MDSEQ: SEQUENCE-LEVEL VERIFICATION FOR MULTI-DRAFT SPECULATIVE DECODING

Anonymous authors
Paper under double-blind review

ABSTRACT

Speculative decoding has emerged as a pivotal paradigm for accelerating large language model inference through a draft-then-verify mechanism. To further enhance efficiency, modern frameworks have transitioned from single-draft to multi-draft to increase the expected acceptance length. However, existing methods primarily rely on token-level verification, which leads to suboptimal performance. In this work, we propose MDSeq, a novel multi-draft sequence-level verification framework. By carefully designing the acceptance probabilities and residual distributions, MDSeq combines the improved candidate coverage of multi-draft speculation with the benefits of jointly verifying entire sequences. We provide theoretical guarantees that MDSeq is valid and attains a near-optimal expected acceptance length. Extensive experiments across various models, scales and tasks demonstrate that MDSeq consistently outperforms current state-of-the-art methods, providing substantial acceleration gains. Ablation studies examine the impact of various hyperparameters on performance.

1 INTRODUCTION

While large language models (LLMs) have demonstrated remarkable capabilities (Chang et al., 2024; Kasneci et al., 2023), their widespread adoption is fundamentally constrained by the substantial latency of autoregressive generation. As models scale to hundreds of billions of parameters, the generation process—which requires a full forward pass for every single token—becomes a critical bottleneck (Sheng et al., 2023; Zhou et al., 2024). To mitigate this, speculative decoding (SD) (Chen et al., 2023; Leviathan et al., 2023; Xia et al., 2023) has emerged as a pivotal valid acceleration paradigm. SD employs a lightweight drafter to generate a sequence of candidate tokens, which are subsequently verified in parallel by the target LLM via rejection sampling.

To further enhance the efficiency of SD, recent studies have expanded the search space from a single candidate to multiple candidates to increase the probability of draft token acceptance (Hu et al., 2025; Jeon et al., 2024; Sun et al., 2023; Thomas & Pal, 2026; Yang et al., 2025). Nevertheless, a critical gap remains because current verification protocols predominantly operate at the token level rather than the sequence level. Specifically, when a token is rejected, all subsequent tokens are discarded, even if they are accurate, effectively negating any potential speedup.

Recently, some studies have explored sequence-level verification to address these limitations. Sun et al. (2024b) pioneered the replacement of token-level verification with sequence-level verification. By selecting the longest accepted sub-sequence, this approach enables the acceptance of more tokens. However, it is strictly confined to a single-sequence draft. While Traversal (Weng et al., 2025) attempts to integrate sequence-level verification with multi-draft by introducing a backward verification scheme, the method lacks a rigorous proof of optimality and entails significant engineering complexity. Consequently, there remains a gap for algorithms that integrate multi-candidate drafting with sequence-level verification while offering rigorous theoretical guarantees.

To bridge this gap, we propose MDSeq, a novel **M**ulti-**D**raft **Seq**uence-level verification framework. MDSeq improves the expected acceptance length by coupling the acceptance states of individual draft tokens and considering sub-sequence probabilities. It introduces a global scaling factor to ensure that, with multiple i.i.d. draft sequences, the verified output maintains identical distribution with the target model. Furthermore, this algorithm guarantees an expected acceptance length that achieves the optimal up to a multiplicative constant factor.

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

Our main contributions are summarized as follows:

- **Problem formulation for multi-sequence draft.** We formalize multi-sequence draft verification and characterize its validity condition. We further derive a universal upper bound on the expected acceptance length of any valid verification algorithm.
- **Novel valid algorithm with theoretical guarantees.** We propose MDSeq, a novel valid SD algorithm. We rigorously prove that it preserves the exact target distribution and attains a near-optimal expected acceptance length. To our knowledge, this is the first work to establish formal theoretical guarantees for multi-draft sequence-level verification. Besides, MDSeq is a drafter-agnostic, plug-and-play solution that integrates seamlessly with other popular frameworks, e.g., EAGLE (Li et al., 2024a) by modifying only the verification phase.
- **Extensive empirical validation.** We conduct extensive experiments across two model families at varying model scales on the Spec-Bench benchmark (Xia et al., 2024), covering a wide range of datasets. The results consistently demonstrate that MDSeq outperforms existing baseline methods, achieving state-of-the-art acceleration performance.

2 PRELIMINARY

In this section, we provide background knowledge on standard SD and sequence-level verification.

2.1 SPECULATIVE DECODING

SD accelerates LLM inference by replacing sequential token generation with a draft-then-verify pipeline. In each step, a small draft model p first generates a sequence of L tokens $x^L = \{x_1, \ldots, x_L\}$ autoregressively, conditioned on prefix c. Then, the target model q performs a single parallel forward pass to verify these candidates. To ensure the output distribution remains identical to the target model (valid), a rejection sampling scheme is employed. A draft token x_i is accepted with probability:

$$\alpha_i = \min\left(1, \frac{q(x_i|c, x^{i-1})}{p(x_i|c, x^{i-1})}\right).$$

If a token is rejected, all subsequent tokens are discarded, and a replacement token is sampled from a residual distribution: $p_{\text{res}}(x) = \text{norm}(\max(0, q(x) - p(x)))$. If all L tokens are accepted, an additional token is sampled from the target distribution $q(\cdot|c, x^L)$. This paradigm significantly improves throughput by generating multiple tokens per target model forward pass.

2.2 SEQUENCE-LEVEL VERIFICATION

In the standard SD framework (Leviathan et al., 2023), the verification of a draft sequence is sub-optimal as it treats each token independently and fails to consider the joint probability distribution over the entire block. To further improve the acceleration, Sun et al. (2024b) proposed greedy block verification, which shifts the verification logic from the token level to the sequence level. The core objective is to maximize the expected acceptance length in a single iteration by jointly verifying the draft tokens. Specifically, the algorithm maintains a recursive acceptance probability $\min\{1, \tilde{\alpha}_i\}$ for each sub-block x^i:

$$\tilde{\alpha}_i = \tilde{\alpha}_{i-1} \cdot \frac{q(x_i|c, x^{i-1})}{p(x_i|c, x^{i-1})}, \quad \text{with } \tilde{\alpha}_0 = 1.$$

Unlike token-level verification, greedy block verification achieves a higher expected acceptance length by coordinating randomness across the entire block. To preserve the identical distribution, the algorithm incorporates a distribution modification step (Algorithm 5 in Sun et al. (2024b)). Greedy block verification achieves the optimal expected acceptance length within a single SD iteration.

3 PROBLEM FORMULATION FOR MULTI-SEQUENCE DRAFT VERIFICATION

In this section, we formalize the multi-sequence draft SD problem and derive a universal upper bound on the expected acceptance length achievable by any valid verification algorithm. We consider a two-stage framework consisting of multi-sequence draft construction and verification. Following

prior multi-draft approaches (Hu et al., 2025; Jeon et al., 2024; Sun et al., 2023; Thomas & Pal, 2026; Yang et al., 2025), we assume that candidate sequences are sampled i.i.d. from the draft distribution.

Definition 1 (Multi-sequence draft)**.** *Given a prefix c and a draft distribution p, a multi-sequence draft $\mathcal{T}$ of length L is a set of K i.i.d. candidate sequences $\{x_1^L, \ldots, x_K^L\}$. For any sequence $k \in \{1, \ldots, K\}$, we denote its prefix of length i as $x_k^i = (x_{k,1}, \ldots, x_{k,i}) \in \mathcal{V}^i$, where x_k^L represents the full sequence and $\mathcal{V}$ denotes the vocabulary. The sequences are generated autoregressively such that for each $x_k^L \in \mathcal{T}$, the joint probability of any prefix x_k^i (for $1 \leq i \leq L$) is given by:*

$$p(x_k^i \mid c) = \prod_{j=1}^{i} p(x_{k,j} \mid c, x_k^{j-1}),$$

where x_k^0 is defined as the empty sequence. In the following, since our discussion often focuses on a specific candidate sequence, we omit the subscript k for brevity (denoting x_k^i as x^i).

We then formalize valid multi-sequence draft verification, requiring the output distribution to be identical to the target distribution.

Definition 2 (Valid multi-sequence draft verification)**.** *A multi-sequence draft verification algorithm `VERIFY` takes the draft $\mathcal{T}$ defined in Definition 1, draft distributions $p(\cdot|x^{i-1}, c)$ for $i \leq L$, and target distributions $q(\cdot|x^{i-1}, c)$ for $i \leq L+1$ as inputs. It outputs an accepted prefix $x^\tau = x_k^\tau$ of some candidate sequence $x_k^L \in \mathcal{T}$, where $\tau \leq L$, together with additional tokens $y^{L+1-\tau}$. The algorithm `VERIFY` is called valid if its output distribution satisfies:*

$$(x^\tau, y^{L+1-\tau}) = \texttt{VERIFY}(\mathcal{T}, p, q) \sim q(\cdot|c),$$

where the right side denotes the joint distribution of an autoregressive sequence generated by the target model q.

We restrict our analysis to algorithms satisfying Definition 2, ensuring validity. For example, when $y_1 \sim p_{\text{res}}$ and $y_2, .., y_{L+1-\tau} \sim q$, it represents the valid verification definition in Sun et al. (2024b). Within this framework, we show that any such multi-sequence draft verification algorithm admits a theoretical upper bound on the expected acceptance length.

Theorem 1 (Upper bound of expected acceptance length)**.** *Let τ denote the acceptance length in a single SD iteration. For any valid multi-sequence draft verification algorithm satisfying Definition 2, and for all p, q, L, and K, the expected acceptance length admits the following upper bound*[1]*:*

$$\mathbb{E}_{\textit{Verify}}[\tau] \leq \sum_{i=1}^{L} \sum_{x^i \in \mathcal{V}^i} \min\{1 - (1 - p(x^i))^K, q(x^i)\}.$$

The upper bound established above reveals that the expected acceptance length scales with both draft length L and sequence count K. Specifically, a longer draft (L) increases the potential length of accepted sequences, while a larger number of sequences (K) further extends it by expanding the candidate space and increasing the path probability, as reflected in $\min\{1 - (1 - p(x^i))^K, q(x^i)\}$. In the single-sequence case ($K = 1$), this bound reduces to the one derived by Sun et al. (2024b). The proof is provided in Appendix D.

