Università
della
Svizzera
italiana

# Faculty of Informatics

Stefano Gonçalves Simao
Student
goncast@usi.ch

Prof. Rolf Krause
Advisor
rolf.krause@usi.ch

Dr. Alena Kopaničáková
Co-Advisor
alena.kopanicakova@usi.ch

# Implementation of a hybrid data parallel algorithm for deep neural network training with reduced communication targeted to GPU-based supercomputers

## GOAL

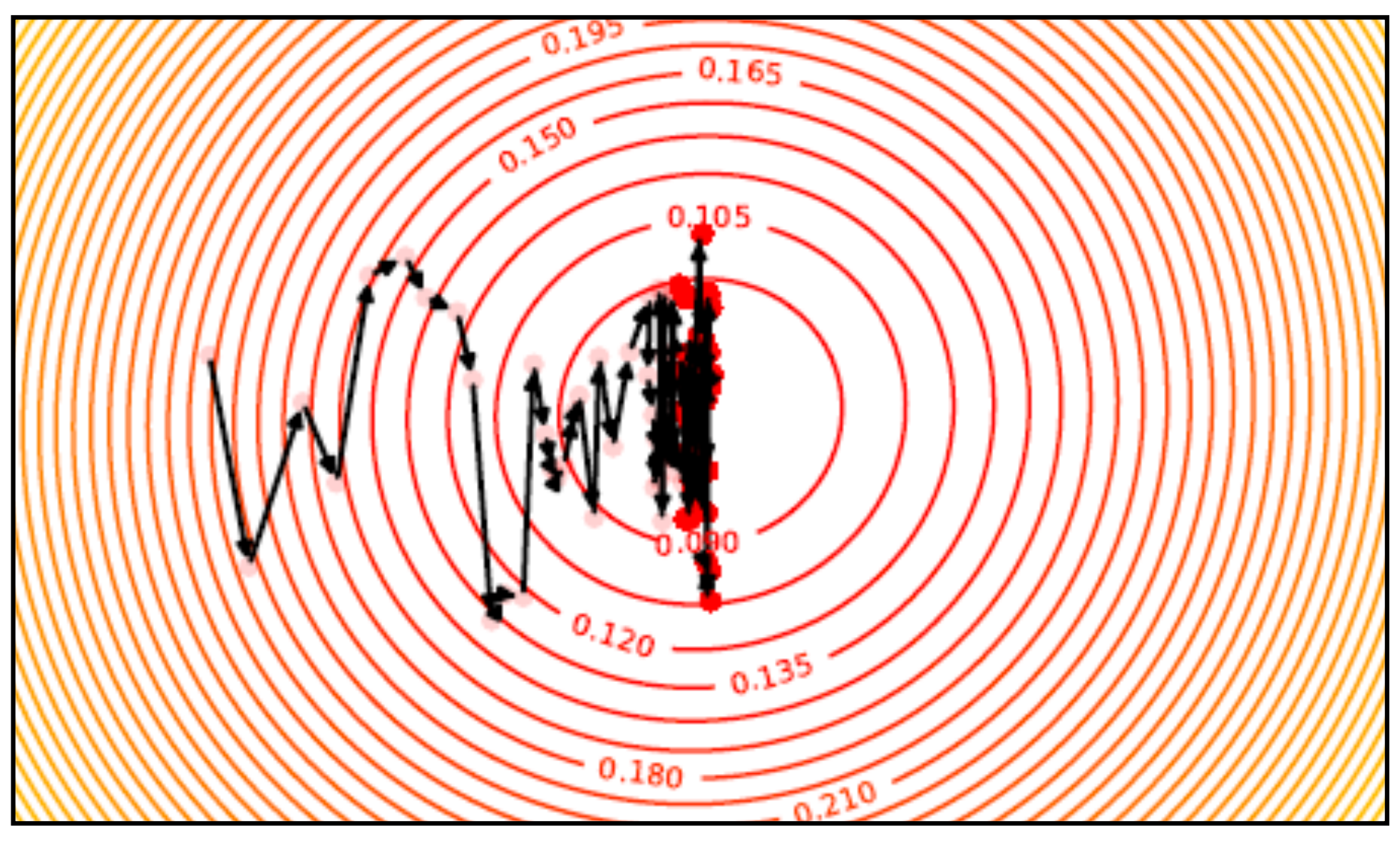
The goal of this project is to investigate existing Parallel Programming strategies to distribute the work of Machine Learning algorithms for training Deep Neural Networks and propose a novel algorithm that reduces communication complexity.

