Unifying Sparsity and Discreteness: One-Shot Pruning for Quantized LLMs via Discrete Optimization

Haozhen Zhang[1] **Hanyuan Zheng**[1] **Teng Hou**[1] **Zhaogeng Liu**[1]
Yi Chang[1,2,3*] **Bin Gu**[1*]
[1]School of Artificial Intelligence, Jilin University, China
[2]International Center of Future Science, Jilin University, China
[3]Engineering Research Center of Knowledge-Driven Human-Machine Intelligence, MOE, China
{haozhen23, hyzheng25, houteng26}@mails.jlu.edu.cn
{liuzg25, yichang, gubin}@jlu.edu.cn

Abstract

Pruning and quantization are two dominant techniques that effectively address the computational and storage burdens of large language model (LLM) inference on edge devices. Recently, one-shot pruning has gained particular attention for its ability to identify weight supports via optimization without retraining. However, integrating such methods seamlessly with quantization for further model lightweighting is non-trivial. The discreteness of quantization shatters their required variable continuity, inevitably collapsing this integration into a decoupled pruning-quantization pipeline with inherently suboptimal outcomes. To tackle this problem, we propose Quantization-aware One-shot Pruning (QOP), which directly optimizes the quantized weights under sparsity and discreteness constraints, thereby explicitly capturing the impact of quantization on pruning within its objective. Specifically, QOP generalizes the alternating direction method of multipliers to sparse-constrained discrete optimization, enabling the identification of the high-quality support and the update of quantized weights. Theoretically, via the well-conditioned approximation obtained by a slight perturbation, we establish convergence to a stationary feasible point and provide the convergence rate. Extensive experiments across various LLMs, sparsity levels, and quantization settings demonstrate that QOP consistently outperforms existing baselines in terms of both average accuracy and perplexity under weight quantization.

1 Introduction

Large language models (LLMs) [21, 51, 41, 47] have achieved impressive performance across various natural language processing (NLP) tasks. Yet, these gains are accompanied by increased model scale and substantial inference computation. The heavy scale of LLMs poses substantial demands on computational resources and severely hinders their deployment on resource-constrained devices [4, 53, 10, 12]. To mitigate this issue, model quantization [45, 38, 29] reduces numerical precision to low-bit representations, enabling significant efficiency improvements for LLMs. This has positioned quantized LLMs as a critical pathway toward reducing the compute and memory footprint.

On the other hand, in parallel with quantization, network pruning [25, 19, 17] is widely regarded as another mainstream lightweighting technique for LLMs. It aims to eliminate redundant parameters to reduce model size, providing a promising solution for alleviating the computational and storage burdens of LLMs. Conventional pruning methods [30, 28, 2] typically require expensive retraining on the whole network to recover performance after pruning. Although such methods have proven

*Corresponding Authors.

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

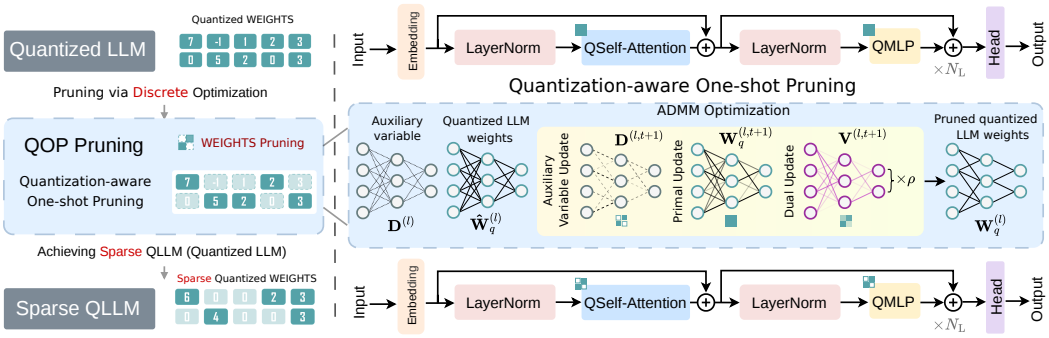


Figure 1: Overview of the proposed Quantization-aware One-shot Pruning (QOP) framework. $\mathbf{W}_q$ denotes the quantized weights, $\mathbf{D}$ represents an auxiliary variable that satisfies the sparse-constrained discrete condition, $\mathbf{V}$ denotes the dual variable, and N_{L} denotes the number of transformer layers.

effective for smaller-scale models, this resource-intensive pruning paradigm renders them impractical for LLMs. Therefore, one-shot pruning [11, 39, 12] that can be completed in a single forward pass of LLM without requiring any retraining has become increasingly prominent. Concretely, these methods decompose LLM pruning into layer-wise pruning subproblems and solve them using only a small amount of calibration data drawn from a single dataset. Building on this paradigm, recent advances [32, 6, 7, 43] focus on leveraging optimization techniques to identify the support of weights and update weights, thereby minimizing the reconstruction error of each subproblem.

Nevertheless, seamlessly integrating such optimization-driven methods with quantized LLMs to further enhance model lightweighting represents a non-trivial endeavor. These methods [32, 43] are developed for full-precision weights, i.e., such optimization algorithms that assume continuity are unable to handle the inherent discreteness of quantization. As a result, when combined with quantization, these methods inevitably result in a pruning–quantization pipeline [27, 15]. However, this decoupled execution overlooks the impact of quantization on pruning. Specifically, it causes conventional one-shot pruning to operate in a quantization-agnostic isolation and fail to capture the quantization-induced shift in weight importance, resulting in the suboptimal pruning decision.

Departing from such a straightforward pruning–quantization pipeline, we introduce a paradigm shift toward quantization-aware pruning. This approach fundamentally reformulates the pruning process, integrating quantization into the core objective instead of relegating it to a standalone stage. While this paradigm offers intuitive advantages, pruning directly under the quantization setting poses significant challenges because quantization restricts weights to a finite set of discrete values. The introduction of this discreteness constraint transforms pruning, traditionally optimized over continuous variables, into a sparse-constrained discrete optimization problem. Consequently, such optimization methods that rely on the continuity of full-precision weights are no longer directly applicable.

To tackle this challenge, we propose Quantization-aware One-shot Pruning (QOP), as illustrated in Figure 1, a method tailored for weight pruning in quantized LLMs. First, we propose a unified problem formulation that integrates layer-wise pruning and weight quantization into a single optimization problem. Such formulation incorporates the quantization-dequantization process into the objective, enabling pruning that explicitly accounts for quantization effects. Furthermore, its general discreteness constraint ensures versatility across various quantization settings, ranging from uniform integer to non-uniform floating-point types. In essence, this constitutes an NP-hard sparse-constrained discrete optimization problem, i.e., an ℓ_0-constrained optimization problem while simultaneously enforcing an additional discreteness constraint induced by quantization. To solve this problem, motivated by the success of the Alternating Direction Method of Multipliers (ADMM) [5] in sparsification [32, 6, 7], we propose a novel sparse-constrained discrete optimization framework grounded in ADMM. Specifically, we derive the closed-form solutions for each subproblem from scratch within this more intricate problem setup, thereby extending the ADMM algorithm to deliver an efficient and effective solution. With this solver, our method directly addresses the sparse-constrained discrete optimization problem arising in quantization-aware pruning and yields a high-performance sparse and quantized LLM in a one-shot manner, without any further fine-tuning or retraining.

Our contributions are summarized as follows:

- We propose a quantization-aware one-shot pruning method, called QOP, which provides the first unified formulation for pruning and quantization via formulating quantized weight pruning as a sparse-constrained discrete optimization problem. By extending ADMM, QOP effectively tackles this NP-hard problem.
- We provide the convergence guarantee for the ADMM algorithm used in QOP. We rigorously establish a proof of the algorithm's convergence to a stationary feasible point of the approximated problem and provide a corresponding convergence rate.
- We conduct extensive evaluations across quantized LLMs at multiple model scales, sparsity levels, and quantization settings. In terms of average metrics across a range of tasks, QOP consistently achieves state-of-the-art performance.

2 Preliminary

2.1 Layer-wise Pruning in LLM

The unstructured pruning problem for LLMs can be decomposed into layer-wise pruning subproblems [32, 43]. Specifically, we define $\hat{\mathbf{W}} \in \mathbb{R}^{N_{in} \times N_{out}}$ as the weight matrix of the original dense layer and $\mathbf{W} \in \mathbb{R}^{N_{in} \times N_{out}}$ for pruned sparse layer, where N_{in} and N_{out} represent the input and output dimension of the layer, respectively. Given N calibration samples with the sequence length of L and corresponding input activations $\mathbf{X} \in \mathbb{R}^{NL \times N_{in}}$, the optimization objective for each pruning subproblem is defined as the ℓ_2 error between the outputs of the original dense layer and the pruned sparse layer, as follows:

$$\min_{\mathbf{W}} \frac{1}{2}\|\mathbf{X}\hat{\mathbf{W}} - \mathbf{X}\mathbf{W}\|_F^2 + \frac{\lambda}{2}\|\hat{\mathbf{W}} - \mathbf{W}\|_F^2 \qquad \text{s.t.} \quad \|\mathbf{W}\|_0 \leq k, \tag{1}$$

where $\lambda > 0$ is the regularization parameter, $\|\cdot\|_0$ denotes the ℓ_0 norm (i.e., the number of non-zero elements in the weight matrix), and $\|\cdot\|_F$ denotes the Frobenius norm of the matrix. By solving the subproblem in Eq.(1), each dense layer can be pruned independently, after which the overall pruned model is obtained by integrating the pruned layers.

2.2 Quantization

Quantization reduces the precision of model parameters by mapping high-precision values to discrete levels, thereby saving memory and reducing latency. In this paper, we mainly focus on weight quantization, whose quantization and dequantization processes can be expressed as follows:

$$\mathbf{W}_q = Q(\frac{\hat{\mathbf{W}}}{\mathrm{s}}), \quad \tilde{\mathbf{W}} = \mathrm{s} \cdot \mathbf{W}_q, \tag{2}$$

where $\mathbf{W}_q$ and $\tilde{\mathbf{W}}$ denote the quantized and dequantized weights, respectively, and s is the scaling factor. For **uniform integer quantization** [20], $Q(\cdot) = \text{clamp}\left(\text{round}\left(\cdot\right), -2^{n-1}, 2^{n-1}-1\right)$, where n is the number of bits, clamp$(\cdot)$ is the clipping operator, round$(\cdot)$ is the rounding operator, and $[-2^{n-1}, 2^{n-1}-1]$ is the quantization range determined by n. For **non-uniform floating-point quantization** [24], $Q(\cdot)$ denotes the quantization function that maps high-precision values to the nearest representative points in the low-precision floating-point format (e.g., E4M3 in FP8). In this paper, we further consider finer-grained quantization, where different scaling factors are applied to each channel of weights, also referred to as per-channel quantization.

3 Methodology

3.1 Revisiting One-shot Pruning for Full-Precision LLM via ADMM

The optimization of the problem in Eq.(1) is intractable due to the sparsity constraint. Recent works [32, 6, 7] propose leveraging ADMM to overcome this challenge for one-shot pruning, with ALPS [32] being the most representative among them. Under its framework, an auxiliary variable $\mathbf{D}$ is incorporated to decouple the sparsity constraint from the objective:

$$\min_{\mathbf{W},\mathbf{D}} \frac{1}{2}\|\mathbf{X}\hat{\mathbf{W}} - \mathbf{X}\mathbf{W}\|_F^2 + \frac{\lambda}{2}\|\hat{\mathbf{W}} - \mathbf{W}\|_F^2 + I_{\mathcal{S}}(\mathbf{D}) \qquad \text{s.t.} \quad \mathbf{W} = \mathbf{D}, \tag{3}$$

where $I_{\mathcal{S}}(\mathbf{D})$ is an indicator function that is 0 if $\mathbf{D}$ is within $\mathcal{S} = \{\mathbf{D} \mid \|\mathbf{D}\|_0 \leq k\}$. To solve this problem, the augmented Lagrangian function is considered:

$$L_\rho(\mathbf{W}, \mathbf{D}, \mathbf{V}) = \frac{1}{2}\|\mathbf{X}\hat{\mathbf{W}} - \mathbf{X}\mathbf{W}\|_F^2 + \frac{\lambda}{2}\|\hat{\mathbf{W}} - \mathbf{W}\|_F^2 + I_{\mathcal{S}}(\mathbf{D}) + \langle \mathbf{V}, \mathbf{W} - \mathbf{D}\rangle + \frac{\rho}{2}\|\mathbf{W} - \mathbf{D}\|_F^2, \tag{4}$$

where $\mathbf{V}$ is the dual variable and ρ is the penalty parameter. The ADMM optimization proceeds by iteratively alternating between the minimization of the augmented Lagrangian with respect to $\mathbf{W}$ and $\mathbf{D}$ and subsequently performing a dual ascent on $\mathbf{V}$:

$$\begin{aligned}
\mathbf{W}^{(t+1)} &= \arg\min_{\mathbf{W}} L_\rho(\mathbf{W}, \mathbf{D}^{(t)}, \mathbf{V}^{(t)}) = (\mathbf{H} + \rho\mathbf{I})^{-1}(\mathbf{H}\hat{\mathbf{W}} - \mathbf{V}^{(t)} + \rho\mathbf{D}^{(t)}), \\
\mathbf{D}^{(t+1)} &= \arg\min_{\mathbf{D}} L_\rho(\mathbf{W}^{(t+1)}, \mathbf{D}, \mathbf{V}^{(t)}) = P_{\mathcal{S}}(\mathbf{W}^{(t+1)} + \frac{\mathbf{V}^{(t)}}{\rho}), \\
\mathbf{V}^{(t+1)} &= \mathbf{V}^{(t)} + \rho(\mathbf{W}^{(t+1)} - \mathbf{D}^{(t+1)}),
\end{aligned} \tag{5}$$

where $\mathbf{H} = \mathbf{X}^\top\mathbf{X} + \lambda\mathbf{I}$, and $P_{\mathcal{S}}(\cdot)$ is the projection operator that maps the input to the closest element in $\mathcal{S}$ in terms of Euclidean distance.

3.2 Motivation

Existing one-shot pruning methods [12, 32, 15], including the framework based on ADMM detailed in Section 3.1, are constrained by the assumption of weight continuity. Consequently, they are fundamentally trapped in a decoupled pruning-quantization pipeline when integrated with quantization. Such decoupling inherently yields suboptimal solutions. Since the pruning decisions remain oblivious to subsequent quantization errors, the pipeline fails to capture the shifts in weight importance triggered by discretization, which we refer to as quantization-induced shifts, ultimately trapping the model in a suboptimal state. To expose this vulnerability, we conduct a diagnostic experiment using Wanda [39] to trace weight importance before and after weight quantization. As shown in Table 1, we observe that the transition from FP16 to INT4 triggers a substantial alteration in the weight mask, which provides striking evidence of the pitfalls of decoupled execution.

Table 1: The different ratio of weight mask across the Llama-2-7B self-attention output projection layers at 50% unstructured sparsity.

Layer ID	**Diff Ratio (%)**
1	12.63
2	15.43
3	13.23
4	12.73
5	13.38
Average	*13.48*

3.3 Unified Problem Formulation

As discussed previously, sequential pruning–quantization pipelines fail to account for the impact of quantization on pruning, often leading to suboptimal performance in sparse quantized LLMs. Accordingly, we consider the unified treatment of both, which we refer to as quantization-aware one-shot pruning. By explicitly modeling the quantization operation to extend the pruning objective in Eq.(1) from full-precision weights to the per-channel quantization setting, we can obtain the following objective of layer-wise quantization-aware pruning:

$$\min_{\mathbf{W}_q} \frac{1}{2}\|\mathbf{X}\hat{\mathbf{W}} - \mathbf{X}\mathbf{s} \odot \mathbf{W}_q\|_F^2 + \frac{\lambda}{2}\|\hat{\mathbf{W}} - \mathbf{s} \odot \mathbf{W}_q\|_F^2 \quad \text{s.t.} \quad \|\mathbf{W}_q\|_0 \leq k, \;\; \forall i, j \; \mathbf{W}_{q,ij} \in C, \tag{6}$$

where $\mathbf{s} \in \mathbb{R}^{1\times N_{out}}$ is the scaling factor of the dequantization, $\mathbf{W}_q \in \mathbb{R}^{N_{in}\times N_{out}}$ is the quantized weights, $\odot$ denotes element-wise multiplication with broadcasting, and C denotes the discrete value set to which the quantized weights belong. It is evident that weight quantization introduces additional dequantization operations and discrete constraints into the objective, while also causing the objective to involve both reconstruction error and quantization error. Accordingly, the problem complexity is further increased compared to the sparsity-only formulation. A more detailed explanation of the symbols used in methodology is provided in Appendix B.

