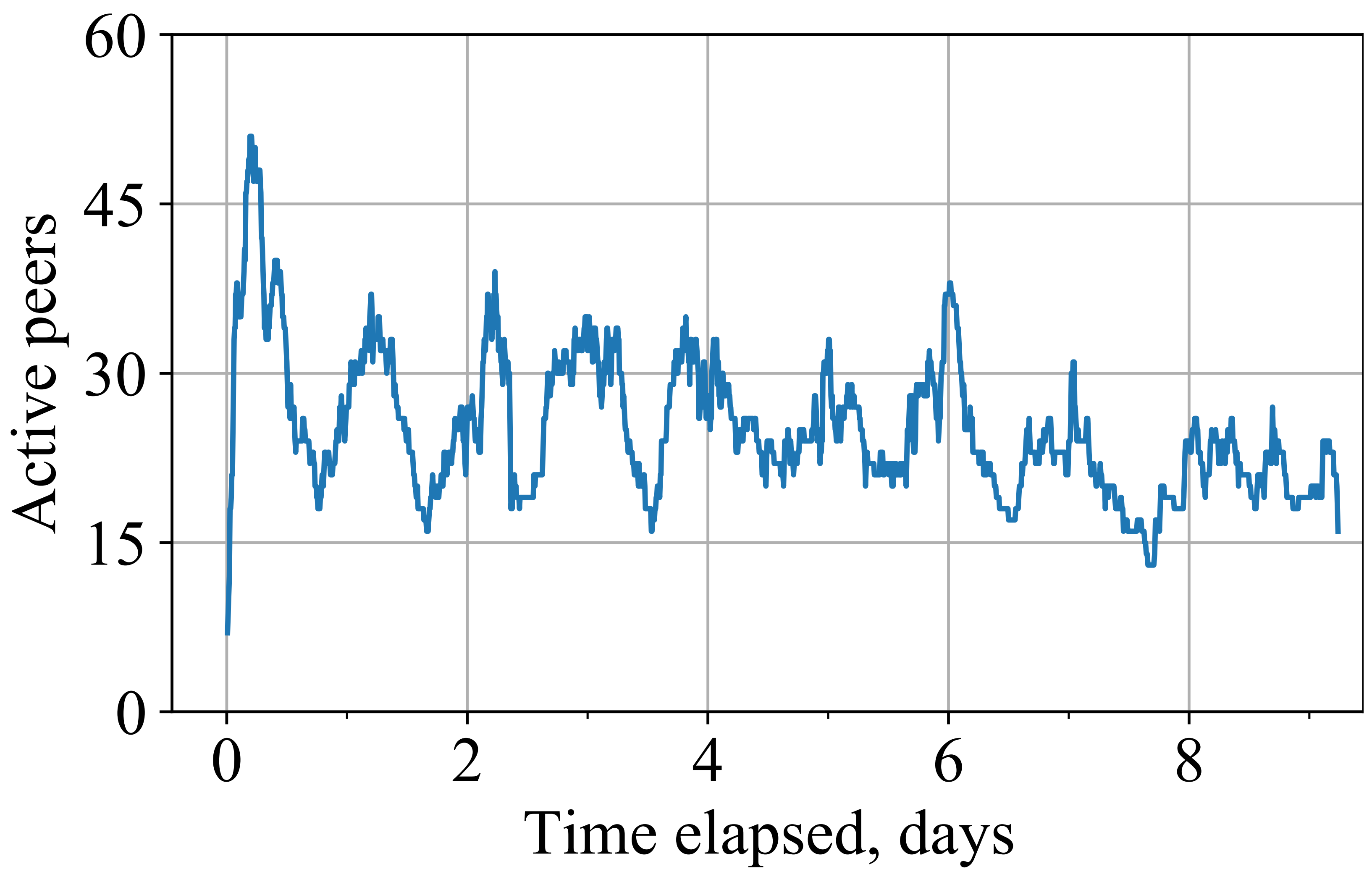
Experiments: network performance

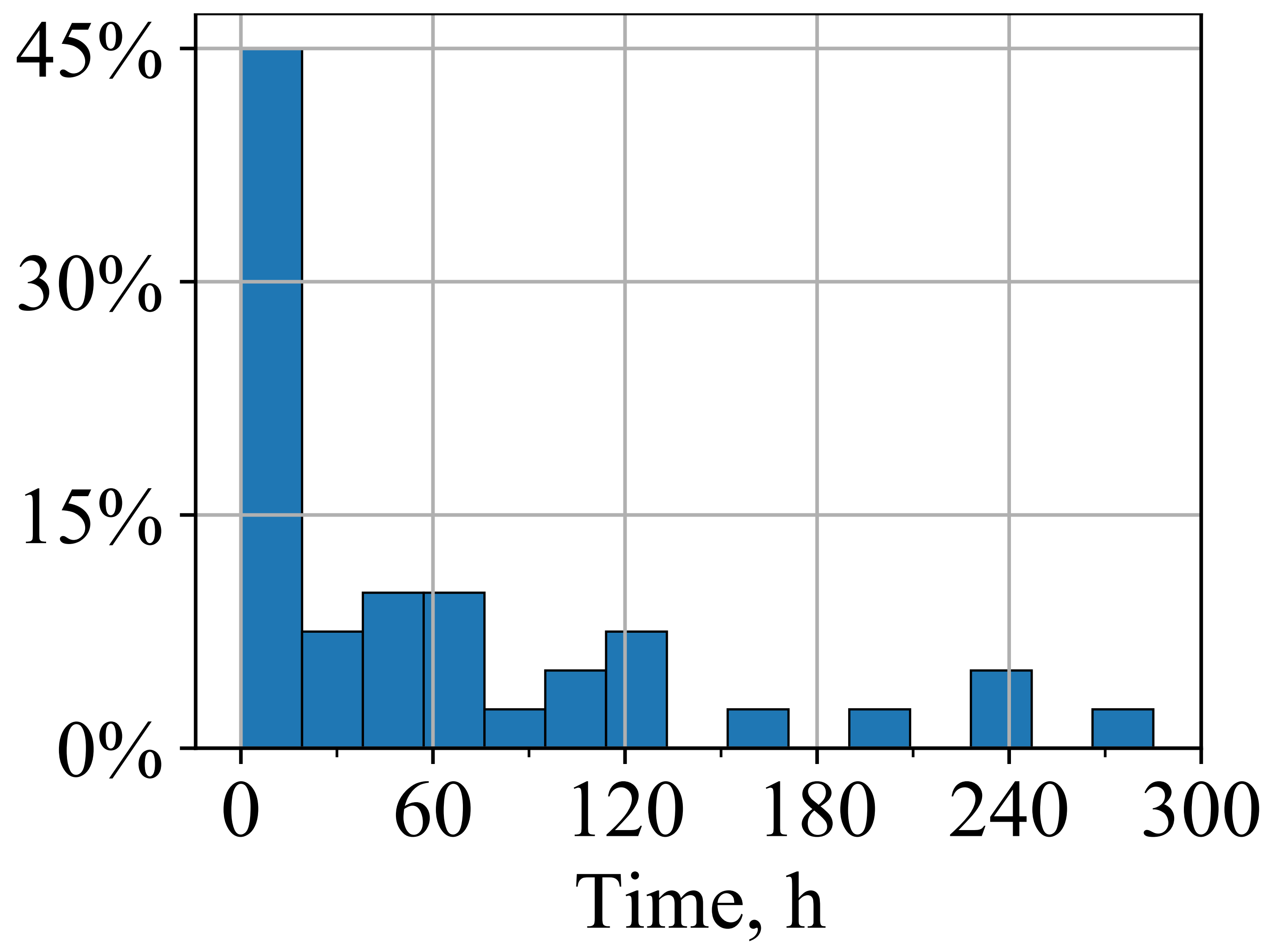
- Pretrain ALBERT on T4 nodes with different network speeds
- Load balancing, CPU-only and part-time peers help significantly!