4 METHODOLOGY

To combine the benefit of multi-draft and sequence-level verification, we explore the design of a sequence-level verification algorithm to bridge the gap toward the upper bound of the expected acceptance length. In this section, we first introduce our MDSeq algorithm, which achieves a nearly optimal expected acceptance length through the meticulous design of sub-sequence acceptance probabilities and their corresponding residual distributions. We then provide theoretical guarantees for MDSeq. Specifically, we prove its validity in Theorem 2 and establish a lower bound for the expected acceptance length in Theorem 3. The overall pseudocode of MDSeq is provided in Algorithms 1 and 2 with the corresponding SD iterations in Algorithm 3.

[1]For brevity, we omit the conditioning on c in subsequent sections.

4.1 MDSeq: Multi-draft Sequence-level Verification

The MDSeq algorithm begins with a sequence verification phase (Lines 2–22 in Algorithm 1), where the candidate sequences are examined sequentially using the hierarchical acceptance probabilities h_{ik} defined in Eq. 1. Within each candidate sequence, MDSeq evaluates sub-sequences of different lengths and records the longest accepted prefix. If the entire sequence is accepted, MDSeq samples an additional token from the target distribution q and terminates immediately. Otherwise, once a candidate sequence has a nonempty accepted prefix, MDSeq selects this sequence, samples a correction token y from the residual distribution p_{res} in Eq. 3, and returns (x^τ, y). If none of the K candidate sequences has a nonempty accepted prefix, MDSeq instead returns an empty prefix together with a correction token sampled from p_{res}.

Algorithm 1 MDSeq Verification

Require: Multi-sequence draft $\mathcal{T} = \{x_k^L\}_{k=1}^K$; draft distributions $\{p(\cdot \mid x_k^{i-1})\}$; target distributions $\{q(\cdot \mid x_k^{i-1})\}$; division factor $\rho \in [1, K]$.
Ensure: Accepted draft prefix x^τ and correction token y.

1: Compute $\beta_1(\rho) = \sum_{x^1 \in \mathcal{V}^1} \min\left\{\frac{q(x^1)}{\rho}, p(x^1)\right\}$.
2: **for** $k = 1, \ldots, K$ **do**
3: Set $\tau_k \leftarrow 0$ and $x^{\tau_k} \leftarrow \emptyset$.
4: **for** $i = 1, \ldots, L-1$ **do**
5: Compute h_{ik} according to Equation 1.
6: Sample $\eta \sim \mathcal{U}(0,1)$.
7: **if** $\eta < h_{ik}$ **then**
8: $\tau_k \leftarrow i$ and $x^{\tau_k} \leftarrow x_k^i$. ▷ Record the accepted prefix.
9: **end if**
10: **end for**
11: Compute h_{Lk} according to Equation 2.
12: Sample $\eta \sim \mathcal{U}(0,1)$.
13: **if** $\eta < h_{Lk}$ **then**
14: $\tau \leftarrow L$ and $x^\tau \leftarrow x_k^L$.
15: Sample $y \sim q(\cdot \mid x^\tau)$.
16: **return** x^τ, y. ▷ Accept the entire sequence.
17: **else if** $\tau_k \geq 1$ **then**
18: $\tau \leftarrow \tau_k$ and $x^\tau \leftarrow x^{\tau_k}$.
19: Sample $y \sim p_{\text{res}}(\cdot \mid x^\tau)$ according to Equation 3.
20: **return** x^τ, y. ▷ Select the first sequence with a nonempty accepted prefix.
21: **end if**
22: **end for**
23: Set $\tau \leftarrow 0$ and $x^\tau \leftarrow \emptyset$.
24: Sample $y \sim p_{\text{res}}(\cdot)$ according to Equation 3.
25: **return** x^τ, y. ▷ No sequence is accepted.

Algorithm 2 Distribution Modification

Require: Draft distribution p; target distribution q; draft length L; division factor ρ; tokens x^τ, y returned by Algorithm 1.
Ensure: Modified target distribution q_{new}.

1: For every $i \leq L - \tau - 1$ and $z^i = (z_1, \ldots, z_i) \in \mathcal{V}^i$, define

$$q_{\text{new}}\big(z_i \mid x^\tau, y, z^{i-1}\big) = \frac{q(x^\tau, y, z^i) - \frac{1-(1-\beta_1(\rho))^K}{\beta_1(\rho)} \min\left\{\frac{q(x^\tau, y, z^i)}{\rho}, p(x^\tau, y, z^i)\right\}}{q(x^\tau, y, z^{i-1}) - \frac{1-(1-\beta_1(\rho))^K}{\beta_1(\rho)} \sum_{z \in \mathcal{V}} \min\left\{\frac{q(x^\tau, y, z^{i-1}, z)}{\rho}, p(x^\tau, y, z^{i-1}, z)\right\}}.$$

2: For every $i > L - \tau - 1$, define

$$q_{\text{new}}\big(z_i \mid x^\tau, y, z^{i-1}\big) = q\big(z_i \mid x^\tau, y, z^{i-1}\big).$$

3: **return** q_{new}.

Algorithm 3 Speculative Decoding with MDSeq

Require: Prefix c; target model distribution q; draft model distribution p; draft length L; number of draft sequences K; division factor ρ.
Ensure: Decoded sequence.
1: **while** EOS $\notin c$ **do**
2: Sample $\mathcal{T} = \{x_k^L\}_{k=1}^K$ i.i.d. from $p(\cdot \mid c)$. ▷ Obtain a multi-sequence draft.
3: Compute $q(\cdot \mid c, x_k^{i-1})$ for $i = 1, \ldots, L+1$ and $k = 1, \ldots, K$ in parallel. ▷ Parallel scoring.
4: $(x^\tau, y) \leftarrow \text{MDSEQ}(\mathcal{T}, p, q, \rho)$. ▷ Draft verification and correction.
5: $c \leftarrow (c, x^\tau, y)$. ▷ Add decoded tokens to the prefix.
6: $q \leftarrow \text{DISTRIBUTIONMODIFICATION}(p, q, L, \rho, x^\tau, y)$. ▷ Modify target distribution.
7: **end while**
8: **return** c.

Drawing inspiration from greedy block verification (Sun et al., 2024b), MDSeq couples the acceptance decisions of individual tokens within each sequence by jointly considering draft sub-sequences of varying lengths. Specifically, each sub-sequence is accepted with probability h_{ik} as defined in Eq. 1. A fundamental departure from token-level verification is that the verification of a candidate sequence continues when a sub-sequence fails verification, rather than terminating at the first rejection. Consequently, MDSeq can identify a longer accepted prefix even when one of its shorter prefixes fails verification. To ensure validity under multi-sequence drafting, we introduce a scaling parameter $\rho \in [1, K]$ and adjust the acceptance probabilities accordingly.

Algorithm 2 presents the distribution modification following MDSeq verification. Given the accepted prefix x^τ and correction token y, it constructs a modified target distribution for the next $L - \tau - 1$ positions to compensate for the probability mass consumed during verification, while leaving subsequent distributions unchanged. Algorithm 3 delineates the outer loop of the standard SD framework, which requires minimal modifications for our implementation. In the next section, we establish that MDSeq preserves the target distribution and achieves an expected acceptance length lower-bounded by a constant fraction of the optimum.

$$h_{ik} = \frac{\min\{\frac{q(x^i)}{\rho}, p(x^i)\} - \sum_x \min\{\frac{q(x^i,x)}{\rho}, p(x^i,x)\}}{p(x^i) - \sum_x \min\{\frac{q(x^i,x)}{\rho}, p(x^i,x)\}}, \tag{1}$$

$$h_{LK} = \min\{\frac{q(x^L)}{\rho \cdot p(x^L)}, 1\}, \tag{2}$$

$$p_{\text{res}}(y \mid x^\tau) = \frac{q(x^\tau, y) - \frac{1-(1-\beta_1(\rho))^K}{\beta_1(\rho)} \min\left\{\frac{q(x^\tau,y)}{\rho}, p(x^\tau, y)\right\}}{q(x^\tau) - \frac{1-(1-\beta_1(\rho))^K}{\beta_1(\rho)} \sum_{y' \in \mathcal{V}} \min\left\{\frac{q(x^\tau,y')}{\rho}, p(x^\tau, y')\right\}}. \tag{3}$$

4.2 Theoretical Analysis

Theorem 2 (Validity guarantee). *For each depth $i = 1, \ldots, L$, we define the depth-level acceptance probability $\beta_i(\rho) = \sum_{x^i \in \mathcal{V}^i} \min\left\{\frac{q(x^i)}{\rho}, p(x^i)\right\}$, and let ρ^* be the solution to*

$$\rho\beta_1(\rho) = 1 - (1 - \beta_1(\rho))^K .$$

For any $\rho \geq \rho^*$, MDSeq is a valid verification algorithm. Moreover, ρ^* can be computed up to accuracy δ in time $O(|\mathcal{V}| \log((K-1)/\delta))$ by Sun et al. (2023). Specifically, for any prefix c, distributions p and q, draft length L, and multi-sequence draft $\mathcal{T} = \{x_1^L, \ldots, x_K^L\}$, the output $O = (x^\tau, y^{L+1-\tau})$ induced by Algorithms 1 and 2 satisfies

$$O \sim q(\cdot \mid c).$$

Here, x^τ is the accepted draft prefix, y_1 is the correction token sampled from $p_{\text{res}}(\cdot \mid x^\tau)$ as defined in Eq. 3 when $\tau < L$ and from $q(\cdot \mid x^\tau)$ when $\tau = L$, and the remaining tokens $y_2, \ldots, y_{L+1-\tau}$ are generated according to the modified distributions defined in Algorithm 2.