3.4 Quantization-aware One-shot Pruning based on ADMM Algorithm

As discreteness leads to immense combinatorial complexity, direct optimization of such a formulation in Eq.(6) is inherently intractable. Inspired by the validity of ADMM in Section 3.1, we generalize

the ADMM-based optimization framework to accommodate our unified formulation. Unlike the framework in Section 3.1, which only focuses on the sparsity of continuous weights, our purpose in introducing the auxiliary variable $\mathbf{D} = \mathbf{W}_q$ is to handle both sparsity and discreteness constraints. The problem is reformulated as follows:

$$\min_{\mathbf{W}_q, \mathbf{D}} \frac{1}{2}\|\mathbf{X}\hat{\mathbf{W}} - \mathbf{X}\mathbf{s} \odot \mathbf{W}_q\|_F^2 + \frac{\lambda}{2}\|\hat{\mathbf{W}} - \mathbf{s} \odot \mathbf{W}_q\|_F^2 + I_{\mathcal{A}}(\mathbf{D}) \qquad \text{s.t.} \quad \mathbf{W}_q = \mathbf{D}, \tag{7}$$

where $\mathcal{A}$ is defined as $\{\mathbf{D} \mid \|\mathbf{D}\|_0 \leq k,\ \mathbf{D}_{ij} \in C,\ \forall i, j\}$. Typically, we further consider the augmented Lagrangian function of this problem:

$$\begin{aligned} L_\rho(\mathbf{W}_q, \mathbf{D}, \mathbf{V}) =& \frac{1}{2}\|\mathbf{X}\hat{\mathbf{W}} - \mathbf{X}\mathbf{s} \odot \mathbf{W}_q\|_F^2 + \frac{\lambda}{2}\|\hat{\mathbf{W}} - \mathbf{s} \odot \mathbf{W}_q\|_F^2 + I_{\mathcal{A}}(\mathbf{D}) \\ &+ \langle \mathbf{V}, \mathbf{W}_q - \mathbf{D} \rangle + \frac{\rho}{2}\|\mathbf{W}_q - \mathbf{D}\|_F^2. \end{aligned} \tag{8}$$

Given the modified augmented Lagrangian, the subproblems yield the following closed-form update, with detailed derivations deferred to Appendix C.

$$\begin{aligned} \mathbf{D}_{ij}^{(t+1)} &= \begin{cases} \big((\mathbf{W}_q^{(t)} + \frac{\mathbf{V}^{(t)}}{\rho})_C\big)_{ij}, & \text{if } (i,j) \in \text{Topk}(\mathbf{Y}), \\ 0, \quad \text{otherwise}, \end{cases} \\ \mathbf{W}_{q,:j}^{(t+1)} &= (\mathbf{s}_j^2 \mathbf{H} + \rho \mathbf{I})^{-1} \mathbf{B}_{:j}^{(t)}, \\ \mathbf{V}^{(t+1)} &= \mathbf{V}^{(t)} + \rho(\mathbf{W}_q^{(t+1)} - \mathbf{D}^{(t+1)}), \end{cases} \end{aligned} \tag{9}$$

where $(\cdot)_C$ is the operator that projects each element of input to the nearest element of discrete set C, $\mathbf{Y}$ is a matrix constructed from $\mathbf{Y}_{ij} = (\mathbf{W}_q^{(t)} + \frac{\mathbf{V}^{(t)}}{\rho})_{ij}^2 - \big(((\mathbf{W}_q^{(t)} + \frac{\mathbf{V}^{(t)}}{\rho})_C)_{ij} - (\mathbf{W}_q^{(t)} + \frac{\mathbf{V}^{(t)}}{\rho})_{ij}\big)^2$, $\text{Topk}(\cdot)$ is the index set of the k largest elements in the input matrix, $\mathbf{B}_{:j}^{(t)}$ is the j-th column of $\mathbf{B}^{(t)} = \mathbf{H}\hat{\mathbf{W}}\mathbf{S} + \rho\mathbf{D}^{(t+1)} - \mathbf{V}^{(t)}$, and $\mathbf{S} = \text{diag}(\mathbf{s})$ is a diagonal matrix constructed from $\mathbf{s}$. Algorithm 1 outlines the optimization procedure for the proposed quantization-aware one-shot pruning framework. $\hat{\mathbf{W}}_q^{(l)}$ is obtained by applying the per-channel quantization process to $\hat{\mathbf{W}}^{(l)}$ of the l-th linear layer. L_w is the number of linear layers. By leveraging Algorithm 1, our method jointly identifies the high-quality support of the weights and refines the quantized weights.

Algorithm 1 Quantization-aware One-shot Pruning for Quantized LLM

Require: Quantized weights $\{\hat{\mathbf{W}}_q^{(l)}\}_{l=1}^{L_w}$, ADMM iteration T_{ADMM}.

1: **for** $l \in \{1, \ldots, L_w\}$ **do**
2: Initialize $\mathbf{W}_q^{(l,0)} = \hat{\mathbf{W}}_q^{(l)}$, $\mathbf{V}^{(l,0)} = \mathbf{0}$.
3: **for** $t = 0, 1, \ldots, T_{\text{ADMM}} - 1$ **do**
4: Update $\mathbf{D}^{(l,t+1)}$, $\mathbf{W}_q^{(l,t+1)}$, $\mathbf{V}^{(l,t+1)}$ by Eq.(9).
5: **end for**
6: Obtain $\mathbf{W}_q^{(l)}$ by applying the projection operator in the $\mathbf{D}$-update to $\mathbf{W}_q^{(l,T_{\text{ADMM}})}$.
7: **end for**
8: **return** Pruned quantized weights $\{\mathbf{W}_q^{(l)}\}_{l=1}^{L_w}$.

Remark 1. *QOP inherits the ADMM backbone of those pruning-only methods but necessitates a comprehensive re-derivation of the closed-form update rules to accommodate the joint constraints in Eq.(6). For example, unlike the simple magnitude-based top-k strategy in the pruning-only methods in Section 3.1, QOP proposes a novel projection operator in the first equation of Eq.(9) to minimize the joint quantization and sparsity error, enabling the optimization in the discrete-sparse landscape. Further analysis regarding the proposed QOP is available in Appendix D.*

4 Convergence

Algorithm 1 sequentially prunes all linear layers. Without loss of generality, we analyze the convergence of pruning for a single layer (lines 2-5 in Algorithm 1). Constrained by the sparsity and

discreteness constraints, the problem addressed by QOP is fundamentally an NP-hard combinatorial optimization problem. Following the approach in [46] to facilitate a rigorous theoretical analysis, we introduce a slight perturbation to relax the original problem into a continuous optimization framework. We then rigorously prove that the algorithm converges to a stationary feasible point of the approximated problem and provide a corresponding convergence rate.

Specifically, by introducing a perturbation term $\epsilon > 0$ and a continuous variable $\mathbf{U}$ into the constraint of Eq.(7), we construct the following approximated problem:

$$\begin{aligned} \min_{\mathbf{W}_q, \mathbf{D}, \mathbf{U}} \quad & \frac{1}{2}\|\mathbf{X}\hat{\mathbf{W}} - \mathbf{X}\mathbf{s} \odot \mathbf{W}_q\|_F^2 + \frac{\lambda}{2}\|\hat{\mathbf{W}} - \mathbf{s} \odot \mathbf{W}_q\|_F^2 + \frac{\epsilon}{2}\|\mathbf{U}\|_F^2 + I_{\mathcal{A}}(\mathbf{D}) \\ & \text{s.t.} \quad \mathbf{W}_q + \epsilon\mathbf{U} = \mathbf{D}. \end{aligned} \tag{10}$$

Evidently, the introduction of the ϵ-perturbation renders the originally discrete feasible set continuous. As $\epsilon \to 0^+$, Eq.(10) serves as a well-conditioned approximation of the original problem. To streamline subsequent notation, we denote the smooth objective function as $f(\mathbf{W}_q) = \frac{1}{2}\|\mathbf{X}\hat{\mathbf{W}} - \mathbf{X}\mathbf{s} \odot \mathbf{W}_q\|_F^2 + \frac{\lambda}{2}\|\hat{\mathbf{W}} - \mathbf{s} \odot \mathbf{W}_q\|_F^2$. Let the composite variable be $\tilde{\mathbf{W}}_q = [\mathbf{W}_q^\top, \mathbf{U}^\top]^\top$, and define $\hat{f}(\tilde{\mathbf{W}}_q) = f(\mathbf{W}_q) + \frac{\epsilon}{2}\|\mathbf{U}\|_F^2$. Employing QOP to solve the approximated problem in Eq.(10), the corresponding augmented Lagrangian function is defined as:

$$L_\epsilon(\tilde{\mathbf{W}}_q, \mathbf{D}, \mathbf{V}) = \hat{f}(\tilde{\mathbf{W}}_q) + I_{\mathcal{A}}(\mathbf{D}) + \langle \mathbf{V}, \mathbf{W}_q + \epsilon\mathbf{U} - \mathbf{D}\rangle + \frac{\rho}{2}\|\mathbf{W}_q + \epsilon\mathbf{U} - \mathbf{D}\|_F^2. \tag{11}$$

To facilitate the subsequent analysis, we recall the subgradient and Kurdyka-Łojasiewicz (KŁ) property, which plays a pivotal role in our analysis.

Definition 1. *For a lower semi-continuous function $f : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$, its subgradient set at $x \in \mathrm{dom} f$ is defined as:*

$$\partial f(x) := \left\{ v \in \mathbb{R}^m : \exists x^k \to x,\ v^k \xrightarrow{f} v,\ \liminf_{z \to x^k} \frac{f(z) - f(x^k) - \langle v^k, z - x^k\rangle}{\|z - x^k\|_F} \geq 0,\ \forall k \right\}.$$

Lemma 1 (KŁ Property [3, 26, 46])**.** *Every proper closed semi-algebraic function $f : \mathbb{R}^m \to \mathbb{R} \cup \{+\infty\}$ is a KŁ function with some exponent $\theta \in [0, 1)$. That is, for any $x^* \in \mathrm{dom}\,\partial f$, there exist constants $\eta \in (0, +\infty]$ and a neighborhood V such that for all $x \in V$ satisfying $f(x^*) < f(x) < f(x^*) + \eta$, the following KŁ inequality holds:*

$$\varphi'\big(f(x) - f(x^*)\big) \cdot \mathrm{dist}\big(0, \partial f(x)\big) = a(1-\theta)\big(f(x) - f(x^*)\big)^{-\theta} \cdot \mathrm{dist}\big(0, \partial f(x)\big) \geq 1,$$

where $a > 0$ is a constant depending on f, $\varphi(s) = as^{1-\theta}$, and $\mathrm{dist}(y, \Omega)$ denotes the Euclidean distance from point y to the set Ω.

Remark 2. *The augmented Lagrangian function L_ϵ, after the introduction of the ϵ-perturbation, is evidently a semi-algebraic function. Consequently, it satisfies the KŁ property as stated in Lemma 1.*

Under the above KŁ Property, we establish the global convergence and the convergence rate of QOP in the following theorems. The detailed proof is deferred to Appendix E.

Theorem 1 (Global Convergence)**.** *$\forall \epsilon > 0$, there exists $\rho > 0$, the sequence $\{(\tilde{\mathbf{W}}_q^{(t)}, \mathbf{D}^{(t)}, \mathbf{V}^{(t)})\}$ generated by the QOP algorithm for the problem defined in Eq.(10) globally converges to a stationary feasible point $(\tilde{\mathbf{W}}_q^*, \mathbf{D}^*, \mathbf{V}^*)$ of the approximated problem in Eq.(10).*

Theorem 2 (Convergence Rate)**.** *Suppose the KŁ exponent of the augmented Lagrangian function L_ϵ is $\theta \in [0, 1)$. Then the sequence $\{\mathbf{Z}^{(t)} = (\tilde{\mathbf{W}}_q^{(t)}, \mathbf{D}^{(t)}, \mathbf{V}^{(t)})\}$ in Theorem 1 converges to the stationary point $\mathbf{Z}^* = (\tilde{\mathbf{W}}_q^*, \mathbf{D}^*, \mathbf{V}^*)$ at a rate of at most $O(t^{-c})$, where the constant $c > 0$ is determined by θ. Specifically:*

1. *If $\theta = 0$, the sequence converges to a stationary point in a finite number of iterations;*
2. *If $0 < \theta \leq \frac{1}{2}$, the sequence converges at a linear rate, i.e., there exist constants $C_1, C_2 > 0$ (with $C_2 < 1$) such that $\|\mathbf{Z}^{(t)} - \mathbf{Z}^*\|_F \leq C_1 C_2^t$;*
3. *If $\frac{1}{2} < \theta < 1$, the sequence converges at a sublinear rate, i.e., there exists a constant $C_3 > 0$ such that $\|\mathbf{Z}^{(t)} - \mathbf{Z}^*\|_F \leq C_3 t^{-\frac{1-\theta}{2\theta-1}}$.*

5 Experiments

Experimental Design. We conduct extensive experiments to evaluate the effectiveness of our proposed QOP method for pruning quantized LLMs. To validate QOP, we present the main results by comparing it with pruning-quantization pipeline baselines on $\sim$7B models at 50%–70% unstructured sparsity. We extend this evaluation to other quantization settings and conduct ablation studies for a more comprehensive analysis. Further experimental setups and implementation details for these experiments are provided in Appendix F.1 and F.2, respectively. As a complementary comparison, we also evaluate QOP against the unified quantization and pruning baseline. Our code is available at https://github.com/ZHZ-JLU/PruningQANN.

5.1 Main Results

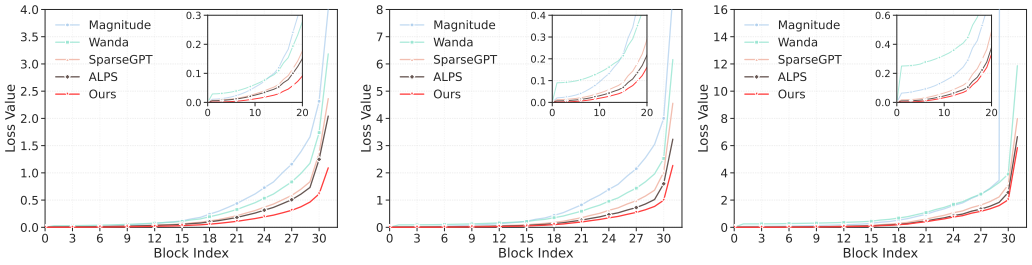
We focus on the perplexity and zero-shot accuracy results under unstructured pruning with sparsity levels ranging from 50% to 70% for W4A16-quantized versions of Llama-2-7B, Llama-7B, and OPT-6.7B. The results on Llama-2-7B are presented in Table 2. The results on Llama-7B and OPT-6.7B are reported in Table 10 and Table 11 of Appendix F.3, respectively. It can be observed that QOP outperforms all baselines, and as sparsity increases, the improvement in PPL also grows progressively. These results confirm the superiority of our proposed quantization-aware pruning framework over the pruning-quantization pipeline. Compared to the pruning-quantization pipeline, QOP effectively captures the shift in weight importance caused by weight quantization, while also reducing quantization error through updates to the discrete weights. Furthermore, this also demonstrates the effectiveness of QOP in addressing the sparse-constrained discrete optimization problem. We additionally present in Figure 2 the loss curves that measure the discrepancy between the outputs of each transformer block and the corresponding ANN outputs for all methods. Specifically, we compute the MSE loss using the ANN output as the ground truth for each block. Compared to other methods, QOP exhibits significantly lower loss under various sparsity levels.

Table 2: Perplexity and accuracy of quantized Llama-2-7B (W4A16) at 50%–70% unstructured sparsity. Highlighting indicates the best result, excluding the dense ANN and QANN references.

