BATTLE-TESTING SWARM

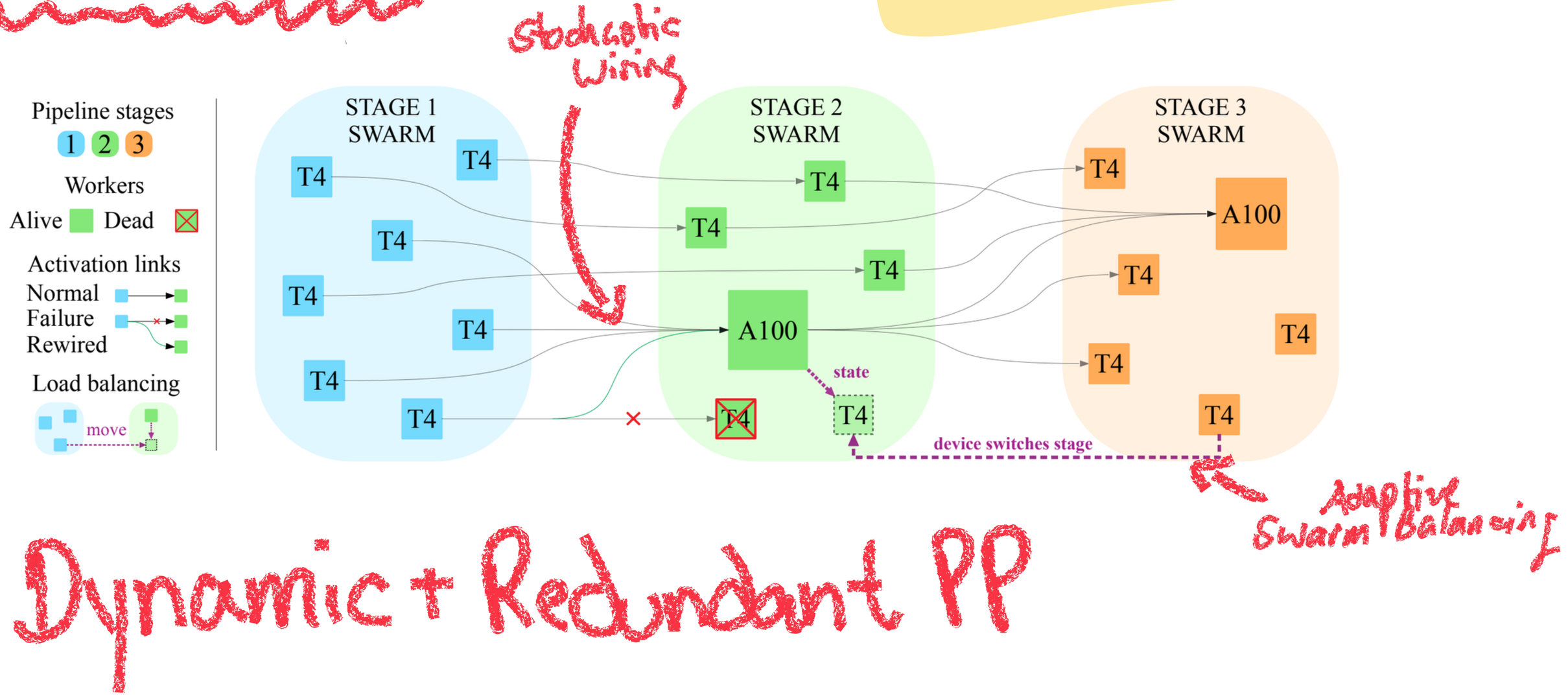
A RESEARCH OUTLINE

MIKA SENGHAAS

EPFL

SWARM

Remember?