Theorem 2 establishes that MDSeq is a valid multi-sequence draft verification algorithm as per Definition 2. We defer the proof to Appendix E. We next characterize the expected acceptance length of MDSeq under a mild assumption on how the acceptance probability evolves with sequence depth.

Assumption 1 (Relative depth-level acceptance)**.** *We assume that, for every $i = 1, \ldots, L$,*

$$\frac{\beta_i(\rho^*)}{\beta_1(\rho^*)} \geq \frac{\beta_i(K)}{\beta_1(K)}. \tag{4}$$

Note that $\frac{\beta_i(\rho)}{\beta_1(\rho)} = P_\rho(\tau \geq i \mid \tau \geq 1)$. Thus, Assumption 1 requires that, conditioned on accepting the first token, verification with ρ^* is at least as likely to reach depth i as verification with factor K. This condition is mild because the scaling factor is applied uniformly to prefixes at every depth. Since $\rho^* \leq K$, it relaxes the acceptance criterion throughout the sequence hierarchy rather than only at the first token. When draft–target likelihood ratios evolve gradually across adjacent positions, the resulting acceptance gain naturally persists at later depths.

Under Assumption 1, we obtain the following guarantee on the expected acceptance length of MDSeq.

Theorem 3 (Lower bound of expected acceptance length of MDSeq)**.** *Let τ denote the acceptance length in a single SD iteration. When $\rho = \rho^*$, the expected acceptance length of MDSeq satisfies:*

$$\mathbb{E}_{\textit{MDSeq}}[\tau] \geq \left(1 - \left(1 - \frac{1}{K}\right)^K\right) \sum_{i=1}^{L} \sum_{x^i \in \mathcal{V}^i} \min\{1 - (1 - p(x^i))^K, q(x^i)\}.$$

We defer the proof to Appendix F.

Remark 1 (Comparison with prior work)**.** *Notably, MDSeq exploits sequence-level coupling, yielding the term $\min\{1 - (1 - p(x^i))^K, q(x^i)\}$, rather than the product of token-level coupling terms $\prod_{k=1}^{i} \min\{1 - (1 - p(x_k \mid x^{k-1}))^K, q(x_k \mid x^{k-1})\}$. This avoids compounding the coupling loss across token positions. Together with $1 - (1 - \frac{1}{K})^K \geq 1 - e^{-1}$, it guarantees that the expected acceptance length of MDSeq is at least a $(1 - e^{-1})$ fraction of the theoretical upper bound in Theorem 1. Specifically, MDSeq generalizes existing methods, reducing to greedy block verification ($K = 1$) (Sun et al., 2024b) or SpecTr ($L = 1$) (Sun et al., 2023) under specific configurations. Although Traversal (Weng et al., 2025) also examines the entire draft, its theoretical results only prove that the expected acceptance length exceeds token-level verification in the special case of single-sequence draft. Moreover, Traversal incurs substantial computational overhead in practice due to the need for frequent modifications of probability distributions.*

We provide a toy example in Appendix A to illustrate that MDSeq exactly preserves the target distribution while achieving a strictly larger expected acceptance length than both SpecTr and Block Verification under the same setting. This example highlights the benefit of MDSeq in combining multi-draft sampling with sequence-level verification, while also verifying its validity guarantee in Theorem 2 and expected acceptance length guarantee in Theorem 3.

5 Experiments

In this section, we empirically demonstrate the superiority of MDSeq over state-of-the-art verification algorithms. Furthermore, we conduct extensive ablation studies to investigate the impact of key hyperparameters, e.g., draft length L, sequence count K, and temperature T.

5.1 Experimental Setup

Datasets. We evaluate MDSeq on the Spec-Bench dataset (Xia et al., 2024), which is a widely adopted benchmark in the field of SD (Weng et al., 2025; Li et al., 2024a). Spec-Bench comprises 80 instances across each of its six diverse domains: multi-turn conversation (**MT-Bench** (Zheng et al., 2023)), machine translation (**WMT14 DE-EN** (Bojar et al., 2014)), text summarization (**CNN/Daily Mail** (Nallapati et al., 2016)), question answering (**Natural Questions** (Kwiatkowski et al., 2019)), mathematical reasoning (**GSM8K** (Cobbe et al., 2021)), and retrieval-augmented generation (**DPR** (Karpukhin et al., 2020)).

Target and draft models. To demonstrate the plug-and-play nature and ease of integration of our algorithm, we implement it within the EAGLE framework (Li et al., 2024a). Specifically, we employ **EAGLE-Vicuna-7B-v1.3 (0.24B)** (Li et al., 2024a) and **EAGLE-Llama2-chat-7B (0.24B)** (Li et al., 2024a) as the draft models, paired with **Vicuna-7B-v1.3** (Zheng et al., 2023) and **Llama2-chat-7B** (Touvron et al., 2023) as their respective target models. This setup follows the standard evaluation protocols widely adopted in existing works (Li et al., 2024a; Weng et al., 2025; Sun et al., 2024a). We also evaluate across different model scales using **Tiny-Vicuna-1B** (Pan, 2023) as the drafter and **Vicuna-13B-v1.3** (Zheng et al., 2023) as the target model.

Baselines. We compare MDSeq with six representative baselines: **Vanilla SD** (Leviathan et al., 2023), **Block Verification** (Sun et al., 2024b), recursive rejection sampling with replacement (**RRSw**), the verification algorithm used by EAGLE (Li et al., 2024a), **SpecTr** (Sun et al., 2023), **SpecHub** (Sun et al., 2024a), **Traversal** (Weng et al., 2025).

Metrics. We evaluate MDSeq and baselines using two metrics: **block efficiency (BE)** and **wall-clock speedup (WS)**. BE measures the number of tokens generated per SD iteration, reflecting theoretical speedup, while WS captures end-to-end performance, accounting for practical overheads.

Implementation. To ensure a fair comparison, all methods are configured with a fixed draft length of 5. Specifically, vanilla SD and block verification utilize a single-sequence draft, SpecHub employs a binary tree, Traversal adopts the EAGLE sparse tree, SpecTr , RRSw and MDSeq utilize 5 i.i.d. sequences with a total draft node count equivalent to that of the EAGLE sparse tree. All experiments are conducted on a single NVIDIA H20 GPU with the PyTorch backend. To account for the inherent randomness in sampling, we perform three independent runs and report the average results.

5.2 Main Results

Table 1 presents the block efficiency and wall-clock speedup results across the Spec-Bench datasets at temperature $T = 1.0$. We evaluate two target models, Vicuna-7B-v1.3 and Llama2-chat-7B, paired with their respective 0.24B EAGLE draft models.

MDSeq achieves the highest average BE and WS on both model families. On the Vicuna model pair, MDSeq improves the average BE from 2.86 to 3.02 and the average WS from 1.87 to 1.94 over the strongest baselines, corresponding to gains of 5.6% and 3.7%, respectively. On the Llama model pair, MDSeq improves the average BE from 3.04 to 3.10 and the average WS from 1.82 to 1.86, yielding gains of 2.0% and 2.2%, respectively. These results demonstrate that MDSeq provides consistent overall acceleration across model families and task domains. MDSeq also outperforms its two special cases—Block Verification and SpecTr—on average in terms of both BE and WS, demonstrating the benefit of combining multi-draft speculation with sequence-level verification. We also observe that wall-clock speedups are generally lower than block efficiency. This is expected because practical speedup also accounts for the non-negligible costs of draft generation and verification.

As shown in Table 5 of Appendix C, compared with the SpecHub baseline, our method consistently outperforms it in terms of WS, while slightly lagging behind in BE. This discrepancy arises because, at depth 5, the binary tree constructed by SpecHub contains $2^6 - 2 = 62$ draft nodes, which exceeds MDSeq's budget of $K \times L = 10 \times 5 = 50$ nodes. Although this larger draft budget enables SpecHub to achieve a higher BE than MDSeq, it also incurs additional computational overhead, which offsets its advantage in block efficiency and leads to lower wall-clock speedup.

We also provide additional results for Tiny-Vicuna-1B vs Vicuna-13B-v1.3 in Appendix B, further demonstrating that MDSeq's gains generalize across model scales and draft–target configurations.

5.3 Ablation Study

In this section, we investigate the sensitivity of BE and WS to draft length L, sequence count K, and sampling temperature T. These evaluations are conducted using the EAGLE-Vicuna-7B-v1.3 and Vicuna-7B-v1.3 model pair, with the experiments on L and K performed on the GSM8K dataset.

Impact of draft length. We evaluate the impact of varying draft lengths by setting $L \in \{3, 4, 5, 8\}$, while maintaining a fixed sequence count of $K = 5$ and temperature $T = 1.0$. As shown in the left part of Table 2, BE scales positively with draft length, consistent with the lower bound of the

Table 1: BE and WS results on Spec-Bench using EAGLE-Vicuna-7B-v1.3/Vicuna-7B-v1.3 and EAGLE-Llama2-chat-7b/Llama2-chat-7b as the draft/target model pairs. The best results for each dataset are highlighted in bold.