Sparsity	**Method**	**Zero-Shot Accuracy (%)↑**					**Perplexity (PPL)↓**				
		Wino	ArcC	ArcE	PiQA	Avg.	Wiki2	C4	Pile	Red	Avg.
0%	ANN	67.25	40.53	53.58	76.99	59.59	5.47	6.97	4.43	5.61	5.62
	QANN	66.46	38.57	50.59	74.59	57.55	7.13	8.81	5.36	7.30	7.15
50%	Magnitude	57.14	35.15	45.33	68.82	51.61	23.67	30.05	21.29	25.24	25.06
	Wanda	61.17	36.78	49.12	73.12	55.05	8.83	10.85	6.37	8.98	8.76
	SparseGPT	64.88	35.32	49.83	73.99	56.01	8.21	10.24	5.85	8.45	8.19
	ALPS	64.09	37.20	52.02	75.57	57.22	8.06	10.05	5.71	8.30	8.03
	QOP (Ours)	65.11	37.88	51.22	75.57	**57.45**	7.26	9.06	5.21	7.50	**7.26**
60%	Magnitude	50.75	27.82	36.24	60.45	43.82	>100	>100	>100	>100	>100
	Wanda	57.78	31.14	44.49	69.04	50.61	16.85	19.64	10.00	16.18	15.67
	SparseGPT	60.22	33.45	46.51	68.12	52.08	12.98	15.38	7.64	12.95	12.24
	ALPS	61.96	32.59	47.60	69.80	52.99	11.48	13.66	6.95	11.56	10.91
	QOP (Ours)	63.77	33.87	47.01	70.62	**53.82**	10.20	12.15	6.32	10.28	**9.73**
70%	Magnitude	47.99	27.21	27.94	50.92	38.52	>100	>100	>100	>100	>100
	Wanda	50.67	23.12	28.58	52.61	38.75	>100	>100	56.59	>100	>100
	SparseGPT	52.01	21.25	33.84	59.41	41.63	34.50	39.77	14.53	33.93	30.68
	ALPS	55.11	27.30	39.56	60.17	45.54	24.31	27.33	10.94	24.14	21.68
	QOP (Ours)	56.75	27.99	41.37	62.08	**47.05**	23.76	24.78	10.03	21.50	**20.02**

5.2 Other Quantization Settings

Non-uniform Floating-Point Quantization. Conducting experiments with a broader range of quantization settings strengthens the evidence for our method's effectiveness. Consequently, we evaluate QOP under FP8 non-uniform floating-point quantization in comparison with various baselines. Notably, despite FP8 being non-uniform, its adaptation only involves modifying the projection on set C at the implementation level, without affecting the previous discussion of QOP. This demonstrates the



(a) Sparsity = 50% (b) Sparsity = 60% (c) Sparsity = 70%

Figure 2: The layer-wise MSE loss on Llama-2-7B (W4A16) across different methods at 50%–70% unstructured sparsity.

generality of QOP. Results in Table 3 show that QOP outperforms the baseline methods, indicating that the proposed framework is not restricted to uniform quantization settings.

Table 3: Perplexity and accuracy of quantized OPT-6.7B (FP8, E4M3) at 60% unstructured sparsity. Highlighting indicates the best result, excluding the dense QANN reference.

Method	**Zero-Shot Accuracy (%)↑**					**Perplexity (PPL)↓**				
	Wino	ArcC	ArcE	PiQA	Avg.	Wiki2	C4	Pile	Red	Avg.
QANN	65.04	34.30	60.06	76.50	58.98	10.85	11.75	5.98	9.36	9.49
Magnitude	49.64	24.66	26.05	51.47	37.96	>100	>100	>100	>100	>100
Wanda	59.27	28.92	52.02	72.03	53.06	16.51	17.39	8.40	13.85	14.04
SparseGPT	61.72	30.72	54.92	73.39	55.19	14.54	14.74	6.91	11.67	11.97
ALPS	62.43	31.83	55.22	73.67	55.79	13.95	14.42	6.73	11.41	11.62
QOP (Ours)	63.22	32.17	55.51	74.05	**56.24**	13.88	14.42	6.71	11.37	**11.59**

Activation Quantization. We further evaluate the effectiveness of QOP under the setting of activation quantization. Table 4 presents the performance of various methods under the W4A8 setting at 60% sparsity. As shown, all methods experience a performance drop. This is attributed to the use of integer quantization for activations. QOP retains its performance edge over baselines in this scenario, once again proving its efficacy.

Table 4: Perplexity and accuracy of quantized OPT-6.7B (W4A8) at 60% unstructured sparsity. Highlighting indicates the best result, excluding the dense QANN reference.

Method	**Zero-Shot Accuracy (%)↑**					**Perplexity (PPL)↓**				
	Wino	ArcC	ArcE	PiQA	Avg.	Wiki2	C4	Pile	Red	Avg.
QANN	58.64	29.01	49.33	70.73	51.93	16.88	17.19	9.14	13.94	14.29
Magnitude	50.67	25.17	24.66	52.12	38.16	>100	>100	>100	>100	>100
Wanda	51.78	22.61	35.98	58.65	42.26	>100	>100	51.27	>100	>100
SparseGPT	52.72	23.21	37.21	58.16	42.82	90.21	64.14	23.88	48.65	56.72
ALPS	53.20	26.02	44.74	64.25	47.05	46.52	35.18	14.99	28.30	31.25
QOP (Ours)	53.20	30.12	51.18	70.29	**51.20**	20.21	20.94	9.84	16.83	**16.96**

5.3 Ablation Studies

Effects of Calibration Data. Calibration data is a critical factor for one-shot pruning methods. In the experiments for Table 2, we utilize a small subset of samples from the Pile dataset as calibration data. To verify the robustness of QOP across different calibration sources, we replace it with the C4 dataset for the ablation study. The results in Table 5 show that when switching to the C4 dataset, the PPL of all methods improves to some extent on C4, while QOP maintains its advantage over other baselines. This phenomenon demonstrates the robustness of QOP across different calibration datasets.

Table 5: Perplexity and accuracy of quantized Llama-2-7B (W4A16) at 50% unstructured sparsity using C4 as the calibration dataset.

Method	Zero-Shot Accuracy (%)↑					Perplexity (PPL)↓				
	Wino	ArcC	ArcE	PiQA	Avg.	Wiki2	C4	Pile	Red	Avg.
Wanda	61.48	36.09	49.28	73.56	55.10	8.80	10.68	6.57	8.91	8.74
SparseGPT	64.48	35.67	47.31	74.54	55.50	8.30	10.02	6.27	8.44	8.26
ALPS	64.72	37.12	48.82	75.52	56.55	8.11	9.70	6.52	8.32	8.16
QOP (Ours)	66.06	37.46	49.49	75.08	**57.02**	7.37	8.79	5.83	7.49	**7.37**

Effects of Regularization Parameter λ. The regularization parameter λ, which balances the activation reconstruction error and the weight reconstruction error, is a core hyperparameter of the algorithm. To evaluate the robustness of QOP to this parameter, we conduct an ablation study by adjusting its values. The results are reported in Table 6. Notably, the performance remains highly stable across different λ values. This consistency proves the QOP's robustness to regularization parameter selection.

Table 6: Ablation study on different regularization parameter λ under quantized Llama-2-7B (W4A16) at 50% unstructured sparsity. We report the average accuracy and perplexity.

λ	Avg. Acc ↑	Avg. PPL ↓
$0.01\text{Tr}(\mathbf{X}^T\mathbf{X})$	57.45	7.26
$0.005\text{Tr}(\mathbf{X}^T\mathbf{X})$	57.33	7.34
$0.05\text{Tr}(\mathbf{X}^T\mathbf{X})$	57.51	7.24
$0.1\text{Tr}(\mathbf{X}^T\mathbf{X})$	57.27	7.25

5.4 Comparisons with Unified Quantization and Pruning Baseline

To further demonstrate the effectiveness of QOP, we provide a complementary comparison with the unified quantization and pruning baseline. We add OBR [15] to our comparison, and the results are shown in Table 7. Following OBR's setup, we include HellaSwag (HellaS) [50] and BoolQ [8] in the accuracy comparison, with perplexity evaluated only on Wiki2. To ensure a fair comparison between the two methods (for example, to avoid the impact of some factors such as different versions of lm_eval in the two codebases on the evaluation results), we integrate QOP into the OBR framework[1]. OBR treats pruning and quantization as separate stages. In contrast, QOP considers the interaction between them, providing a new perspective on solving this problem. The results show the advantage of QOP over OBR. Additional comparisons with OBR are provided in Appendix F.5.

Table 7: Comparisons with unified quantization and pruning baseline OBR on Llama-2-7B at 50% unstructured sparsity. Following OBR, we use the Wiki2 dataset as the calibration dataset.

Bits	Method	Wino	ArcC	ArcE	PiQA	HellaS	BoolQ	Avg.	Wiki2
W4A4 KV4	OBR_RTN	59.27	31.74	53.79	69.10	58.77	66.64	56.55	8.75
	OBR_GPTQ	61.88	32.76	58.84	69.42	58.31	69.08	58.38	8.49
	QOP (Ours)	60.85	35.32	59.60	69.97	61.26	66.76	**58.96**	**7.76**
W4A4 KV16	OBR_RTN	59.12	32.25	55.09	70.08	60.12	69.20	57.64	8.26
	OBR_GPTQ	61.48	34.30	58.96	69.86	59.66	68.99	58.88	8.01
	QOP (Ours)	62.98	35.15	60.86	71.87	62.59	67.00	**60.08**	**7.39**

6 Conclusion

In this paper, we investigate the problem of one-shot pruning quantized LLMs and highlight the suboptimal performance inherent in combining existing optimization-driven one-shot pruning methods with quantization. To address this challenge, we propose QOP, which provides the first unified formulation for pruning and quantization by framing it as a sparse-constrained discrete optimization problem. By extending ADMM to this discrete setting, QOP jointly identifies high-quality supports and updates quantized weights while admitting efficient closed-form updates and a theoretical convergence guarantee. A comprehensive suite of experiments substantiates QOP's effectiveness, scalability, and robustness. In future work, we aim to extend QOP to multimodal architectures and spiking neural networks in order to deepen our understanding of the applicability of quantization-aware pruning.

[1] https://github.com/csguoh/OBR

Acknowledgments and Disclosure of Funding

Dr. Yi Chang was supported by the National Key R&D Program of China (Grant No. 2023YFF0905400), the National Natural Science Foundation of China (Grant No. U2341229), the Fundamental and Interdisciplinary Disciplines Breakthrough Plan of the Ministry of Education of China (JYB2025XDXM903), and the New Cornerstone Science Foundation through the XPLORER PRIZE.

References

[1] Yonatan Bisk, Rowan Zellers, Ronan Le Bras, Jianfeng Gao, and Yejin Choi. Piqa: Reasoning about physical commonsense in natural language. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pages 7432–7439, 2020.

[2] Davis Blalock, Jose Javier Gonzalez Ortiz, Jonathan Frankle, and John Guttag. What is the state of neural network pruning? *Proceedings of machine learning and systems*, 2:129–146, 2020.

[3] Jérôme Bolte, Aris Daniilidis, Adrian Lewis, and Masahiro Shiota. Clarke subgradients of stratifiable functions. *SIAM Journal on Optimization*, 18(2):556–572, 2007.

[4] Rishi Bommasani, Drew A Hudson, Ehsan Adeli, Russ Altman, Simran Arora, Sydney von Arx, Michael S Bernstein, Jeannette Bohg, Antoine Bosselut, Emma Brunskill, et al. On the opportunities and risks of foundation models. *arXiv preprint arXiv:2108.07258*, 2021.

[5] Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine learning*, 3(1):1–122, 2011.

[6] Vladimír Boža. Fast and effective weight update for pruned large language models. *arXiv preprint arXiv:2401.02938*, 2024.

[7] Weiyu Chen, Hansi Yang, Yunhao Gou, Han Shi, Enliang Hu, Zhenguo Li, and James Kwok. Multi-objective one-shot pruning for large language models. *Advances in Neural Information Processing Systems*, 38:91960–91992, 2025.

[8] Christopher Clark, Kenton Lee, Ming-Wei Chang, Tom Kwiatkowski, Michael Collins, and Kristina Toutanova. Boolq: Exploring the surprising difficulty of natural yes/no questions. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 2924–2936, 2019.

[9] Peter Clark, Isaac Cowhey, Oren Etzioni, Tushar Khot, Ashish Sabharwal, Carissa Schoenick, and Oyvind Tafjord. Think you have solved question answering? try arc, the ai2 reasoning challenge. *arXiv preprint arXiv:1803.05457*, 2018.

[10] Ahmad Faiz, Sotaro Kaneda, Ruhan Wang, Rita Osi, Prateek Sharma, Fan Chen, and Lei Jiang. Llmcarbon: Modeling the end-to-end carbon footprint of large language models. *arXiv preprint arXiv:2309.14393*, 2023.

[11] Elias Frantar and Dan Alistarh. Optimal brain compression: A framework for accurate post-training quantization and pruning. *Advances in Neural Information Processing Systems*, 35:4475–4488, 2022.

[12] Elias Frantar and Dan Alistarh. Sparsegpt: Massive language models can be accurately pruned in one-shot. In *International conference on machine learning*, pages 10323–10337. PMLR, 2023.

[13] Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. Gptq: Accurate post-training quantization for generative pre-trained transformers. *arXiv preprint arXiv:2210.17323*, 2022.

[14] Leo Gao, Stella Biderman, Sid Black, Laurence Golding, Travis Hoppe, Charles Foster, Jason Phang, Horace He, Anish Thite, Noa Nabeshima, et al. The pile: An 800gb dataset of diverse text for language modeling. *arXiv preprint arXiv:2101.00027*, 2020.

[15] Hang Guo, Yawei Li, and Luca Benini. Optimal brain restoration for joint quantization and sparsification of llms. *arXiv preprint arXiv:2509.11177*, 2025.

[16] Yiwen Guo, Anbang Yao, and Yurong Chen. Dynamic network surgery for efficient dnns. *Advances in neural information processing systems*, 29, 2016.

[17] Song Han, Jeff Pool, John Tran, and William Dally. Learning both weights and connections for efficient neural network. *Advances in neural information processing systems*, 28, 2015.

[18] Babak Hassibi and David Stork. Second order derivatives for network pruning: Optimal brain surgeon. *Advances in neural information processing systems*, 5, 1992.

[19] Babak Hassibi, David G Stork, and Gregory J Wolff. Optimal brain surgeon and general network pruning. In *IEEE international conference on neural networks*, pages 293–299. IEEE, 1993.

[20] Benoit Jacob, Skirmantas Kligys, Bo Chen, Menglong Zhu, Matthew Tang, Andrew Howard, Hartwig Adam, and Dmitry Kalenichenko. Quantization and training of neural networks for efficient integer-arithmetic-only inference. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 2704–2713, 2018.

[21] Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213, 2022.

[22] Eldar Kurtic, Daniel Campos, Tuan Nguyen, Elias Frantar, Mark Kurtz, Benjamin Fineran, Michael Goin, and Dan Alistarh. The optimal bert surgeon: Scalable and accurate second-order pruning for large language models. *arXiv preprint arXiv:2203.07259*, 2022.

[23] Aditya Kusupati, Vivek Ramanujan, Raghav Somani, Mitchell Wortsman, Prateek Jain, Sham Kakade, and Ali Farhadi. Soft threshold weight reparameterization for learnable sparsity. In *International conference on machine learning*, pages 5544–5555. PMLR, 2020.

[24] Andrey Kuzmin, Mart Van Baalen, Yuwei Ren, Markus Nagel, Jorn Peters, and Tijmen Blankevoort. Fp8 quantization: The power of the exponent. *Advances in Neural Information Processing Systems*, 35:14651–14662, 2022.

[25] Yann LeCun, John Denker, and Sara Solla. Optimal brain damage. *Advances in neural information processing systems*, 2, 1989.

[26] Guoyin Li and Ting Kei Pong. Global convergence of splitting methods for nonconvex composite optimization. *SIAM Journal on Optimization*, 25(4):2434–2460, 2015.

[27] Jing Liu, Toshiaki Koike-Akino, Ye Wang, Hassan Mansour, and Matthew Brand. Awp: Activation-aware weight pruning and quantization with projected gradient descent. *arXiv preprint arXiv:2506.10205*, 2025.

[28] Zechun Liu, Haoyuan Mu, Xiangyu Zhang, Zichao Guo, Xin Yang, Kwang-Ting Cheng, and Jian Sun. Metapruning: Meta learning for automatic neural network channel pruning. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 3296–3305, 2019.

[29] Zechun Liu, Changsheng Zhao, Igor Fedorov, Bilge Soran, Dhruv Choudhary, Raghuraman Krishnamoorthi, Vikas Chandra, Yuandong Tian, and Tijmen Blankevoort. Spinquant: Llm quantization with learned rotations. *arXiv preprint arXiv:2405.16406*, 2024.

[30] Zhuang Liu, Mingjie Sun, Tinghui Zhou, Gao Huang, and Trevor Darrell. Rethinking the value of network pruning. *arXiv preprint arXiv:1810.05270*, 2018.

[31] Xinyin Ma, Gongfan Fang, and Xinchao Wang. Llm-pruner: On the structural pruning of large language models. *Advances in neural information processing systems*, 36:21702–21720, 2023.

[32] Xiang Meng, Kayhan Behdin, Haoyue Wang, and Rahul Mazumder. Alps: Improved optimization for highly sparse one-shot pruning for large language models. *Advances in Neural Information Processing Systems*, 37:37594–37625, 2024.

[33] Stephen Merity, Caiming Xiong, James Bradbury, and Richard Socher. Pointer sentinel mixture models. *arXiv preprint arXiv:1609.07843*, 2016.

