Beyond A Single AI Cluster: A Survey of Decentralized LLM Training

Haotian Dong[1]*, **Jingyan Jiang**[2]*, **Rongwei Lu**[1], **Jiajun Luo**[1],
Jiajun Song[1], **Bowen Li**[1], **Ying Shen**[3]†, **Zhi Wang**[1]†
[1]Shenzhen International Graduate School, Tsinghua University
[2]Shenzhen Technology University [3]China Central Depository & Clearing Co., Ltd.
donght24@mails.tsinghua.edu.cn, jiangjingyan@sztu.edu.cn
wangzhi@sz.tsinghua.edu.cn

Abstract

The emergence of large language models (LLMs) has revolutionized AI development, yet their resource demands beyond a single cluster or even datacenter, limiting accessibility to well-resourced organizations. Decentralized training has emerged as a promising paradigm to leverage dispersed resources across clusters, datacenters and even regions, offering the potential to democratize LLM development for broader communities. As the first comprehensive exploration of this emerging field, we present decentralized LLM training as a resource-driven paradigm and categorize existing efforts into community-driven and organizational approaches. We further clarify this through: (1) a comparison with related paradigms, (2) characterization of decentralized resources, and (3) a taxonomy of recent advancements. We also provide up-to-date case studies and outline future directions to advance research in decentralized LLM training.

1 Introduction

The rapid advancement of LLMs has yielded remarkable progress across a wide range of domains (DeepSeek-AI et al., 2025b). With model scales expanding from GPT-3's (Brown et al., 2020) 175 billion to DeepSeek-R1's (DeepSeek-AI et al., 2025a) 660 billion parameters, the computing resource demands of training LLMs have increased dramatically (Jiang et al., 2024).

However, this exponential growth in computational requirements poses significant challenges. For individual researchers and small laboratories with limited resources, the demands are particularly prohibitive. Even for well-resourced organizations, confining LLM training within a single AI cluster faces challenges like: geographically distributed services required for latency optimization (McMahan et al., 2017), inherent hardware bottlenecks limiting single-cluster scalability (Athlur et al., 2022; Grattafiori et al., 2024), economic patterns requiring placement adaptation (Liu et al., 2023), etc.

These challenges underscore the necessity for innovative resource management approaches to enhance the accessibility of LLM training. One such approach is *decentralized LLM training*, a distributed paradigm that leverages decentralized resources at varying scales to achieve greater scalability and cost-efficiency.

Training LLMs with decentralized resources faces inherent challenges across different scenarios. For individual researchers and communities collaborating in decentralized environments, key challenges include dynamic resource availability, limited bandwidth in wide area networks (WANs), and heterogeneous computing capabilities (Borzunov et al., 2023; Yuan et al., 2022; Yang et al., 2024). For well-resourced organizations managing multiple clusters or datacenters, it is essential to consider not only communication efficiency but also energy consumption minimization and cross-datacenter workload scheduling coordination (Park et al., 2024; Choudhury et al., 2024). Furthermore, the inherent complexity of the LLM training process exacerbates these challenges. Modern LLM training relies on a hybrid of parallelization strategies to efficiently coordinate computing resources (Narayanan et al., 2021), which significantly amplifies the difficulties when operating within a decentralized infrastructure.

This survey systematically investigates challenges and solutions for decentralized LLM training. Compared to prior surveys on other distributed paradigms, our paper centers around LLM training with decentralized resources, as shown in Table 1. We aim to characterize the utilization of decentralized resources and analyze optimization methods in decentralized LLM training, thereby exploring potential research opportunities. Figure 1 illustrates

* Equal Contribution.
† Corresponding Author.

Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing, pages 29187–29201
November 4-9, 2025

Surveys	LLM-Focused	Resource-Driven	Cross-Regional	Paradigms
(Duan et al., 2024)	✓	✓		Efficient LLM Training, Centralized Infrastructures
(Khan et al., 2023)			✓	Decentralized Machine Learning, Geo-Distributed Machine Learning
(Woisetschläger et al., 2024)	✓		✓	Federated Learning, Efficient Foundation Model Training
Ours	✓	✓	✓	Decentralized LLM Training, Decentralized Infrastructures

Table 1: Comparison with related surveys. These paradigms overlap in terms of scenarios and optimization targets. To illustrate decentralized LLM training and resources, comparison and analysis are presented in §2 and §3.

the utilization paradigms of resources in decentralized LLM training and optimization objectives we categorize in this paper. To the best of our knowledge, our survey is the *first* to study recent advances in decentralized LLM training. Our work contains three distinct contributions:

- **A resource-driven position of decentralized LLM training**. We position decentralized LLM training as a resource-driven paradigm by comparing with related paradigms and examining the characteristics of decentralized resources.
- **A novel taxonomy on decentralization paradigms and optimization objectives of decentralized LLM training**. We classify the utilization of decentralized resources into two paradigms: community-driven and organizational. Then we taxonomize recent studies based on optimization objectives and review related methodologies.
- **Up-to-date case studies and prospective future research directions**. We compare two LLMs trained under different decentralized paradigms: one leveraging organizational resources and the other utilizing scattered, fragmented resources. We also propose potential directions for future research.

2 Background

The concept of decentralized LLM training intersects with several distributed machine learning paradigms, including *Geographically-Distributed Machine Learning (Geo-ML)*, *Federated Learning (FL)*, *Decentralized Machine Learning (De-ML)*, and *Efficient LLM Training*. We review these paradigms to highlight the distinctive characteristics of decentralized LLM training as a resource-driven paradigm.

2.1 Geo-ML and FL

Geo-ML and FL both tackle challenges from data decentralization, yet each approach has its distinct focus. Geo-ML mainly tackles service latency and regulatory requirements by optimizing training processes across datacenters, leveraging hierarchical network topologies (e.g., high-bandwidth local area networks (LANs) within datacenters and limited-bandwidth WANs between datacenters) to utilize decentralized data and computational resources efficiently (Hsieh et al., 2017). In contrast, FL focuses more on privacy protection (McMahan et al., 2017), employing either centralized parameter-server or decentralized methods (Lalitha et al., 2019; Xing et al., 2021), with resources from decentralized clusters or edge devices.

2.2 De-ML

De-ML has a dual meaning in terms of decentralization. From the distributed strategy perspective, it mitigates communication bottlenecks inherent in parameter server architectures through peer-to-peer networking (Hegedűs et al., 2019; Warnat-Herresthal et al., 2021). From the resource utilization perspective, it offers a cost-effective and flexible strategy that maximizes the utilization of geographically dispersed computing resources (Yuan et al., 2022). While both De-ML and Geo-ML involve training architectures, they differ fundamentally in their approach—the former adopts peer-to-peer topology where nodes communicate directly, while the latter relies on hierarchical centralized architecture both across and within datacenters.

2.3 Efficient LLM Training

Currently, LLM training primarily employs distributed methods with centralized resources. However, efficient resource utilization remains challenging even within a single cluster. Efficient LLM training methods (e.g. 3D parallelism (Narayanan et al., 2021), mixed-precision training (Micikevicius et al., 2018), gradient compression (Lu et al., 2024b), etc.) have been proposed to address scalability, efficiency, and reliability challenges (Duan et al., 2024). These approaches are also essential for decentralized LLM training, where resource constraints are more severe and complex.

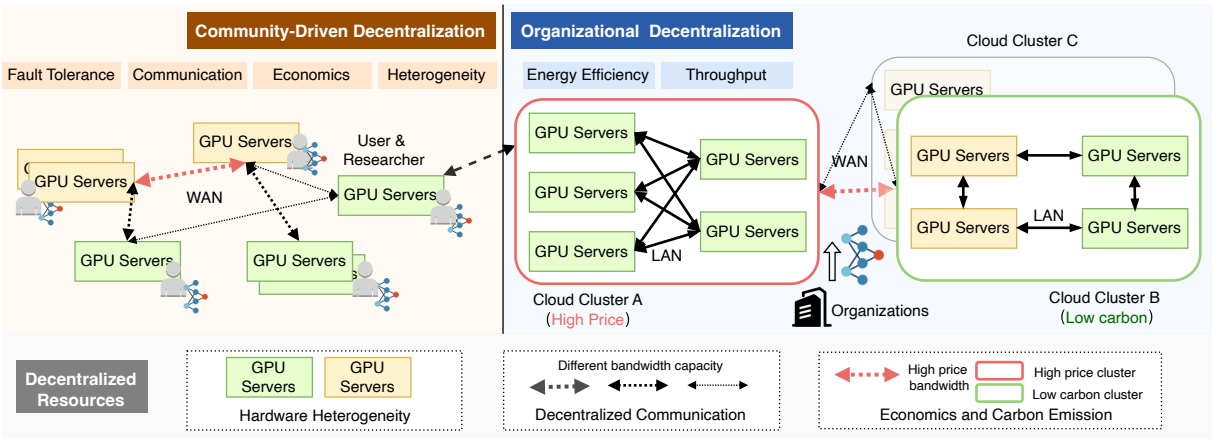

Figure 1: Decentralized LLM training paradigms of both community-driven and organizational decentralization. **Community-driven decentralization** utilizes pooling resources from communities with independent entities. These pooled resources can include local servers from researchers, instances from cloud services or volunteer communities, etc. **Organizational decentralization** involves consolidating resources from multiple clusters or even multiple datacenters managed by well-resourced organizations (e.g., technology giants and governments).