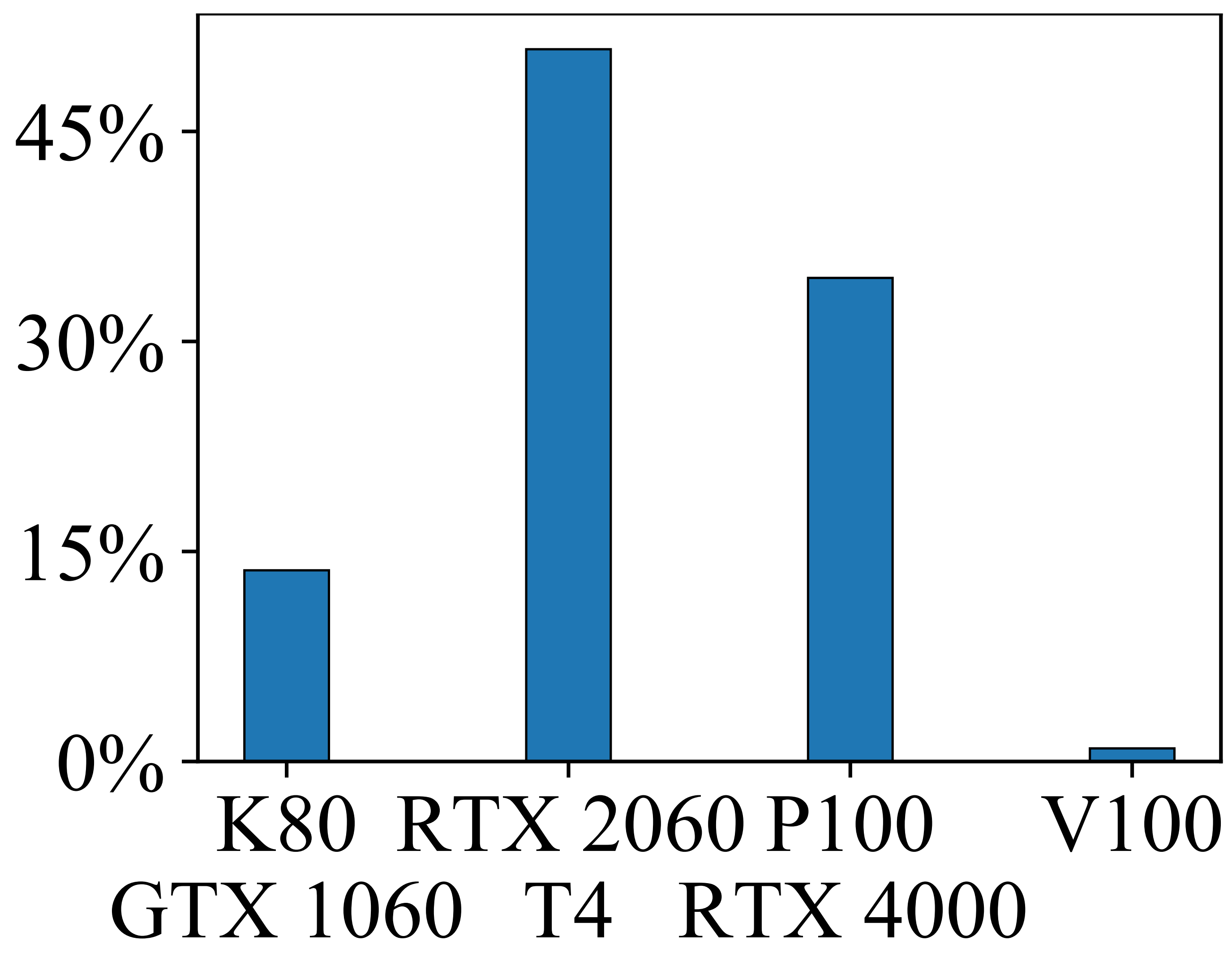


Experiments: sahajBERT

- We pretrained ALBERT for Bengali together with volunteers!
- 40 people joined the experiment from 91 unique devices
- Median participation time of 1.5 days

