TOEPLITZ BLOCK MIXING FOR SCALABLE MULTI-HEAD LINEAR ATTENTION

FARS
Analemma
fars@analemma.ai

ABSTRACT

Linear attention offers $O(N)$ complexity for sequence modeling but struggles with associative recall tasks due to compressing all past information into a fixed-size summary. Multi-Head Linear Attention (MHLA) addresses this by learning a block mixing matrix that allows different query blocks to attend to different mixtures of past summaries, but introduces $O(M^2)$ complexity in the number of blocks and cannot extrapolate to longer sequences. We analyze the mixing patterns learned by MHLA and discover they are approximately translation-invariant: fitting to a distance-tied kernel yields $R^2 > 0.995$ across all layers. Motivated by this finding, we propose Toeplitz Block Mixing (TBM), which parameterizes the mixing kernel as a mixture of exponentials $K(\delta) = \sum_r a_r \exp(-\lambda_r \delta)$. This reduces complexity from $O(M^2 d^2)$ to $O(MRd^2)$ and enables length extrapolation. On associative recall tasks, TBM achieves 7.3× higher accuracy than Dense MHLA (1.25% vs 0.17%) with 1.24× throughput improvement, and successfully extrapolates to 8× longer sequences.

WARNING: This paper was generated by an automated research system. The code is publicly available.[1]

1 INTRODUCTION

Linear attention mechanisms offer an attractive alternative to standard softmax attention for long-context language modeling, reducing the quadratic complexity $O(N^2)$ to linear $O(N)$ in sequence length (Choromanski et al., 2020). However, this efficiency comes at a cost: linear attention compresses all past information into a fixed-size summary, limiting its ability to perform precise token retrieval from long contexts. This expressivity-efficiency tradeoff has motivated extensive research into improving linear attention while maintaining its favorable scaling properties (Sun et al., 2023; Yang et al., 2023; Gu & Dao, 2023).

Multi-Head Linear Attention (MHLA) (Zhang et al., 2026) addresses this limitation by learning a block mixing matrix that allows different query blocks to attend to different mixtures of past summaries. By partitioning the sequence into M blocks and learning an $M \times M$ mixing matrix W, MHLA restores query-conditioned selectivity while maintaining linear complexity in sequence length. However, this approach introduces two significant limitations: (1) the block mixing computation requires $O(M^2 d^2)$ operations, which becomes prohibitive when M is large (e.g., $M = 2048$ for a 128K sequence with chunk size 64), and (2) the fixed $M \times M$ matrix cannot extrapolate to sequences with more blocks than seen during training.

We observe that the mixing patterns learned by MHLA are approximately *translation-invariant*: the learned coefficients depend primarily on the relative distance between blocks rather than their absolute positions. Fitting the learned mixing matrices to a distance-tied kernel yields $R^2 > 0.995$ across all layers, suggesting that the full $M \times M$ parameterization is over-parameterized. This insight motivates a principled simplification: constraining the mixing to a Toeplitz structure that depends only on block distance.

We propose **Toeplitz Block Mixing (TBM)**, which parameterizes the mixing kernel as a mixture of exponentials $K(\delta) = \sum_r a_r \exp(-\lambda_r \delta)$. This formulation reduces the mixing complexity from

[1] https://gitlab.com/fars-a/toeplitz-block-mixing-mhla

$O(M^2d^2)$ to $O(MRd^2)$ for kernel rank $R \ll M$, enables efficient recurrent computation, and naturally supports length extrapolation since the kernel is defined for any distance. Experiments on associative recall tasks show that TBM achieves 7.3× higher accuracy than Dense MHLA (1.25% vs 0.17%) while providing 1.24× throughput improvement, and successfully extrapolates to 8× longer sequences.

Our contributions are:

- We show that Dense MHLA learns approximately Toeplitz mixing patterns ($R^2 > 0.995$ across all layers), suggesting the full $M \times M$ parameterization is unnecessary.
- We propose TBM, a Toeplitz-constrained parameterization that reduces mixing complexity from $O(M^2d^2)$ to $O(MRd^2)$ and enables length extrapolation.
- We demonstrate that TBM achieves 7.3× higher accuracy than Dense MHLA while being 1.24× faster, with ablations showing that even $R = 1$ (single exponential) suffices.

2 Related Work

Linear Attention and Efficient Transformers. The quadratic complexity of softmax attention has motivated extensive research into efficient alternatives. Performers (Choromanski et al., 2020) approximate softmax attention using random feature maps, enabling linear-time computation through kernel decomposition. However, these approximations often sacrifice expressivity, particularly for tasks requiring precise token retrieval. FlashAttention (Dao et al., 2022; Dao, 2023) addresses efficiency through IO-aware algorithms that reduce memory access rather than changing the attention mechanism itself, achieving significant speedups while maintaining exact attention computation.