2.4 Decentralized LLM Training

Position. *We position decentralized LLM training as the convergence of decentralized ML and efficient LLM training, primarily driven by resources distribution akin to Geo-ML.*

Decentralized LLM training leverages distributed resources at varying scales to meet the substantial computational demands. Resource-limited communities rely on fragmented, globally contributed resources to meet LLM training demands. For well-resourced organizations, the resource demands for training ultra-scale LLMs necessitate the combination of consolidated resources from multiple clusters or datacenters to enable relatively decentralized training (Qian et al., 2024; Grattafiori et al., 2024). Based on these resource utilization paradigms, we categorize decentralized LLM training into: *community-driven decentralization* and *organizational decentralization*. These two decentralization paradigms are depicted in Figure 1.

The fundamental distinction between utilizing decentralized resources for training LLMs and general models lies in the demand of more sophisticated parallelization strategies. In LLM training, the substantial model parameter scale coupled with extensive activation values generated by auto-regressive Transformer architecture (e.g., GPT-3's activation memory during training exceeds its model parameters by more than 5× (Narayanan et al., 2021)), rendering simple Data Parallelism (DP) inadequate. This necessitates the adoption of strategies like Tensor Parallelism (TP) and Pipeline Parallelism (PP) to effectively distribute activation memory across GPUs, typically forming a comprehensive 3D parallelism. Decentralized LLM training leverage a broader range of resources compared to traditional distributed LLM training, yet it still employ similar parallel strategies. DP and PP are currently the primary parallel strategies[1] for training LLMs with decentralized resources (Yuan et al., 2022; Ryabinin et al., 2023). Due to intensive communication requirements, TP is more constrained within both LANs and WANs.

3 Decentralized Resources

Training LLMs with decentralized resources efficiently is challenging due to resource constraints in communication, computational heterogeneity, and cost disparities. We present characteristics of decentralized resources to illustrate these constraints in this section.

3.1 Communication Constraint

In distributed training, computational devices need to communicate frequently. However, decentralized resources have more limited communication capabilities compared to centralized ones.

Community-driven decentralized resources, combining with local servers and cloud instances, communicate through LANs or even WANs during model training with bandwidths under 10 Gb/s (Azurespeed, 2024). In contrast, organization-driven decentralized training can reach over 500

[1]More details about parallel strategies of distributed training are presented in Appendix A

GB/s of bandwidth within a server by NVLink, or over 200 Gb/s within a cluster using InfiniBand. However, when the scale of resources expands across multiple geographically distributed clusters or datacenters, the bandwidth can drop to only hundreds of Mb/s (Xiang et al., 2022). To alleviate communication bottlenecks, strategies such as gradient compression (Lu et al., 2024b, 2025b) and delayed aggregation (Zhu et al., 2021; Lu et al., 2025a) are often employed during distributed training.

3.2 Hardware Heterogeneity

In distributed training, computational devices are typically configured with identical specifications (e.g., memory capacity and FLOPS[2]) to prevent slower ones from becoming bottlenecks (Shen et al., 2024).

However, in decentralized paradigms, when aggregating resources across multiple nodes, clusters, or datacenters, the heterogeneity of resources becomes significant. This is particularly evident in communication, computation, and hardware architecture (Yuan et al., 2022; Yang et al., 2024). For instance, in geographically distributed clusters, computational devices like GPUs and NPUs with different architectures may coexist, and the communication bandwidth between clusters varies (Xiang et al., 2022). Due to the fast-paced evolution of computing devices, hardware heterogeneity is almost inevitable (Tang et al., 2023).

3.3 Cost Volatility

Training ultra-scale LLMs typically requires tens of thousands of GPUs running for thousands of hours, resulting in significant computational costs and energy consumption (Grattafiori et al., 2024).

However, reducing costs with decentralized resources is often more challenging than centralized setups. In community-driven decentralization, fragmented resources such as preemptive cloud instances are relatively cheaper, but their prices are highly volatile (Lee and Son, 2017). Additionally, resource instability can lead to re-computation and inter-node waiting, prolonging training process and increasing overall costs. For organizational decentralization, accurately modeling energy consumption and optimizing resource scheduling become increasingly complex, owing to the demand for massive resources and the inherently cross-datacenter nature of this paradigm (Faiz et al., 2024; Choudhury et al., 2024).

[2]FLOPS (Floating-Point Operations Per Second) measures the number of floating-point arithmetic operations (e.g., addition, subtraction, multiplication, division) that a computational unit can perform in one second.

4 Community-Driven Decentralization

Training LLMs under community-driven paradigm often encounters challenges, including hardware heterogeneity, limited communication bandwidth, resource instability, and economic volatility. This chapter focuses on these challenges and provides an in-depth discussion of related research efforts. We summarize and compare selected studies in Table 2. A more comprehensive paper list is presented in Figure 3 of Appendix B.

4.1 Communication Optimization

Due to bandwidth limitations of LANs and WANs, it is essential to optimize communication efficiency for decentralized LLM training to alleviate bottleneck. Primary strategies can be divided into optimizations at both temporal and spatial levels, corresponding to: (1) reduce communication frequency; (2) reduce communication intensity.

Reduce Communication Frequency. During community-driven decentralization, the fragmented resources used are often distributed across regions or even globally. As a result, the communication process involves local and global levels, in which the global often becomes the bottleneck.

In DP-only strategy, the primary communication payload comes from gradient transmission. Gaia (Hsieh et al., 2017) dynamically eliminate insignificant gradients to reduce communication across datacenters. Co-learning (Mi et al., 2020) enlarge the number of local epochs dynamically to reduce global synchronization frequency between datacenters. DeDloc (Diskin et al., 2021) adopt large local batches while training to allow peers to communicate less frequently. DiLoCo (Douillard et al., 2024) only synchronize globally once after 500 local optimization steps, effectively reducing communication frequency.

In PP strategy, gradients are exchanged within a stage, while both gradients and activations are transmitted across stages. Varuna (Athlur et al., 2022) leverages rule-based policy to adjust PP depth based on the available GPU count to better accommodate bandwidth constraints. To allocating tasklets requiring high communication volume to computing units with faster connections,

Papers	Parallel Strategy	Model Type	Resource Scale	Comm.	Hete.	Faul.	Econ.
L@H (Ryabinin and Gusev, 2020)	Decentralized Mixture-of-Experts	Transformer-XL (1.3B, MoE)	4 GTX 1080 GPUs	✓	✓	✓	
DeDloc (Diskin et al., 2021)	DP with extremely large batches	BERT (170M)	91 devices (RTX 2060, K80, etc.)	✓	✓	✓	
Moshpit (Ryabinin et al., 2021)	DP with dynamic local groups	ALBERT (18M)	8 V100 GPUs and 66 GPU instances	✓	✓	✓	
Varuna (Athlur et al., 2022)	PP with inner-stage DP	GPT-2 (200B)	Low-priority spot VMs with 300 GPUs	✓		✓	
AQ-SGD (Wang et al., 2022)	PP with inner-stage DP	GPT2 (1.5B)	32 V100 GPU instances	✓			
DTFM (Yuan et al., 2022)	PP with inner-stage DP	GPT3 (1.3B)	64 V100 GPUs of 8 nodes	✓	✓		
SWARM (Ryabinin et al., 2023)	PP with inner-stage DP	Transformer (1.01B)	Spot instances with 400 T4 GPUs	✓	✓	✓	
Petals (Borzunov et al., 2023)	PP with dynamic stage	BLOOM (176B)	27 GPUs (A100, RTX3090, A4000, etc.)	✓	✓	✓	
FusionAI (Tang et al., 2023)	PP with load-balance scheduling	GPT-3	50 RTX 3080 GPUs	✓		✓	
Ravnest (Menon et al., 2024)	PP with inner-stage DP	BERT-base (110M)	10 nodes (A10G, V100, T4) in 4 clusters	✓	✓	✓	
StellaTrain (Lim et al., 2024)	DP with dynamic batch	GPT-2 (123.6M)	10 GPUs (V100, RTX4090, etc.)	✓	✓		✓
Holmes (Yang et al., 2024)	PP with inner-stage DP and TP	GPT (7.5B)	64 A100 GPUs of 8 nodes	✓	✓		
Atom (Wu et al., 2024)	DP with memory swapping	GPT-3 (13B)	12 GPUs (V100, RTX 1080Ti, etc.) of 3 nodes		✓	✓	
Positon (Lu et al., 2024a)	PP with layer skipping	Bloom (7B)	6 nodes (A40, V100, etc.)			✓	
MLTC (Strati et al., 2024)	PP and DP comparison	OPT (30B)	85 A100 GPUs				✓
DiLoCo (Douillard et al., 2024)	DP with large local steps	Simple Transformer (400M)	8 A100 nodes (16GPUs each)	✓		✓	
HowCW (Erben et al., 2024)	DP with target batch size	RoBERTa-XLM (560.1M)	8 VMs cross continents			✓	✓