[34] Decebal Constantin Mocanu, Elena Mocanu, Peter Stone, Phuong H Nguyen, Madeleine Gibescu, and Antonio Liotta. Scalable training of artificial neural networks with adaptive sparse connectivity inspired by network science. *Nature communications*, 9(1):2383, 2018.

[35] Pavlo Molchanov, Stephen Tyree, Tero Karras, Timo Aila, and Jan Kautz. Pruning convolutional neural networks for resource efficient inference. *arXiv preprint arXiv:1611.06440*, 2016.

[36] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

[37] Keisuke Sakaguchi, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. Winogrande: An adversarial winograd schema challenge at scale. *Communications of the ACM*, 64(9):99–106, 2021.

[38] Wenqi Shao, Mengzhao Chen, Zhaoyang Zhang, Peng Xu, Lirui Zhao, Zhiqian Li, Kaipeng Zhang, Peng Gao, Yu Qiao, and Ping Luo. Omniquant: Omnidirectionally calibrated quantization for large language models. *arXiv preprint arXiv:2308.13137*, 2023.

[39] Mingjie Sun, Zhuang Liu, Anna Bair, and J Zico Kolter. A simple and effective pruning approach for large language models. *arXiv preprint arXiv:2306.11695*, 2023.

[40] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*, 2023.

[41] Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint arXiv:2307.09288*, 2023.

[42] Maurice Weber, Dan Fu, Quentin Anthony, Yonatan Oren, Shane Adams, Anton Alexandrov, Xiaozhong Lyu, Huu Nguyen, Xiaozhe Yao, Virginia Adams, et al. Redpajama: an open dataset for training large language models. *Advances in neural information processing systems*, 37:116462–116492, 2024.

[43] Canhong Wen, Yihong Zuo, and Wenliang Pan. An efficient pruner for large language model with theoretical guarantee. In *Forty-second International Conference on Machine Learning*, 2025.

[44] Wei Wen, Chunpeng Wu, Yandan Wang, Yiran Chen, and Hai Li. Learning structured sparsity in deep neural networks. *Advances in neural information processing systems*, 29, 2016.

[45] Guangxuan Xiao, Ji Lin, Mickael Seznec, Hao Wu, Julien Demouth, and Song Han. Smoothquant: Accurate and efficient post-training quantization for large language models. In *International conference on machine learning*, pages 38087–38099. PMLR, 2023.

[46] Huan Xiong, Mengyang Yu, Li Liu, Fan Zhu, Jie Qin, Fumin Shen, and Ling Shao. A generalized method for binary optimization: Convergence analysis and applications. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(9):4524–4543, 2021.

[47] An Yang, Anfeng Li, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Gao, Chengen Huang, Chenxu Lv, et al. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025.

[48] An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge,

Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan. Qwen2 technical report, 2024.

[49] Lu Yin, You Wu, Zhenyu Zhang, Cheng-Yu Hsieh, Yaqing Wang, Yiling Jia, Gen Li, Ajay Jaiswal, Mykola Pechenizkiy, Yi Liang, et al. Outlier weighed layerwise sparsity (owl): A missing secret sauce for pruning llms to high sparsity. *arXiv preprint arXiv:2310.05175*, 2023.

[50] Rowan Zellers, Ari Holtzman, Yonatan Bisk, Ali Farhadi, and Yejin Choi. Hellaswag: Can a machine really finish your sentence? In *Proceedings of the 57th annual meeting of the association for computational linguistics*, pages 4791–4800, 2019.

[51] Susan Zhang, Stephen Roller, Naman Goyal, Mikel Artetxe, Moya Chen, Shuohui Chen, Christopher Dewan, Mona Diab, Xian Li, Xi Victoria Lin, et al. Opt: Open pre-trained transformer language models. *arXiv preprint arXiv:2205.01068*, 2022.

[52] Yingtao Zhang, Haoli Bai, Haokun Lin, Jialin Zhao, Lu Hou, and Carlo Vittorio Cannistraci. Plug-and-play: An efficient post-training pruning method for large language models. In *International Conference on Learning Representations*, volume 2024, pages 50490–50508, 2024.

[53] Yuxin Zhang, Lirui Zhao, Mingbao Lin, Yunyun Sun, Yiwu Yao, Xingjia Han, Jared Tanner, Shiwei Liu, and Rongrong Ji. Dynamic sparse no training: Training-free fine-tuning for sparse llms. *arXiv preprint arXiv:2310.08915*, 2023.

[54] Aojun Zhou, Yukun Ma, Junnan Zhu, Jianbo Liu, Zhijie Zhang, Kun Yuan, Wenxiu Sun, and Hongsheng Li. Learning n: m fine-grained structured sparse neural networks from scratch. *arXiv preprint arXiv:2102.04010*, 2021.

Appendix

A Related Work

Network Pruning. Network pruning is a widely recognized paradigm for developing lightweight models [19, 17]. Its common categorization is based on training requirements, which categorizes network pruning into one-shot pruning, gradual pruning, and training from scratch. One-shot pruning requires no additional training to recover performance [11, 39]. In comparison, gradual pruning [35, 22] and training from scratch [34, 23] necessitate varying degrees of training. The former alternates between pruning and fine-tuning, while the latter learns a sparse network from scratch. Furthermore, depending on the pruning granularity, pruning methods can also be categorized into structured and unstructured pruning. The former operates at coarse-grained structures such as channels [44, 31], whereas the latter removes weights at a fine-grained individual-parameter level [16, 52]. In addition, semi-structured pruning (e.g., N:M sparsity, i.e., N out of M weights kept) serves as a compromise between the two, balancing the hardware efficiency of structured pruning with the superior performance retention of unstructured pruning [54]. This work primarily investigates one-shot unstructured pruning for LLMs.

One-shot Unstructured Pruning for LLMs. Post-training one-shot unstructured pruning methods for LLMs can be broadly categorized into heuristic-based and optimization-based approaches. (i) Heuristic-based methods [17, 39, 49, 12, 52] typically estimate the importance of each weight using simple statistics and prune less important weights by setting them to zero. Given a weight matrix $\hat{\mathbf{W}} = (\hat{w}_{ij}) \in \mathbb{R}^{N_{\text{in}} \times N_{\text{out}}}$, magnitude pruning [17] computes an importance score based on weight magnitude, i.e., $S_{ij} = |\hat{w}_{ij}|$, and sparsifies weights accordingly. Wanda [39] combines weight magnitude with activation statistics computed on a calibration set, typically formulated as $S_{ij} = |\hat{w}_{ij}| \cdot \|x_i\|_2$. Outlier Weighed Layerwise sparsity [49] considers the outlier distribution in large language models and applies a non-uniform layer-wise sparsity strategy, where importance scores are computed by combining weights and activations. SparseGPT [12] approximates the OBS [18] algorithm via partial weight updates and adaptive mask selection, avoiding the expensive computation of the Hessian. Relative importance and activation [52] defines a relative importance score $\text{RIA}_{ij} = (\frac{|\hat{w}_{ij}|}{\sum_{i'}|\hat{w}_{i'j}|} + \frac{|\hat{w}_{ij}|}{\sum_{j'}|\hat{w}_{ij'}|}) \times (\|x_i\|_2)^a$, which evaluates the importance of weights based on the relative scales of their input and output channels and uses this score to guide pruning. (ii) Optimization-based methods [32, 43, 7] formulate reconstruction objectives under sparsity constraints and perform pruning by approximately minimizing these objectives (typically in a layer-wise or block-wise manner). ALPS [32] formulates pruning as an ℓ_0-constrained reconstruction problem, uses ADMM to identify a high-quality support, and then refines the weights on this support with a modified PCG solver. mAIHT [43] similarly introduces an ℓ_0 constraint and minimizes the pruning error via iterative hard-thresholding updates. MOSP [7] considers one-shot pruning of LLMs under the multi-objective setting and addresses diverse user preferences through a multi-stage process.

B Definition of Key Notation in Methodology

Key notation used in methodology is provided in Table 8.

C Derivation of the Closed-form Updates in Eq.(9)

D-update. The update for $\mathbf{D}$ aims to minimize the following part of the augmented Lagrangian function:

$$I_{\mathcal{A}}(\mathbf{D}) + \left\langle \mathbf{V}^{(t)}, -\mathbf{D} \right\rangle + \frac{\rho}{2}\|\mathbf{W}_q^{(t)} - \mathbf{D}\|_F^2. \tag{12}$$

This is equivalent to minimizing the objective as follows:

$$I_{\mathcal{A}}(\mathbf{D}) + \frac{\rho}{2}\|\mathbf{D} - (\mathbf{W}_q^{(t)} + \frac{\mathbf{V}^{(t)}}{\rho})\|_F^2. \tag{13}$$

Table 8: Key notation used in methodology.

Notation	Description
N_{in}, N_{out}	The input and output dimension of the layer.
N	The number of calibration samples.
$\mathbf{X}$	Input activations of calibration samples.
L	Sequence length of the calibration sample.
L_w	The number of linear layers.
$\hat{\mathbf{W}}$	The weight matrix of the original dense layer.
$\hat{\mathbf{W}}_q$	The quantized weight of the original weight matrix $\hat{\mathbf{W}}$.
λ	Regularization parameter.
$\mathbf{W}_q$	Quantized weight.
$\tilde{\mathbf{W}}$	Dequantized weight.
$\mathbf{s}$	Scaling factor of the dequantization.
$\mathbf{S}$	The diagonal matrix constructed from $\mathbf{s}$.
k	The number of non-zero elements in the weight matrix.
n	The number of bits.
C	The discrete value set to which the quantized weights belong.
$\mathbf{D}$	Auxiliary variable in ADMM.
$\mathbf{V}$	Dual variable in ADMM.
I	Indicator function.
ρ	Penalty parameter.
$L_\rho(\mathbf{W}_q, \mathbf{D}, \mathbf{V})$	Augmented Lagrangian function.

Thus, it leads to the following update for $\mathbf{D}$:

$$\begin{aligned}\mathbf{D}^{(t+1)} &= \arg\min_{\mathbf{D}} I_{\mathcal{A}}(\mathbf{D}) + \frac{\rho}{2}\|\mathbf{D} - (\mathbf{W}_q^{(t)} + \frac{\mathbf{V}^{(t)}}{\rho})\|_F^2 \\ &= \arg\min_{\mathbf{D}} I_{\mathcal{A}}(\mathbf{D}) + \|\mathbf{D} - (\mathbf{W}_q^{(t)} + \frac{\mathbf{V}^{(t)}}{\rho})\|_F^2.\end{aligned} \tag{14}$$

It can be readily shown by contradiction that $\forall i,j\ \mathbf{D}_{ij}^{(t+1)} \in \{0, \left((\mathbf{W}_q^{(t)} + \frac{\mathbf{V}^{(t)}}{\rho})_C\right)_{ij}\}$, where $(\cdot)_C$ is the operator that projects each element of input to the nearest element of discrete set C. The error values induced by the two choices are $(\mathbf{W}_q^{(t)} + \frac{\mathbf{V}^{(t)}}{\rho})_{ij}^2$ and $\left(((\mathbf{W}_q^{(t)} + \frac{\mathbf{V}^{(t)}}{\rho})_C)_{ij} - (\mathbf{W}_q^{(t)} + \frac{\mathbf{V}^{(t)}}{\rho})_{ij}\right)^2$, respectively.

When $\forall i,j\ \mathbf{D}_{ij}^{(t+1)} = 0$, for the fixed $\mathbf{W}_q^{(t)} + \frac{\mathbf{V}^{(t)}}{\rho}$, the error value is fixed at $\|\mathbf{W}_q^{(t)} + \frac{\mathbf{V}^{(t)}}{\rho}\|_F^2$. Therefore, the problem can be reformulated as changing k elements in $\mathbf{D}^{(t+1)}$ from 0 to $\left((\mathbf{W}_q^{(t)} + \frac{\mathbf{V}^{(t)}}{\rho})_C\right)_{ij}$ so as to maximize the reduction in error. We define $\mathbf{Y}_{ij} = (\mathbf{W}_q^{(t)} + \frac{\mathbf{V}^{(t)}}{\rho})_{ij}^2 - \left(((\mathbf{W}_q^{(t)} + \frac{\mathbf{V}^{(t)}}{\rho})_C)_{ij} - (\mathbf{W}_q^{(t)} + \frac{\mathbf{V}^{(t)}}{\rho})_{ij}\right)^2$ and denote Topk$(\cdot)$ as the index set of the k largest elements in the input matrix. Then,

$$\mathbf{D}_{ij}^{(t+1)} = \begin{cases} \left((\mathbf{W}_q^{(t)} + \frac{\mathbf{V}^{(t)}}{\rho})_C\right)_{ij}, & \text{if} \quad (i,j) \in \text{Topk}(\mathbf{Y}), \\ 0, & \text{otherwise.} \end{cases} \tag{15}$$

$\mathbf{W}_q$-update. To simplify the augmented Lagrangian in Eq.(8), we define $\mathbf{S} = \text{diag}(\mathbf{s})$, i.e., a diagonal matrix constructed from $\mathbf{s}$, as shown below:

$$\begin{aligned}L_\rho(\mathbf{W}_q, \mathbf{D}, \mathbf{V}) =& \frac{1}{2}\|\mathbf{X}\hat{\mathbf{W}} - \mathbf{X}\mathbf{W}_q\mathbf{S}\|_F^2 + \frac{\lambda}{2}\|\hat{\mathbf{W}} - \mathbf{W}_q\mathbf{S}\|_F^2 + I_{\mathcal{A}}(\mathbf{D}) \\ &+ \langle \mathbf{V}, \mathbf{W}_q - \mathbf{D}\rangle + \frac{\rho}{2}\|\mathbf{W}_q - \mathbf{D}\|_F^2.\end{aligned} \tag{16}$$

Similarly, the update for $\mathbf{W}_q$ is obtained by minimizing $L_\rho(\mathbf{W}_q, \mathbf{D}^{(t+1)}, \mathbf{V}^{(t)})$. Taking its derivative with respect to $\mathbf{W}_q$ and setting it to zero yields the following equation:

$$\mathbf{H}\mathbf{W}_q\mathbf{S}^2 + \rho\mathbf{W}_q = \mathbf{H}\hat{\mathbf{W}}\mathbf{S} + \rho\mathbf{D}^{(t+1)} - \mathbf{V}^{(t)}. \tag{17}$$

Since $\mathbf{S}$ is a diagonal matrix, we can derive a closed-form solution by decoupling the columns as follows:

$$(\mathbf{s}_j^2\mathbf{H} + \rho\mathbf{I})\mathbf{W}_{q,:j} = \mathbf{B}_{:j}^{(t)}, \tag{18}$$

where $\mathbf{B}_{:j}^{(t)}$ is the j-th column of $\mathbf{B}^{(t)} = \mathbf{H}\hat{\mathbf{W}}\mathbf{S} + \rho\mathbf{D}^{(t+1)} - \mathbf{V}^{(t)}$. The update for $\mathbf{W}_q$ is given by

$$\mathbf{W}_{q,:j}^{(t+1)} = (\mathbf{s}_j^2\mathbf{H} + \rho\mathbf{I})^{-1}\mathbf{B}_{:j}^{(t)}. \tag{19}$$

$\mathbf{V}$-update. The dual variable is updated as

$$\mathbf{V}^{(t+1)} = \mathbf{V}^{(t)} + \rho(\mathbf{W}_q^{(t+1)} - \mathbf{D}^{(t+1)}). \tag{20}$$

D Analysis for QOP

Avoiding Column-wise Update. The update of $\mathbf{W}_q$ in Eq.(9) admits a closed-form solution in a column-wise manner, which is less amenable to efficient implementation. To avoid column-wise update, we first perform eigenvalue decomposition for $\mathbf{H}$ as ALPS [32] does, yielding $\mathbf{H} = \mathbf{Q}\mathbf{O}\mathbf{Q}^\top$. The update rule can then be reformulated as follows:

$$\mathbf{W}_{q,:j}^{(t+1)} = \mathbf{Q}(\mathbf{s}_j^2\mathbf{O} + \rho\mathbf{I})^{-1}\mathbf{Q}^\top\mathbf{B}_{:j}^{(t)}. \tag{21}$$

Notably, $(\mathbf{s}_j^2\mathbf{O} + \rho\mathbf{I})^{-1}$ can be readily obtained by taking element-wise reciprocals. Furthermore, due to its diagonal structure, the update rule can be further reformulated as follows:

$$\mathbf{W}_{q,:j}^{(t+1)} = \mathbf{Q}\big(\mathbf{p}_j \odot (\mathbf{Q}^\top\mathbf{B}_{:j}^{(t)})\big), \tag{22}$$

where $\mathbf{p}_j$ is the column vector formed by the diagonal elements of $(\mathbf{s}_j^2\mathbf{O} + \rho\mathbf{I})^{-1}$. Ultimately, we can derive the update rule for the entire matrix $\mathbf{W}_q$ rather than the original column-wise formulation in Eq.(9), as shown below:

$$\mathbf{W}_q^{(t+1)} = \mathbf{Q}\big(\mathbf{P} \odot (\mathbf{Q}^\top\mathbf{B}^{(t)})\big), \tag{23}$$

where $\mathbf{P} = [\mathbf{p}_1, \mathbf{p}_2, \cdots, \mathbf{p}_{N_{out}}]$.