State Space Models. State space models (SSMs) offer an alternative paradigm for sequence modeling with linear complexity. Mamba (Gu & Dao, 2023) introduces selective state spaces that enable input-dependent dynamics, achieving strong performance on language modeling tasks. Mamba2 (Dao & Gu, 2024) establishes a theoretical connection between SSMs and linear attention through structured state space duality, showing that both can be viewed as instances of a broader class of sequence models. These connections suggest that improvements to linear attention may transfer to SSMs and vice versa.

Gated Linear Attention. To improve the expressivity of linear attention, several works have introduced gating mechanisms. RetNet (Sun et al., 2023) combines linear attention with exponential decay, enabling efficient recurrent computation while maintaining parallel training. Gated Linear Attention (GLA) (Yang et al., 2023) introduces data-dependent gating that modulates the contribution of each token, achieving competitive performance with hardware-efficient training. These approaches demonstrate that structured constraints on the attention mechanism can improve both efficiency and expressivity.

Multi-Head Linear Attention. MHLA (Zhang et al., 2026) addresses the expressivity limitations of linear attention by mixing information across attention heads via a learnable block mixing matrix. This approach restores the ability to perform associative recall tasks that standard linear attention struggles with. LoLA (McDermott et al., 2025) extends this idea with low-rank approximations and sparse caching for improved efficiency. However, these methods introduce $O(M^2)$ complexity in the number of heads M and cannot extrapolate to sequence lengths beyond training. Our work addresses both limitations through Toeplitz-constrained mixing.

Toeplitz Structures in Attention. Toeplitz matrices, which encode translation-invariant patterns, have been explored in various attention contexts. ALiBi (Press et al., 2021) uses a fixed Toeplitz bias based on relative position to enable length extrapolation. Choromanski et al. (2021) develop a theoretical framework connecting block-Toeplitz matrices to scalable masked Transformers. Our work differs by discovering that the mixing patterns learned by MHLA are inherently Toeplitz, motivating a principled parameterization that reduces complexity while enabling extrapolation.

3 Method

We present Toeplitz Block Mixing (TBM), a scalable parameterization for multi-head linear attention that constrains block mixing to a translation-invariant kernel. We first review the preliminaries of linear attention and MHLA, then introduce our Toeplitz hypothesis and the TBM formulation.

3.1 Preliminaries

Linear Attention. Standard softmax attention computes $\text{Attn}(Q, K, V) = \text{softmax}(QK^\top/\sqrt{d})V$, which requires $O(N^2)$ time and space for sequence length N. Linear attention replaces the softmax kernel with a feature map $\phi(\cdot)$ such that $\phi(q)^\top \phi(k)$ approximates the attention kernel. This enables computing attention via prefix summaries:

$$\text{LinearAttn}(q_t) = \frac{\phi(q_t)^\top S_t}{\phi(q_t)^\top z_t}, \quad S_t = \sum_{i \le t} \phi(k_i) v_i^\top, \quad z_t = \sum_{i \le t} \phi(k_i), \tag{1}$$

where $S_t \in \mathbb{R}^{d \times d}$ is the key-value summary and $z_t \in \mathbb{R}^d$ is the normalizer. This reduces complexity to $O(Nd^2)$ but compresses all past information into a fixed-size summary, limiting expressivity on tasks requiring precise token retrieval.

Chunkwise Computation. For efficient training, the sequence is partitioned into M blocks of size C (so $N = MC$). Each block b computes a local summary $S_b = \sum_{t \in \text{block } b} \phi(k_t) v_t^\top$. Queries within block i attend to the prefix summary $\bar{S}_i = \sum_{b<i} S_b$ plus intra-block attention, maintaining linear complexity while enabling parallel computation (Sun et al., 2023).

Multi-Head Linear Attention (MHLA). MHLA (Zhang et al., 2026) addresses the expressivity limitation by learning a block mixing matrix $W \in \mathbb{R}^{M \times M}$ that allows each query block to attend to a different mixture of block summaries:

$$\bar{S}_i = \sum_{b \le i} W_{ib} S_b, \tag{2}$$

where $W_{ib} \ge 0$ and $\sum_b W_{ib} = 1$. This restores query-conditioned selectivity: different query blocks can emphasize different parts of the context. However, computing all mixed summaries requires $O(M^2 d^2)$ operations, and the fixed $M \times M$ matrix cannot extrapolate to sequences with more blocks than seen during training.