Table 2: Comparison of related works in community-driven decentralization. The symbol ✓indicates whether a paper primarily focuses on a specific optimization objective. Comm: communication efficiency; Hete: network or device heterogeneity; Faul: fault tolerance; Econ: economics. For the model type, the largest language model used in each paper is selected. Papers where language models were not explicitly used are included in Appendix B.

DTFM (Yuan et al., 2022) uses a two-level approach: a balanced graph partitioning problem for DP within each stage, and a joint graph matching and traveling salesman problem for the entire PP process. More dynamically, SWARM (Ryabinin et al., 2023) optimizes the PP process by enabling real-time adjustments during each iteration. In this strategy, each PP stage uses multiple candidate devices. When a device outperforms others, it processes inputs from multiple slower predecessors and distributes outputs to multiple slow successors, maximizing bandwidth utilization.

Reduce Communication Intensity. Similar to training LLMs with centralized resources, employing techniques such as compression and sparsification for gradients or activations to eliminate insignificant values is an effective approach to reducing whole communication intensity.

For gradients, StellaTrain (Lim et al., 2024) leverages gradient sparsity to achieve a 99% compression rate, significantly reducing communication intensity. OpenDiLoCo (Jaghouar et al., 2024b) employs mixed precision training (Micikevicius et al., 2018) with FP16 quantized gradients to reduce communication. For activations, SWARM (Ryabinin et al., 2023) and Petals (Borzunov et al., 2023) use quantization method to reduce activations. Rather than compressing activation values directly, AQ-SGD (Wang et al., 2022) transmits and compresses sparser activation changes. With decentralized mixture of experts (DeMoE) structure, Learning@Home (Ryabinin and Gusev, 2020) diminishes activations through expert selection to reduce communication payload.

Coordinating the communication topology can balance payload with multiple connections, thereby reduce the intensity of a single session. Moshpit (Ryabinin et al., 2021) dynamically forms communication groups to reduce network load during all-reduce. Ravnest (Menon et al., 2024) parallelizes multiple Ring All-Reduce operations simultaneously to accommodate low bandwidth.

Additionally, optimizing parallel strategies to overlap communication with computation can also accommodate bandwidth limitations in decentralized LLM training. In this condition, PP is often employed (Athlur et al., 2022; Yuan et al., 2022; Ryabinin et al., 2023; Lim et al., 2024). We regard it as one method to reduce communication intensity as PP often leads to small-batch communications.

Discussion. One of the most significant challenges in decentralized LLM training lies in communication constraints. The effectiveness of PP in such settings highlights that developing intelligent communication protocols tailored to LLM-specific features (e.g. 3D parallelism, mixture-of-experts (MoE) and inter-layer parameter sparsity, etc.) is vital for improving the performance and scalability of LLMs in decentralized paradigms.

4.2 Heterogeneity Awareness

Since network bandwidth across regions in WANs varies significantly and computational devices differ in capacity, both network and device heterogeneity exist within decentralized LLM training.

Network Heterogeneity. Bandwidth constraints necessitate optimization of communication volume in decentralized LLM training, while network heterogeneity introduces additional challenges in communication scheduling.

To deal with network heterogeneity, BA-Combo (Jiang et al., 2020) uses a bandwidth-aware worker selection strategy, enabling efficient gradient splitting and scheduling across multiple connections. DTFM (Yuan et al., 2022) employs a communication matrix to model bandwidth differences, and it solves the hierarchical optimization problem to minimize communication cost. In contrast to static scheduling, DeDLOC (Diskin et al., 2021) dynamically adapts its strategy by switching between All-Reduce, parameter servers, and decentralized SGD based on real-time network conditions. To further optimize communication paths, NETSTOR (Li et al., 2024) employs a multi-root adaptive synchronization topology to dynamically allocate tasks based on bandwidth and balancing communication loads with auxiliary paths. Additionally, hardware characteristics can be integrated into the scheduling process. Holmes (Yang et al., 2024) automatically selects heterogeneous network interface cards (NICs, including InfiniBand, RoCE, Ethernet) for diverse distributed strategies (DP, PP and TP) across heterogeneous clusters.

Beyond scheduling, adaptive gradient compression techniques, which dynamically adjust compression levels according to bandwidth, can mitigate straggler issues within heterogeneous networks (Fan et al., 2023).

Device Heterogeneity. The varying computational capabilities of decentralized devices significantly influence the strategies for both data-parallel task allocation and model placement.

For data-parallel task allocation, DLion (Hong and Chandra, 2021) assigns different batch sizes to computational devices in micro-clouds based on their capacities in DP setting. SWARM (Ryabinin et al., 2023) adaptively merges outputs from predecessors to faster devices and distributes to multiple slower successors within a dynamic PP process. For model placement, Learning@home (Ryabinin and Gusev, 2020) employs DeMoE paradigm, distributing expert layers to different consumer-grade devices based on memory capability. ATOM (Wu et al., 2024) performs dynamic model partitioning by jointly considering memory and compute capacity. Petals (Borzunov et al., 2023) also adopts load balancing, dynamically assigning transformer blocks based on device capability for fine-tuning LLMs on heterogeneous devices. Ravnest (Menon et al., 2024) applies a genetic algorithm to group devices with similar memory and bandwidth, implementing PP within each group and proportionally partitioning model based on capacity to optimize heterogeneous device utilization.

Discussion. The effective utilization of decentralized resources is fundamentally constrained by the ability to manage system heterogeneity. Current frameworks rarely account for cross-architecture collaboration (e.g., NVIDIA GPUs with CUDA[3] and Huawei Ascend GPUs with CANN[4]), which limits their potential. Furthermore, existing heterogeneity-aware strategies mainly focus on protocol or topology optimization, overlooking critical semantic aspects of LLM training dynamics, such as layer sensitivity and gradient rank (Refael et al., 2025). Developing semantics-guided, architecture-agnostic frameworks may enhance scalability and resource efficiency in decentralized LLM training.

4.3 Fault Tolerance

Given the inherent instability of computational resources contributed by communities, node failures and communication disruptions are inevitable in decentralized LLM training. Consequently, fault tolerance becomes a critical requirement to ensure stable and efficient training processes.

A fundamental approach to fault tolerance involves periodically saving model checkpoints and migrating tasks to functioning nodes in the event of failures (Lee and Son, 2017; Athlur et al., 2022). While straightforward, this reactive method is often inefficient due to the overhead of migration and global recomputation. To mitigate the impact of single-node failures on global synchronization performance, a hierarchical synchronization strategy has been proposed (Ryabinin et al., 2021; Li et al., 2021) to synchronize locally before global synchronization, which confines recomputation to subgroups, with group size adaptable to failure rates (Diskin et al., 2021). Waiting for failure nodes recomputation is not always necessary, DiLoCo (Douillard et al., 2024) enabling asynchronous local training during communication failures, reducing reliance on global synchronization. Furthermore, if sufficient resources are available, introducing redundant computing replicas to mitigate fault-induced losses can also be a feasible approach (Lu et al., 2024a).

[3] https://developer.nvidia.com/cuda-toolkit
[4] https://www.hiascend.com/software/cann

To implement fault-tolerant mechanisms mentioned above, distributed hash tables (DHTs) have emerged as a core technology. We present how DHTs are leveraged in decentralized LLM training in Appendix C.