Computational Cost. Due to the typically large hidden dimensions of LLMs, the computational complexity of an element-wise matrix multiplication is negligible compared to that of a matrix multiplication. Therefore, the computational cost of Line 2-5 in Algorithm 1 is dominated by the matrix multiplications involved in updating $\mathbf{W}_q$. Since $\mathbf{H}\hat{\mathbf{W}}\mathbf{S}$ is a fixed term during the iterative process and can be pre-computed, the computational overhead of computing $\mathbf{B}^{(t)}$ is negligible. Thus, the dominant computational cost in updating $\mathbf{W}_q$ arises from two matrix multiplications, resulting in a time complexity of $O(N_{in}^2 N_{out})$.

Difference between QOP and ALPS. While both QOP and ALPS [32] are rooted in the ADMM framework, the fundamental distinction of QOP lies in its pursuit of a significantly more challenging optimization objective. Specifically, while ALPS focuses on ℓ_0-constrained optimization within a continuous domain, QOP introduces a unified formulation that integrates pruning and quantization into a single optimization. This shift from a continuous to a discrete feasible grid elevates the task to a combinatorial optimization challenge, tackling an inherently more complex NP-hard problem. A comparison summary between QOP and ALPS can be found in Table 9.

Table 9: Comparison summary between QOP and ALPS.

Method	Objective Function	Variable	Mathematical Constraint	Feasible Set
ALPS	Pruning-only	FP16	$\|\mathbf{W}\|_0 \leq k$	Continuous
QOP	Pruning & Quantization	Quantized	$\|\mathbf{W}_q\|_0 \leq k,\ \forall i, j\ \mathbf{W}_{q,ij} \in C$	Discrete

E Convergence

In light of the perturbation, the iterative update rules for the variables are correspondingly adjusted as:

$$\begin{aligned}\mathbf{D}^{(t+1)} &= \arg\min_{\mathbf{D}} L_\epsilon(\tilde{\mathbf{W}}_q^{(t)}, \mathbf{D}, \mathbf{V}^{(t)}),\\ \tilde{\mathbf{W}}_q^{(t+1)} &= \arg\min_{\tilde{\mathbf{W}}_q} L_\epsilon(\tilde{\mathbf{W}}_q, \mathbf{D}^{(t+1)}, \mathbf{V}^{(t)}),\\ \mathbf{V}^{(t+1)} &= \mathbf{V}^{(t)} + \rho(\mathbf{W}_q^{(t+1)} + \epsilon\mathbf{U}^{(t+1)} - \mathbf{D}^{(t+1)}).\end{aligned} \tag{24}$$

For the convenience of subsequent derivations, we define the mapping matrix $\mathbf{M} = [\mathbf{I}_n, \epsilon\mathbf{I}_n]$, thus we have $\mathbf{M}\tilde{\mathbf{W}}_q = \mathbf{W}_q + \epsilon\mathbf{U}$.

Lemma 2. *For any $\epsilon > 0$, there exist $\rho > 0$ and positive constants c_0, c_1 such that the difference in the augmented Lagrangian function values between the $(t+1)$-th and t-th iterations of the QOP algorithm satisfies:*

$$\begin{aligned}&L_\epsilon(\tilde{\mathbf{W}}_q^{(t+1)}, \mathbf{D}^{(t+1)}, \mathbf{V}^{(t+1)}) - L_\epsilon(\tilde{\mathbf{W}}_q^{(t)}, \mathbf{D}^{(t)}, \mathbf{V}^{(t)})\\ &\le -c_0\|\tilde{\mathbf{W}}_q^{(t+1)} - \tilde{\mathbf{W}}_q^{(t)}\|_F^2\\ &\le -c_1\big(\|\tilde{\mathbf{W}}_q^{(t+1)} - \tilde{\mathbf{W}}_q^{(t)}\|_F^2 + \|\mathbf{D}^{(t+1)} - \mathbf{D}^{(t)}\|_F^2 + \|\mathbf{V}^{(t+1)} - \mathbf{V}^{(t)}\|_F^2\big).\end{aligned} \tag{25}$$

Proof. It is easy to verify that $\hat{f}$ is a positive definite quadratic function in $\tilde{\mathbf{W}}_q$, hence $\hat{f}$ is μ-strongly convex and L_1-smooth, where μ and L_1 are respectively the smallest and largest eigenvalues of its Hessian matrix. In the following we assume the penalty parameter ρ is chosen sufficiently large such that

$$\rho > \frac{L_1^2}{\mu(1+\epsilon^2)}. \tag{26}$$

From the second update rule in Eq.(24), setting the gradient to zero yields:

$$\nabla_{\tilde{\mathbf{W}}_q}\hat{f}(\tilde{\mathbf{W}}_q^{(t+1)}) + \mathbf{M}^\top\mathbf{V}^{(t)} + \rho\mathbf{M}^\top(\mathbf{M}\tilde{\mathbf{W}}_q^{(t+1)} - \mathbf{D}^{(t+1)}) = \mathbf{0}. \tag{27}$$

Leveraging the dual update relation $\mathbf{M}\tilde{\mathbf{W}}_q^{(t+1)} - \mathbf{D}^{(t+1)} = \frac{1}{\rho}(\mathbf{V}^{(t+1)} - \mathbf{V}^{(t)})$, we obtain:

$$\nabla_{\tilde{\mathbf{W}}_q}\hat{f}(\tilde{\mathbf{W}}_q^{(t+1)}) = -\mathbf{M}^\top\mathbf{V}^{(t+1)} = -[(\mathbf{V}^{(t+1)})^\top, \epsilon(\mathbf{V}^{(t+1)})^\top]^\top, \tag{28}$$

thus,

$$\nabla_{\tilde{\mathbf{W}}_q}\hat{f}(\tilde{\mathbf{W}}_q^{(t+1)}) - \nabla_{\tilde{\mathbf{W}}_q}\hat{f}(\tilde{\mathbf{W}}_q^{(t)}) = -\mathbf{M}^\top(\mathbf{V}^{(t+1)} - \mathbf{V}^{(t)}). \tag{29}$$

Taking the Frobenius norm of the above equation and applying the L_1-smoothness property yields:

$$(1+\epsilon^2)\|\mathbf{V}^{(t+1)} - \mathbf{V}^{(t)}\|_F^2 \le L_1^2\|\tilde{\mathbf{W}}_q^{(t+1)} - \tilde{\mathbf{W}}_q^{(t)}\|_F^2. \tag{30}$$

Also, for updating of $\mathbf{V}^{(t)}$ we can obtain:

$$\|\mathbf{D}^{(t+1)} - \mathbf{D}^{(t)}\|_F \le \|\mathbf{M}(\tilde{\mathbf{W}}_q^{(t+1)} - \tilde{\mathbf{W}}_q^{(t)})\|_F + \|\frac{1}{\rho}(\mathbf{V}^{(t+1)} - \mathbf{V}^{(t)})\|_F + \|\frac{1}{\rho}(\mathbf{V}^{(t)} - \mathbf{V}^{(t-1)})\|_F \tag{31}$$

Furthermore, we decompose the difference in the augmented Lagrangian between consecutive iterations into three terms:

$$\begin{aligned}&L_\epsilon(\tilde{\mathbf{W}}_q^{(t+1)}, \mathbf{D}^{(t+1)}, \mathbf{V}^{(t+1)}) - L_\epsilon(\tilde{\mathbf{W}}_q^{(t)}, \mathbf{D}^{(t)}, \mathbf{V}^{(t)})\\ =&\underbrace{[L_\epsilon(\tilde{\mathbf{W}}_q^{(t+1)}, \mathbf{D}^{(t+1)}, \mathbf{V}^{(t+1)}) - L_\epsilon(\tilde{\mathbf{W}}_q^{(t+1)}, \mathbf{D}^{(t+1)}, \mathbf{V}^{(t)})]}_{\text{(I)}}\\ &+\underbrace{[L_\epsilon(\tilde{\mathbf{W}}_q^{(t+1)}, \mathbf{D}^{(t+1)}, \mathbf{V}^{(t)}) - L_\epsilon(\tilde{\mathbf{W}}_q^{(t)}, \mathbf{D}^{(t+1)}, \mathbf{V}^{(t)})]}_{\text{(II)}}\\ &+\underbrace{[L_\epsilon(\tilde{\mathbf{W}}_q^{(t)}, \mathbf{D}^{(t+1)}, \mathbf{V}^{(t)}) - L_\epsilon(\tilde{\mathbf{W}}_q^{(t)}, \mathbf{D}^{(t)}, \mathbf{V}^{(t)})]}_{\text{(III)}}.\end{aligned} \tag{32}$$

For term (I), utilizing the dual variable update formula yields:

$$\text{(I)} = \langle \mathbf{V}^{(t+1)} - \mathbf{V}^{(t)}, \mathbf{M}\tilde{\mathbf{W}}_q^{(t+1)} - \mathbf{D}^{(t+1)} \rangle = \frac{1}{\rho}\|\mathbf{V}^{(t+1)} - \mathbf{V}^{(t)}\|_F^2. \tag{33}$$

For term (II), $\tilde{\mathbf{W}}_q^{(t+1)}$ constitutes the minimizer of $L_\epsilon(\tilde{\mathbf{W}}, \mathbf{D}^{(t+1)}, \mathbf{V}^{(t)})$. By the strong convexity of $L_\epsilon(\cdot)$, it follows that:

$$\text{(II)} \leq -\mu\|\tilde{\mathbf{W}}_q^{(t+1)} - \tilde{\mathbf{W}}_q^{(t)}\|_F^2. \tag{34}$$

For term (III), the optimality condition of the $\mathbf{D}$-update directly gives:

$$\text{(III)} \leq 0. \tag{35}$$

Finally, by Eq.(33),(34),(35),(28),(31) and assumption in Eq.(26), we can derive Eq.(25).

This completes the proof. □

Lemma 3. *Under the conditions of Lemma 2, the sequence* $\{(\tilde{\mathbf{W}}_q^{(t)}, \mathbf{D}^{(t)}, \mathbf{V}^{(t)})\}$ *is bounded. Moreover, there exists a cluster point* $(\tilde{\mathbf{W}}_q^*, \mathbf{D}^*, \mathbf{V}^*)$ *such that:*

$$\lim_{t\to+\infty} L_\epsilon(\tilde{\mathbf{W}}_q^{(t)}, \mathbf{D}^{(t)}, \mathbf{V}^{(t)}) = L_\epsilon(\tilde{\mathbf{W}}_q^*, \mathbf{D}^*, \mathbf{V}^*). \tag{36}$$

Proof. First, Lemma 2 implies that the sequence $\{L_\epsilon(\tilde{\mathbf{W}}_q^{(t)}, \mathbf{D}^{(t)}, \mathbf{V}^{(t)})\}$ is monotonically decreasing with a finite initial value, hence:

$$+\infty > L_\epsilon(\tilde{\mathbf{W}}_q^{(0)}, \mathbf{D}^{(0)}, \mathbf{V}^{(0)}) \geq L_\epsilon(\tilde{\mathbf{W}}_q^{(t)}, \mathbf{D}^{(t)}, \mathbf{V}^{(t)}). \tag{37}$$

According to the definition of the augmented Lagrangian Eq.(11) and Eq.(28), when ρ satisfies Eq.(26), there exists constant $c_2 > 0$ such that:

$$\begin{aligned}
+\infty &> L_\epsilon(\tilde{\mathbf{W}}_q^{(0)}, \mathbf{D}^{(0)}, \mathbf{V}^{(0)}) \\
&\geq L_\epsilon(\tilde{\mathbf{W}}_q^{(t)}, \mathbf{D}^{(t)}, \mathbf{V}^{(t)}) \\
&= \hat{f}(\tilde{\mathbf{W}}_q^{(t)}) + I_{\mathcal{A}}(\mathbf{D}^{(t)}) + \langle \mathbf{V}^{(t)}, \mathbf{M}\tilde{\mathbf{W}}_q^{(t)} - \mathbf{D}^{(t)} \rangle + \frac{\rho}{2}\|\mathbf{M}\tilde{\mathbf{W}}_q^{(t)} - \mathbf{D}^{(t)}\|_F^2 \\
&= \hat{f}(\tilde{\mathbf{W}}_q^{(t)}) + I_{\mathcal{A}}(\mathbf{D}^{(t)}) + \frac{\rho}{2}\left\|\mathbf{M}\tilde{\mathbf{W}}_q^{(t)} - \mathbf{D}^{(t)} + \frac{1}{\rho}\mathbf{V}^{(t)}\right\|_F^2 - \frac{1}{2\rho}\|\mathbf{V}^{(t)}\|_F^2 \\
&\geq \hat{f}(\tilde{\mathbf{W}}_q^{(t)}) - \frac{1}{2\rho}\|\mathbf{V}^{(t)}\|_F^2 \\
&= \hat{f}(\tilde{\mathbf{W}}_q^{(t)}) - \frac{1}{2\rho(1+\epsilon^2)}\|\nabla_{\tilde{\mathbf{W}}_q}\hat{f}(\tilde{\mathbf{W}}_q^{(t)})\|_F^2 \\
&\geq c_2\|\tilde{\mathbf{W}}_q^{(t)}\|_F^2.
\end{aligned} \tag{38}$$

Since $\hat{f}$ is a quadratic function that is μ-strongly convex and L_1-smooth, and $\|\mathbf{V}^{(t)}\|_F^2 = \frac{1}{1+\epsilon^2}\|\nabla_{\tilde{\mathbf{W}}_q}\hat{f}(\tilde{\mathbf{W}}_q^{(t)})\|_F^2$ by Eq.(28), for ρ satisfying $\rho(1+\epsilon^2) > \frac{L_1^2}{\mu} \geq L_1$, the Hessian of the combined quadratic form $\hat{f}(\tilde{\mathbf{W}}_q) - \frac{1}{2\rho(1+\epsilon^2)}\|\nabla_{\tilde{\mathbf{W}}_q}\hat{f}(\tilde{\mathbf{W}}_q)\|_F^2$ remains uniformly positive definite for all t. Hence, there exists a constant $c_2 > 0$ independent of t such that the lower bound holds. Finally, this indicates that the sequences $\{\|\tilde{\mathbf{W}}_q^{(t)}\|_F^2\}$ and $\{L_\epsilon(\tilde{\mathbf{W}}_q^{(t)}, \mathbf{D}^{(t)}, \mathbf{V}^{(t)})\}$ are bounded.

Hence, the sequence $\{L_\epsilon(\tilde{\mathbf{W}}_q^{(t)}, \mathbf{D}^{(t)}, \mathbf{V}^{(t)})\}$ converges to a finite limit. The descent inequality Eq.(25) directly forces

$$\lim_{t\to\infty}\|\tilde{\mathbf{W}}_q^{(t+1)} - \tilde{\mathbf{W}}_q^{(t)}\|_F = 0, \quad \lim_{t\to\infty}\|\mathbf{D}^{(t+1)} - \mathbf{D}^{(t)}\|_F = 0, \quad \lim_{t\to\infty}\|\mathbf{V}^{(t+1)} - \mathbf{V}^{(t)}\|_F = 0. \tag{39}$$

Furthermore, from the dual variable update rule (the third equation in Eq.(24)):

$$\mathbf{V}^{(t+1)} = \mathbf{V}^{(t)} + \rho(\mathbf{M}\tilde{\mathbf{W}}_q^{(t+1)} - \mathbf{D}^{(t+1)}), \tag{40}$$

which can be equivalently rewritten as:

$$\mathbf{D}^{(t+1)} = \mathbf{M}\tilde{\mathbf{W}}_q^{(t+1)} - \frac{1}{\rho}(\mathbf{V}^{(t+1)} - \mathbf{V}^{(t)}). \tag{41}$$

Given the boundedness of $\{\tilde{\mathbf{W}}_q^{(t)}\}$ and Eq.(28), the sequence $\{\mathbf{V}^{(t)}\}$ is also bounded, which consequently implies the boundedness of $\{\mathbf{D}^{(t)}\}$.