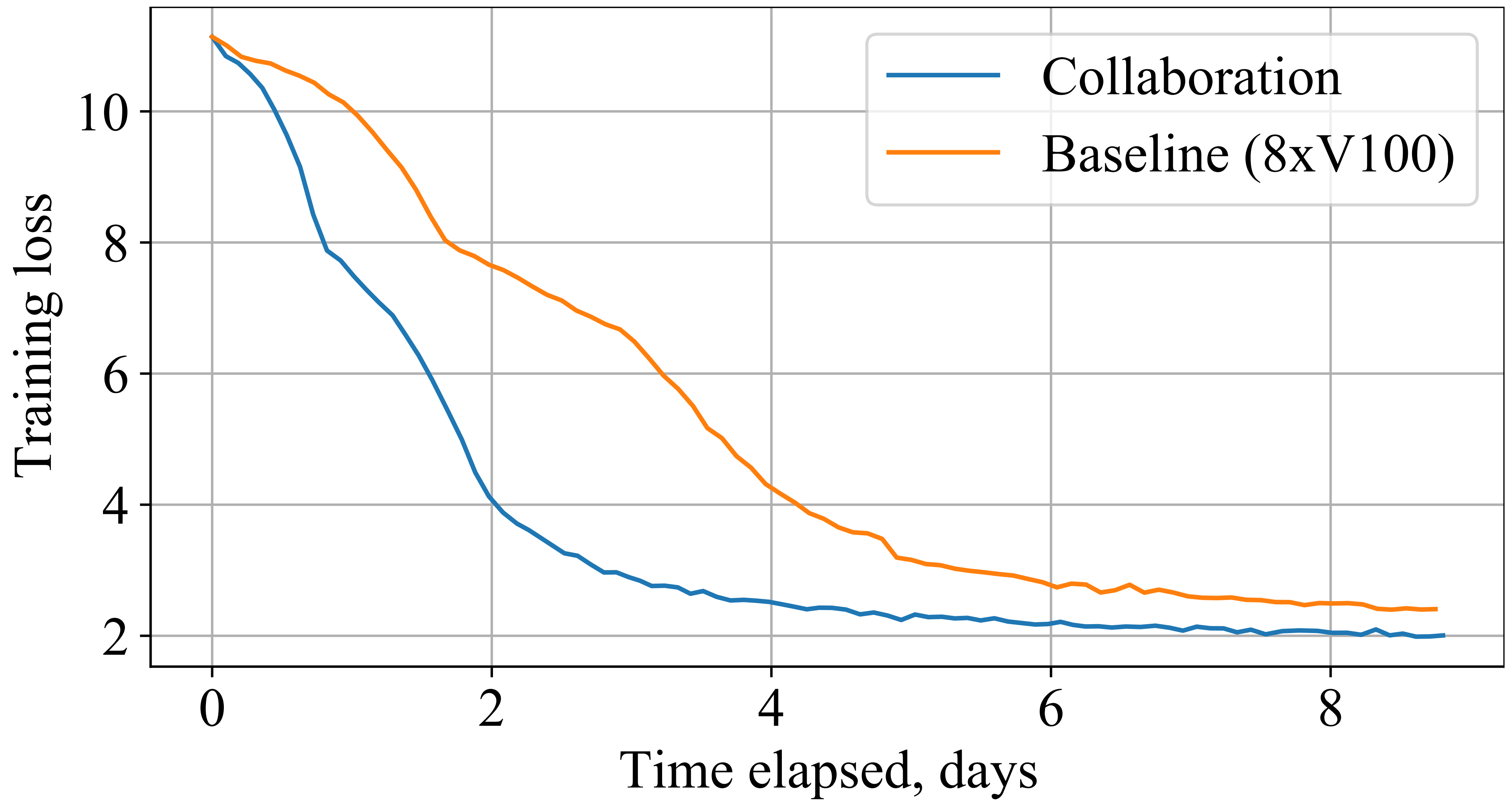






sahajBERT: results

- The model converged in ~8 days
- Outperforms very strong baselines, both cross-lingual and Bengali-only



Model	Wikiann F1	NCC Accuracy
bnRoBERTa	82.32 ± 0.67	80.94 ± 0.45
IndicBERT	92.52 ± 0.45	74.46 ± 1.91
XLM-R	96.48 ± 0.22	90.05 ± 0.38
sahajBERT	95.45 ± 0.53	91.97 ± 0.47
sahajBERT-XL	$\mathbf{96.59} \pm \mathbf{0.26}$	$\mathbf{92.91} \pm \mathbf{0.43}$

Conclusion

- We propose a practical method for collaborative training!
- Learn more:

Paper	Blog post	Code
arxiv.org/abs/2106.10207	huggingface.co/blog/collaborative-training	github.com/yandex-research/DeDLOC