Data	Metric	Vanilla	Block	SpecTr	RRSw	Traversal	MDSeq
Draft: EAGLE-Vicuna-7B-v1.3, Target: Vicuna-7B-v1.3, Temperature=1							
Con.	WS	$1.98_{\pm 0.01}$	$2.02_{\pm 0.01}$	$1.96_{\pm 0.02}$	$2.07_{\pm 0.03}$	$1.74_{\pm 0.01}$	$\mathbf{2.22}_{\pm 0.03}$
	BE	$2.83_{\pm 0.01}$	$2.92_{\pm 0.00}$	$2.96_{\pm 0.02}$	$2.96_{\pm 0.01}$	$3.01_{\pm 0.02}$	$\mathbf{3.36}_{\pm 0.03}$
Math	WS	$1.98_{\pm 0.00}$	$2.08_{\pm 0.02}$	$1.99_{\pm 0.04}$	$2.13_{\pm 0.09}$	$1.77_{\pm 0.02}$	$\mathbf{2.21}_{\pm 0.04}$
	BE	$2.82_{\pm 0.01}$	$2.98_{\pm 0.00}$	$3.18_{\pm 0.03}$	$3.03_{\pm 0.04}$	$3.14_{\pm 0.01}$	$\mathbf{3.40}_{\pm 0.04}$
Trans.	WS	$1.65_{\pm 0.03}$	$1.61_{\pm 0.01}$	$1.72_{\pm 0.00}$	$1.68_{\pm 0.03}$	$1.52_{\pm 0.01}$	$\mathbf{1.74}_{\pm 0.02}$
	BE	$2.33_{\pm 0.01}$	$2.32_{\pm 0.00}$	$2.73_{\pm 0.01}$	$2.64_{\pm 0.02}$	$\mathbf{2.74}_{\pm 0.01}$	$2.69_{\pm 0.01}$
Sum.	WS	$1.67_{\pm 0.03}$	$1.71_{\pm 0.01}$	$1.76_{\pm 0.03}$	$1.82_{\pm 0.03}$	$1.54_{\pm 0.03}$	$\mathbf{1.87}_{\pm 0.03}$
	BE	$2.48_{\pm 0.01}$	$2.56_{\pm 0.00}$	$2.76_{\pm 0.02}$	$2.72_{\pm 0.01}$	$2.80_{\pm 0.01}$	$\mathbf{2.93}_{\pm 0.01}$
QA	WS	$1.64_{\pm 0.02}$	$1.67_{\pm 0.00}$	$1.73_{\pm 0.04}$	$1.75_{\pm 0.10}$	$1.56_{\pm 0.04}$	$\mathbf{1.80}_{\pm 0.04}$
	BE	$2.27_{\pm 0.04}$	$2.36_{\pm 0.00}$	$2.53_{\pm 0.02}$	$2.53_{\pm 0.03}$	$2.65_{\pm 0.06}$	$\mathbf{2.72}_{\pm 0.04}$
RAG	WS	$1.70_{\pm 0.01}$	$1.77_{\pm 0.01}$	$1.74_{\pm 0.01}$	$1.78_{\pm 0.03}$	$1.59_{\pm 0.02}$	$\mathbf{1.80}_{\pm 0.04}$
	BE	$2.49_{\pm 0.06}$	$2.63_{\pm 0.00}$	$2.76_{\pm 0.03}$	$2.71_{\pm 0.05}$	$2.82_{\pm 0.03}$	$\mathbf{3.02}_{\pm 0.03}$
AVG	WS	1.77	1.81	1.82	1.87	1.62	**1.94**
	BE	2.54	2.63	2.82	2.77	2.86	**3.02**
Draft: EAGLE-Llama2-chat-7b, Target: Llama2-chat-7b, Temperature=1							
Con.	WS	$1.94_{\pm 0.03}$	$1.98_{\pm 0.01}$	$1.93_{\pm 0.02}$	$1.98_{\pm 0.02}$	$1.69_{\pm 0.02}$	$\mathbf{2.01}_{\pm 0.03}$
	BE	$2.81_{\pm 0.02}$	$2.81_{\pm 0.00}$	$3.10_{\pm 0.00}$	$3.02_{\pm 0.02}$	$3.03_{\pm 0.01}$	$\mathbf{3.19}_{\pm 0.00}$
Math	WS	$2.01_{\pm 0.02}$	$2.05_{\pm 0.04}$	$2.00_{\pm 0.06}$	$2.01_{\pm 0.02}$	$1.65_{\pm 0.08}$	$\mathbf{2.06}_{\pm 0.01}$
	BE	$3.17_{\pm 0.02}$	$3.12_{\pm 0.03}$	$3.48_{\pm 0.00}$	$3.33_{\pm 0.02}$	$3.37_{\pm 0.01}$	$\mathbf{3.50}_{\pm 0.00}$
Trans.	WS	$1.66_{\pm 0.03}$	$1.75_{\pm 0.03}$	$1.80_{\pm 0.03}$	$1.77_{\pm 0.04}$	$1.57_{\pm 0.04}$	$\mathbf{1.81}_{\pm 0.01}$
	BE	$2.51_{\pm 0.00}$	$2.53_{\pm 0.01}$	$2.90_{\pm 0.06}$	$2.75_{\pm 0.01}$	$\mathbf{2.95}_{\pm 0.02}$	$2.92_{\pm 0.00}$
Sum.	WS	$1.57_{\pm 0.01}$	$1.65_{\pm 0.03}$	$1.71_{\pm 0.03}$	$1.68_{\pm 0.01}$	$1.44_{\pm 0.04}$	$\mathbf{1.74}_{\pm 0.01}$
	BE	$2.51_{\pm 0.01}$	$2.57_{\pm 0.00}$	$2.90_{\pm 0.01}$	$2.75_{\pm 0.00}$	$2.87_{\pm 0.03}$	$\mathbf{2.96}_{\pm 0.00}$
QA	WS	$1.71_{\pm 0.01}$	$1.66_{\pm 0.01}$	$1.66_{\pm 0.03}$	$\mathbf{1.76}_{\pm 0.03}$	$1.58_{\pm 0.01}$	$\mathbf{1.76}_{\pm 0.02}$
	BE	$2.57_{\pm 0.03}$	$2.56_{\pm 0.00}$	$2.82_{\pm 0.03}$	$2.78_{\pm 0.02}$	$2.94_{\pm 0.04}$	$\mathbf{2.94}_{\pm 0.00}$
RAG	WS	$1.67_{\pm 0.00}$	$1.71_{\pm 0.00}$	$1.67_{\pm 0.04}$	$1.74_{\pm 0.01}$	$1.54_{\pm 0.04}$	$\mathbf{1.74}_{\pm 0.01}$
	BE	$2.63_{\pm 0.03}$	$2.65_{\pm 0.00}$	$3.06_{\pm 0.07}$	$2.88_{\pm 0.01}$	$2.90_{\pm 0.01}$	$\mathbf{3.07}_{\pm 0.00}$
AVG	WS	1.76	1.80	1.80	1.82	1.58	**1.86**
	BE	2.70	2.71	3.04	2.92	3.02	**3.10**

expected acceptance length in Theorem 3. Across all draft lengths, MDSeq consistently outperforms the token-level method SpecTr and RRSw in both BE and WS.

Impact of sequence count. We evaluate performance with varying sequence counts $K \in \{1, 3, 5, 7\}$, while fixing the draft length at $L = 5$ and the temperature at $T = 1.0$. As shown in the right columns of Table 2, increasing K generally improves the BE of all methods, while the corresponding WS tends to saturate as drafting and verification overhead increases. These results confirm the benefit of multi-candidate drafting: a larger candidate set improves coverage and acceptance efficiency, although its marginal practical gain diminishes at larger values of K. Across all sequence counts, MDSeq consistently outperforms the token-level method SpecTr and RRSw in both BE and WS.

Impact of temperature. To evaluate the performance across different generation settings, we examine three representative sampling temperatures $T = 0.4, 0.6, 1.0$ with the draft length L and sequence count K both fixed at 5. As shown in Table 3, MDSeq consistently achieves high BE and WS across various temperature settings, demonstrating the robustness of our algorithm.

Table 2: Ablation study on the effects of draft length L and sequence count K. We report WS and BE of SpecTr, RRSw, and MDSeq under different configurations.

	SpecTr		RRSw		MDSeq			SpecTr		RRSw		MDSeq	
L	WS	BE	WS	BE	WS	BE	K	WS	BE	WS	BE	WS	BE
3	1.82	2.76	2.01	2.79	2.07	2.91	1	1.93	2.81	1.93	2.80	2.01	2.92
4	2.01	3.01	2.01	2.93	2.08	3.18	3	1.98	3.03	2.01	3.05	2.09	3.24
5	2.03	3.15	2.02	3.06	2.11	3.40	5	2.03	3.15	2.02	3.06	2.11	3.40
8	1.95	3.24	1.87	3.23	1.96	3.53	7	1.94	3.17	2.03	3.11	2.15	3.46

Table 3: Ablation study of MDSeq under different sampling temperatures across multiple datasets. We report WS and BE on six datasets.

	Con.		Math		QA		RAG		Sum		Trans	
Temp.	WS	BE	WS	BE	WS	BE	WS	BE	WS	BE	WS	BE
0.4	2.00	3.36	1.87	3.27	1.56	2.58	1.62	3.05	1.71	2.92	1.48	2.63
0.6	1.86	3.42	1.82	3.36	1.75	2.78	1.58	2.91	1.65	2.91	1.64	2.69
1.0	2.20	3.36	2.21	3.40	1.80	2.72	1.80	3.02	1.87	2.93	1.74	2.69

6 Related Work

Speculative Decoding accelerates LLM inference by generating draft tokens with a lightweight model and verifying them using a larger target model (Chen et al., 2023; Xia et al., 2023; Leviathan et al., 2023). Existing efforts to improve acceptance efficiency mainly focus on three directions: enhancing draft-target alignment (Li et al., 2024a; Cai et al., 2024; Ankner et al., 2024), optimizing draft topologies (Chen et al., 2024; Li et al., 2024b), and designing effective verification algorithms (Cheng, 2023; Sun et al., 2023; 2024a;b; Weng et al., 2025). Our work focuses on the last direction.

Multi-candidate Verification expands the search space by verifying multiple draft candidates simultaneously. SpecTr (Sun et al., 2023) formulates this process as an optimal transport problem, while SpecInfer (Miao et al., 2024) and its variants (Lin et al., 2024; Jeon et al., 2024; Yang et al., 2025) employ recursive rejection sampling without replacement to avoid redundant candidates. Hybrid approaches such as SpecHub (Sun et al., 2024a) and Greedy Sampling (Hu et al., 2025) further combine deterministic top-K selection with probabilistic sampling to improve efficiency.