## KEY WORDS

Supervised Learning Algorithm

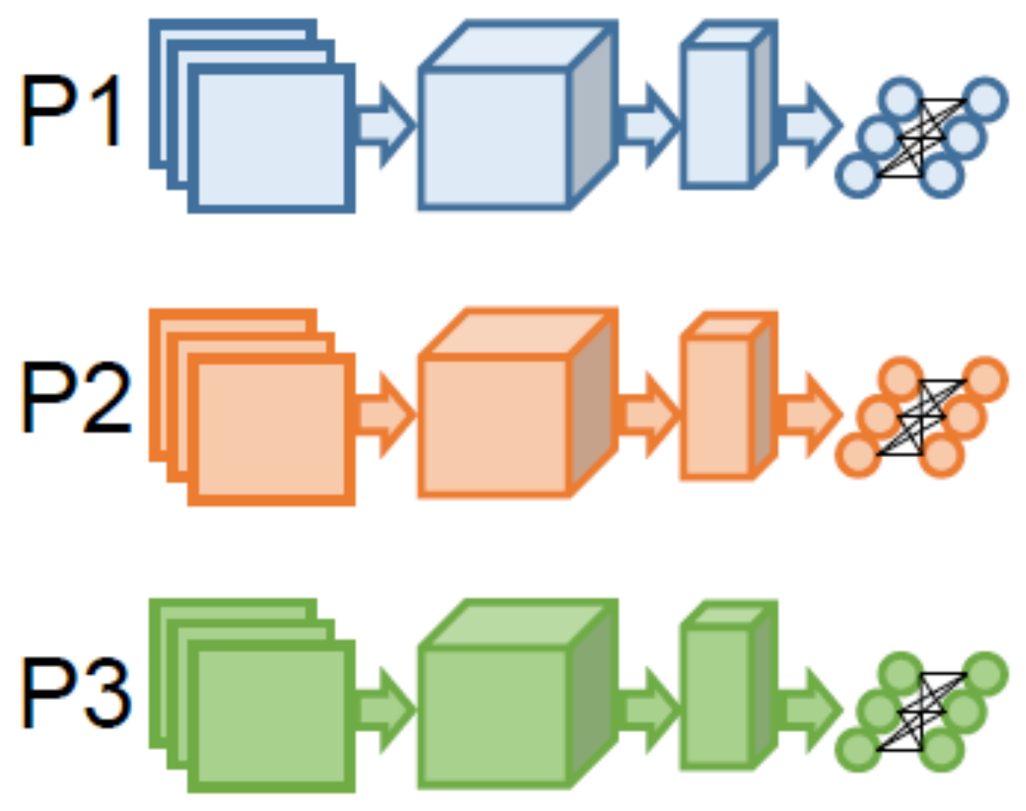
### Stochastic Gradient Descent

$$\min_{\theta} L := \sum_{i=1}^{n} l(g(x_i;\theta), y_i),$$



$$\theta_{k+1} = \theta_k - \alpha_k \nabla L_k,$$

