MLLM Pipeline Bubble Modeling for Large-Scale Training

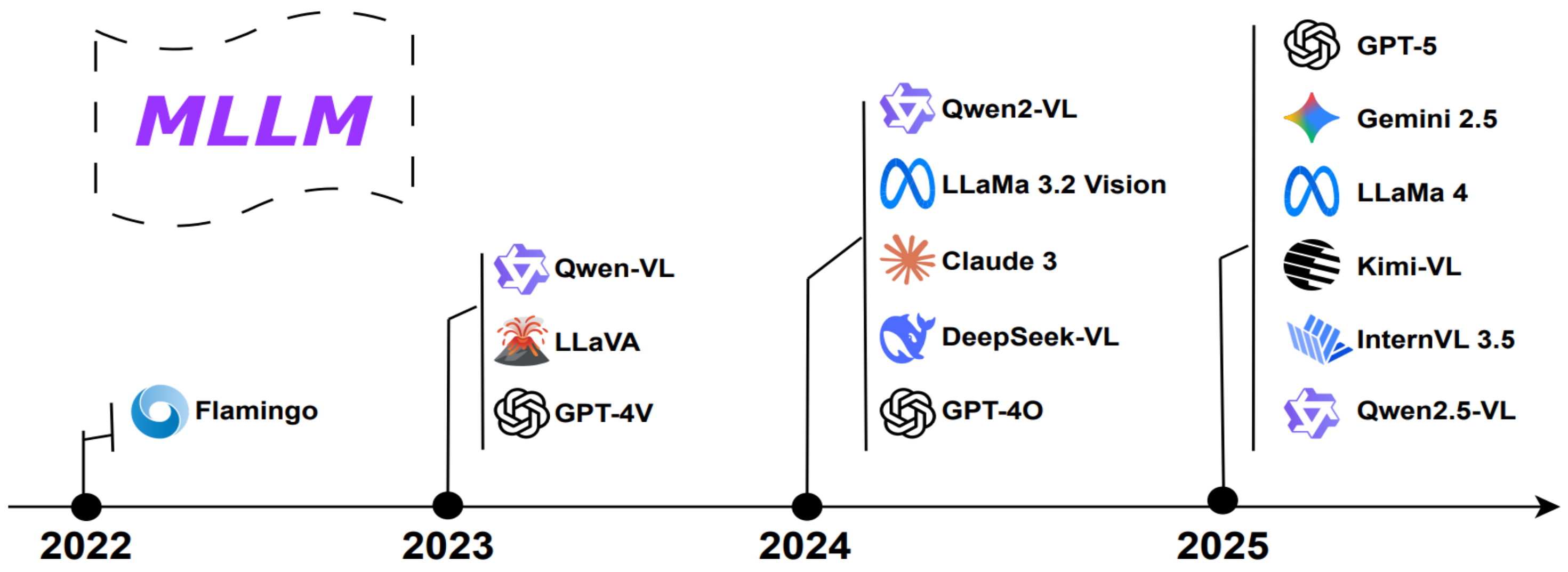
Zhengdao Yu[1,2], Ruiwen Wang[1,2], Nelson Lossing[2], Chong Li[2]