Sequence-level Verification validates entire sequences instead of individual tokens to improve acceptance efficiency. Block Verification (Sun et al., 2024b) first demonstrated the superiority of sequence-level schemes, though only in the single-sequence setting. Traversal (Weng et al., 2025) extended this idea to tree-based drafts via backward verification, but lacks rigorous guarantees and incurs substantial overhead from frequent distribution modifications. GBV (Thomas & Pal, 2026) further considers multi-draft verification, yet verifies only a single optimal path, underutilizing the remaining sampled sequences. SpecTr-GBV (Lin et al., 2026) extends sequence-level verification to the multi-draft setting. However, its validity relies on a conditionally independent coupling assumption, which is generally not satisfied in standard LLM decoding settings.

7 Conclusion

In this work, we introduced MDSeq, a novel sequence-level verification framework designed to bridge the efficiency gap in multi-draft speculative decoding. We theoretically formalized the problem for multi-sequence draft verification and established a universal upper bound on the expected acceptance length. By coupling acceptance decisions across token positions and employing a global scaling mechanism, MDSeq provides a rigorous guarantee for valid acceleration while achieving near-optimal expected acceptance length. Our extensive empirical evaluations on Spec-Bench across various model families and scales demonstrate that MDSeq consistently outperforms current state-of-the-art baselines, such as SpecTr and RRSw, across various models and temperatures.

AI USE STATEMENT

AI-based tools were used during the preparation of this manuscript for limited purposes, including language editing, proofreading, and assisting with code implementation. All mathematical derivations, theoretical results, algorithmic designs, and scientific conclusions were developed by the authors. All AI-assisted code was manually reviewed, verified, and validated by the authors, and all experimental implementations and empirical analyses were independently checked by the authors.

REFERENCES

Zachary Ankner, Rishab Parthasarathy, Aniruddha Nrusimha, Christopher Rinard, Jonathan Ragan-Kelley, and William Brandon. Hydra: Sequentially-dependent draft heads for medusa decoding. *arXiv preprint arXiv:2402.05109*, 2024.

Ondřej Bojar, Christian Buck, Christian Federmann, Barry Haddow, Philipp Koehn, Johannes Leveling, Christof Monz, Pavel Pecina, Matt Post, Herve Saint-Amand, et al. Findings of the 2014 workshop on statistical machine translation. In *Proceedings of the ninth workshop on statistical machine translation*, pp. 12–58, 2014.

Tianle Cai, Yuhong Li, Zhengyang Geng, Hongwu Peng, Jason D Lee, Deming Chen, and Tri Dao. Medusa: Simple llm inference acceleration framework with multiple decoding heads. *arXiv preprint arXiv:2401.10774*, 2024.

Yupeng Chang, Xu Wang, Jindong Wang, Yuan Wu, Linyi Yang, Kaijie Zhu, Hao Chen, Xiaoyuan Yi, Cunxiang Wang, Yidong Wang, et al. A survey on evaluation of large language models. *ACM transactions on intelligent systems and technology*, 15(3):1–45, 2024.

Charlie Chen, Sebastian Borgeaud, Geoffrey Irving, Jean-Baptiste Lespiau, Laurent Sifre, and John Jumper. Accelerating large language model decoding with speculative sampling. *arXiv preprint arXiv:2302.01318*, 2023.

Zhuoming Chen, Avner May, Ruslan Svirschevski, Yuhsun Huang, Max Ryabinin, Zhihao Jia, and Beidi Chen. Sequoia: Scalable, robust, and hardware-aware speculative decoding. *arXiv preprint arXiv:2402.12374*, 2024.

Xinhao Cheng. Specinfer: Accelerating generative large language model serving with speculative inference and token tree verification. Master's thesis, Carnegie Mellon University, 2023.

Karl Cobbe, Vineet Kosaraju, Mohammad Bavarian, Mark Chen, Heewoo Jun, Lukasz Kaiser, Matthias Plappert, Jerry Tworek, Jacob Hilton, Reiichiro Nakano, et al. Training verifiers to solve math word problems. *arXiv preprint arXiv:2110.14168*, 2021.

Zhengmian Hu, Tong Zheng, Vignesh Viswanathan, Ziyi Chen, Ryan A Rossi, Yihan Wu, Dinesh Manocha, and Heng Huang. Towards optimal multi-draft speculative decoding. *arXiv preprint arXiv:2502.18779*, 2025.

Wonseok Jeon, Mukul Gagrani, Raghavv Goel, Junyoung Park, Mingu Lee, and Christopher Lott. Recursive speculative decoding: Accelerating llm inference via sampling without replacement. *arXiv preprint arXiv:2402.14160*, 2024.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pp. 6769–6781, 2020.

Enkelejda Kasneci, Kathrin Seßler, Stefan Küchemann, Maria Bannert, Daryna Dementieva, Frank Fischer, Urs Gasser, Georg Groh, Stephan Günnemann, Eyke Hüllermeier, et al. Chatgpt for good? on opportunities and challenges of large language models for education. *Learning and individual differences*, 103:102274, 2023.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, et al. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466, 2019.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. Fast inference from transformers via speculative decoding. In *International Conference on Machine Learning*, pp. 19274–19286. PMLR, 2023.

Yuhui Li, Fangyun Wei, Chao Zhang, and Hongyang Zhang. Eagle: Speculative sampling requires rethinking feature uncertainty. *arXiv preprint arXiv:2401.15077*, 2024a.

Yuhui Li, Fangyun Wei, Chao Zhang, and Hongyang Zhang. Eagle-2: Faster inference of language models with dynamic draft trees. In *Proceedings of the 2024 conference on empirical methods in natural language processing*, pp. 7421–7432, 2024b.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. Awq: Activation-aware weight quantization for on-device llm compression and acceleration. *Proceedings of machine learning and systems*, 6: 87–100, 2024.

Yijun Lin, Jinhao Sheng, Qingyue Cai, and Feng Zhou. Spectr-gbv: Multi-draft block verification accelerating speculative decoding. *arXiv preprint arXiv:2604.25925*, 2026.

Xupeng Miao, Gabriele Oliaro, Zhihao Zhang, Xinhao Cheng, Zeyu Wang, Zhengxin Zhang, Rae Ying Yee Wong, Alan Zhu, Lijie Yang, Xiaoxiang Shi, et al. Specinfer: Accelerating large language model serving with tree-based speculative inference and verification. In *Proceedings of the 29th ACM International Conference on Architectural Support for Programming Languages and Operating Systems, Volume 3*, pp. 932–949, 2024.

Ramesh Nallapati, Bowen Zhou, Cicero Dos Santos, Çağlar Gulçehre, and Bing Xiang. Abstractive text summarization using sequence-to-sequence rnns and beyond. In *Proceedings of the 20th SIGNLL conference on computational natural language learning*, pp. 280–290, 2016.

Jiayi Pan. Tiny-vicuna-1b. `https://huggingface.co/Jiayi-Pan/Tiny-Vicuna-1B`, 2023. Hugging Face model repository.

Ying Sheng, Lianmin Zheng, Binhang Yuan, Zhuohan Li, Max Ryabinin, Beidi Chen, Percy Liang, Christopher Ré, Ion Stoica, and Ce Zhang. Flexgen: High-throughput generative inference of large language models with a single gpu. In *International Conference on Machine Learning*, pp. 31094–31116. PMLR, 2023.

Ryan Sun, Tianyi Zhou, Xun Chen, and Lichao Sun. Spechub: Provable acceleration to multi-draft speculative decoding. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 20620–20641, 2024a.

Ziteng Sun, Ananda Theertha Suresh, Jae Hun Ro, Ahmad Beirami, Himanshu Jain, and Felix Yu. Spectr: Fast speculative decoding via optimal transport. *Advances in Neural Information Processing Systems*, 36:30222–30242, 2023.

Ziteng Sun, Uri Mendlovic, Yaniv Leviathan, Asaf Aharoni, Jae Hun Ro, Ahmad Beirami, and Ananda Theertha Suresh. Block verification accelerates speculative decoding. *arXiv preprint arXiv:2403.10444*, 2024b.

Rahul Thomas and Arka Pal. Greedy multi-path block verification for faster decoding in speculative sampling. *arXiv preprint arXiv:2602.16961*, 2026.

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.

Yepeng Weng, Qiao Hu, Xujie Chen, Li Liu, Dianwen Mei, Huishi Qiu, Jiang Tian, and Zhongchao Shi. Traversal verification for speculative tree decoding. *arXiv preprint arXiv:2505.12398*, 2025.

Heming Xia, Tao Ge, Peiyi Wang, Si-Qing Chen, Furu Wei, and Zhifang Sui. Speculative decoding: Exploiting speculative execution for accelerating seq2seq generation. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 3909–3925, 2023.

Heming Xia, Zhe Yang, Qingxiu Dong, Peiyi Wang, Yongqi Li, Tao Ge, Tianyu Liu, Wenjie Li, and Zhifang Sui. Unlocking efficiency in large language model inference: A comprehensive survey of speculative decoding. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 7655–7671, 2024.

Sen Yang, Shujian Huang, Xinyu Dai, and Jiajun Chen. Multi-candidate speculative decoding. In *CCF International Conference on Natural Language Processing and Chinese Computing*, pp. 335–348. Springer, 2025.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, et al. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623, 2023.

Zixuan Zhou, Xuefei Ning, Ke Hong, Tianyu Fu, Jiaming Xu, Shiyao Li, Yuming Lou, Luning Wang, Zhihang Yuan, Xiuhong Li, et al. A survey on efficient inference for large language models. *arXiv preprint arXiv:2404.14294*, 2024.

A A TOY EXAMPLE OF MDSEQ

We provide a concrete example to illustrate that MDSeq preserves the target distribution while achieving a strictly larger expected acceptance length than SpecTr and Block Verification. All quantities are obtained by directly executing Algorithms 1 and 2, without invoking the theoretical guarantees established for MDSeq.