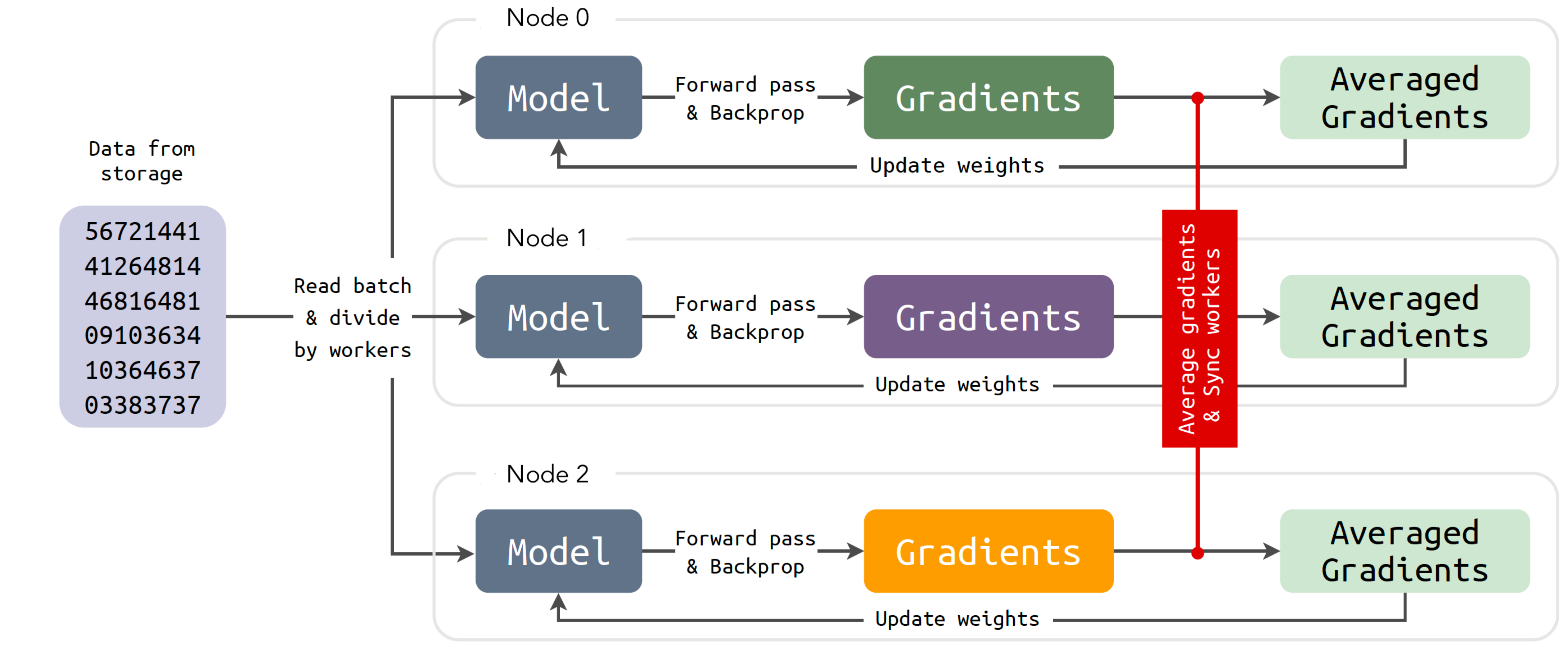
### Data Parallelism



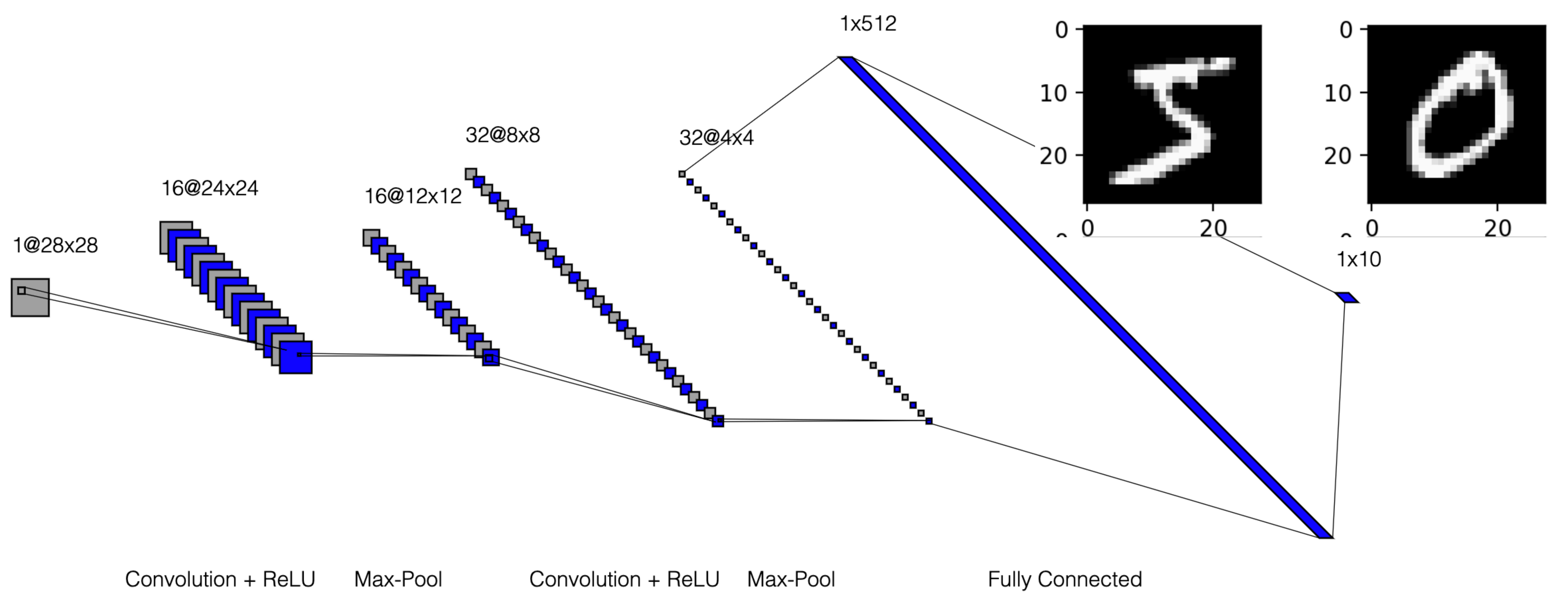
### GPU-based supercomputers

(CSCS - Piz Daint)
5704 x Intel® Xeon® E5-2690 v3 @2.60GHz
(12 cores, 64GB RAM),
NVIDIA® Tesla® P100 16GB