In summary, the sequence $\{(\tilde{\mathbf{W}}_q^{(t)}, \mathbf{D}^{(t)}, \mathbf{V}^{(t)})\}$ is bounded. By the Bolzano-Weierstrass theorem, we can extract a convergent subsequence $\{t_k\}$ with

$$(\tilde{\mathbf{W}}_q^{(t_k)}, \mathbf{D}^{(t_k)}, \mathbf{V}^{(t_k)}) \to (\tilde{\mathbf{W}}_q^{*}, \mathbf{D}^{*}, \mathbf{V}^{*}) \tag{42}$$

as $k \to \infty$.

Finally, by leveraging the convergence of the augmented Lagrangian function values, together with the properties of subsequence limits, lower semicontinuity, and continuity on its domain, we obtain

$$\begin{aligned} \lim_{t\to\infty} L_\epsilon(\tilde{\mathbf{W}}_q^{(t)}, \mathbf{D}^{(t)}, \mathbf{V}^{(t)}) &= \lim_{k\to\infty} L_\epsilon(\tilde{\mathbf{W}}_q^{(t_k)}, \mathbf{D}^{(t_k)}, \mathbf{V}^{(t_k)}) \\ &= L_\epsilon(\tilde{\mathbf{W}}_q^{*}, \mathbf{D}^{*}, \mathbf{V}^{*}) > -\infty. \end{aligned} \tag{43}$$

This completes the proof. □

Lemma 4. *Under the conditions of Lemma 2, there exists a subgradient* $\omega^{(t+1)} \in \partial_{(\tilde{\mathbf{W}}_q, \mathbf{D}, \mathbf{V})} L_\epsilon(\tilde{\mathbf{W}}_q^{(t+1)}, \mathbf{D}^{(t+1)}, \mathbf{V}^{(t+1)})$ *and a constant* $d > 0$ *such that:*

$$\left\|\omega^{(t+1)}\right\|_F \le d \left\|\tilde{\mathbf{W}}_q^{(t+1)} - \tilde{\mathbf{W}}_q^{(t)}\right\|_F. \tag{44}$$

Proof. Let the subgradient be denoted by $\omega^{(t+1)} = (\omega_{\tilde{\mathbf{W}}_q}, \omega_{\mathbf{D}}, \omega_{\mathbf{V}}) \in \partial_{(\tilde{\mathbf{W}}_q, \mathbf{D}, \mathbf{V})} L_\epsilon(\tilde{\mathbf{W}}_q^{(t+1)}, \mathbf{D}^{(t+1)}, \mathbf{V}^{(t+1)})$. We estimate the norm of each component separately.

First, consider the $\tilde{\mathbf{W}}_q$ component. According to the update rule in Eq.(24), $\tilde{\mathbf{W}}_q^{(t+1)}$ minimizes $L_\epsilon(\cdot, \mathbf{D}^{(t+1)}, \mathbf{V}^{(t)})$, implying its gradient vanishes. Thus:

$$\begin{aligned} \|\omega_{\tilde{\mathbf{W}}_q}\|_F^2 &= \left\|\nabla_{\tilde{\mathbf{W}}_q} L_\epsilon(\tilde{\mathbf{W}}_q^{(t+1)}, \mathbf{D}^{(t+1)}, \mathbf{V}^{(t+1)}) - \nabla_{\tilde{\mathbf{W}}_q} L_\epsilon(\tilde{\mathbf{W}}_q^{(t+1)}, \mathbf{D}^{(t+1)}, \mathbf{V}^{(t)})\right\|_F^2 \\ &= \left\|\mathbf{M}^\top(\mathbf{V}^{(t+1)} - \mathbf{V}^{(t)})\right\|_F^2 \\ &= (1+\epsilon^2)\|\mathbf{V}^{(t+1)} - \mathbf{V}^{(t)}\|_F^2 \\ &\le L_1^2 \|\tilde{\mathbf{W}}_q^{(t+1)} - \tilde{\mathbf{W}}_q^{(t)}\|_F^2. \end{aligned} \tag{45}$$

Second, for the $\mathbf{V}$ component, the dual update rule directly yields:

$$\omega_{\mathbf{V}} = \nabla_{\mathbf{V}} L_\epsilon = \mathbf{M}\tilde{\mathbf{W}}_q^{(t+1)} - \mathbf{D}^{(t+1)} = \frac{1}{\rho}(\mathbf{V}^{(t+1)} - \mathbf{V}^{(t)}). \tag{46}$$

Finally, regarding the $\mathbf{D}$ component, we derive its update by projection onto $\mathcal{A}$. The optimality condition for $\mathbf{D}^{(t+1)}$ implies

$$0 \in \partial_{\mathbf{D}} I_{\mathcal{A}}(\mathbf{D}^{(t+1)}) - \mathbf{V}^{(t)} + \rho(\mathbf{D}^{(t+1)} - \mathbf{M}\tilde{\mathbf{W}}_q^{(t)}). \tag{47}$$

Meanwhile, the subgradient set at $(\tilde{\mathbf{W}}_q^{(t+1)}, \mathbf{D}^{(t+1)}, \mathbf{V}^{(t+1)})$ is given by:

$$\partial_{\mathbf{D}} L_\epsilon(\tilde{\mathbf{W}}_q^{(t+1)}, \mathbf{D}^{(t+1)}, \mathbf{V}^{(t+1)}) = \partial_{\mathbf{D}} I_{\mathcal{A}}(\mathbf{D}^{(t+1)}) - \mathbf{V}^{(t+1)} + \rho(\mathbf{D}^{(t+1)} - \mathbf{M}\tilde{\mathbf{W}}_q^{(t+1)}).$$

Combining these two relations, we deduce the existence of $\omega_{\mathbf{D}} \in \partial_{\mathbf{D}} L_\epsilon(\tilde{\mathbf{W}}_q^{(t+1)}, \mathbf{D}^{(t+1)}, \mathbf{V}^{(t+1)})$ satisfying:

$$\omega_{\mathbf{D}} = -\left[(\mathbf{V}^{(t+1)} - \mathbf{V}^{(t)}) + \rho\mathbf{M}(\tilde{\mathbf{W}}_q^{(t+1)} - \tilde{\mathbf{W}}_q^{(t)})\right]. \tag{48}$$

Combining Eq.(45),(46),(48),(30), we conclude that there exists a constant $d > 0$, such that $\|\omega^{(t+1)}\|_F \le d\|\tilde{\mathbf{W}}_q^{(t+1)} - \tilde{\mathbf{W}}_q^{(t)}\|_F$.

This completes the proof. □

For notational brevity, we introduce the following compact notation:

$$\mathbf{Z}^{(t)} := (\tilde{\mathbf{W}}_q^{(t)}, \mathbf{D}^{(t)}, \mathbf{V}^{(t)}), \quad \mathbf{Z}^* := (\tilde{\mathbf{W}}_q^*, \mathbf{D}^*, \mathbf{V}^*), \quad L_\epsilon^{(t)} := L_\epsilon(\mathbf{Z}^{(t)}), \quad L_\epsilon^* := L_\epsilon(\mathbf{Z}^*), \tag{49}$$

and define the closed ball $\mathcal{B}(\mathbf{Z}^*, r) := \{\mathbf{X} : \|\mathbf{X} - \mathbf{Z}^*\|_F \leq r\}$. Applying Lemma 1 to the augmented Lagrangian L_ϵ at the cluster point $\mathbf{Z}^*$, we select a radius $0 < r < \delta$ such that $\mathcal{B}(\mathbf{Z}^*, \delta) \subset V$, where V denotes the neighborhood specified in the KŁ property. Furthermore, we choose $\eta > 0$ sufficiently small to satisfy $\eta < c_1(\delta - r)^2$.

Lemma 5 (Relation between Iteration Step Size and the KŁ Property [46])**.** *Under the conditions of Lemmas 1–4, when $\mathbf{Z}^{(t)} \in \mathcal{B}(\mathbf{Z}^*, r)$ and $L_\epsilon^* < L_\epsilon^{(t)} < L_\epsilon^* + \eta$, the following holds:*

$$2\|\mathbf{Z}^{(t+1)} - \mathbf{Z}^{(t)}\|_F \leq \|\mathbf{Z}^{(t)} - \mathbf{Z}^{(t-1)}\|_F + \frac{ad}{c_1}\left[(L_\epsilon^{(t)} - L_\epsilon^*)^{1-\theta} - (L_\epsilon^{(t+1)} - L_\epsilon^*)^{1-\theta}\right], \tag{50}$$

where $a > 0$ is the constant from the KŁ property, and d, c_1 are derived from Lemmas 4 and 2, respectively.

Proof. The derivation of this inequality relies on the KŁ inequality combined with the descent estimates from Lemmas 1–4. The detailed proof aligns exactly with Lemma 9 in [46] and is therefore omitted here for conciseness. □

Lemma 6 (Finite Length and Asymptotic Properties of the Sequence [46])**.** *Under the conditions of Lemmas 1–4, there exists an iteration index $t_0 \in \mathbb{N}$ such that for all $t \geq t_0$, the sequence $\{\mathbf{Z}^{(t)}\}$ satisfies:*

$$\begin{aligned} &\|\mathbf{Z}^{(t+1)} - \mathbf{Z}^{(t)}\|_F + \sum_{i=t_0+1}^{t} \|\mathbf{Z}^{(i+1)} - \mathbf{Z}^{(i)}\|_F \\ &\leq \|\mathbf{Z}^{(t_0+1)} - \mathbf{Z}^{(t_0)}\|_F + \frac{ad}{c_1}[(L_\epsilon^{(t_0+1)} - L_\epsilon^*)^{1-\theta} - (L_\epsilon^{(t+1)} - L_\epsilon^*)^{1-\theta}] < +\infty. \end{aligned} \tag{51}$$

In particular, $\sum_{t=1}^{\infty} \|\mathbf{Z}^{(t+1)} - \mathbf{Z}^{(t)}\|_F < +\infty$, and $\lim_{t\to\infty} \mathbf{Z}^{(t)} = \mathbf{Z}^*$.

Proof. The proof of this result follows identically to Lemma 10 in [46] and is thus omitted. □

Proof of Theorem 1: Lemma 6 implies that $\sum_{t=1}^{\infty} \|\mathbf{Z}^{(t+1)} - \mathbf{Z}^{(t)}\|_F < +\infty$, which means that $\mathbf{Z}^{(t)} = (\tilde{\mathbf{W}}_q^{(t)}, \mathbf{D}^{(t)}, \mathbf{V}^{(t)})$ converges to $\mathbf{Z}^* = (\tilde{\mathbf{W}}_q^*, \mathbf{D}^*, \mathbf{V}^*)$. Since the feasible set of the approximated problem is closed, let t tend to infinity in the updating rules, we derive that the limit point $\mathbf{Z}^* = (\tilde{\mathbf{W}}_q^*, \mathbf{D}^*, \mathbf{V}^*)$ satisfies

$$\begin{aligned} \mathbf{D}^* &= \arg\min_{\mathbf{D}} \left\{ I_{\mathcal{A}}(\mathbf{D}) + \left\langle \mathbf{V}^*, \mathbf{M}\tilde{\mathbf{W}}_q^* - \mathbf{D} \right\rangle + \frac{\rho}{2} \left\| \mathbf{M}\tilde{\mathbf{W}}_q^* - \mathbf{D} \right\|_F^2 \right\}, \\ \tilde{\mathbf{W}}_q^* &= \arg\min_{\tilde{\mathbf{W}}_q} \left\{ \hat{f}(\tilde{\mathbf{W}}_q) + \left\langle \mathbf{V}^*, \mathbf{M}\tilde{\mathbf{W}}_q - \mathbf{D}^* \right\rangle + \frac{\rho}{2} \left\| \mathbf{M}\tilde{\mathbf{W}}_q - \mathbf{D}^* \right\|_F^2 \right\}, \\ \mathbf{V}^* &= \mathbf{V}^* + \rho\left(\mathbf{M}\tilde{\mathbf{W}}_q^* - \mathbf{D}^*\right) \implies \mathbf{M}\tilde{\mathbf{W}}_q^* = \mathbf{D}^*. \end{aligned} \tag{52}$$

Therefore, $(\tilde{\mathbf{W}}_q^*, \mathbf{D}^*, \mathbf{V}^*)$ is a feasible point of the approximated problem in Eq.(10). Also, the second equality of Eq.(52) implies that

$$\nabla_{\tilde{\mathbf{W}}_q} \hat{f}(\tilde{\mathbf{W}}_q^*) + \mathbf{M}^\top \mathbf{V}^* + \rho \mathbf{M}^\top \left(\mathbf{M}\tilde{\mathbf{W}}_q^* - \mathbf{D}^*\right) = \mathbf{0},$$

which, combined with the feasibility condition $\mathbf{M}\tilde{\mathbf{W}}_q^* = \mathbf{D}^*$, is equivalent to

$$\nabla_{\tilde{\mathbf{W}}_q} \hat{f}(\tilde{\mathbf{W}}_q^*) + \mathbf{M}^\top \mathbf{V}^* = \mathbf{0}.$$

Similarly, the third equality of Eq.(52) directly implies that

$$\nabla_{\mathbf{V}} L_\epsilon(\tilde{\mathbf{W}}_q^*, \mathbf{D}^*, \mathbf{V}^*) = \mathbf{M}\tilde{\mathbf{W}}_q^* - \mathbf{D}^* = \mathbf{0}.$$

For the $\mathbf{D}$-subproblem, the first equality of Eq.(52) gives the first-order optimality condition

$$\mathbf{0} \in \partial_{\mathbf{D}}[I_{\mathcal{A}}(\mathbf{D}^*) + \left\langle \mathbf{V}^*, \mathbf{M}\tilde{\mathbf{W}}_q^* - \mathbf{D}^* \right\rangle + \frac{\rho}{2} \left\| \mathbf{M}\tilde{\mathbf{W}}_q^* - \mathbf{D}^* \right\|_F^2],$$

which simplifies to

$$\mathbf{0} \in \partial_{\mathbf{D}} L_\epsilon(\tilde{\mathbf{W}}_q^*, \mathbf{D}^*, \mathbf{V}^*).$$

Therefore, $(\tilde{\mathbf{W}}_q^*, \mathbf{D}^*, \mathbf{V}^*)$ is a stationary point of the approximated problem in Eq.(10). □

Proof of Theorem 2: Building upon the preceding lemmas and the theorem, although the proof of Theorem 2 in this paper differs from that in Theorem 2 of [46] only in terms of notation, we still provide an analogous proof below **purely to ensure this paper is self-contained**.

As L_ϵ is a proper closed semi-algebraic function, it admits the KŁ property with $\theta \in [0, 1)$ and $a > 0$ by Lemma 1. The proof is organized into the following three parts.

By Lemma 1 and Lemma 4, we can obtain:

$$\begin{aligned} 1 &\leq a(1-\theta)(L_\epsilon^{(t)} - L_\epsilon^*)^{-\theta} dist(0, \partial_{(\tilde{\mathbf{W}}_\mathbf{q}, \mathbf{D}, \mathbf{V})} L_\epsilon^{(t)}) \\ &\leq a(1-\theta)(L_\epsilon^{(t)} - L_\epsilon^*)^{-\theta} \|\omega^{(t)}\|_F \\ &\leq ad(1-\theta)(L_\epsilon^{(t)} - L_\epsilon^*)^{-\theta} \|\mathbf{Z}^{(t)} - \mathbf{Z}^{(t-1)}\|_F \end{aligned} \tag{53}$$

(i) $\theta = 0$: If $\mathbf{Z}^{(t+1)} \neq \mathbf{Z}^{(t)}$ for all $t \in \mathbb{N}$, we have $\|\mathbf{Z}^{(t)} - \mathbf{Z}^{(t-1)}\|_F \geq \frac{1}{ad}$ for Eq.(53), which contradicts the fact that $\|\mathbf{Z}^{(t)} - \mathbf{Z}^{(t-1)}\|_F \to 0$. Therefore, in this case, $\mathbf{Z}^{(t)}$ converges in a finite number of iterations.