Setup. Consider a binary vocabulary $\mathcal{V} = \{a, b\}$ and position-independent draft and target distributions

$$p(a) = \frac{2}{3}, \qquad p(b) = \frac{1}{3},$$

and

$$q(a) = \frac{1}{3}, \qquad q(b) = \frac{2}{3}.$$

We draw $K = 2$ independent candidate sequences of length $L = 2$. Define

$$r := \rho^* = \frac{5 + \sqrt{13}}{6}.$$

Since $1 < r < 2$, we have

$$\begin{aligned}\beta(r) &= \min\left\{\frac{q(a)}{r}, p(a)\right\} + \min\left\{\frac{q(b)}{r}, p(b)\right\} \\ &= \frac{1}{3r} + \frac{1}{3} = \frac{r+1}{3r}.\end{aligned}$$

Moreover, r satisfies

$$r\beta(r) = 1 - (1 - \beta(r))^2,$$

and hence

$$\frac{1 - (1 - \beta(r))^2}{\beta(r)} = r.$$

Executing MDSeq verification. For a length-one prefix, the quantities used in Eq. 1 are

$$\min\left\{\frac{q(a)}{r}, p(a)\right\} = \frac{1}{3r}, \qquad \min\left\{\frac{q(b)}{r}, p(b)\right\} = \frac{1}{3}.$$

For the four length-two sequences, we obtain

x^2	aa	ab	ba	bb
$p(x^2)$	$\frac{4}{9}$	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{1}{9}$
$q(x^2)$	$\frac{1}{9}$	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{4}{9}$
$\min\{q(x^2)/r, p(x^2)\}$	$\frac{1}{9r}$	$\frac{2}{9r}$	$\frac{2}{9r}$	$\frac{1}{9}$

Substituting these values into Eq. 1 gives

$$h_1(a) = \frac{\frac{1}{3r} - \left(\frac{1}{9r} + \frac{2}{9r}\right)}{\frac{2}{3} - \left(\frac{1}{9r} + \frac{2}{9r}\right)} = 0,$$

and

$$h_1(b) = \frac{\frac{1}{3} - \left(\frac{2}{9r} + \frac{1}{9}\right)}{\frac{1}{3} - \left(\frac{2}{9r} + \frac{1}{9}\right)} = 1.$$

Thus, a length-one prefix is accepted if and only if its first token is b.

At depth two, Eq. 2 gives

$$\begin{aligned}h_2(aa) &= \frac{q(aa)}{rp(aa)} = \frac{1}{4r}, \\ h_2(ab) &= \frac{q(ab)}{rp(ab)} = \frac{1}{r}, \\ h_2(ba) &= \frac{q(ba)}{rp(ba)} = \frac{1}{r}, \\ h_2(bb) &= \min\left\{\frac{q(bb)}{rp(bb)}, 1\right\} = 1.\end{aligned}$$

Therefore, for one candidate sequence, the probabilities of fully accepting the four possible sequences are

x^2	aa	ab	ba	bb
$p(x^2)h_2(x^2)$	$\frac{1}{9r}$	$\frac{2}{9r}$	$\frac{2}{9r}$	$\frac{1}{9}$

and the probability of full acceptance is

$$P_{\text{full}} = \frac{1}{9r} + \frac{2}{9r} + \frac{2}{9r} + \frac{1}{9} = \frac{r+5}{9r}.$$

The only event in which exactly one token is accepted occurs when the candidate is ba, its first token is accepted, and the entire sequence is rejected. This event has probability

$$P_{\text{partial}} = p(ba)\big(1 - h_2(ba)\big) = \frac{2}{9}\left(1 - \frac{1}{r}\right) = \frac{2(r-1)}{9r}.$$

Consequently, a single candidate has a nonempty accepted prefix with probability

$$P_{\text{full}} + P_{\text{partial}} = \frac{r+1}{3r} = \beta(r).$$

Algorithm 1 examines the two candidates sequentially and selects the first candidate with a nonempty accepted prefix. Therefore,

$$\begin{aligned}\Pr_{\text{MDSeq}}(\tau \geq 1) &= \beta(r) + (1 - \beta(r))\beta(r) \\ &= \beta(r)\big(2 - \beta(r)\big) \\ &= r\beta(r) = \frac{r+1}{3}.\end{aligned}$$

Similarly, a full sequence is accepted either from the first candidate or from the second candidate after the first candidate has no accepted prefix. Thus,

$$\begin{aligned}\Pr_{\text{MDSeq}}(\tau = 2) &= P_{\text{full}} + (1 - \beta(r))P_{\text{full}} \\ &= P_{\text{full}}\big(2 - \beta(r)\big) \\ &= rP_{\text{full}} = \frac{r+5}{9}.\end{aligned}$$

The expected acceptance length is therefore

$$\begin{aligned}\mathbb{E}_{\text{MDSeq}}[\tau] &= \Pr(\tau \geq 1) + \Pr(\tau \geq 2) \\ &= \frac{r+1}{3} + \frac{r+5}{9} \\ &= \frac{4r+8}{9} \\ &= \frac{34 + 2\sqrt{13}}{27} \approx 1.52634.\end{aligned}$$

Verifying losslessness. We next compute the output distribution produced by the same execution. For a single candidate, the accepted probability masses of the first token are

$$\frac{1}{9r} + \frac{2}{9r} = \frac{1}{3r}$$

for a, and

$$\frac{2}{9r} + \frac{1}{9} + \frac{2(r-1)}{9r} = \frac{1}{3}$$

for b. Since the second candidate is examined only when the first candidate has no accepted prefix, the corresponding two-candidate accepted masses are multiplied by

$$1 + (1 - \beta(r)) = 2 - \beta(r) = r.$$

They are consequently

$$\frac{1}{3} \quad \text{for } a, \qquad \frac{r}{3} \quad \text{for } b.$$

Both candidates have empty accepted prefixes with probability

$$(1-\beta(r))^2 = 1 - r\beta(r) = \frac{2-r}{3}.$$

In this event, substituting the distributions into Eq. 3 gives

$$p_{\text{res}}(a) = 0, \qquad p_{\text{res}}(b) = 1.$$

The correction token is therefore deterministically b. Combining the accepted and residual masses yields

$$\Pr(O_1 = a) = \frac{1}{3} = q(a),$$

and

$$\Pr(O_1 = b) = \frac{r}{3} + \frac{2-r}{3} = \frac{2}{3} = q(b).$$

Thus, the first output token follows the target distribution exactly.

For length-two sequences, the probability that MDSeq fully accepts each sequence is its single-candidate acceptance probability multiplied by $2 - \beta(r) = r$:

x^2	aa	ab	ba	bb
$\Pr(O^2 = x^2, \tau = 2)$	$\frac{1}{9}$	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{r}{9}$
$q(x^2)$	$\frac{1}{9}$	$\frac{2}{9}$	$\frac{2}{9}$	$\frac{4}{9}$

The fully accepted sequences already provide the exact target masses for aa, ab, and ba. The missing target mass is concentrated on bb and equals

$$q(bb) - \frac{r}{9} = \frac{4-r}{9}.$$

We now verify that every execution with $\tau < 2$ produces bb. If $\tau = 1$, the selected candidate must be ba. Given the accepted prefix b, direct substitution into Eq. 3 gives

$$p_{\text{res}}(a \mid b) = 0, \qquad p_{\text{res}}(b \mid b) = 1.$$

Hence, the correction token is b. If $\tau = 0$, the first correction token is also b, as shown above, and Algorithm 2 gives

$$q_{\text{new}}(a \mid b) = 0, \qquad q_{\text{new}}(b \mid b) = 1.$$

Thus, every non-full-acceptance execution outputs bb. Its total probability is

$$1 - \Pr(\tau = 2) = 1 - \frac{r+5}{9} = \frac{4-r}{9},$$

which exactly matches the missing target mass. Therefore,

$$\Pr(O^2 = aa) = \frac{1}{9},$$
$$\Pr(O^2 = ab) = \frac{2}{9},$$
$$\Pr(O^2 = ba) = \frac{2}{9},$$
$$\Pr(O^2 = bb) = \frac{r}{9} + \frac{4-r}{9} = \frac{4}{9}.$$

Consequently,

$$O^2 \sim q^2.$$

When the full draft is accepted, Algorithm 1 samples the additional token directly from q. Otherwise, Algorithm 2 restores the original target distribution after completing the modified positions. Consequently, the third token follows q in either case, and hence

$$O^3 \sim q^3.$$

This explicitly verifies the losslessness of MDSeq by following the algorithmic execution.