Discussion. Fault tolerance should balance robustness and resource efficiency, especially when employing intricate parallel strategies. For instance, in PP, node failures can lead to complete pipeline stalls. Although coarse-grained redundancy and checkpointing mitigate the impact of node failures, they often result in substantial resource inefficiencies. Optimizing this trade-off between performance and resource utilization is crucial for enabling reliable and scalable decentralized training of ultra-large LLMs, ultimately making such training more accessible and reliable.

4.4 Economics

While decentralized resources offer cost advantages compared to dedicated cluster, the volatile pricing of cloud resources, particularly GPU instances, necessitates strategic economic optimization for effective cost control and reduction. Prior work shows that 50 commodity RTX 3080 GPUs can deliver throughput comparable to four H100 GPUs, revealing a favorable trade-off between cost and performance (Tang et al., 2023).

One effective cost-saving strategy is the use of low-priority or preemptible instances, which are substantially cheaper than dedicated servers. For instance, Varuna (Athlur et al., 2022) leverages low-priority virtual machines (VMs) that cost approximately 5× less than dedicated GPU servers, without sacrificing training throughput. Similarly, (Strati et al., 2024) demonstrates that spot instances, which are 60%–90% cheaper than on-demand alternatives, can be effectively used when combined with robust fault-tolerance mechanisms.

Cross-region and multi-cloud resource selection also plays a crucial role in optimizing training economics. DeepSpotCloud (Lee and Son, 2017) monitors real-time GPU Spot pricing and selects optimal placements to maximize cost-effectiveness. Erben et al. (2024) evaluates a hybrid deployment strategy across four continents, finding that using distributed spot instances outperforms centralized DGX-2 or LambdaLabs A10 setups in terms of cost efficiency. StellaTrain (Lim et al., 2024) further explores the hybrid cloud/on-premise setting, reporting a 64.5% reduction in cloud costs through workload-aware scheduling.

Additionally, several studies propose analytical cost models to guide deployment decisions. Strati et al. (2024) develops a training cost estimator that highlights the benefits of intra-region communication for improving throughput and minimizing expenses. Phalak et al. (2024) extends this to a multi-cloud, multi-geography scenario, incorporating serverless compute and VM selection into a unified model for performance-cost optimization.

5 Organizational Decentralization

In organizational paradigms, besides the optimizations involved in community-driven decentralization, well-resourced organizations consider objectives that extend beyond individual systems to include datacenter-level enhancements, such as energy efficiency, system throughput[5] within and across datacenters. Representative related papers are compactly summarized in Figure 2.

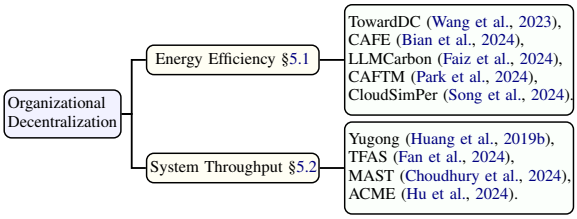

Figure 2: Taxonomy of related papers based on optimization objectives of organizational decentralization

5.1 Energy Efficiency

Training LLMs within datacenters entails substantial energy consumption, making carbon efficiency a critical concern for large organizations. For a single job, predicting the carbon footprint of current training methods for large models can help optimize training strategies, thereby reducing carbon emissions (Faiz et al., 2024). Resource selection across datacenters also plays a vital role, enabling a trade-off between model performance and environmental impact (Bian et al., 2024) At the datacenter level, implementing job scheduling strategies that are aware of carbon emissions can further decrease the overall carbon footprint (Park et al., 2024; Song et al., 2024). In addition, digital twin technologies hold significant potential for enabling real-time monitoring, control, and optimization of datacen-

[5] System throughput refers to the amount of data or tasks a training system can process per unit of time, reflecting its overall processing capacity and efficiency.

ter operations, thereby enhancing energy efficiency during LLM training (Wang et al., 2023).

5.2 System Throughput

When training large models across datacenters, it is crucial to design resource scheduling policies that account for the characteristics of training processes, such as 4D parallelism and synchronization requirements, as well as the infrastructure characteristics including low-bandwidth inter-datacenter communication (Fan et al., 2024; Hu et al., 2024). Traditional cross-datacenter schedulers often fall short, as they struggle to adapt to the dynamic and resource-intensive nature of LLM training workloads (Huang et al., 2019b). To address these challenges, MAST (Choudhury et al., 2024), a global scheduling system, effectively orchestrates the training of LLama-3 (Grattafiori et al., 2024) across 16,000 H100 GPUs, achieving both load balancing and fault tolerance across datacenters.

Discussion. Organizational decentralization is the preferred approach for large, well-resourced organizations to leverage massive computational resources for training ultra-large LLMs. While the global-scale distributed model training job scheduler MAST can coordinate tens of thousands of GPUs across multiple datacenters, the 16,000 GPUs used for training Llama-3 were confined to a single massive datacenter (Grattafiori et al., 2024). Expanding such training frameworks to span multiple data centers in the future represents a pivotal pathway for developing ultra-scale LLMs.

6 Case Study

This section examines two representative models, Llama-3 and INTELLECT-1, which we considered as examples of organizational and community-driven decentralization, respectively. Comparative configurations are listed in Table 3.

Llama-3. Llama-3, an open-source foundation model family from Meta, scales up to 405 billion parameters (Grattafiori et al., 2024). It is trained on 15 T multilingual tokens, requiring a significant amount of FLOPs for computation. To achieve this, Llama-3 utilizes 16,000 H100 GPUs with a global-scale scheduler (Choudhury et al., 2024) and 4D parallel training (as illustrated in Appendix A), achieving 38-43% Model Fractional Utilization (MFU). Storage is managed by the Tectonic distributed file system (Pan et al., 2021), offering 240 PB capacity and optimized throughput to reduce GPU idle time. For networking, Llama-3 leverages RoCE and InfiniBand within a three-layer topology (Lee et al., 2024; Gangidi et al., 2024), enhanced by load balancing and congestion control for efficient communication across 24,000 GPUs. As a representative open-source LLM, Llama-3 leverages interconnected clusters orchestrated by a global-scale scheduler, providing critical insights for organizational decentralized LLM training paradigms.

INTELLECT–1. INTELLECT-1 (Jaghouar et al., 2024a) , an open-source 10-billion-parameter LLM trained with decentralized resources. It employs hierarchical parameter aggregation and int8 quantization to minimize bandwidth usage, while VPN is integrated to ensure stability in low-bandwidth networks. For fault tolerance, INTELLECT-1 utilizes ElasticDeviceMesh for node management and phased checkpointing to minimize training interruptions, with peer-to-peer transfer enabling rapid checkpoint recovery. To optimize memory utilization, INTELLECT-1 integrates FSDP2 (Zhao et al., 2023) and CPU offloading, enhancing the system efficiency and scalability. Furthermore, Intellect-2 (Team et al., 2025), the successor model of INTELLECT-1, extends the capabilities of decentralized LLMs through decentralized post-training with reinforce learning. This community-driven decentralized paradigm democratizes AI model development, preventing monopolization and fostering open-source innovation with decentralized resources.

Model	**Llama-3**	**INTELLECT-1**
Parameter Scale	405 B	10 B
Resource Scale	16K H100 GPUs	112 H100 GPUs
Resource Distribution	Across pods, each with 3072 GPUs, in one datacenter	Across servers from 5 countries and 3 continents
Parallel Mechanism	4D parallelism	Hybrid DiLoCo-FSDP2
Training Time	54 days	42 days
Effective Training Time	$\geq 90\%$	83%
Processed Tokens	15 T	1 T

Table 3: Comparison of training configurations of Llama-3 and INTELLECT-1

7 Summary and Future Directions

7.1 Summary

In this survey, we classify decentralized LLM training into two paradigms based on resource utilization: community-driven decentralization and organizational decentralization. By analyzing the characteristics of decentralized resources, review-

ing relevant optimization methods from the literature, and examining two model cases, representing the applications of community-driven and organizational decentralization respectively, we provide a systematic overview of the current development landscape in decentralized LLM training.

7.2 Future Directions

We outline potential research directions spanning resource organization, model architecture, and training paradigms.