[1]Sorbonne University, [2]Huawei Paris Research Center

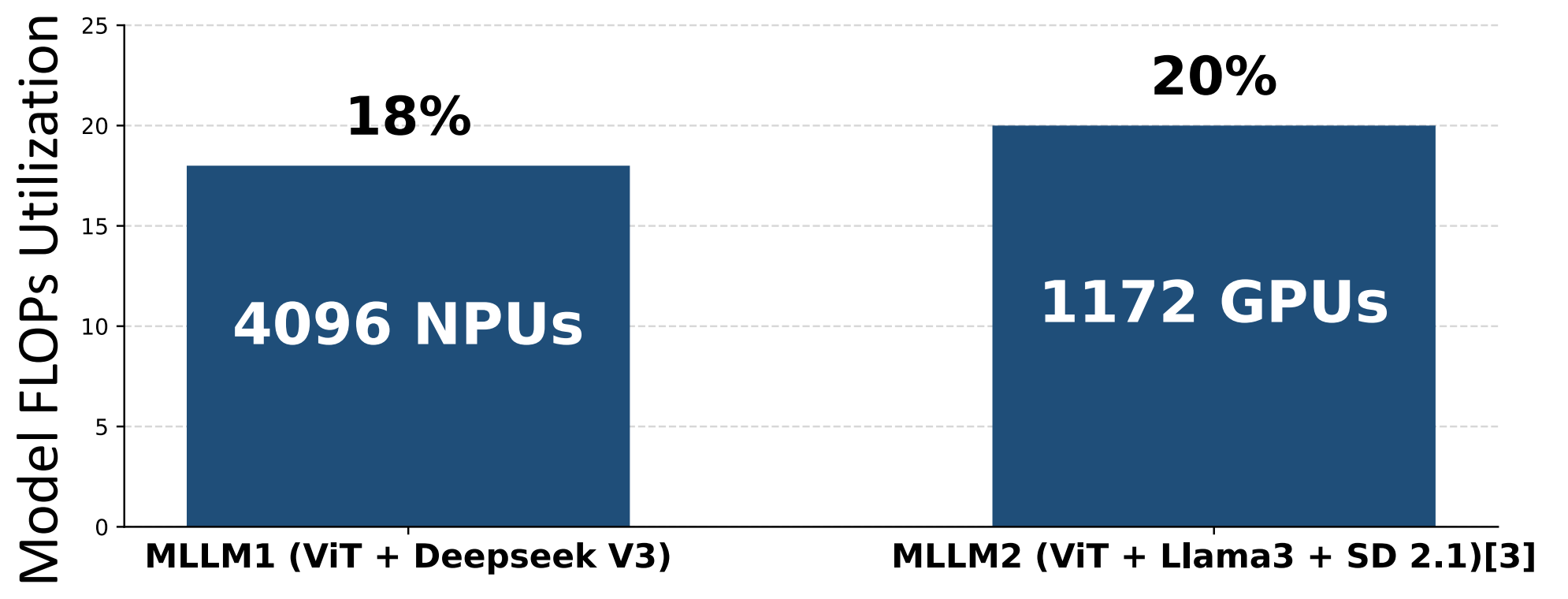
SCA/HPC Asia 2026

Background

Multimodal Large Language Models (MLLMs) have become a rising research focus. With recent models scaling to **hundreds of billions of parameters (B),** training MLLMs requires large-scale accelerator clusters (often with **thousands of accelerators**).



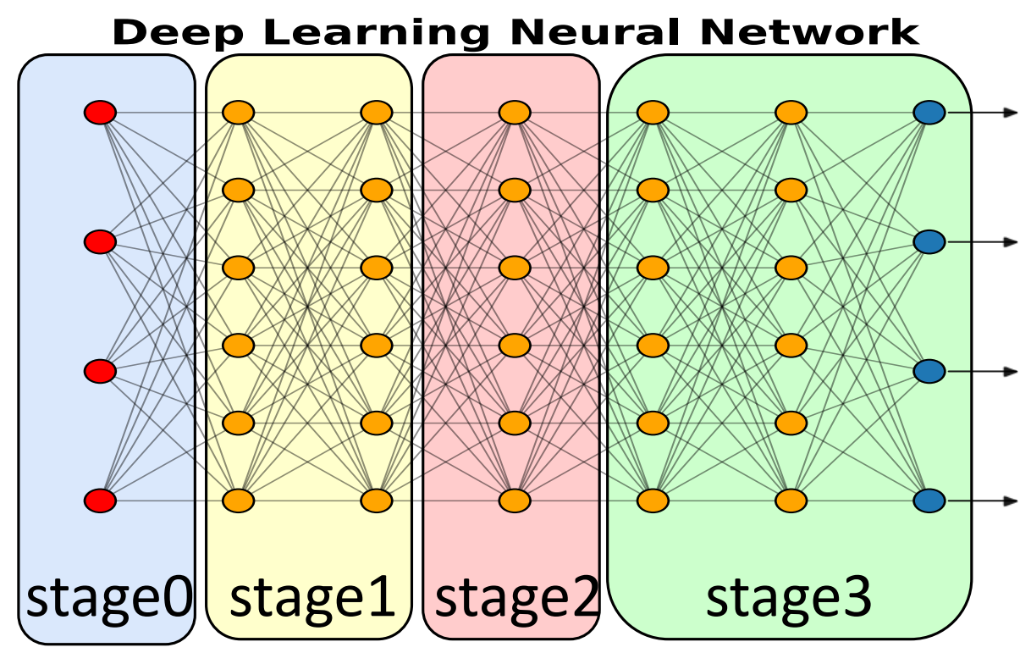
However, MLLM training often exhibits **low cluster utilization.**