Comparison with SpecTr. We finally apply SpecTr to the same two candidate sequences. With two candidates, its division factor is also r, and its first-token acceptance probability is

$$\Pr_{\text{SpecTr}}(\tau \geq 1) = r\beta(r) = \frac{r+1}{3} = \frac{11+\sqrt{13}}{18}.$$

Let N denote the number of candidate sequences sharing the accepted first-token prefix. The two mixed ordered draft pairs, (a, b) and (b, a), contain only one candidate consistent with the selected token. Since token b is accepted with probability one, these pairs contribute

$$\Pr(N = 1, \tau \geq 1) = 2p(a)p(b) = \frac{4}{9}.$$

For the pair (a, a), each occurrence of a is accepted with probability

$$\frac{q(a)}{rp(a)} = \frac{1}{2r}.$$

Its contribution to the event $N = 2$ is therefore

$$\frac{4}{9}\left[1 - \left(1 - \frac{1}{2r}\right)^2\right].$$

The pair (b, b) is accepted with probability one and contributes $1/9$. Hence,

$$\begin{aligned}\Pr(N = 2, \tau \geq 1) &= \frac{4}{9}\left[1 - \left(1 - \frac{1}{2r}\right)^2\right] + \frac{1}{9} \\ &= \frac{1}{6} + \frac{\sqrt{13}}{18}.\end{aligned}$$

With one remaining candidate, the next-token acceptance probability is the single-draft overlap

$$\sum_{x \in \mathcal{V}} \min\{p(x), q(x)\} = \frac{2}{3}.$$

With two remaining candidates, the acceptance probability remains $(r + 1)/3$. Therefore,

$$\begin{aligned}\Pr_{\text{SpecTr}}(\tau \geq 2) &= \frac{4}{9} \cdot \frac{2}{3} + \left(\frac{1}{6} + \frac{\sqrt{13}}{18}\right)\frac{r+1}{3} \\ &= \frac{71 + 7\sqrt{13}}{162}.\end{aligned}$$

The expected acceptance length of SpecTr is consequently

$$\mathbb{E}_{\text{SpecTr}}[\tau] = \Pr(\tau \geq 1) + \Pr(\tau \geq 2) = \frac{85 + 8\sqrt{13}}{81} \approx 1.40549.$$

We thus obtain

$$\mathbb{E}_{\text{MDSeq}}[\tau] - \mathbb{E}_{\text{SpecTr}}[\tau] = \frac{17 - 2\sqrt{13}}{81} \approx 0.12085 > 0.$$

Although the two methods have the same first-token acceptance probability, MDSeq has a higher probability of accepting the complete sequence. MDSeq jointly verifies prefixes of different lengths, whereas SpecTr restricts subsequent token-level verification to candidates consistent with the previously selected token. This example therefore isolates the advantage of sequence-level coupling while explicitly confirming that MDSeq preserves the target distribution.

Comparison with Block Verification. Under the same setting, Block Verification (Sun et al., 2024b) achieves an expected acceptance length of

$$\mathbb{E}_{\text{Block}}[\tau] = \frac{11}{9},$$

which is also strictly smaller than $\mathbb{E}_{\text{MDSeq}}[\tau]$. This comparison further demonstrates the benefit of combining sequence-level verification with multiple draft sequences.

B EXPERIMENTS ON VICUNA 13B

Standard speculative decoding and model-scale expansion. To evaluate the generalizability of MDSeq beyond EAGLE, we further conduct experiments using a standard speculative decoding framework. We use Tiny-Vicuna-1B as the draft model and Vicuna-13B-v1.3 as the target model, with draft length $L = 7$, sequence count $K = 5$, and temperature $T = 1.0$. As shown in Table 4, MDSeq consistently outperforms all baselines, corroborating our main results under a larger target model and a different drafter architecture.

Table 4: BE and WS results on Spec-Bench using Tiny-Vicuna-1B and Vicuna-13B-v1.3 as the draft and target model pairs. The best result for each dataset and metric is highlighted in bold.

Data	Metric	Vanilla	SpecTr	RRSw	BV	MDSeq
Draft: Tiny-Vicuna-1B, Target: Vicuna-13B-v1.3						
Con.	WS	1.83	2.30	1.98	2.09	**2.32**
	BE	1.76	2.95	2.43	2.16	**3.03**
Math	WS	2.26	2.37	2.24	2.42	**2.51**
	BE	2.16	2.83	2.65	2.42	**3.05**
Trans.	WS	1.77	1.84	1.71	1.67	**1.88**
	BE	1.89	2.37	2.17	1.71	**2.48**
Sum.	WS	2.11	2.43	2.24	2.33	**2.57**
	BE	1.92	2.80	2.48	2.17	**2.97**
QA	WS	1.65	**1.97**	1.96	1.93	1.95
	BE	1.91	**2.99**	2.92	2.45	2.95
RAG	WS	2.20	2.47	2.20	2.28	**2.61**
	BE	1.94	2.77	2.39	2.08	**2.96**
AVG	WS	1.97	2.23	2.06	2.12	**2.31**
	BE	1.93	2.79	2.51	2.17	**2.91**

C EXPERIMENTS ON SPECHUB

Setting. Because SpecHub constructs a binary draft tree, we fix its maximum draft depth to 5, matching the draft length of MDSeq. For MDSeq, we set the number of candidate sequences to $K = 10$ to approximately match the draft budget of SpecHub. We use **EAGLE-Vicuna-7B-v1.3 (0.24B)** (Li et al., 2024a) as the draft model and **Vicuna-7B-v1.3** (Zheng et al., 2023) as the target model, with temperature $T = 1.0$.

Experimental results. At depth 5, the binary tree constructed by SpecHub contains $2^6 - 2 = 62$ draft nodes, exceeding the MDSeq budget of $KL = 10 \times 5 = 50$ nodes. This larger draft budget allows SpecHub to achieve higher BE than MDSeq. However, the additional drafting and verification costs introduce substantial computational overhead, resulting in lower WS than MDSeq. These results highlight that higher BE does not necessarily translate into greater end-to-end acceleration.

Table 5: WS and BE results of MDSeq and SpecHub on Spec-Bench. The best result for each dataset and metric is highlighted in bold.

Data	MDSeq		SpecHub	
	WS	BE	WS	BE
Math	**2.19**	3.54	2.13	**3.65**
QA	**1.84**	**2.94**	1.69	2.87
Trans.	**1.67**	2.80	**1.67**	**2.89**
Sum.	**1.83**	3.08	1.77	**3.16**
MT-Bench	**2.19**	3.57	2.12	**3.65**
RAG	1.71	3.04	**1.77**	**3.24**
AVG	**1.91**	3.16	1.86	**3.24**

D Proof of Theorem 1

Theorem 4 (Upper bound of expected acceptance length)**.** *Let τ be the acceptance length, for any valid multi-sequence draft verification algorithm in Definition 2, we have $\forall p, q, L, K$ the upper bound of expected acceptance length is:*

$$\mathbb{E}_{\textit{Verify}}[\tau] \leq \sum_{i=1}^{L} \sum_{x^i} \min\{1-(1-p(x^i))^K, q(x^i)\}.$$

Proof.

$$\begin{aligned}
\mathbb{E}[\tau] &= \sum_{i=1}^{L} P(\tau \geq i) \\
&= \sum_{i=1}^{L} \sum_{x^i} P\left(\exists k \in \{1,\dots,K\} : X_k^i = x^i,\ O^i = x^i\right) \\
&\leq \sum_{i=1}^{L} \sum_{x^i} \min\{P(\exists k \in \{1,\dots,K\} : X_k^i = x^i), P(O^i = x^i)\} \\
&= \sum_{i=1}^{L} \sum_{x^i} \min\{1-(1-p(x^i))^K, q(x^i)\},
\end{aligned} \tag{5}$$

where the inequality comes from the fact that the joint probability is upper bounded by the minimum of the marginals. The last equality follows from the i.i.d. assumption of the sequences.

□

E Proof of Theorem 2

Lemma 5 (Probability of Accepting Sub-block in Single Sequence)**.** *For all $i \in [1, L]$ and X^L, in MDSeq, we have*

$$P(X_1^i = x^i,\, \tau^{X_1} \geq i) = \min\{\frac{q(x^i)}{\rho}, p(x^i)\}.$$

where τ^{X_1} denotes the acceptance length of token sequence X_1.

Proof. We prove this by induction in the backward direction. When $i = L$, this holds due to the maximal coupling step for the entire block x^L in Algorithm 1 since

$$\begin{aligned} P(X_1^L = x^L, \tau^{X_1} \geq L) &= P(X_1^L = x^L) P\left(\tau^{X_1} = L \mid X_1^L = x^L\right) \\ &= p(x^L) \cdot \min\{\frac{q(x^L)}{\rho \cdot p(x^L)}, 1\} \\ &= \min\{\frac{q(x^L)}{\rho}, p(x^L)\}. \end{aligned}$$

where $P\left(X_1^L = x^L, \tau^{X_1} \geq L\right)$ denotes h_{Lk} in Eq. equation 2.

Suppose the equation holds for $i \geq i_0$. For $i = i_0 - 1$, we have

$$\begin{aligned} P(X_1^{i_0-1} = x^{i_0-1}, \tau^{X_1} \geq i_0 - 1) =& P(X_1^{i_0-1} = x^{i_0-1}, \tau^{X_1} \geq i_0) \\ &+ P(X_1^{i_0-1} = x^{i_0-1}, \tau^{X_1} = i_0 - 1) \end{aligned}$$

For the first term, by the induction assumption:

$$\begin{aligned} P(X_1^{i_0-1} = x^{i_0-1}, \tau^{X_1} \geq i_0) &= \sum_x P\left(X_1^{i_0} = (x^{i_0-1}, x), \tau^{X_1} \geq i_0\right) \\ &= \sum_x \min\{\frac{q(x^{i_0-1}, x)}{\rho}, p(x^{i_0-1}, x)\} \end{aligned} \tag{6}$$

For the second term, define the acceptance probability:

$$h \triangleq P(\text{accept } x^{i_0-1} \mid X_1^{i_0-1} = x^{i_0-1}).$$

Then:

$$\begin{aligned} P(X_1^{i_0-1} = x^{i_0-1}, \tau^{X_1} = i_0 - 1) &= P(X_1^{i_0-1} = x^{i_0-1}, \tau^{X_1} < i_0) P(\text{accept } x^{i_0-1} \mid X_1^{i_0-1} = x^{i_0-1}) \\ &= \left(P(X_1^{i_0-1} = x^{i_0-1}) - P(X_1^{i_0-1} = x^{i_0-1}, \tau^{X_1} \geq i_0)\right) h. \end{aligned} \tag{7}$$

The marginal probability is:

$$P(X_1^{i_0-1} = x^{i_0-1}) = p(x^{i_0-1}). \tag{8}$$

Substituting Eq. equation 8 into Eq. equation 7:

$$P(X_1^{i_0-1} = x^{i_0-1}, \tau^{X_1} = i_0 - 1) = \left(p(x^{i_0-1}) - \sum_x \min\{\frac{q(x^{i_0-1}, x)}{\rho}, p(x^{i_0-1}, x)\}\right) h \tag{9}$$