Dynamic + Redundant PP

Decentralised Setting

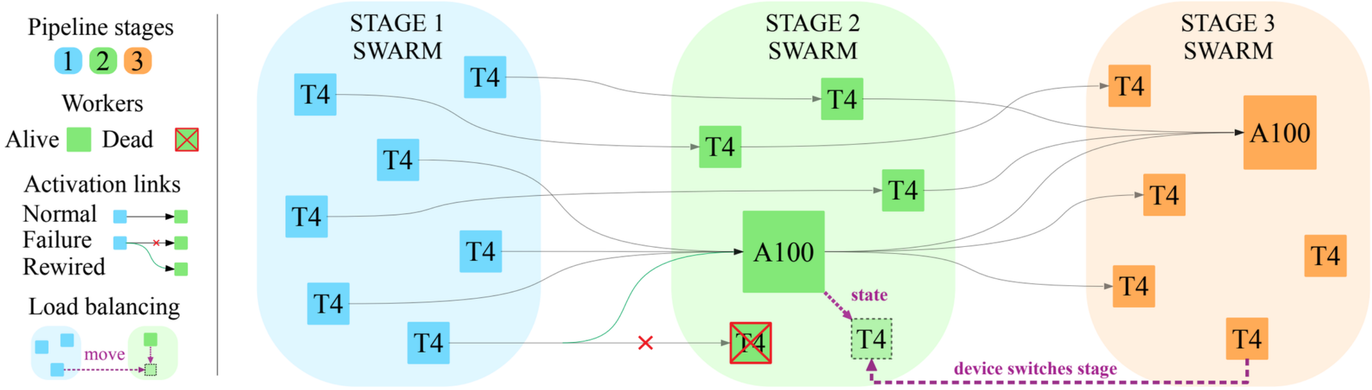
1. Heterogeneous Hardware
2. Heterogeneous Network
3. Reliability
4. Scale