(ii) For $0 < \theta \leq \dfrac{1}{2}$, we sum Eq.(50) over $j = t, t+1, t+2, \ldots$ with $t > t_0$, where the constant t_0 is defined in Lemma 6,

$$\sum_{j=t}^{+\infty} \|\mathbf{Z}^{(j+1)} - \mathbf{Z}^{(j)}\|_F \leq \|\mathbf{Z}^{(t)} - \mathbf{Z}^{(t-1)}\|_F + \frac{ad}{c_1} \left(L_\epsilon^{(t)} - L_\epsilon^*\right)^{1-\theta}. \tag{54}$$

We define $\sigma_j = \sum_{t=j}^{+\infty} \|\mathbf{Z}^{(t+1)} - \mathbf{Z}^{(t)}\|_F$. Then, Eq.(54) becomes

$$\sigma_t \leq \sigma_{t-1} - \sigma_t + \frac{ad}{c_1} \left(L_\epsilon^{(t)} - L_\epsilon^*\right)^{1-\theta}. \tag{55}$$

By combining Eq.(53) and Eq.(55), we obtain

$$\sigma_t \leq \sigma_{t-1} - \sigma_t + c_3(\sigma_{t-1} - \sigma_t)^{\frac{1-\theta}{\theta}}, \tag{56}$$

where $c_3 = \frac{ad}{c_1} \cdot (ad(1-\theta))^{\frac{1-\theta}{\theta}}$. In the case where $0 < \theta \leq \frac{1}{2}$, Eq.(56) leads to

$$\sigma_t \leq c_4(\sigma_{t-1} - \sigma_t), \tag{57}$$

for some constant $c_4 > 0$ and each $t > t_0$ since $\sigma_t \to 0$. Consequently,

$$\sigma_t \leq \frac{c_4}{1+c_4} \sigma_{t-1}, \tag{58}$$

and we can obtain

$$\sigma_t \leq c_5 \cdot \left(\frac{c_4}{1+c_4}\right)^t, \tag{59}$$

for some constant $c_5 > 0$.

(iii) $\frac{1}{2} < \theta < 1$: From Eq.(56), it follows that

$$\sigma_t^{\frac{\theta}{1-\theta}} \leq c_6(\sigma_{t-1} - \sigma_t), \tag{60}$$

for some constant $c_6 > 0$ and each $t > t_0$. Since $\sigma_t \to 0$, there exists some $t_1 > t_0$ and some constant $c_7 > 0$ such that

$$(2^{-\frac{1-2\theta}{\theta}} - 1)\sigma_{t-1}^{\frac{1-2\theta}{1-\theta}} > c_7, \quad \forall t \geq t_1.$$

We define $c_8 = \min\{c_7, \frac{2\theta-1}{2c_6(1-\theta)}\}$. Subsequently, we demonstrate that

$$\sigma_t^{\frac{1-2\theta}{1-\theta}} - \sigma_{t-1}^{\frac{1-2\theta}{1-\theta}} \geq c_8, \quad \forall t \geq t_1. \tag{61}$$

To prove the above inequality, we proceed by considering the following cases:

On the one hand, if $\sigma_t^{-\frac{\theta}{1-\theta}} > 2\sigma_{t-1}^{-\frac{\theta}{1-\theta}}$, we obtain

$$\sigma_t^{\frac{1-2\theta}{1-\theta}} - \sigma_{t-1}^{\frac{1-2\theta}{1-\theta}} \geq (2^{-\frac{1-2\theta}{\theta}} - 1)\sigma_{t-1}^{\frac{1-2\theta}{1-\theta}} > c_7 \geq c_8. \tag{62}$$

On the other hand, if $\sigma_t^{-\frac{\theta}{1-\theta}} \leq 2\sigma_{t-1}^{-\frac{\theta}{1-\theta}}$, by Eq.(60) we have

$$\begin{aligned} 1 &\leq c_6(\sigma_{t-1} - \sigma_t)\sigma_t^{-\frac{\theta}{1-\theta}} \leq 2c_6(\sigma_{t-1} - \sigma_t)\sigma_{t-1}^{-\frac{\theta}{1-\theta}} \\ &\leq 2c_6 \int_{\sigma_t}^{\sigma_{t-1}} \tau^{-\frac{\theta}{1-\theta}} d\tau \leq \frac{2c_6(1-\theta)}{2\theta-1}(\sigma_t^{\frac{1-2\theta}{1-\theta}} - \sigma_{t-1}^{\frac{1-2\theta}{1-\theta}}). \end{aligned} \tag{63}$$

Therefore, we have

$$\sigma_t^{\frac{1-2\theta}{1-\theta}} - \sigma_{t-1}^{\frac{1-2\theta}{1-\theta}} \geq \frac{2\theta-1}{2c_6(1-\theta)} \geq c_8.$$

Then, for each $t \geq t_1$, inequality Eq.(61) always holds, and we can obtain $\sigma_t^{\frac{1-2\theta}{1-\theta}} - \sigma_{t_1}^{\frac{1-2\theta}{1-\theta}} \geq (t-t_1)c_8$ and

$$\sigma_t \leq \left(\sigma_{t_1}^{\frac{1-2\theta}{1-\theta}} + (t-t_1)c_8\right)^{\frac{1-\theta}{1-2\theta}} \leq c_9 t^{-\frac{1-\theta}{2\theta-1}},$$

for some constant $c_9 > 0$ and each $t \geq t_1$.

Combining the cases (i), (ii), (iii), it follows that for any $0 \leq \theta < 1$ there always exist some constants $c_{10}, c_{11} > 0$ and some $t_2 \in \mathbb{N}$ such that

$$\sigma_t \leq c_{10} t^{-c_{11}}, \quad \forall t \geq t_2.$$

Therefore, recalling the definition of σ_t, we obtain

$$\|\mathbf{Z}^{(t)} - \mathbf{Z}^*\|_F \leq \sigma_t \leq c_{10} t^{-c_{11}}, \quad \forall t \geq t_2.$$

Finally, the proof is complete due to $\mathbf{Z}^{(t)} = (\tilde{\mathbf{W}}_q^{(t)}, \mathbf{D}^{(t)}, \mathbf{V}^{(t)})$ and $\mathbf{Z}^* = (\tilde{\mathbf{W}}_q^*, \mathbf{D}^*, \mathbf{V}^*)$. □

F More Experimental Results

F.1 More Experimental Setups

Datasets and Model Configurations. We evaluate all methods using two primary metrics: perplexity (PPL) and zero-shot accuracy. For PPL evaluation, we utilize widely adopted language modeling benchmarks, including Wikitext-2 (Wiki2) [33], C4 [36], the Pile [14], and RedPajama (Red) [42]. For zero-shot evaluation, we assess performance on WinoGrande (Wino) [37], ARC-Challenge (ArcC), ARC-Easy (ArcE) [9], and PiQA [1]. We report the accuracy for WinoGrande and the accuracy norm for the others. To effectively validate QOP, we first conduct experiments on ∼7B-scale models, including Llama-7B [40], Llama-2-7B [41], and OPT-6.7B [51]. We subsequently scale our evaluation to larger models (Llama-2-13B, Llama-30B, and Llama-2-70B) to demonstrate the robustness and scalability of QOP.

Baselines. We compare QOP against established baselines, including Magnitude [17], Wanda [39], SparseGPT [12], and ALPS [32]. A key limitation is that these methods do not natively support quantization-aware pruning. For a fair comparison, we evaluate them using a pruning-quantization pipeline with consistent experimental settings.

F.2 Implementation Details

For Magnitude, Wanda, SparseGPT, and ALPS, we adopt a pruning-quantization pipeline, where pruning is performed on the FP16 LLM followed by quantization. For QOP, we choose $\lambda = 0.01\text{Tr}(\mathbf{X}^\top\mathbf{X})$ and incorporate the ρ update scheme and data pre-processing in [32] as empirical tricks to further enhance performance. For scaling factor $\mathbf{s}$, analogous to $\mathbf{W}_q$, we can derive a closed-form update via a similar procedure that $\mathbf{s}_j = \frac{(\mathbf{X}\mathbf{W}_{q,:j})^\top(\mathbf{X}\hat{\mathbf{W}}_{:j})+\lambda\mathbf{W}_{q,:j}^\top\hat{\mathbf{W}}_{:j}}{\|\mathbf{X}\mathbf{W}_{q,:j}\|_2^2+\lambda\|\mathbf{W}_{q,:j}\|_2^2}$. We update $\mathbf{s}$ once before running Algorithm 1 and keep it fixed throughout the subsequent iterations. Unless otherwise noted, all methods are evaluated using the Pile dataset for calibration, with a model sequence length of 2048 and 128 calibration samples. All reported results are based on a single run. The experiments are conducted on a machine equipped with a cluster of NVIDIA A100-SXM4-80GB GPUs.

F.3 More Results On 7B-Scale Model

To evaluate the effectiveness and scalability of QOP on different models, we conduct experiments on Llama-7B and OPT-6.7B under different sparsity levels. The results are provided in Table 10 and 11.

Table 10: Perplexity and accuracy of quantized Llama-7B (W4A16) at 50%–70% unstructured sparsity. Highlighting indicates the best result, excluding the dense ANN and QANN references.

Sparsity	Method	Zero-Shot Accuracy (%)↑					Perplexity (PPL)↓				
		Wino	ArcC	ArcE	PiQA	Avg.	Wiki2	C4	Pile	Red	Avg.
0%	ANN	67.09	41.38	52.48	77.42	59.59	5.68	7.08	4.58	5.86	5.80
	QANN	63.06	39.51	51.51	75.35	57.36	6.84	8.37	5.25	7.02	6.87
50%	Magnitude	52.57	32.51	42.76	70.02	49.47	23.66	30.81	20.81	25.75	25.26
	Wanda	61.09	35.32	46.46	72.52	53.85	8.63	10.24	6.17	8.53	8.39
	SparseGPT	61.96	36.77	48.36	72.63	54.93	8.59	10.24	5.95	8.55	8.33
	ALPS	61.80	36.60	47.64	72.80	54.71	8.29	10.02	5.82	8.38	8.13
	QOP (Ours)	64.25	38.65	50.97	74.37	**57.06**	7.49	9.12	5.36	7.64	**7.40**
60%	Magnitude	51.07	27.13	34.26	60.17	43.16	>100	>100	>100	>100	>100
	Wanda	56.04	31.23	42.38	67.74	49.35	13.71	15.88	8.66	13.21	12.87
	SparseGPT	58.41	31.31	43.34	67.90	50.24	12.81	14.99	7.67	12.67	12.04
	ALPS	59.84	32.25	43.18	69.53	51.20	11.38	13.06	6.89	11.20	10.63
	QOP (Ours)	62.19	34.04	46.00	70.95	**53.30**	10.07	11.75	6.32	10.02	**9.54**
70%	Magnitude	48.54	25.68	27.61	52.12	38.49	>100	>100	>100	>100	>100
	Wanda	49.64	25.09	29.57	51.41	38.93	>100	>100	53.77	>100	>100
	SparseGPT	50.75	24.83	33.71	55.39	41.17	47.14	48.65	17.31	41.02	38.53
	ALPS	55.01	28.16	38.13	60.94	45.56	24.12	25.63	10.76	22.79	20.83
	QOP (Ours)	54.78	28.50	41.29	62.62	**46.80**	21.11	22.70	9.72	20.06	**18.40**

F.4 Results on Another Representative Model and Setting

Considering that evaluating our method on another mainstream model and a different quantization setting strengthens the evidence for its scalability, we further provide experimental results on Qwen-2-7B [48] with W8A16 quantization in Table 12. The results demonstrate that QOP still outperforms the baselines, further validating its effectiveness across other models and quantization settings.

F.5 More Results on the Unified Quantization and Pruning Baseline Comparison

To provide a broader evaluation across quantization settings and model families, we extend the comparison with OBR beyond the W4A4KV4 and W4A4KV16 settings on Llama-2-7B in the main text. Specifically, we evaluate QOP and OBR on Qwen-2-7B under W4A8KV8 and W4A8KV16 at 50% unstructured sparsity, as shown in Table 13. The results consistently show that QOP outperforms OBR across different quantization settings and model families, demonstrating its effectiveness and generality.

Table 11: Perplexity and accuracy of quantized OPT-6.7B (W4A16) at 50%–70% unstructured sparsity. Highlighting indicates the best result, excluding the dense ANN and QANN references.

Sparsity	Method	Zero-Shot Accuracy (%)↑					Perplexity (PPL)↓				
		Wino	ArcC	ArcE	PiQA	Avg.	Wiki2	C4	Pile	Red	Avg.
0%	ANN	65.19	34.64	60.14	76.50	59.12	10.86	11.74	5.93	9.36	9.47
	QANN	59.67	32.94	55.93	75.95	56.12	13.82	16.96	8.54	13.43	13.19
50%	Magnitude	49.72	23.98	30.13	55.22	39.76	>100	>100	>100	>100	>100
	Wanda	52.88	24.57	46.13	67.25	47.71	61.97	>100	55.95	>100	>100
	SparseGPT	57.54	30.72	54.00	72.36	53.66	21.92	26.65	11.36	19.83	19.94
	ALPS	61.88	30.97	54.34	72.63	54.96	18.24	22.43	10.15	17.46	17.07
	QOP (Ours)	62.51	31.83	57.32	74.37	**56.51**	12.41	13.01	6.44	10.32	**10.55**
60%	Magnitude	49.72	25.77	27.40	50.38	38.32	>100	>100	>100	>100	>100
	Wanda	51.93	23.21	36.36	59.58	42.77	>100	>100	>100	>100	>100
	SparseGPT	53.51	26.88	48.32	69.21	49.48	31.89	40.20	14.59	30.17	29.21
	ALPS	56.75	29.86	51.22	69.64	51.87	27.37	36.78	13.71	28.05	26.48
	QOP (Ours)	60.77	30.97	54.67	72.63	**54.76**	14.43	14.79	6.90	11.71	**11.96**
70%	Magnitude	49.72	26.88	25.63	48.53	37.69	>100	>100	>100	>100	>100
	Wanda	50.59	23.98	29.88	52.56	39.25	>100	>100	>100	>100	>100
	SparseGPT	48.54	21.25	36.45	58.05	41.07	>100	>100	>100	>100	>100
	ALPS	49.41	24.74	44.32	63.55	45.51	>100	94.00	25.11	68.88	72.43
	QOP (Ours)	56.83	28.58	50.38	67.73	**50.88**	22.59	22.14	8.65	17.49	**17.72**

Table 12: Perplexity and accuracy of quantized Qwen-2-7B (W8A16) at 60% unstructured sparsity. Highlighting indicates the best result, excluding the dense ANN and QANN references.

Method	Zero-Shot Accuracy (%)↑					Perplexity (PPL)↓				
	Wino	ArcC	ArcE	PiQA	Avg.	Wiki2	C4	Pile	Red	Avg.
ANN	72.30	49.91	74.58	81.07	69.47	7.14	9.90	5.12	8.52	7.67
QANN	71.74	50.26	73.82	80.58	69.10	7.18	9.96	5.14	8.57	7.71
Magnitude	48.22	23.72	33.84	50.44	39.06	>100	>100	>100	>100	>100
Wanda	61.01	37.03	61.36	70.62	57.51	32.59	37.56	15.69	34.79	30.16
SparseGPT	67.56	41.38	66.62	74.21	62.44	11.68	15.27	6.92	13.42	11.82
ALPS	68.11	42.66	68.06	74.43	63.32	10.98	14.46	6.59	12.70	11.18
QOP (Ours)	69.38	40.96	68.94	75.03	**63.58**	10.77	14.37	6.58	12.63	**11.09**

Table 13: Comparisons with unified quantization and pruning baseline OBR on Qwen-2-7B at 50% unstructured sparsity. Following OBR, we use the Wiki2 dataset for calibration.

Bits	Method	Wino	ArcC	ArcE	PiQA	HellaS	BoolQ	Avg.	Wiki2
W4A8 KV8	OBR_RTN	68.82	45.48	74.03	76.93	69.05	80.00	69.05	8.35
	OBR_GPTQ	69.06	45.22	74.07	77.64	68.77	75.26	68.34	8.25
	QOP (Ours)	70.56	46.76	74.20	77.86	70.61	79.85	**69.97**	**7.96**
W4A8 KV16	OBR_RTN	69.14	45.48	73.74	76.82	68.96	80.15	69.05	8.32
	OBR_GPTQ	70.01	45.56	73.99	77.37	68.76	75.29	68.50	8.23
	QOP (Ours)	70.56	46.93	73.86	78.18	70.56	79.88	**70.00**	**7.94**

F.6 Results on N:M Sparsity

We further validate the effectiveness of QOP under N:M semi-structured sparsity pattern. By modifying the solution of the $\mathbf{D}$-update to satisfy the requirement of N:M sparsity, QOP can be extended to the N:M sparsity pattern. Specifically, similar to [32], we divide the input to the projection operator into multiple groups of size M, and perform the original projection independently within each group. Table 14 reports results for pruning the W4A16-quantized version of Llama-2-7B under 2:4 and 4:8 sparsity settings. Despite the increased complexity of N:M sparsity, QOP still outperforms other methods in terms of both average PPL and average accuracy. These results demonstrate that our method remains effective when extended from unstructured pruning to N:M semi-structured pruning.

Table 14: Perplexity and accuracy of quantized Llama-2-7B (W4A16) at $2:4$ and $4:8$ semi-structured sparsity.