Combining terms Eqs. equation 6 and equation 9 yields:

$$\begin{aligned} &P(X_1^{i_0-1} = x^{i_0-1}, \tau^{X_1} \geq i_0 - 1) \\ &= P(X_1^{i_0-1} = x^{i_0-1}, \tau^{X_1} \geq i_0) + P(X_1^{i_0-1} = x^{i_0-1}, \tau^{X_1} = i_0 - 1) \\ &= \sum_x \min\{\frac{q(x^{i_0-1}, x)}{\rho}, p(x^{i_0-1}, x)\} + \left(p(x^{i_0-1}) - \sum_x \min\{\frac{q(x^{i_0-1}, x)}{\rho}, p(x^{i_0-1}, x)\}\right) h. \end{aligned} \tag{10}$$

As in Eq. equation 1, $h_{(i_0-1)k}$ equals to

$$h = \frac{\min\{\frac{q(x^{i_0-1})}{\rho}, p(x^{i_0-1})\} - \sum_x \min\{\frac{q(x^{i_0-1}, x)}{\rho}, p(x^{i_0-1}, x)\}}{p(x^{i_0-1}) - \sum_x \min\{\frac{q(x^{i_0-1}, x)}{\rho}, p(x^{i_0-1}, x)\}},$$

By the verification mechanism, substituting h into Eq. equation 10 yields:

$$P(X_1^{i_0-1} = x^{i_0-1}, \tau^{X_1} \geq i_0 - 1) = \min\{\frac{q(x^{i_0-1})}{\rho}, p(x^{i_0-1})\},$$

which completes the proof. □

Lemma 6 (Probability of Accepting Sub-block in Multi-Sequence Draft)**.** *For all $i \in [1, L]$ and X^L, in MDSeq, we have*

$$P(O^i = x^i, \tau \geq i) = \frac{1-(1-\beta_1(\rho))^K}{\beta_1(\rho)} \cdot \min\{\frac{q(x^i)}{\rho}, p(x^i)\}$$

Proof. We denote $\beta_1(\rho) = P(\tau^X \geq 1) = \sum_{x^1 \in \mathcal{V}^1} \min\{\frac{q(x^1)}{\rho}, p(x^1)\}$ as the acceptance rate of the first token. By Lemma 5, we have:

$$\begin{aligned}
&P(O^i = x^i, \tau \geq i) \\
=&\sum_{j=1}^{K} P(\tau^{X_1} \leq 0, ..., \tau^{X_{j-1}} \leq 0, X_j^i = x^i, \tau^{X_j} \geq i) \\
=&\sum_{j=1}^{K} (1-\beta_1(\rho))^{j-1} P(X_j^i = x^i, \tau^{X_j} \geq i) \\
=&\frac{1-(1-\beta_1(\rho))^K}{\beta_1(\rho)} \cdot \min\{\frac{q(x^i)}{\rho}, p(x^i)\},
\end{aligned}$$

where the last equality comes from Lemma 5. □

Here, we continue to prove the Theorem 2 which demonstrates that complete procedure of MDSeq, induced by Algorithm 1 and 2, preserves the distribution consistency of target model distribution. That is, for $\forall i \leq L$ and sub-block x^i, we have $P(O^i = x^i) = q(x^i)$.

Proof. We prove the claim by induction. Note that the claim $P(O^i = x^i) = q(x^i)$ holds for $i = 0$, which is the trivial case and both sides are equal to 1. Suppose the claim holds for $i \leq i_0$. This means that for any sub-block $\forall x^{i_0}$, we have

$$P(O^{i_0} = x^{i_0}) = q(x^{i_0}).$$

When $i = i_0 + 1$, by the algorithm, we have that either $\tau \geq i_0 + 1$, where $O^{(i_0+1)}$ is an accepted token, or $\tau \leq i_0$, where $O^{(i_0+1)}$ is sampled according to $p_{\text{res}}(\cdot \mid O^{i_0})$. We have

$$\begin{aligned}
&P(O^{i_0+1} = x^{i_0+1}) \\
=&P(O^{i_0+1} = x^{i_0+1}, \tau \geq i_0+1) + P(O^{i_0+1} = x^{i_0+1}, \tau \leq i_0) \\
=&P(O^{i_0+1} = x^{i_0+1}, \tau \geq i_0+1) + P(O^{i_0} = x^{i_0}, \tau \leq i_0) \cdot p_{\text{res}}(x_{i_0+1} \mid x^{i_0}) \\
=&P(O^{i_0+1} = x^{i_0+1}, \tau \geq i_0+1) \\
&+ \big(P(O^{i_0} = x^{i_0}) - P(O^{i_0} = x^{i_0}, \tau \geq i_0+1)\big) \cdot p_{\text{res}}(x_{i_0+1} \mid x^{i_0}) \\
=&\frac{1-(1-\beta_1(\rho))^K}{\beta_1(\rho)} \cdot \min\{\frac{q(x^{i_0+1})}{\rho}, p(x^{i_0+1})\} \\
&+ \left(q(x^{i_0}) - \sum_x \frac{1-(1-\beta_1(\rho))^K}{\beta_1(\rho)} \cdot \min\{\frac{q(x^{i_0}, x)}{\rho}, p(x^{i_0}, x)\}\right) \cdot p_{\text{res}}(x_{i_0+1} \mid x^{i_0}) \\
=&q(x^{i_0+1}),
\end{aligned}$$

where $p_{\text{res}}(x_{i_0+1} \mid x^{i_0})$ is defined in Eq. equation 3. The fourth equality comes from Lemma 6.

Furthermore, when $\rho \geq \rho^*$, we have $p_{\text{res}}(x_{i_0+1} \mid x^{i_0}) \geq 0$, ensuring validity.

This completes the induction. Therefore, for all $i \leq L$, we have

$$P(O^i = x^i) = q(x^i).$$

It remains to consider the final position $L + 1$, which is not covered by the preceding induction argument. Regardless of whether $\tau = L$ or $\tau < L$, the token at position $L + 1$ is sampled from the

original target conditional distribution. Specifically, when $\tau = L$, Algorithm 1 directly samples the additional token from $q(\cdot \mid O^L)$. When $\tau < L$, the distribution modification applies only through position L, after which the original target distribution is restored. Therefore,

$$\begin{aligned} P\big(O^{L+1} = x^{L+1}\big) &= P\big(O^L = x^L\big)\, q\big(x_{L+1} \mid x^L\big) \\ &= q(x^L) q\big(x_{L+1} \mid x^L\big) \\ &= q(x^{L+1}). \end{aligned}$$

Hence, the output sequence produced by MDSeq follows the target distribution for all positions up to $L+1$. □

F Proof of Theorem 3

Theorem 7 (Lower bound of expected acceptance length of MDSeq)**.** *Let τ denote the number of accepted tokens in a single SD iteration, when $\rho = \rho^*$, the expected acceptance length of MDSeq is*

$$\mathbb{E}_{MDSeq}[\tau] \geq \left(1 - \left(1 - \frac{1}{K}\right)^K\right) \sum_{i=1}^{L} \sum_{x^i} \min\{1 - (1 - p(x^i))^K, q(x^i)\}.$$

Proof. We define $\beta_i(\rho^*) = \sum_{x^i \in \mathcal{V}^i} \min\{\frac{q(x^i)}{\rho^*}, p(x^i)\}$

$$\begin{aligned} \mathbb{E}[\tau] &= \sum_{i=1}^{L} P(\tau \geq i) \\ &= \sum_{i=1}^{L} \sum_{x^i} P(O^i = x^i, \text{acceptance length} \geq i) \\ &= \sum_{i=1}^{L} \sum_{x^i} \sum_{k=1}^{K} P(X_1, ..., X_{k-1} \text{ are rejected, } X_k = x^i \text{ is accepted}) \\ &= \sum_{i=1}^{L} \sum_{x^i} \sum_{k=1}^{K} (1 - \beta_1(\rho^*))^{k-1} P(X_k = x^i, \tau^{x^i} \geq i) \\ &= \sum_{i=1}^{L} \sum_{x^i} \frac{1 - (1 - \beta_1(\rho^*))^K}{\beta_1(\rho^*)} \min\left\{p(x^i), \frac{q(x^i)}{\rho^*}\right\} \\ &= \sum_{i=1}^{L} \frac{1 - (1 - \beta_1(\rho^*))^K}{\beta_1(\rho^*)} \beta_i(\rho^*) \\ &\geq \sum_{i=1}^{L} \frac{1 - (1 - \beta_1(\rho^*))^K}{\beta_1(K)} \beta_i(K) \\ &\geq \sum_{i=1}^{L} \frac{1 - (1 - \beta_1(K))^K}{\beta_1(K)} \beta_i(K) \\ &= \sum_{i=1}^{L} \frac{1 - (1 - \beta_1(K))^K}{K\beta_1(K)} \sum_{x^i} \min\left\{Kp(x^i), q(x^i)\right\} \\ &\geq \left(1 - \left(1 - \frac{1}{K}\right)^K\right) \sum_{i=1}^{L} \sum_{x^i} \min\left\{Kp(x^i), q(x^i)\right\} \\ &\geq \left(1 - \left(1 - \frac{1}{K}\right)^K\right) \sum_{i=1}^{L} \sum_{x^i} \min\left\{1 - \left(1 - p(x^i)\right)^K, q(x^i)\right\}. \end{aligned}$$

The first inequality follows from Assumption 1. The second inequality follows from $\rho^* \leq K$, which implies $\beta_1(\rho^*) \geq \beta_1(K)$. For the third inequality, note that $\beta_1(K) \leq 1/K$ and that $f(z) = \frac{1-(1-z)^K}{Kz}$ is monotonically decreasing on $(0, 1/K]$. Therefore, $\frac{1-(1-\beta_1(K))^K}{K\beta_1(K)} \geq 1 - \left(1 - \frac{1}{K}\right)^K$. The final inequality follows from $Kz \geq 1 - (1-z)^K$ for every $z \in [0, 1]$.

□