Scaling Law of Decentralized LLM Training. The scaling law for centralized LLM training primarily focus on computational power, data volume, and model size to optimize training strategies (Hoffmann et al., 2022; Grattafiori et al., 2024). In decentralized paradigms, however, the interplay between computational and network resources becomes significantly more complex, necessitating their inclusion in the scaling laws. For a given topology of computational resources with constrained bandwidth, there may exist a practical limit on scaling efficiency. Beyond this point, further increasing model size or local resources does not yield proportional improvements in global performance. Exploring the scaling law for decentralized LLM training is essential for effectively coordinating global decentralized resources and enhancing their utilization efficiency.

Decentralized Resources Governance. Current efforts in decentralized LLM training primarily focus on resource utilization and management for individual model training processes. As the community expands, effective governance of decentralized resources will become crucial for sustaining the development of decentralized LLM training. Challenges like communication bottlenecks, resource heterogeneity, and instability may stem from the resources themselves or arise from inefficient coordination at the resource abstraction layer of decentralized systems. Future research could prioritize developing mechanisms to optimize pricing and scheduling of decentralized resources in multi-tenant environments, thereby facilitating the proliferation of open-source models powered by decentralized infrastructure.

Training MLLM with Decentralized Resources. Decentralized training of multi-modal large language models (MLLMs) presents both significant opportunities and unique challenges compared to conventional LLM training. The inherent complexity of training MLLMs stems from different data types (e.g. text, images, audio), each requiring specialized processing modules. These heterogeneous modules complicate communication scheduling and resource allocation for distributed training (Huang et al., 2024). When training MLLMs with decentralized resources, these heterogeneities can be further exacerbated. Addressing these challenges through future research could unlock the full potential of decentralized multi-modal data, enabling scalable and efficient utilization of decentralized resources and significantly advancing the development of multi-modal AI systems.

Post-training with Decentralized Resources. Current research on utilizing decentralized resources for LLM training predominantly focuses on the pre-training stage. However, the post-training phase incorporating reinforcement learning (RL) is crucial for enhancing the reasoning capabilities of LLMs (DeepSeek-AI et al., 2025a). One distinctive characteristic of this phase lies in the intensive rollout generation (Team et al., 2025), which is computation-intensive inference process without backward. Therefore, during RL-based post-training, decentralized methods should also optimize LLM inference serving on weaker nodes (e.g., serving LLMs with frameworks like vLLM (Kwon et al., 2023) or SGLang (Zheng et al., 2024) on consumer GPUs, which is non-trivial), while pre-training can only consider training period optimization. Future research could focus on the joint optimization of inference rollout and backward during LLM post-training with decentralized resources, which can expand the capability boundaries of decentralized LLMs, thereby enhancing accessibility to LLM services for broader communities.

Limitations

This survey focuses on decentralized LLM training from a resource-driven perspective, but several limitations should be noted. First, important issues such as data distribution and privacy protection are not covered, as they diverge from the scope of our survey. Second, given the rapid advancements in LLM development, some recent developments may have been inadvertently overlooked despite our efforts to include more relevant research.

This paper is a survey and does not involve the development of new artifacts or data collection. Therefore, it poses no direct potential risks.

Acknowledgment

This work is supported in part by National Key Research and Development Project of China (Grant No. 2023YFF0905502), National Natural Science Foundation of China(Grant No. 92467204 and 62472249), Shenzhen Science and Technology Program (Grant No. JCYJ20220818101014030 and KJZD20240903102300001) and Natural Science Foundation of Top Talent of SZTU(Grant No. GDRC202413). We thank the anonymous reviewers for their efforts, which have helped improve the quality of this paper.

References

Sanjith Athlur, Nitika Saran, Muthian Sivathanu, Ramachandran Ramjee, and Nipun Kwatra. 2022. Varuna: scalable, low-cost training of massive deep learning models. In *Proceedings of the Seventeenth European Conference on Computer Systems*, EuroSys '22, page 472–487, New York, NY, USA. Association for Computing Machinery.

Azurespeed. 2024. Azure speed test - latency. https://www.azurespeed.com/Azure/Latency.

Jieming Bian, Lei Wang, Shaolei Ren, and Jie Xu. 2024. Cafe: Carbon-aware federated learning in geographically distributed data centers. In *Proceedings of the 15th ACM International Conference on Future and Sustainable Energy Systems*, e-Energy '24, page 347–360, New York, NY, USA. Association for Computing Machinery.

Alexander Borzunov, Max Ryabinin, Artem Chumachenko, Dmitry Baranchuk, Tim Dettmers, Younes Belkada, Pavel Samygin, and Colin Raffel. 2023. Distributed inference and fine-tuning of large language models over the internet. In *Proceedings of the 37th International Conference on Neural Information Processing Systems*, NIPS '23, Red Hook, NY, USA. Curran Associates Inc.

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel M. Ziegler, Jeffrey Wu, Clemens Winter, and 12 others. 2020. Language models are few-shot learners. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Arnab Choudhury, Yang Wang, Tuomas Pelkonen, Kutta Srinivasan, Abha Jain, Shenghao Lin, Delia David, Siavash Soleimanifard, Michael Chen, Abhishek Yadav, Ritesh Tijoriwala, Denis Samoylov, and Chunqiang Tang. 2024. Mast: global scheduling of ml training across geo-distributed datacenters at hyperscale. In *Proceedings of the 18th USENIX Conference on Operating Systems Design and Implementation*, OSDI'24, USA. USENIX Association.

DeepSeek-AI, Daya Guo, Dejian Yang, Haowei Zhang, Junxiao Song, Ruoyu Zhang, Runxin Xu, Qihao Zhu, Shirong Ma, Peiyi Wang, Xiao Bi, Xiaokang Zhang, Xingkai Yu, Yu Wu, Z. F. Wu, Zhibin Gou, Zhihong Shao, Zhuoshu Li, Ziyi Gao, and 181 others. 2025a. Deepseek-r1: Incentivizing reasoning capability in llms via reinforcement learning. *Preprint*, arXiv:2501.12948.

DeepSeek-AI, Aixin Liu, Bei Feng, Bing Xue, Bingxuan Wang, Bochao Wu, Chengda Lu, Chenggang Zhao, Chengqi Deng, Chenyu Zhang, Chong Ruan, Damai Dai, Daya Guo, Dejian Yang, Deli Chen, Dongjie Ji, Erhang Li, Fangyun Lin, Fucong Dai, and 181 others. 2025b. Deepseek-v3 technical report. *Preprint*, arXiv:2412.19437.

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 Wolff, and Gennady Pekhimenko. 2021. Distributed deep learning in open collaborations. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, Red Hook, NY, USA. Curran Associates Inc.

Arthur Douillard, Qixuan Feng, Andrei A. Rusu, Rachita Chhaparia, Yani Donchev, Adhiguna Kuncoro, Marc'Aurelio Ranzato, Arthur Szlam, and Jiajun Shen. 2024. Diloco: Distributed low-communication training of language models. *Preprint*, arXiv:2311.08105.

Jiangfei Duan, Shuo Zhang, Zerui Wang, Lijuan Jiang, Wenwen Qu, Qinghao Hu, Guoteng Wang, Qizhen Weng, Hang Yan, Xingcheng Zhang, Xipeng Qiu, Dahua Lin, Yonggang Wen, Xin Jin, Tianwei Zhang, and Peng Sun. 2024. Efficient training of large language models on distributed infrastructures: A survey. *Preprint*, arXiv:2407.20018.

Alexander Erben, Ruben Mayer, and Hans-Arno Jacobsen. 2024. How can we train deep learning models across clouds and continents? an experimental study. *Proc. VLDB Endow.*, 17(6):1214–1226.

Ahmad Faiz, Sotaro Kaneda, Ruhan Wang, Rita Chukwunyere Osi, Prateek Sharma, Fan Chen, and Lei Jiang. 2024. LLMCarbon: Modeling the end-to-end carbon footprint of large language models. In *The Twelfth International Conference on Learning Representations*, ICLR'24, Vienna, Austria.

Chenyu Fan, Xiaoning Zhang, Yangming Zhao, Yutao Liu, and Shui Yu. 2023. Self-adaptive gradient quantization for geo-distributed machine learning over heterogeneous and dynamic networks. *IEEE Transactions on Cloud Computing*, 11(4):3483–3496.

Lang Fan, Xiaoning Zhang, Yangming Zhao, Keshav Sood, and Shui Yu. 2024. Online training flow scheduling for geo-distributed machine learning jobs over heterogeneous and dynamic networks. *IEEE Transactions on Cognitive Communications and Networking*, 10(1):277–291.