### DistributedDataParallel - NCCL - AllReduce



### Convolutional Neural Networks - MNIST



### Decentralised Synchronous Communication

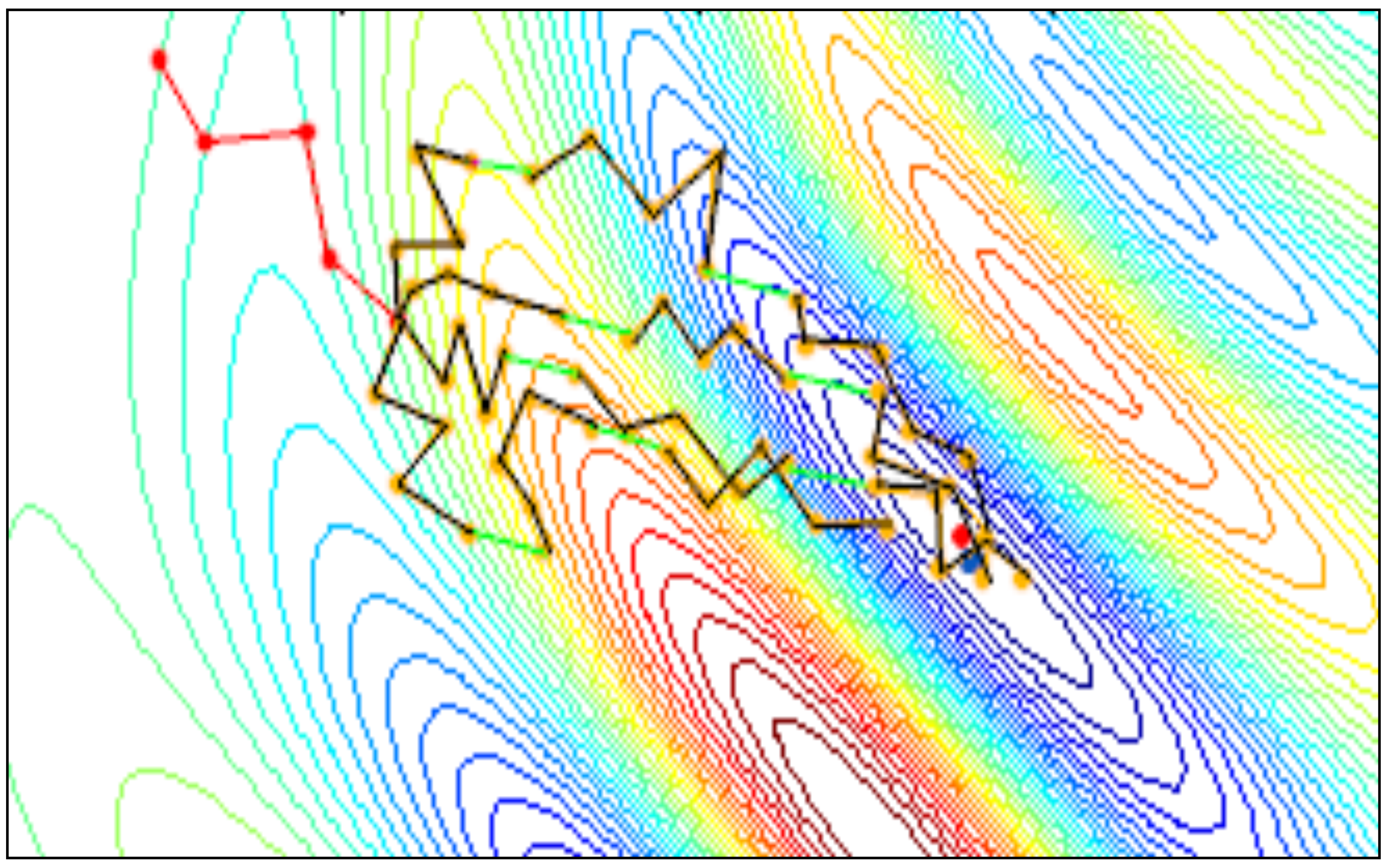
## Hybrid Parallel Stochastic Gradient Descent

