HW7

Tguo67

Question(s):

Read the related materials, and answer the following questions:

1. What are the major features of Dynamo?
2. How does Dynamo select the best-matched prefill worker?
3. What are the evaluation workloads and metrics used in Nvidia's study "Beyond the Buzz"?
4. What factors are included in the conditional disaggregation of Dynamo? Based on the research results of Nvidia's study, why are these factors considered?
5. Critical thinking: Does Nvidia's study completely demonstrate the conditions where P-D disaggregation can be beneficial? What experiments should be added?

Related materials:

Dynamo official website: https://www.nvidia.com/en-us/ai/dynamo/Links to an external site.

Dynamo Git Repo: https://github.com/ai-dynamo/dynamoLinks to an external site. (You may need to check components/backends/vllm/src/dynamo/vllm_prefill_router/__main__.py and lib/llm/src/disagg_router.rs)

Dynamo documentation: https://docs.nvidia.com/dynamo/latest/index.htmlLinks to an external site.

Beyond the Buzz paper: https://arxiv.org/html/2506.05508v1Links to an external site.

Answers:

Major Dynamo features:

Disaggregated prefill & decode (P-D) serving, with conditional disaggregation, KV-aware request routing, dynamic GPU scheduling/planners, and multi-engine support (vLLM/SGLang/TensorRT-LLM). Also: accelerated KV transfer via NIXL, KV cache offloading/tiering (KVBM), request migration/HA router, and multi-node deployments.

How Dynamo picks the "best-matched" prefill worker:

With KV-aware routing, engines publish their KV-block metadata, the router scores candidate workers by prefix/KV overlap. while also considering load, then routes to the worker with the highest score. If KV routing is off, it can fall back to round-robin/random.

The evaluation workloads and metrics used in Nvidia's study "Beyond the Buzz":

Models/workloads: DeepSeek-R1 and Llama-3.1-70B (plus Llama 8B/70B/405B sensitivity), explored across traffic patterns (varying input sequence length ISL and output sequence length OSL) and latency targets, sensitivity to NVLink domain size and parallelism mixes (TP/EP/PP/CPP/TEP). Evaluations are produced by a datacenter-scale GPU inference simulator targeting modern Blackwell systems (FP4).

Metrics: throughput vs. interactivity Pareto frontier; FTL/TTFT (first-token), TTL/ITL (per-token), tokens/s/user; they rate-match prefill:decode capacity subject to FTL/TTL constraints.

What's in Dynamo's conditional disaggregation, and why:

Heuristics: (a) Local prefill length threshold (max-local-prefill-length)—short prompts are prefilled locally; (b) Remote prefill queue depth threshold—if the global prefill queue is long, prefer local prefill to avoid extra waiting. The queue itself is a NATS stream; prefill workers pull and process remote requests.

Why these factors: Nvidia's study finds P-D shines for prefill-heavy traffic and under designs that keep prefill TTFT low while rate-matching decode capacity. When prompts are short or prefill is backlogged, remote prefill can add overhead/queuing, eroding benefits—hence the length/queue conditions

Critical thinking: Does Nvidia's study completely demonstrate the conditions where P-D disaggregation can be beneficial? What experiments should be added?:

It's simulation-driven (explicitly), with assumptions like immediate, overlapped KV transfer; real clusters face NIC/NVLink contention, queueing jitter, failures, and cache-reuse skew. So there is still room for the improvements.

Some experiments like heavy KV transfer or straggler related events can be helpful.