↳ On-/Off Ramping

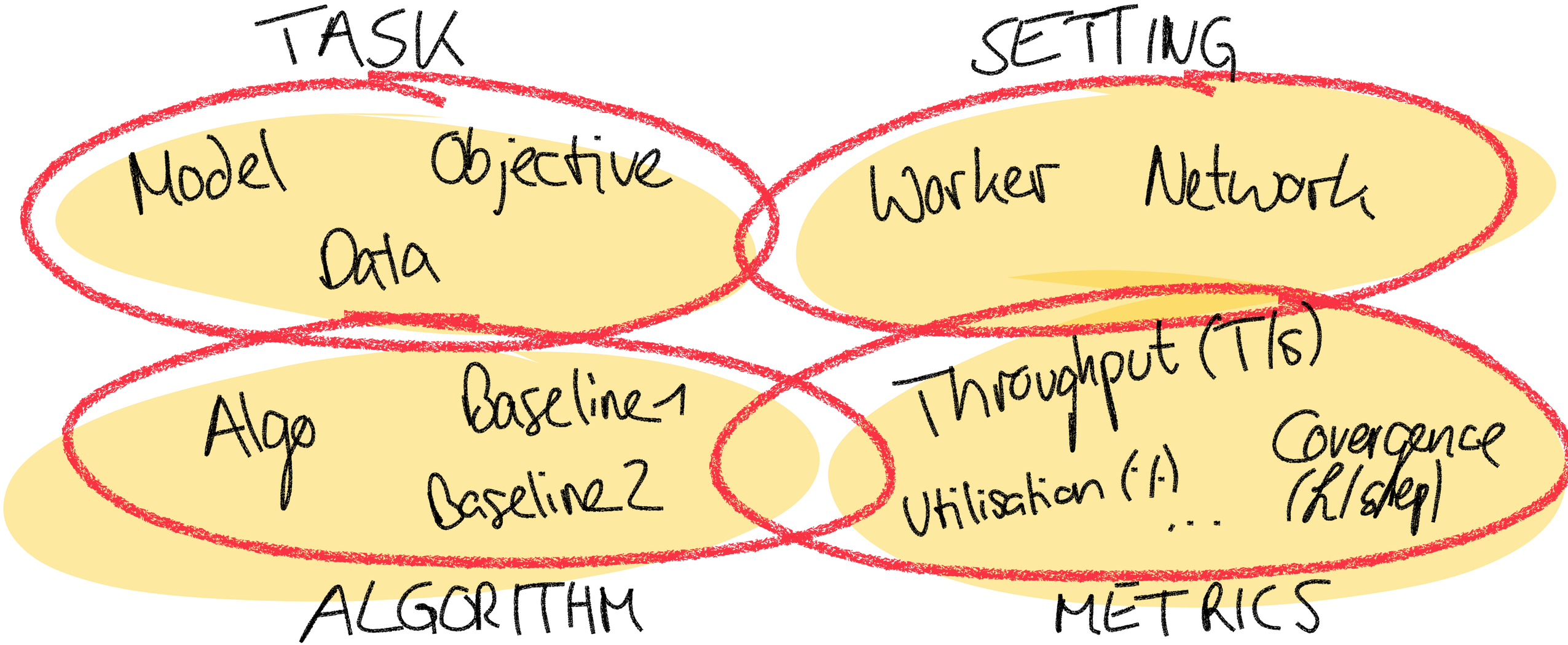
↳ Infinite Model + Data Scale

Decentral + SWARM

1. Heterogeneous Hardware ✓
2. Heterogeneous Network ✓
3. Reliability ✓
4. Scale ✓

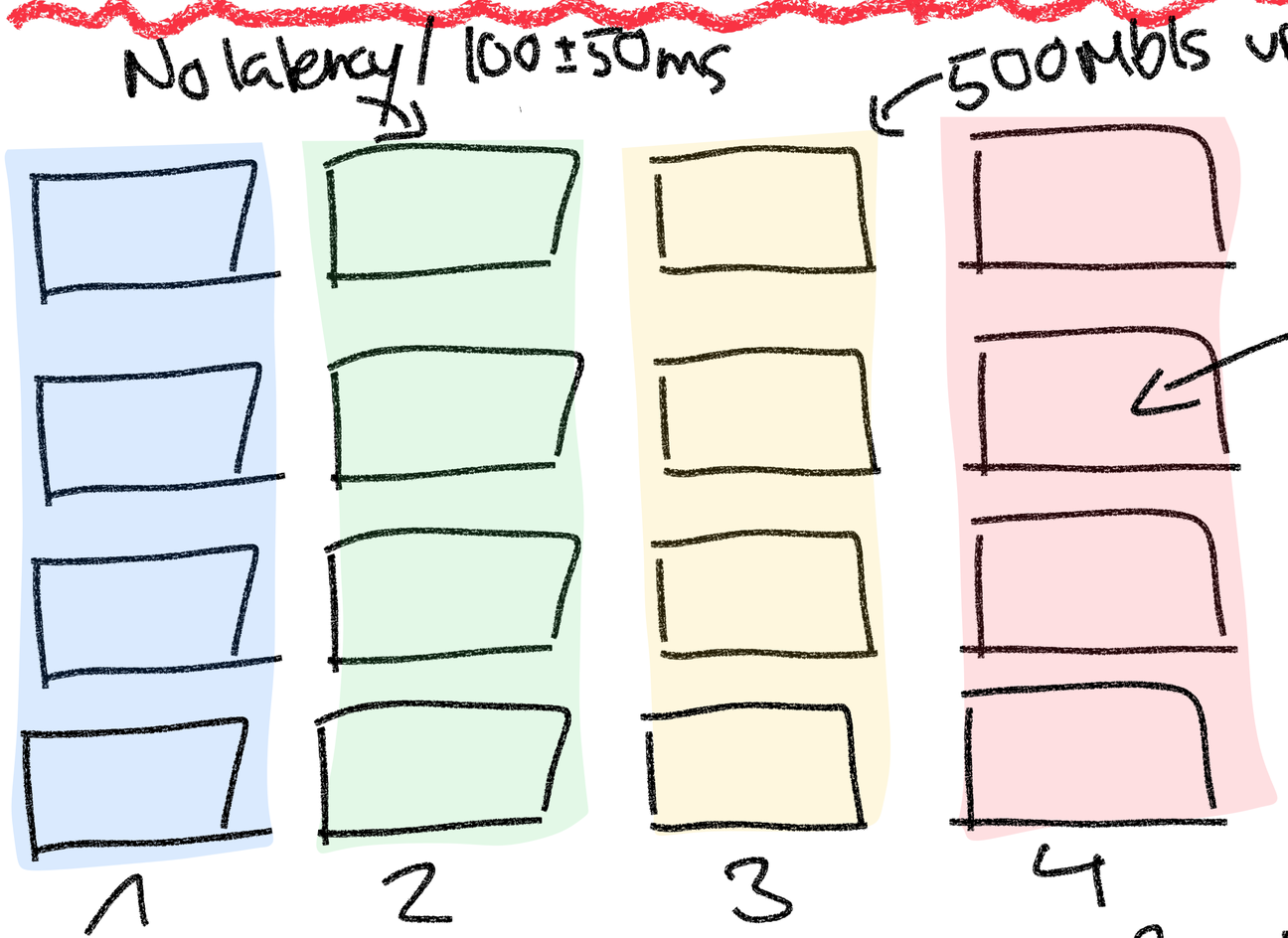


LET'S BENCHMARK


TASK
Model
Objective
Data
SETTING
Worker
Network
Algo
Baseline 1
Baseline 2
Throughput (T/s)
Utilisation (%)
...
Convergence (R/step)
ALGORITHM
METRICS

SWARM Experiment 1

+ Scale of Comm. Efficiency Experiment



V100 (128GB)
holding 4 layers

Metrics

Throughput (Time / 6250 ex)
AllReduce (Time)