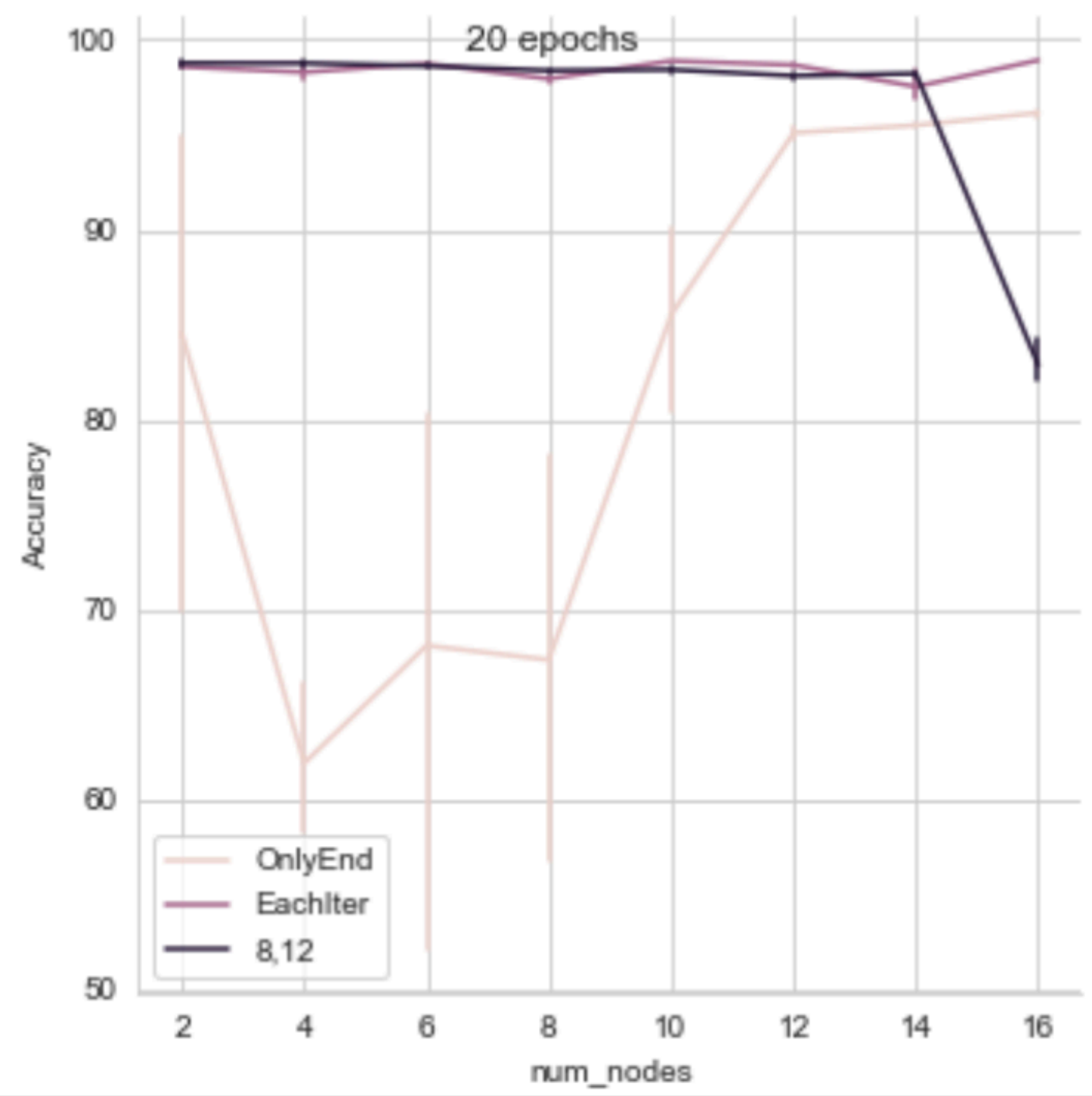
***Algorithm structure:***

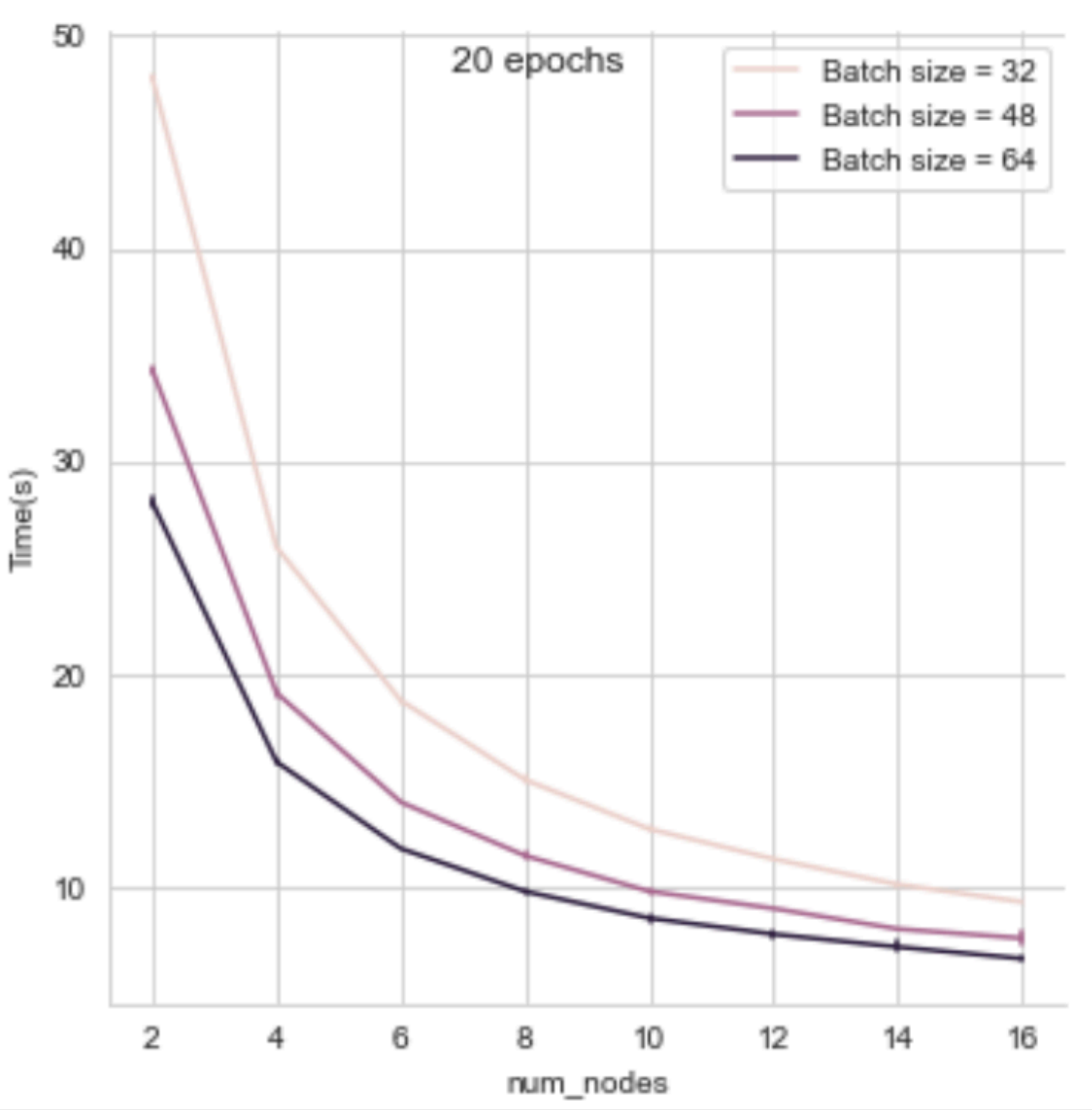
Divide num. epochs by 3