Sparsity	Method	Zero-Shot Accuracy (%)↑					Perplexity (PPL)↓				
		Wino	ArcC	ArcE	PiQA	Avg.	Wiki2	C4	Pile	Red	Avg.
$2:4$	Magnitude	57.14	30.38	44.15	68.61	50.07	>100	>100	90.22	>100	>100
	Wanda	55.49	29.01	42.38	67.57	48.61	18.85	21.88	11.12	18.07	17.48
	SparseGPT	60.62	31.40	46.38	68.77	51.79	15.23	17.57	8.43	14.75	14.00
	ALPS	60.85	32.08	43.69	69.04	51.42	13.39	15.87	7.60	13.53	12.60
	QOP (Ours)	61.01	31.74	46.34	68.28	**51.84**	12.82	14.25	7.10	12.18	**11.59**
$4:8$	Magnitude	55.56	32.59	46.55	70.57	51.32	30.26	60.02	31.53	49.30	42.78
	Wanda	60.03	32.51	45.20	71.71	52.36	12.04	14.42	7.98	11.91	11.59
	SparseGPT	61.17	33.96	47.39	71.38	53.48	10.87	13.10	6.87	10.91	10.44
	ALPS	61.80	34.90	46.17	71.33	53.55	10.22	12.43	6.55	10.43	9.91
	QOP (Ours)	62.83	34.81	47.14	71.22	**54.00**	9.76	11.16	6.22	9.41	**9.14**

F.7 Results on Larger-Scale Model Pruning

We assess the scalability of QOP by extending our evaluation to larger models, including Llama-2-13B, Llama-30B, and Llama-2-70B. We prune the models under the W4A16 quantization setting. The results are reported in Table 15, Table 16, and Table 17 respectively. These results indicate that QOP outperforms other baseline methods, thereby confirming its scalability.

Table 15: Perplexity and accuracy of quantized Llama-2-13B (W4A16) at 50%–70% unstructured sparsity. Highlighting indicates the best result, excluding the dense ANN and QANN references.

Sparsity	Method	Zero-Shot Accuracy (%)↑					Perplexity (PPL)↓				
		Wino	ArcC	ArcE	PiQA	Avg.	Wiki2	C4	Pile	Red	Avg.
0%	ANN	69.61	44.28	57.95	79.05	62.72	4.88	6.47	4.14	5.19	5.17
	QANN	68.35	43.34	57.62	78.35	61.92	5.40	7.01	4.46	5.68	5.64
50%	Magnitude	57.14	36.95	46.17	73.07	53.33	7.50	9.24	5.70	7.60	7.51
	Wanda	66.06	40.36	54.17	76.88	59.37	6.75	8.76	5.22	7.17	6.98
	SparseGPT	67.09	39.68	54.12	76.12	59.25	6.76	8.67	5.09	7.11	6.91
	ALPS	66.38	41.38	55.35	75.79	59.73	6.60	8.54	4.99	7.01	6.79
	QOP (Ours)	69.22	43.43	56.36	76.99	**61.50**	6.17	8.02	4.74	6.58	**6.38**
60%	Magnitude	53.35	29.61	37.12	64.04	46.03	14.95	15.25	10.05	12.87	13.28
	Wanda	59.59	34.04	48.19	72.63	53.61	10.96	13.67	7.30	11.48	10.85
	SparseGPT	63.46	34.98	48.78	72.63	54.96	9.79	12.04	6.35	10.08	9.57
	ALPS	64.17	38.14	51.35	73.39	56.76	8.88	10.99	5.86	9.24	8.74
	QOP (Ours)	67.25	37.71	52.95	74.86	**58.19**	7.93	9.96	5.45	8.31	**7.91**
70%	Magnitude	49.96	26.19	29.38	52.88	39.60	>100	>100	>100	>100	>100
	Wanda	49.41	23.29	28.41	53.97	38.77	68.76	82.23	33.73	70.28	63.75
	SparseGPT	55.17	23.64	39.52	60.61	44.74	30.68	33.93	12.72	28.98	26.58
	ALPS	54.70	28.16	42.89	64.53	47.57	18.45	20.99	9.10	18.78	16.83
	QOP (Ours)	59.12	29.78	45.96	66.92	**50.44**	16.03	18.40	8.24	16.11	**14.70**

F.8 Comparison with Baselines Using a Stronger Quantizer

We further examine whether QOP retains its advantage with a stronger quantizer. Specifically, we use GPTQ [13] and apply the same GPTQ configuration to all methods. For QOP, GPTQ initializes the quantized weights $\mathbf{W}_q$ and scaling factors $\mathbf{s}$. As shown in Table 18, QOP achieves the highest average zero-shot accuracy and the lowest average perplexity, showing that its advantage persists under this stronger quantization setting.

F.9 Wall-Clock Runtime Comparison

We compare the wall-clock runtime of QOP with several baselines on the same NVIDIA A100-SXM4-80GB GPU. As shown in Table 19, QOP takes longer than SparseGPT and ALPS across the four

Table 16: Perplexity and accuracy of quantized Llama-30B (W4A16) at 50%–70% unstructured sparsity. Highlighting indicates the best result, excluding the dense ANN and QANN references.

Sparsity	Method	Zero-Shot Accuracy (%)↑					Perplexity (PPL)↓				
		Wino	ArcC	ArcE	PiQA	Avg.	Wiki2	C4	Pile	Red	Avg.
0%	ANN	72.77	45.39	58.92	80.09	64.29	4.10	5.98	3.98	4.88	4.74
	QANN	71.82	46.50	60.06	79.00	64.35	4.75	6.50	4.27	5.34	5.22
50%	Magnitude	59.91	40.12	51.89	72.69	56.15	9.15	31.83	14.42	19.23	18.66
	Wanda	70.48	43.34	58.50	77.53	62.46	5.85	7.69	4.78	6.32	6.16
	SparseGPT	71.59	43.34	58.67	77.37	62.74	5.89	7.65	4.72	6.41	6.17
	ALPS	71.82	45.65	60.69	77.64	63.95	5.87	7.63	4.68	6.38	6.14
	QOP (Ours)	73.32	46.08	60.06	77.75	**64.30**	5.52	7.22	4.48	6.03	**5.81**
60%	Magnitude	51.07	24.15	32.49	55.93	40.91	57.55	>100	>100	>100	>100
	Wanda	69.30	41.38	54.76	76.39	60.46	7.48	9.47	5.63	7.89	7.62
	SparseGPT	67.72	40.44	53.37	75.79	59.33	7.40	9.47	5.43	7.94	7.56
	ALPS	71.03	43.00	58.16	76.39	62.15	7.25	9.22	5.24	7.73	7.36
	QOP (Ours)	71.67	43.69	58.92	76.39	**62.66**	6.69	8.60	4.95	7.20	**6.86**
70%	Magnitude	50.91	28.16	26.30	50.33	38.93	>100	>100	>100	>100	>100
	Wanda	58.33	30.80	44.74	66.76	50.16	36.76	26.48	13.16	21.94	24.59
	SparseGPT	62.83	33.87	47.98	70.73	53.85	14.81	16.77	7.94	14.39	13.48
	ALPS	65.82	35.75	51.09	71.49	56.04	11.80	13.85	6.87	12.06	11.15
	QOP (Ours)	65.59	36.35	53.03	72.52	**56.87**	11.08	13.25	6.57	11.48	**10.59**

Table 17: Perplexity and accuracy of quantized Llama-2-70B (W4A16) at 60% unstructured sparsity. Highlighting indicates the best result, excluding the dense ANN and QANN references.

Method	Zero-Shot Accuracy (%)↑					Perplexity (PPL)↓				
	Wino	ArcC	ArcE	PiQA	Avg.	Wiki2	C4	Pile	Red	Avg.
ANN	76.87	47.95	59.68	80.85	66.34	3.32	5.52	3.63	4.31	4.20
QANN	74.11	47.01	59.51	80.03	65.17	3.87	5.97	3.87	4.71	4.60
Magnitude	62.27	39.93	53.03	75.79	57.76	9.23	12.13	8.52	9.79	9.92
Wanda	74.03	42.75	55.93	78.89	62.90	5.84	7.94	4.77	6.54	6.27
SparseGPT	72.93	45.31	59.89	77.75	63.97	6.05	8.21	4.72	6.72	6.42
ALPS	73.95	45.14	62.46	78.78	65.08	5.81	7.84	4.53	6.47	6.16
QOP (Ours)	74.90	46.84	61.62	78.07	**65.36**	5.43	7.42	4.35	6.09	**5.82**

Table 18: Perplexity and accuracy of quantized Llama-2-7B (W4A16) at 60% unstructured sparsity using GPTQ. The same GPTQ configuration is applied to all methods.

Method	Zero-Shot Accuracy (%)↑					Perplexity (PPL)↓				
	Wino	ArcC	ArcE	PiQA	Avg.	Wiki2	C4	Pile	Red	Avg.
SparseGPT	61.96	31.74	46.38	67.85	51.98	11.60	14.00	7.07	11.77	11.11
ALPS	64.09	33.45	48.02	70.84	54.10	10.60	12.54	6.45	10.63	10.05
QOP (Ours)	64.09	35.15	49.66	71.55	**55.11**	9.98	11.76	6.16	9.95	**9.46**

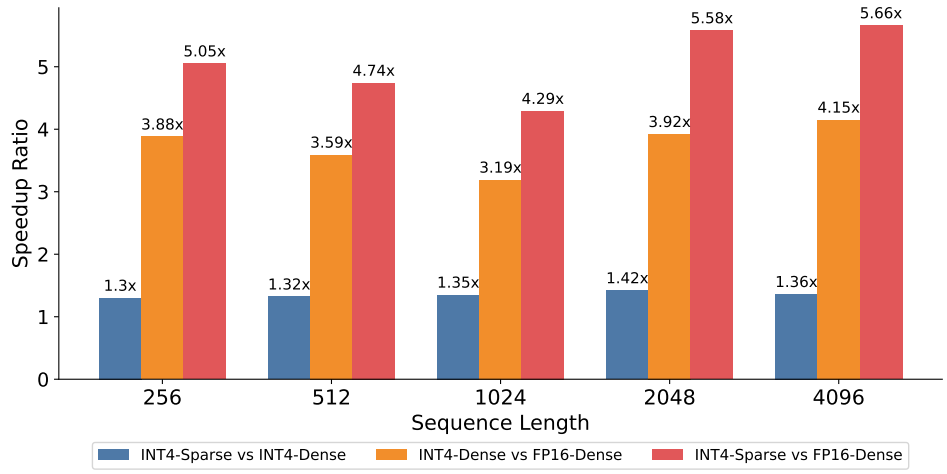
evaluated models. This is expected because QOP addresses a fundamentally more complex sparse-constrained discrete optimization problem. Unlike ALPS, which operates over continuous weights, QOP handles the combinatorial complexity introduced by the discrete quantization constraint and consequently requires more iterations to converge. In comparison with OBR, which also considers both pruning and quantization, QOP runs faster than both OBR variants on Llama-2-7B and Qwen-2-7B, as shown in Table 20. Nevertheless, we emphasize that the model lightweighting procedure is performed only once per model. Once completed, it does not need to be rerun during subsequent inference, so the additional runtime has only a limited impact on the overall use of LLMs.

Table 19: Wall-clock runtime (seconds) compared with SparseGPT and ALPS.

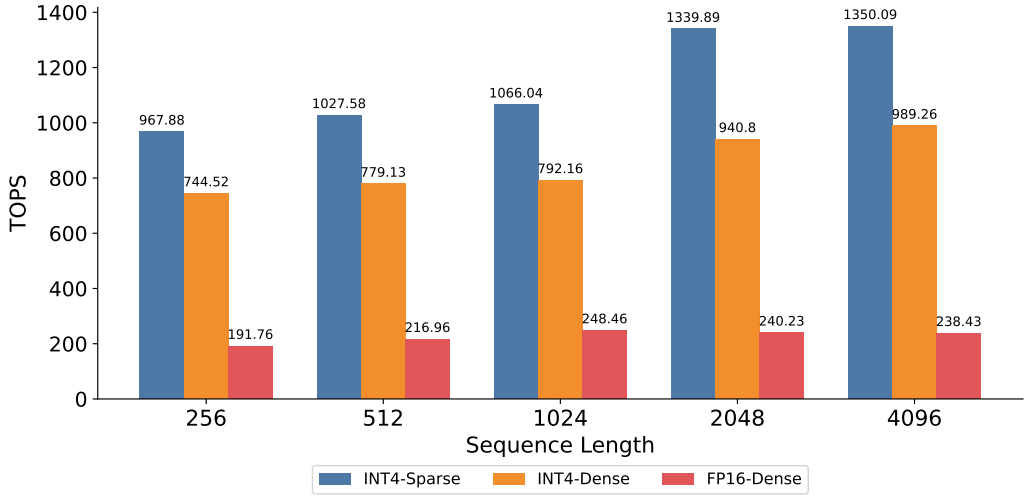
Method	OPT-6.7B	Llama-7B	Llama-2-7B	Llama-2-13B
SparseGPT	618	476	482	847
ALPS	1445	922	937	2029
QOP (Ours)	1578	1629	1563	3692

Table 20: Wall-clock runtime (seconds) compared with OBR.

Method	Llama-2-7B	Qwen-2-7B
OBR_RTN	5752	7684
OBR_GPTQ	5981	7909
QOP (Ours)	1582	2562



(a) Speedup Ratio



(b) TOPS

Figure 3: Comparison of practical speedup ratio (a) and TOPS (b) across different sequence lengths.

F.10 Speedup

Our method employs a combination of standard quantization (e.g., INT4) and sparsification (e.g., 2:4 sparsity) without introducing any auxiliary architectural components into the model or custom operators that might incur additional computational overhead. To ensure the self-contained nature of this work, we use the open-source CUDA kernel implementation and experimental configuration from [15] to reproduce comparative experiments that verify the efficiency under sparsity and low-bit quantization. Specifically, weights are applied 2:4 semi-structured sparsity with INT4 quantization, and activations are quantized to INT4 (INT4-Sparse). We reproduce the speedup ratio and TOPS of INT4-Sparse against INT4-Dense and FP16-Dense baselines on an NVIDIA A100-SXM4-80GB GPU using a 4096×4096 weight matrix and an activation tensor with a batch size of 32. Figure 3 demonstrates that the combination of 2:4 sparsity and INT4 quantization yields substantial improvements in both speedup ratio and TOPS.

G Limitation

While the proposed QOP framework demonstrates efficacy for unstructured pruning, its current formulation is primarily tailored to unstructured pruning and does not straightforwardly extend to structured pruning. Specifically, accommodating structured sparsity constraints would necessitate a non-trivial reformulation of the feasible set and the corresponding projection operator within the $\mathbf{D}$-update step. In addition, QOP relies on per-channel scaling and therefore does not directly support group-wise quantization, which would also require re-deriving the corresponding update rule. We leave the adaptation of QOP to structured pruning and group-wise quantization as important directions for future research.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: The abstract and introduction sections reflect the paper's contributions and scope.

   Guidelines:

   - The answer [N/A] means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A [No] or [N/A] answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: We discuss the limitation in Appendix G.

   Guidelines:

   - The answer [N/A] means that the paper has no limitation while the answer [No] means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [Yes]

Justification: The proofs are provided in Appendix E.

Guidelines:

- The answer [N/A] means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer: [Yes]

   Justification: Code link and experimental details are provided in Section 5 and Appendix F.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - If the paper includes experiments, a [No] answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer: [Yes]

Justification: Data and code are provided in the code link of Section 5.

Guidelines:

- The answer [N/A] means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so [No] is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://neurips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer) necessary to understand the results?

   Answer: [Yes]

   Justification: Experimental details are provided in Appendix F.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [No]

   Justification: We report the average performance across evaluation datasets.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The authors should answer [Yes] if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).
   - It should be clear whether the error bar is the standard deviation or the standard error of the mean.

- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g., negative error rates).
- If error bars are reported in tables or plots, the authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The experiments are conducted on a machine equipped with a cluster of NVIDIA A100-SXM4-80GB GPUs.

   Guidelines:

   - The answer [N/A] means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: The research conforms to the NeurIPS Code of Ethics.

   Guidelines:

   - The answer [N/A] means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer [No], they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [N/A]

    Justification: Our method does not involve any privacy or security concerns.

    Guidelines:

    - The answer [N/A] means that there is no societal impact of the work performed.
    - If the authors answer [N/A] or [No], they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate Deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pre-trained language models, image generators, or scraped datasets)?

    Answer: [N/A]

    Justification: All the datasets and models used in this paper are publicly available.

    Guidelines:

    - The answer [N/A] means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: All the code, datasets, and models used in this paper are publicly available.

    Guidelines:

    - The answer [N/A] means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [N/A]

    Justification: The paper does not release new assets.

    Guidelines:

    - The answer [N/A] means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

Answer: [N/A]

Justification: The core method development in this research does not involve LLMs as any important, original, or non-standard components.

Guidelines:

- The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.