Baselines

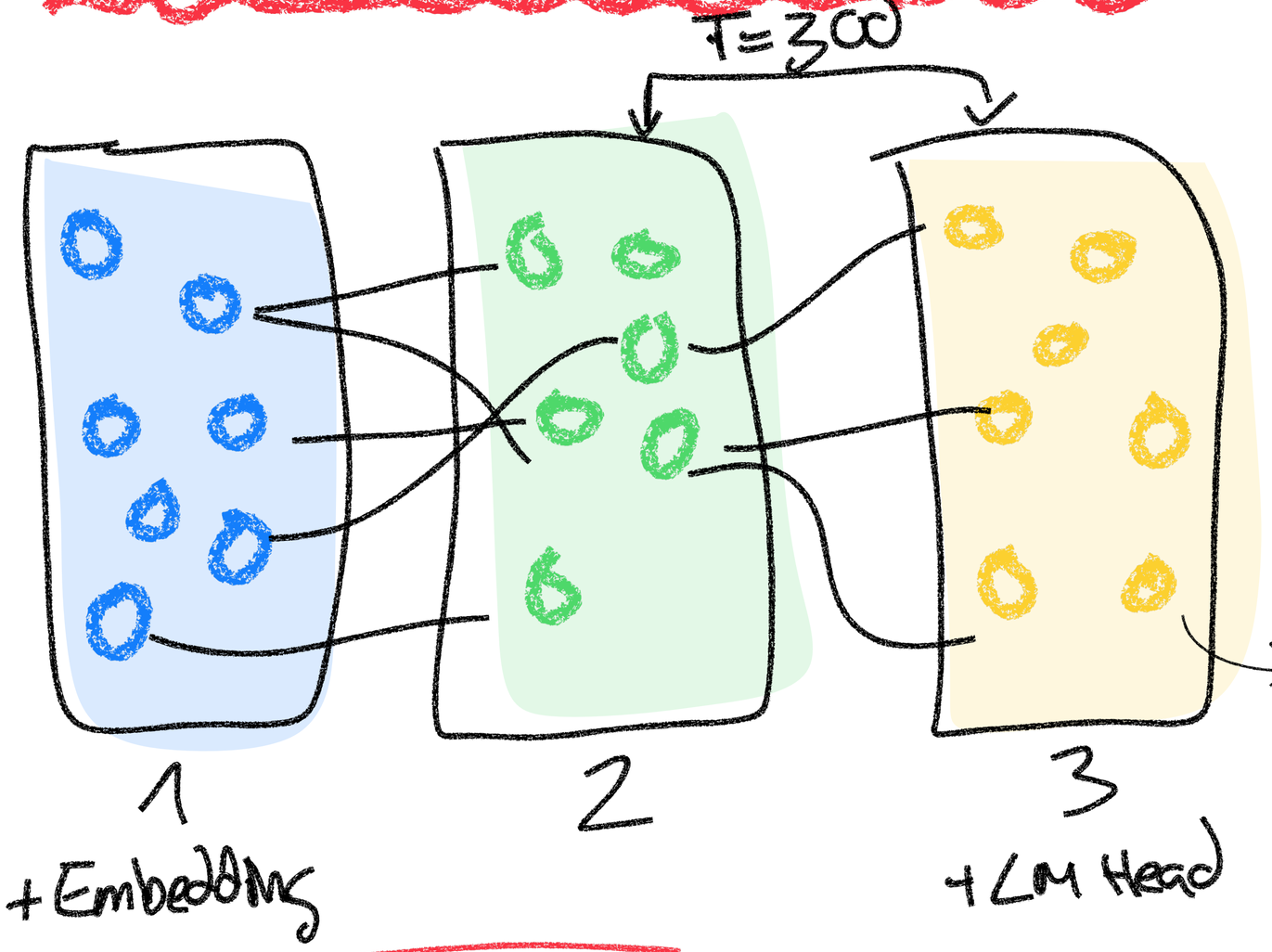
GPipe
ZeRO offload

Models

16 Transformer layers w/

① d=4096, h=32, b=4
② d=12288, h=96, b=1
③ d=4096, h=32, b=1

SWARM Experiment 2



Hardware: 400 T4

Model: 1.1B (effective 13B)