Adithya Gangidi, Rui Miao, Shengbao Zheng, Sai Jayesh Bondu, Guilherme Goes, Hany Morsy, Rohit Puri, Mohammad Riftadi, Ashmitha Jeevaraj Shetty, Jingyi Yang, Shuqiang Zhang, Mikel Jimenez Fernandez, Shashidhar Gandham, and Hongyi Zeng. 2024. Rdma over ethernet for distributed training at meta scale. In *Proceedings of the ACM SIGCOMM 2024 Conference*, ACM SIGCOMM '24, page 57–70, New York, NY, USA. Association for Computing Machinery.

Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, and 542 others. 2024. The llama 3 herd of models. *Preprint*, arXiv:2407.21783.

István Hegedűs, Gábor Danner, and Márk Jelasity. 2019. Gossip learning as a decentralized alternative to federated learning. In *Distributed Applications and Interoperable Systems: 19th IFIP WG 6.1 International Conference, DAIS 2019, Held as Part of the 14th International Federated Conference on Distributed Computing Techniques, DisCoTec 2019, Kongens Lyngby, Denmark, June 17–21, 2019, Proceedings 19*, pages 74–90. Springer.

Jordan Hoffmann, Sebastian Borgeaud, Arthur Mensch, Elena Buchatskaya, Trevor Cai, Eliza Rutherford, Diego de Las Casas, Lisa Anne Hendricks, Johannes Welbl, Aidan Clark, Tom Hennigan, Eric Noland, Katie Millican, George van den Driessche, Bogdan Damoc, Aurelia Guy, Simon Osindero, Karen Simonyan, Erich Elsen, and 3 others. 2022. Training compute-optimal large language models. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Rankyung Hong and Abhishek Chandra. 2021. Dlion: Decentralized distributed deep learning in micro-clouds. In *Proceedings of the 30th International Symposium on High-Performance Parallel and Distributed Computing*, HPDC '21, page 227–238, New York, NY, USA. Association for Computing Machinery.

Kevin Hsieh, Aaron Harlap, Nandita Vijaykumar, Dimitris Konomis, Gregory R. Ganger, Phillip B. Gibbons, and Onur Mutlu. 2017. Gaia: geo-distributed machine learning approaching lan speeds. In *Proceedings of the 14th USENIX Conference on Networked Systems Design and Implementation*, NSDI'17, page 629–647, USA. USENIX Association.

Qinghao Hu, Zhisheng Ye, Zerui Wang, Guoteng Wang, Meng Zhang, Qiaoling Chen, Peng Sun, Dahua Lin, Xiaolin Wang, Yingwei Luo, Yonggang Wen, and Tianwei Zhang. 2024. Characterization of large language model development in the datacenter. In *Proceedings of the 21st USENIX Symposium on Networked Systems Design and Implementation*, NSDI'24, USA. USENIX Association.

Jun Huang, Zhen Zhang, Shuai Zheng, Feng Qin, and Yida Wang. 2024. Distmm: accelerating distributed multimodal model training. In *Proceedings of the 21st USENIX Symposium on Networked Systems Design and Implementation*, NSDI'24, USA. USENIX Association.

Yanping Huang, Youlong Cheng, Ankur Bapna, Orhan Firat, Mia Xu Chen, Dehao Chen, HyoukJoong Lee, Jiquan Ngiam, Quoc V. Le, Yonghui Wu, and Zhifeng Chen. 2019a. Gpipe: efficient training of giant neural networks using pipeline parallelism. In *Proceedings of the 33rd International Conference on Neural Information Processing Systems*, Red Hook, NY, USA. Curran Associates Inc.

Yuzhen Huang, Yingjie Shi, Zheng Zhong, Yihui Feng, James Cheng, Jiwei Li, Haochuan Fan, Chao Li, Tao Guan, and Jingren Zhou. 2019b. Yugong: geo-distributed data and job placement at scale. *Proc. VLDB Endow.*, 12(12):2155–2169.

Sam Ade Jacobs, Masahiro Tanaka, Chengming Zhang, Minjia Zhang, Reza Yazdani Aminabadi, Shuaiwen Leon Song, Samyam Rajbhandari, and Yuxiong He. 2024. System Optimizations for Enabling Training of Extreme Long Sequence Transformer Models. In *IEEE International Parallel and Distributed Processing Symposium (IPDPS) - Workshops*, pages 1206–1208.

Sami Jaghouar, Jack Min Ong, Manveer Basra, Fares Obeid, Jannik Straube, Michael Keiblinger, Elie Bakouch, Lucas Atkins, Maziyar Panahi, Charles Goddard, Max Ryabinin, and Johannes Hagemann. 2024a. Intellect-1 technical report. *Preprint*, arXiv:2412.01152.

Sami Jaghouar, Jack Min Ong, and Johannes Hagemann. 2024b. Opendiloco: An open-source framework for globally distributed low-communication training. *Preprint*, arXiv:2407.07852.

Jingyan Jiang, Liang Hu, Chenghao Hu, Jiate Liu, and Zhi Wang. 2020. Bacombo—bandwidth-aware decentralized federated learning. *Electronics*, 9(3):440.

Ziheng Jiang, Haibin Lin, Yinmin Zhong, Qi Huang, Yangrui Chen, Zhi Zhang, Yanghua Peng, Xiang Li, Cong Xie, Shibiao Nong, Yulu Jia, Sun He, Hongmin Chen, Zhihao Bai, Qi Hou, Shipeng Yan, Ding Zhou, Yiyao Sheng, Zhuo Jiang, and 13 others. 2024. Megascale: scaling large language model training to more than 10,000 gpus. In *Proceedings of the 21st USENIX Symposium on Networked Systems Design and Implementation*, NSDI'24, USA. USENIX Association.

Qazi Waqas Khan, Anam Nawaz Khan, Atif Rizwan, Rashid Ahmad, Salabat Khan, and Do-Hyeun Kim. 2023. Decentralized machine learning training: a survey on synchronization, consolidation, and topologies. *IEEE Access*, 11:68031–68050.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient Memory Management for Large Language Model Serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*, pages 611–626. ACM.

Anusha Lalitha, Osman Cihan Kilinc, Tara Javidi, and Farinaz Koushanfar. 2019. Peer-to-peer federated learning on graphs. *Preprint*, arXiv:1901.11173.

Kevin Lee, Adi Gangidi, and Mathew Oldham. 2024. Building meta's genai infrastructure.

Kyungyong Lee and Myungjun Son. 2017. Deepspotcloud: Leveraging cross-region gpu spot instances for deep learning. In *2017 IEEE 10th International Conference on Cloud Computing (CLOUD)*, pages 98–105.

Shen Li, Yanli Zhao, Rohan Varma, Omkar Salpekar, Pieter Noordhuis, Teng Li, Adam Paszke, Jeff Smith, Brian Vaughan, Pritam Damania, and Soumith Chintala. 2020. Pytorch distributed: experiences on accelerating data parallel training. *Proc. VLDB Endow.*, 13(12):3005–3018.

Shigang Li, Tal Ben-Nun, Giorgi Nadiradze, Salvatore Di Girolamo, Nikoli Dryden, Dan Alistarh, and Torsten Hoefler. 2021. Breaking (global) barriers in parallel stochastic optimization with wait-avoiding group averaging. *IEEE Transactions on Parallel and Distributed Systems*, 32(7):1725–1739.

Zonghang Li, Wenjiao Feng, Weibo Cai, Hongfang Yu, Long Luo, Gang Sun, Hongyang Du, and Dusit Niyato. 2024. Accelerating geo-distributed machine learning with network-aware adaptive tree and auxiliary route. *IEEE/ACM Trans. Netw.*, 32(5):4238–4253.

Hwijoon Lim, Juncheol Ye, Sangeetha Abdu Jyothi, and Dongsu Han. 2024. Accelerating model training in multi-cluster environments with consumer-grade gpus. In *Proceedings of the ACM SIGCOMM 2024 Conference*, ACM SIGCOMM '24, page 707–720, New York, NY, USA. Association for Computing Machinery.

Ruiyun Liu, Weiqiang Sun, and Weisheng Hu. 2023. Placement of high availability geo-distributed data centers in emerging economies. *IEEE Transactions on Cloud Computing*, 11(3):3274–3288.

Lin Lu, Chenxi Dai, Wangcheng Tao, Binhang Yuan, Yanan Sun, and Pan Zhou. 2024a. Position: Exploring the robustness of pipeline-parallelism-based decentralized training. In *Proceedings of the 41st International Conference on Machine Learning*, volume 235 of *Proceedings of Machine Learning Research*, pages 32978–32989. PMLR.