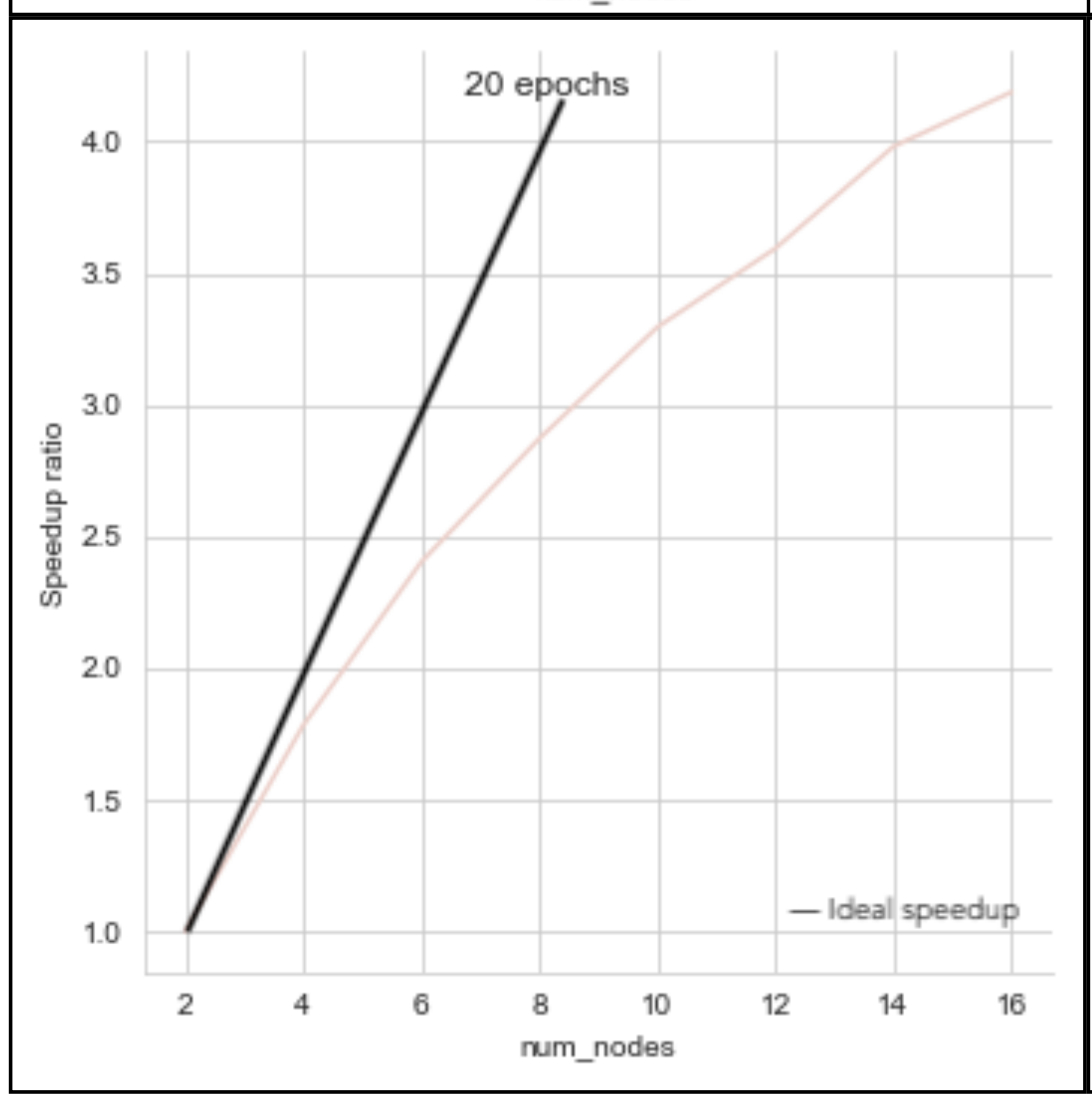
1. Standard Parallel SGD with gradient synchronisation at each iteration
2. Standard Parallel SGD with gradient accumulation
3. Parallel SGD with gradient accumulation without model consistency (independent *influenced* training)