Observation

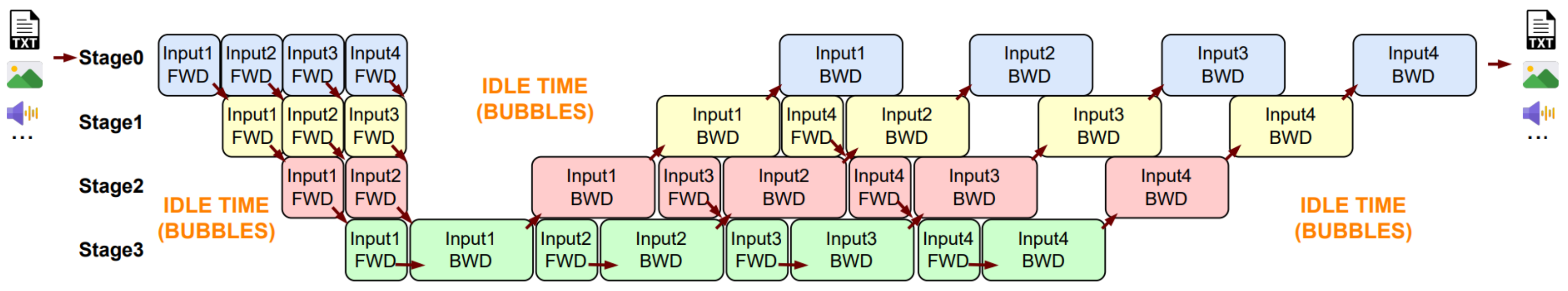
MLLM training faces substantial per-device memory pressure. To alleviate this, it commonly leverages pipeline parallelism (PP), which partitions the MLLM across multiple devices.

Illustration of pipeline parallelism (PP) using 4 stages



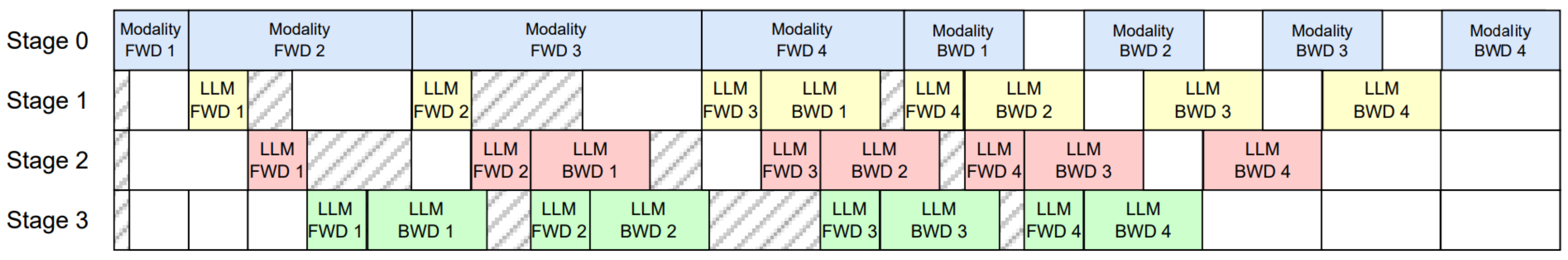
* Each stage corresponds to a device group in the large-scale cluster

The MLLM pipeline is dominated by substantial device idle time, which significantly degrades overall performance.



MLLM Modality-wise Pipeline Bubble Modeling

We first categorize the idle time in the pipeline (white regions).



A simplified MLLM pipeline execution timeline

- **Modality bubbles** B_γ **:**

Since modality inputs are usually unstable (e.g., images with different resolutions, videos with varying temporal lengths, etc.), idle time caused by variable modality inputs is classified as modality bubbles (shown as hatched white regions). To the best of our knowledge, there is no formal model to express MLLM training.

- **PP bubbles** $\bar{B}_\theta$ **:**

By temporarily setting the modal inputs to a constant value, we eliminate the variability in the modal input. As a result, the remaining pipeline idle time is attributed to PP bubbles, which could be reduced by existing pipeline scheduling techniques (e.g., [1][2])

- **Bubbles overlapping** $\bar{B}_\theta \cdot f^m(i)$:

We discovered that modality bubbles could be hidden inside PP bubbles. We thus introduce the overlapped-bubble term in our model.

The following formula introduces an overlapped bubble time term. By maximizing **overlapped-bubble time**, it reduces the exposed modality bubbles and thereby minimizes the step time T.

$$T^{l,m}_{\text{step},i} = \overbrace{(\bar{M}_b+\bar{B}_\theta)\cdot(T^{l,m}_{\text{fwd}}+T^{l,m}_{\text{bwd}})}^{\text{PP bubbles + computation time}} \overbrace{+B_\gamma\cdot\bar{M}_b\cdot T^{l,m}_{\text{fwd}}}^{+\ \text{Modality bubbles time}} \overbrace{-\bar{B}_\theta\cdot f^m(i)\cdot(T^{l,m}_{\text{fwd}}+T^{l,m}_{\text{bwd}})}^{-\ \text{Overlapped bubble time}}$$