Rongwei Lu, Jingyan Jiang, Chunyang Li, Haotian Dong, Xingguang Wei, Delin Cai, and Zhi Wang. 2025a. Deco-sgd: Joint optimization of delay staleness and gradient compression ratio for distributed sgd. *ArXiv*, abs/2507.17346.

Rongwei Lu, Yutong Jiang, Jinrui Zhang, Chunyang Li, Yifei Zhu, Bin Chen, and Zhi Wang. 2025b. γ-fedht: Stepsize-aware hard-threshold gradient compression in federated learning. *IEEE INFOCOM 2025 - IEEE Conference on Computer Communications*, pages 1–10.

Rongwei Lu, Yutong Jiang, and 1 others. 2024b. Data-aware gradient compression for fl in communication-constrained mobile computing. *IEEE Transactions on Mobile Computing*, pages 1–14.

Brendan McMahan, Eider Moore, and 1 others. 2017. Communication-efficient learning of deep networks from decentralized data. In *Artificial intelligence and statistics*, pages 1273–1282. PMLR.

Anirudh Rajiv Menon, Unnikrishnan Menon, and Kailash Ahirwar. 2024. Ravnest: Decentralized asynchronous training on heterogeneous devices. *Preprint*, arXiv:2401.01728.

Haibo Mi, Kele Xu, Dawei Feng, Huaimin Wang, Yiming Zhang, Zibin Zheng, Chuan Chen, and Xu Lan. 2020. Collaborative deep learning across multiple data centers. *Science China Information Sciences*, 63(8):182102.

Paulius Micikevicius, Sharan Narang, and 1 others. 2018. Mixed precision training. In *International Conference on Learning Representations*, ICLR' 18, Vancouver, Canada.

Deepak Narayanan, Mohammad Shoeybi, and 1 others. 2021. Efficient large-scale language model training on gpu clusters using megatron-lm. *SC21: International Conference for High Performance Computing, Networking, Storage and Analysis*, pages 1–14.

NVIDIA. 2023. Megatron core: Context parallelism.

Satadru Pan, Theano Stavrinos, Yunqiao Zhang, Atul Sikaria, Pavel Zakharov, Abhinav Sharma, Shiva Shankar P, Mike Shuey, Richard Wareing, Monika Gangapuram, Guanglei Cao, Christian Preseau, Pratap Singh, Kestutis Patiejunas, JR Tipton, Ethan Katz-Bassett, and Wyatt Lloyd. 2021. Facebook's tectonic filesystem: Efficiency from exascale. In *19th USENIX Conference on File and Storage Technologies (FAST 21)*, pages 217–231. USENIX Association.

Jeonghyeon Park, Daero Kim, and 1 others. 2024. Carbon-aware and fault-tolerant migration of deep learning workloads in the geo-distributed cloud. *2024 IEEE 17th International Conference on Cloud Computing (CLOUD)*, pages 494–501.

Chetan Phalak, Dheeraj Chahal, Manju Ramesh, and Rekha Singhal. 2024. Towards geo-distributed training of ml models in a multi-cloud environment. In *Companion of the 15th ACM/SPEC International Conference on Performance Engineering*, ICPE '24 Companion, page 211–217, New York, NY, USA. Association for Computing Machinery.

Kun Qian, Yongqing Xi, Jiamin Cao, Jiaqi Gao, Yichi Xu, Yu Guan, Binzhang Fu, Xuemei Shi, Fangbo Zhu, Rui Miao, Chao Wang, Peng Wang, Pengcheng Zhang, Xianlong Zeng, Eddie Ruan, Zhiping Yao, Ennan Zhai, and Dennis Cai. 2024. Alibaba hpn: A data center network for large language model training. In *Proceedings of the ACM SIGCOMM 2024 Conference*, ACM SIGCOMM '24, page 691–706, New York, NY, USA. Association for Computing Machinery.

Yehonathan Refael, Jonathan Svirsky, Boris Shustin, Wasim Huleihel, and Ofir Lindenbaum. 2025. Adarankgrad: Adaptive gradient-rank and moments for memory-efficient llms training and fine-tuning. In *The Twelfth International Conference on Learning Representations*, ICLR'25, Singapore.

Max Ryabinin, Tim Dettmers, Michael Diskin, and Alexander Borzunov. 2023. Swarm parallelism: training large models can be surprisingly communication-efficient. In *Proceedings of the 40th International Conference on Machine Learning*, ICML'23. JMLR.org.

Max Ryabinin, Eduard Gorbunov, Vsevolod Plokhotnyuk, and Gennady Pekhimenko. 2021. Moshpit sgd: communication-efficient decentralized training on heterogeneous unreliable devices. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, Red Hook, NY, USA. Curran Associates Inc.

Max Ryabinin and Anton Gusev. 2020. Towards crowdsourced training of large neural networks using decentralized mixture-of-experts. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Li Shen, Yan Sun, Zhiyuan Yu, Liang Ding, Xinmei Tian, and Dacheng Tao. 2024. On efficient training of large-scale deep learning models. *ACM Comput. Surv.*, 57(3).

Jie Song, Peimeng Zhu, Yanfeng Zhang, and Ge Yu. 2024. Cloudsimper: Simulating geo-distributed datacenters powered by renewable energy mix. *IEEE Transactions on Parallel and Distributed Systems*, 35(4):531–547.

Foteini Strati, Paul Elvinger, Tolga Kerimoglu, and Ana Klimovic. 2024. Ml training with cloud gpu shortages: Is cross-region the answer? In *Proceedings of the 4th Workshop on Machine Learning and Systems*, EuroMLSys '24, page 107–116, New York, NY, USA. Association for Computing Machinery.

Zhenheng Tang, Yuxin Wang, Xin He, Longteng Zhang, Xinglin Pan, Qiang Wang, Rongfei Zeng, Kaiyong Zhao, Shaohuai Shi, Bingsheng He, and Xiaowen Chu. 2023. Fusionai: Decentralized training and deploying llms with massive consumer-level gpus. *Preprint*, arXiv:2309.01172.

Prime Intellect Team, Sami Jaghouar, Justus Mattern, Jack Min Ong, Jannik Straube, Manveer Basra, Aaron Pazdera, Kushal Thaman, Matthew Di Ferrante, Felix Gabriel, Fares Obeid, Kemal Erdem, Michael Keiblinger, and Johannes Hagemann. 2025. Intellect-2: A reasoning model trained through globally decentralized reinforcement learning. *Preprint*, arXiv:2505.07291.

Surat Teerapittayanon, Bradley McDanel, and H.T. Kung. 2017. Distributed deep neural networks over the cloud, the edge and end devices. In *2017 IEEE 37th International Conference on Distributed Computing Systems (ICDCS)*, pages 328–339.

Jue Wang, Binhang Yuan, Luka Rimanic, Yongjun He, Tri Dao, Beidi Chen, Christopher Ré, and Ce Zhang. 2022. Fine-tuning language models over slow networks using activation quantization with guarantees. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Ruihang Wang, Deneng Xia, and 1 others. 2023. Toward data center digital twins via knowledge-based model calibration and reduction. *ACM Transactions on Modeling and Computer Simulation*, 33:1 – 24.

Stefanie Warnat-Herresthal, Hartmut Schultze, Krishnaprasad Lingadahalli Shastry, Sathyanarayanan Manamohan, Saikat Mukherjee, Vishesh Garg, Ravi Sarveswara, Kristian Händler, Peter Pickkers, N Ahmad Aziz, and 1 others. 2021. Swarm learning for decentralized and confidential clinical machine learning. *Nature*, 594(7862):265–270.

Herbert Woisetschläger, Alexander Erben, Shiqiang Wang, Ruben Mayer, and Hans-Arno Jacobsen. 2024. A survey on efficient federated learning methods for foundation model training. In *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence*, IJCAI '24.

Xiaofeng Wu, Jia Rao, and Wei Chen. 2024. Atom: Asynchronous training of massive models for deep learning in a decentralized environment. *Preprint*, arXiv:2403.10504.

Yang Xiang, Zhihua Wu, Weibao Gong, Siyu Ding, Xianjie Mo, Yuang Liu, Shuohuan Wang, Peng Liu, Yongshuai Hou, Long Li, Bin Wang, Shaohuai Shi, Yaqian Han, Yue Yu, Ge Li, Yu Sun, Yanjun Ma, and Dianhai Yu. 2022. Nebula-i: A general framework for collaboratively training deep learning models on low-bandwidth cloud clusters. *Preprint*, arXiv:2205.09470.