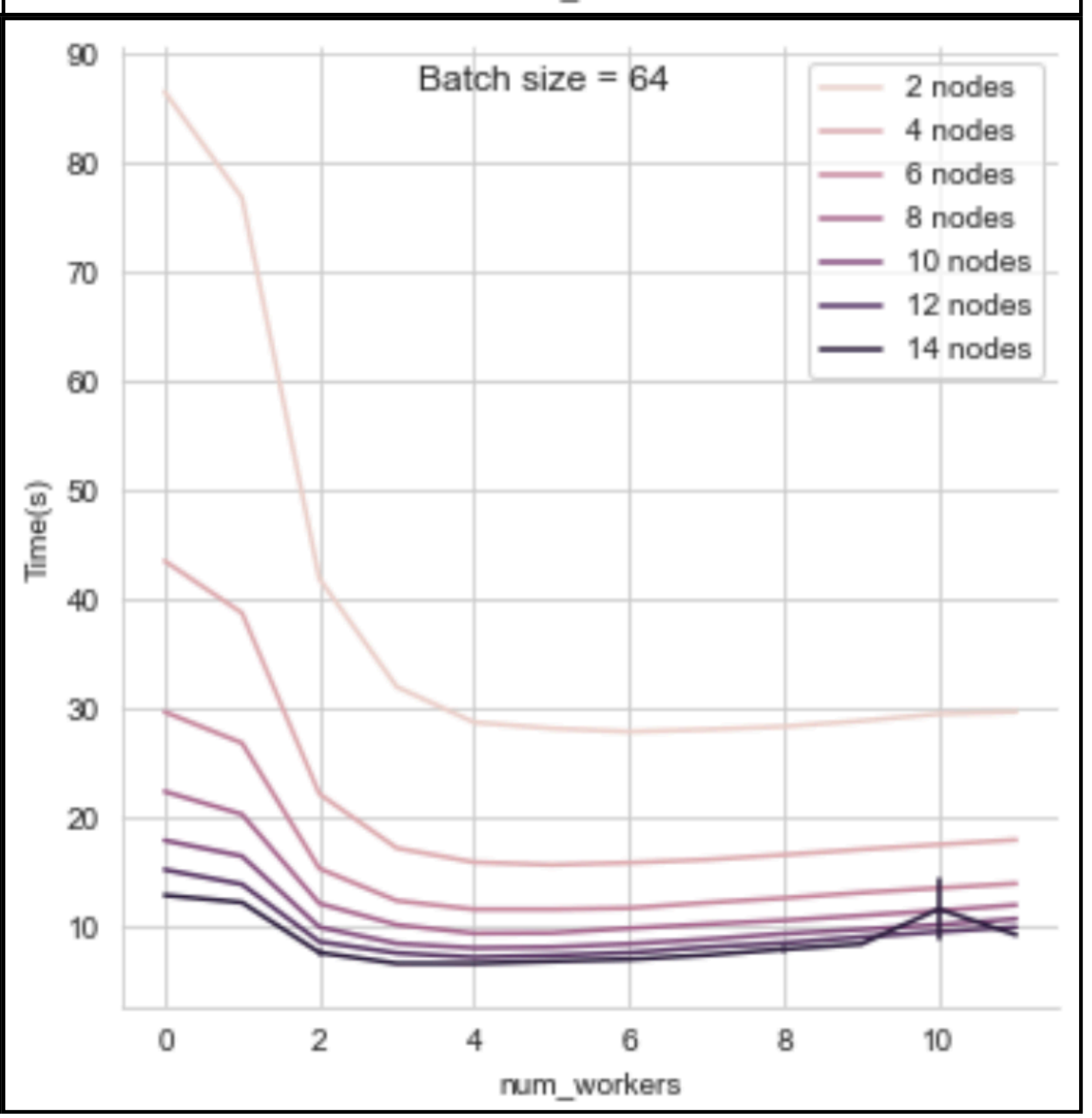
Model averaging




20 epochs
Accuracy
num_nodes
OnlyEnd
EachIter


20 epochs
Time(s)
num_nodes
Batch size = 32
Batch size = 48
Batch size = 64


20 epochs
Speedup ratio
num_nodes
Ideal speedup


Batch size = 64
Time(s)
num_workers
2 nodes
4 nodes
6 nodes
8 nodes
10 nodes
12 nodes
14 nodes