Data: Pile

Baseline: DDP + offloading on 8 A100

Verify Convergence

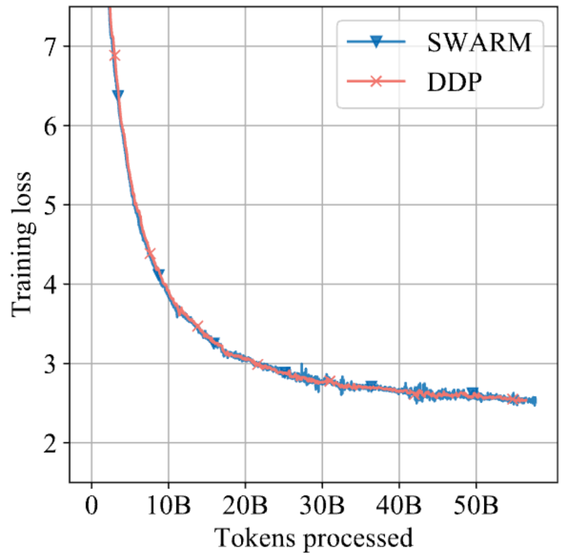
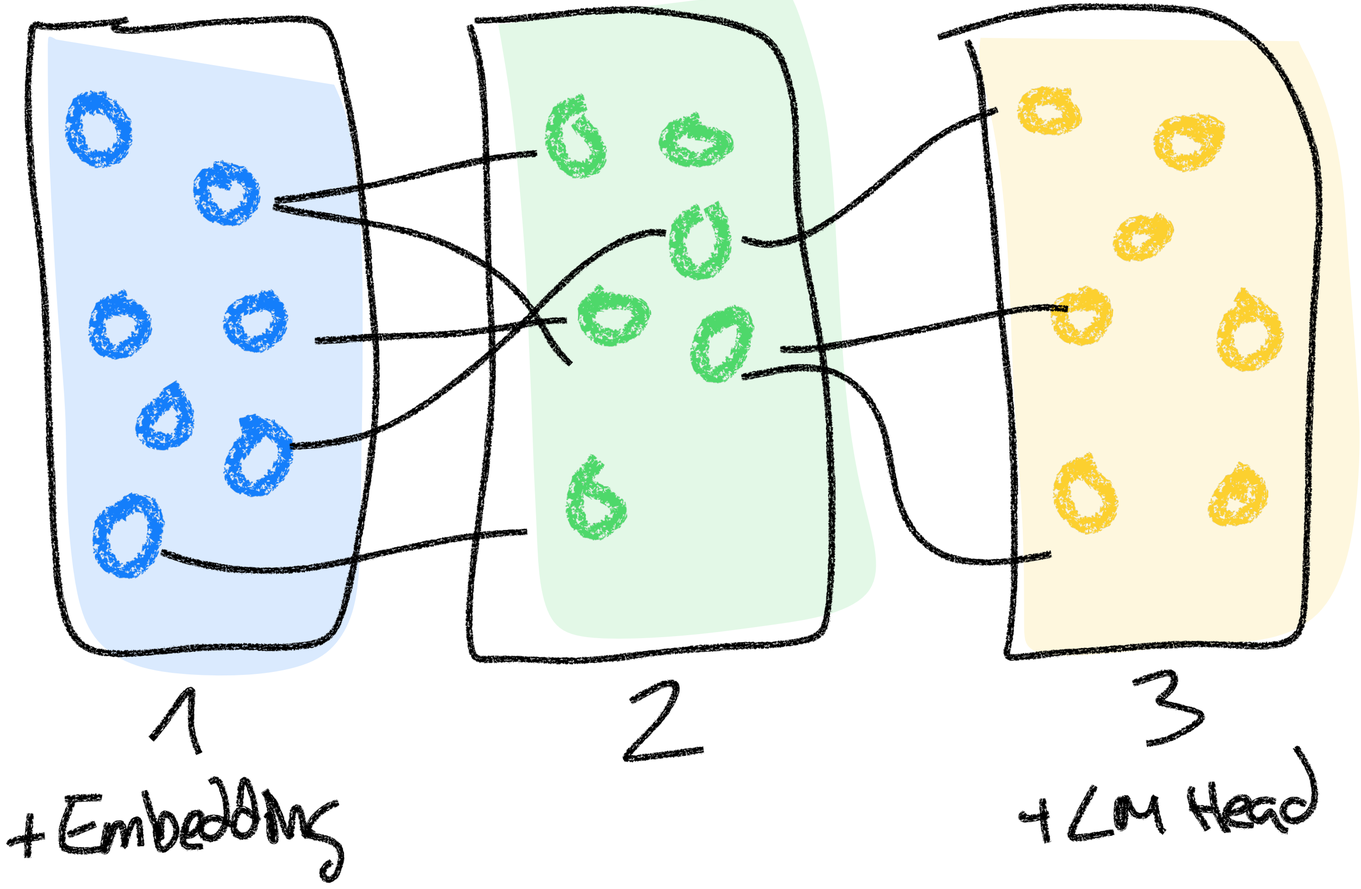


Figure 4: Training convergence comparison.