Hong Xing, Osvaldo Simeone, and Suzhi Bi. 2021. Federated learning over wireless device-to-device

networks: Algorithms and convergence analysis. *IEEE Journal on Selected Areas in Communications*, 39(12):3723–3741.

Fei Yang, Shuang Peng, Ning Sun, Fangyu Wang, Yuanyuan Wang, Fu Wu, Jiezhong Qiu, and Aimin Pan. 2024. Holmes: Towards distributed training across clusters with heterogeneous nic environment. In *Proceedings of the 53rd International Conference on Parallel Processing*, ICPP '24, page 514–523, New York, NY, USA. Association for Computing Machinery.

Binhang Yuan, Yongjun He, Jared Quincy Davis, Tianyi Zhang, Tri Dao, Beidi Chen, Percy Liang, Christopher Re, and Ce Zhang. 2022. Decentralized training of foundation models in heterogeneous environments. In *Proceedings of the 36th International Conference on Neural Information Processing Systems*, NIPS '22, Red Hook, NY, USA. Curran Associates Inc.

Yanli Zhao, Andrew Gu, Rohan Varma, Liang Luo, Chien-Chin Huang, Min Xu, Less Wright, Hamid Shojanazeri, Myle Ott, Sam Shleifer, Alban Desmaison, Can Balioglu, Pritam Damania, Bernard Nguyen, Geeta Chauhan, Yuchen Hao, Ajit Mathews, and Shen Li. 2023. Pytorch fsdp: Experiences on scaling fully sharded data parallel. *Proc. VLDB Endow.*, 16(12):3848–3860.

Lianmin Zheng, Liangsheng Yin, Zhiqiang Xie, Chuyue Sun, Jeff Huang, Cody Hao Yu, Shiyi Cao, Christos Kozyrakis, Ion Stoica, Joseph E. Gonzalez, Clark Barrett, and Ying Sheng. 2024. Sglang: Efficient Execution of Structured Language Model Programs. In *The 38th Annual Conference on Neural Information Processing Systems*.

Ligeng Zhu, Hongzhou Lin, Yao Lu, Yujun Lin, and Song Han. 2021. Delayed gradient averaging: tolerate the communication latency in federated learning. In *Proceedings of the 35th International Conference on Neural Information Processing Systems*, NIPS '21, Red Hook, NY, USA. Curran Associates Inc.

A Parallel Strategies in LLM Training

Training LLMs demands substantial computational resources and advanced parallel strategies to achieve efficiency. To address the challenges posed by the massive scale of LLMs, researchers have developed multi-dimensional parallel strategies, including Data Parallelism (DP), Tensor Parallelism (TP), Pipeline Parallelism (PP), and Context Parallelism (CP). Among these, TP and PP can be regarded as specialized forms of Model Parallelism (MP). Context Parallelism (CP) (NVIDIA, 2023; Jacobs et al., 2024) has emerged as a complementary strategy, which operates at the token level by slicing input sequences across devices, enabling scalable and efficient training of long-context LLMs. DP and PP strategies have been implemented within decentralized LLM training. When these techniques are strategically combined during the training of Llama-3 (Grattafiori et al., 2024), they collectively enhance throughput, reduce memory footprint, and optimize resource utilization. A detailed comparison of these four parallelism strategies is presented in Table 4.

B Literature Summary

More comprehensive studies associated with community-driven decentralization are presented in Figure 3. While this survey encompasses a broad spectrum of research, our main text primarily focuses on works that specifically target LLMs, ensuring a more in-depth and focused analysis.

C Distributed Hash Tables (DHTs)

DHTs are a class of decentralized storage systems designed to provide scalable, fault-tolerant, and efficient key-value lookups across a large set of networked nodes. Unlike traditional centralized hash tables, where a single server manages all mappings between keys and values, DHTs distribute this responsibility across multiple peers, each responsible for a subset of the key space. These features make DHTs an enabling infrastructure for robust and elastic decentralized LLM training systems, especially under environments with high node failure ratios and heterogeneous conditions.

First, DHTs can facilitate recovery from failures by storing training states (Erben et al., 2024). In Learning@home (Ryabinin and Gusev, 2020), expert checkpoints are stored in a DHT, allowing newly joined nodes to retrieve the latest state of failed ones and resume training seamlessly. Similarly, ATOM (Wu et al., 2024) leverages DHTs for asynchronous training, enabling task reallocation and training states recover.

Second, DHTs can act as metadata stores to coordinate task redistribution and enable resilient system reconfiguration. For instance, Petals (Borzunov et al., 2023) uses DHTs to manage model shard placement, allowing the system to rebalance and recover from failures during collaborative fine-tuning. SWARM (Ryabinin et al., 2023) extends this by integrating stochastic pipeline rewiring, allowing the PP process to filter failed nodes and redistribute workloads by iteration. FusionAI (Tang et al., 2023) adopts a similar

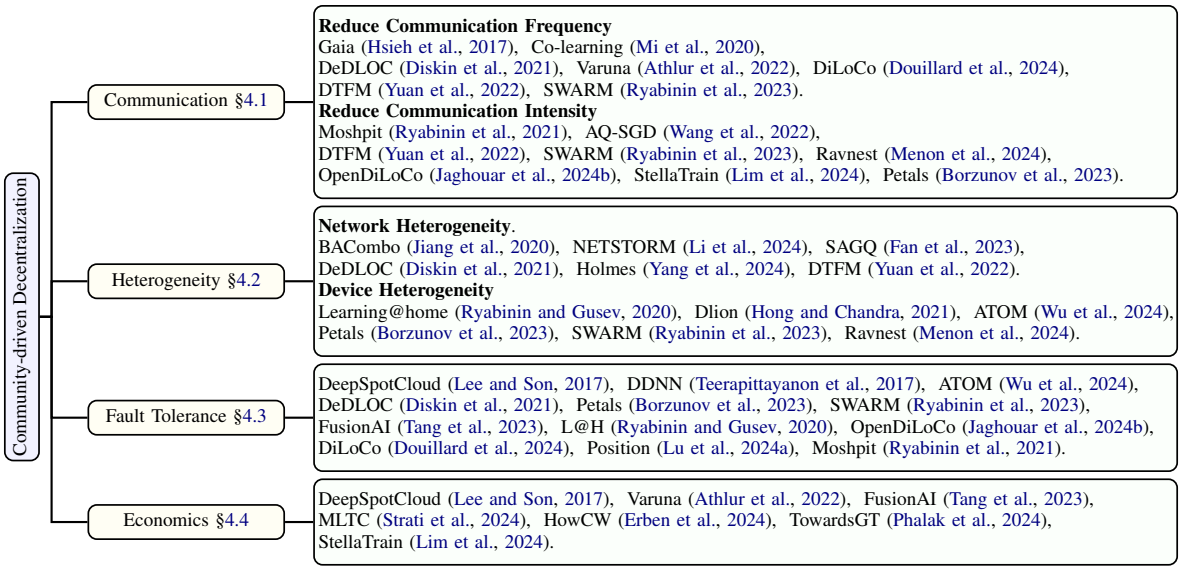

Figure 3: Taxonomy of related papers based on optimization objectives of community-driven decentralization.

Parallel Strategy	**Data Parallelism**	**Pipeline Parallelism**	**Tensor Parallelism**	**Context Parallelism**
Parallel Granularity	Data batches	Model stages and data batches	Intra-layer tensor slices	Partitioned sequences
Model Partition	Full model	Model block partitioned by stage	Model block partitioned by tensor slice	Full model
Communication	All-reduce full gradients	Inter-stage activations and gradients	All-reduce/All-gather intra-layer states	All-to-all attention KV tensors
Memory Usage	Full model duplication	Sharded model and activations	Sharded model and activations	Sharded KV cache and activations
Scalability	Good for large data batches	Good for inter-node communication	Good for intra-node communication	Good for long sequence processing
Example	PyTorch DDP (Li et al., 2020)	GPipe (Huang et al., 2019a)	Megatron-LM (Narayanan et al., 2021)	DeepSpeed-Ulysses (Jacobs et al., 2024)

Table 4: Comparison of four primary types of parallelism in LLM training.

design by combining metadata with an agent to handle task reassignment and node recovery.

Additionally, DHTs enable robust coordination of decentralized communication and update mechanisms, even under high node volatility. Moshpit (Ryabinin et al., 2021) use DHTs to dynamically form groups for gradient averaging, ensuring that partial updates are aggregated reliably despite frequent node failures. These approaches collectively highlight that DHTs can enhance the fault tolerance during decentralized LLM training.