- i pipeline stage index
- (l, m) local assignment of l (LLM) and m (modality) layers to stage i
- $T^{l,m}_{\text{step},i}$ step time of stage i under assignment (l, m)
- $\bar{M}_b$ number of micro-batches (global)
- $T^{l,m}_{\text{fwd}}$ avg. forward time per micro-batch at stage i
- $T^{l,m}_{\text{bwd}}$ avg. backward time per micro-batch at stage i
- $\bar{B}_\theta$ number of PP bubbles (global)
- B_γ number of Modality bubbles at stage i
- $f^m(i)$ the fraction of the B_m at stage i that can be hidden

Optimization target: maximizing overlapped-bubble time

1. Stage position: earlier stages (i↓)
2. Modal layers distribution: more modal layers (m↑)

Layer distribution	PP bubbles time	Computation time	Modality bubbles time	Overlapped bubble time	Step time
m↑	↑	↑	↑	↑↑	↓

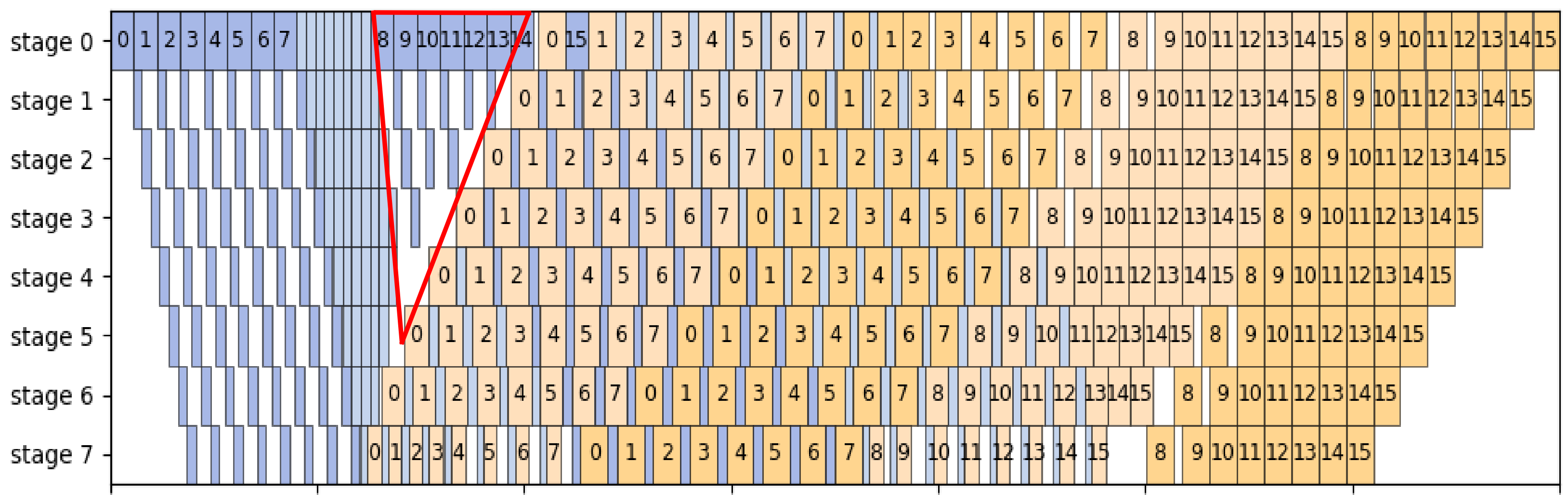
By adjusting (l, m), the gain from overlapped bubble time could outweigh the increased compute and bubble time, reducing the step time and yielding performance benefit.

End-to-End Training Performance Evaluation

Experimental Setup

- **Hardware**: Ascend-910 AI cluster (**4,096 NPUs**)
- **Model**: Qwen2-VL ViT + DeepSeek-V3 600-billion-parameter LLM
- **Framework**: MindSpore
- **Primary metric**: Training throughput
- **Results:** 6.37% improvement of training throughput

Modality bubbles are effectively overlapped by the PP bubbles at the front of the pipeline (red box marks overlap regions).



Pipeline execution timeline on 4,096×NPU cluster

Future Work

We have successfully took advantage of PP bubbles including overlapping with modality bubbles and reducing PP bubbles itself.

Our next step is to **minimize the unmasked modality bubbles** for MLLM training on large-scale AI clusters.

Reference

[1] Narayanan et al. 2021. Efficient large-scale language model training on GPU clusters using megatron-LM (SC '21).

[2] Sun et al. 2025. Seq1F1B: Efficient Sequence-Level Pipeline Parallelism for Large Language Model Training, Luis Chiruzzo, Alan Ritter, and Lu Wang (Eds.).

[3] Zhang et al. 2025. DistTrain: Addressing Model and Data Heterogeneity with Disaggregated Training for Multimodal Large Language Models (SIGCOMM '25).