SWARM Experiment 3



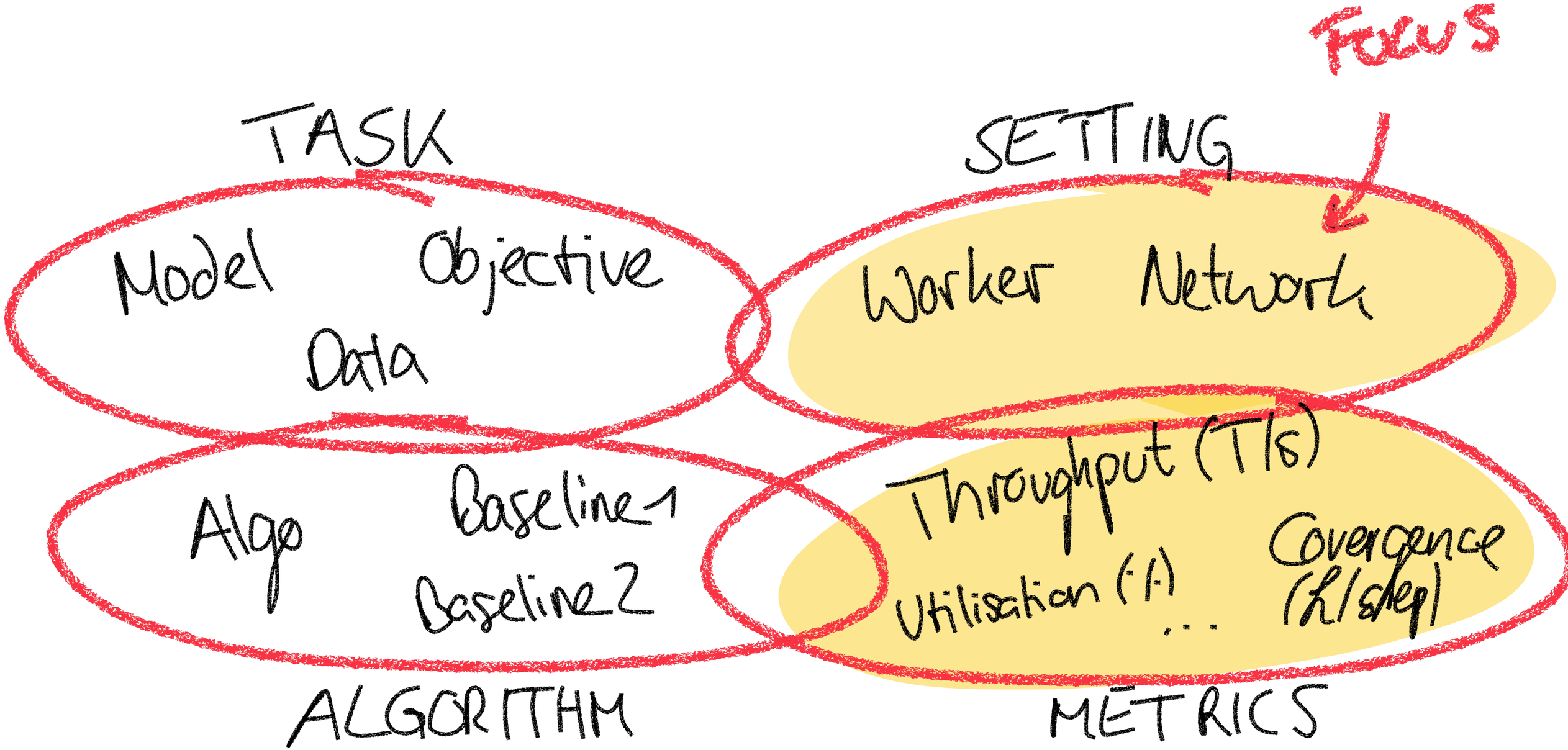
Hardware: 400T4 / 7x8A100 ⊕

Model:

Data: Pile

Baseline: Ideal Throughput (Avg. Processed Tokens/Stage x 3 x Num Nodes)

+Experiment on Rebalancing


FOCUS
TASK
SETTING
Model
Objective
Data
Worker
Network
Throughput (T/s)
Algo
Baseline 1
Baseline 2
Utilisation (%)
...
Convergence (R/step)
ALGORITHM
METRICS

Next Steps

Hivemind / Pytorch / ZeroBand

① Write-up Experiment Setup

→ Llama

→ C4

→ T4 / A100 / H100

Baselines

② Implementation

↳ Existing Frameworks / Code?

↳ MVP Sandbox Implementation

↳ Distribute step-by-step

Questions

1. SWARM
   - When to sync what inter-layer?
   - How to sync intra-layer?
   - Scheduling of backward path?
2. Tooling
   - Hivemind
   - OpenDiLoCo / Petals
3. Resources
   - Prime Compute
   - Local EPFL cluster?
4. Metrics
   - What is relevant?