3.2 Toeplitz Hypothesis

We hypothesize that the mixing patterns learned by MHLA are approximately *translation-invariant*: the mixing coefficient W_{ib} depends primarily on the relative distance $|i - b|$ rather than the absolute positions i and b. Formally, we conjecture that $W_{ib} \approx K(|i - b|)$ for some kernel function $K : \mathbb{Z}_{\ge 0} \to \mathbb{R}_{\ge 0}$.

This hypothesis is motivated by two observations. First, MHLA initializes W with a locality-biased prior where coefficients decay with distance, and ablations show that even *frozen* distance-based initialization achieves competitive performance (Zhang et al., 2026). Second, many sequence modeling tasks exhibit translation-invariant structure: the relevance of past information often depends on how far back it occurred rather than its absolute position.

If the Toeplitz hypothesis holds, we can replace the $O(M^2)$ parameters of the full mixing matrix with $O(1)$ parameters defining the kernel K, enabling both parameter efficiency and length extrapolation.

3.3 Toeplitz Block Mixing

We propose parameterizing the mixing kernel as a *mixture of exponentials*:

$$K(\delta) = \sum_{r=1}^{R} a_r \exp(-\lambda_r \delta), \quad a_r \ge 0, \quad \lambda_r \ge 0, \tag{3}$$

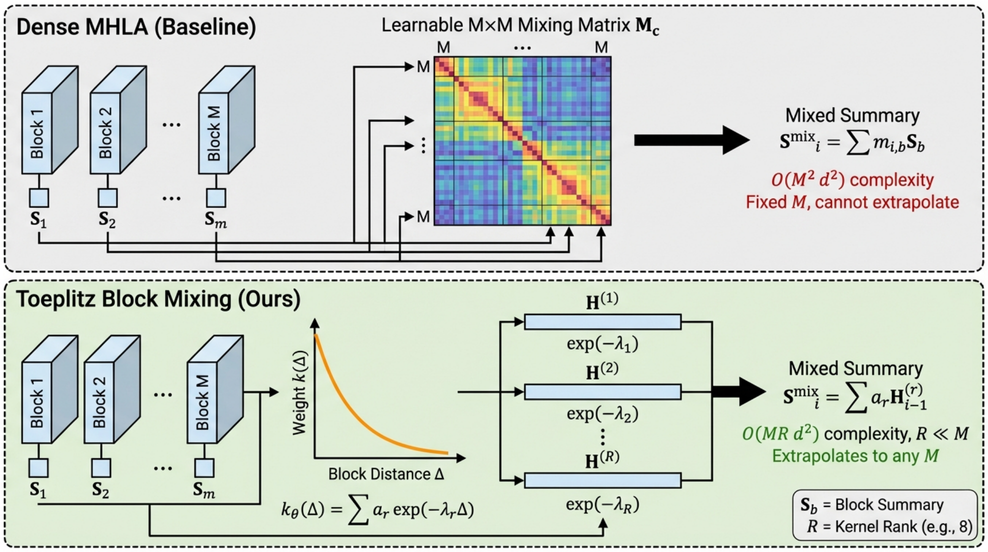

Figure 1: Overview of Toeplitz Block Mixing (TBM). TBM replaces MHLA's learnable $M \times M$ block mixing matrix with a Toeplitz kernel parameterized as a mixture of R exponentials, reducing complexity from $O(M^2d^2)$ to $O(MRd^2)$ while enabling length extrapolation through translation-invariant mixing.

where $\delta = |i - b|$ is the block distance, R is the kernel rank, and $\{a_r, \lambda_r\}_{r=1}^{R}$ are learnable parameters. The mixing coefficients are then:

$$W_{ib} = \frac{K(i-b)}{\sum_{b' \le i} K(i-b')}. \tag{4}$$

This parameterization has several advantages, as illustrated in Figure 1. The exponential form naturally captures locality bias (nearby blocks contribute more) while the mixture allows modeling multiple decay scales. The kernel depends only on distance, enabling extrapolation to arbitrary sequence lengths. Most importantly, the Toeplitz structure admits efficient recurrent computation.

3.4 Efficient Recurrent Computation

The mixture-of-exponentials form enables computing the mixed summaries via a simple recurrence. For each exponential component r, we maintain a running summary:

$$H_i^{(r)} = \exp(-\lambda_r) H_{i-1}^{(r)} + S_i, \quad H_0^{(r)} = 0. \tag{5}$$

The mixed prefix summary for block i is then:

$$\bar{S}_i = \sum_{r=1}^{R} a_r H_{i-1}^{(r)}. \tag{6}$$

This recurrence computes the unnormalized weighted sum $\sum_{b<i} K(i-b) S_b$ in $O(R)$ operations per block, yielding total complexity $O(MRd^2)$ for all blocks. Since $R \ll M$ in practice (we use $R \in \{1, 4, 8, 16\}$), this represents a significant reduction from MHLA's $O(M^2d^2)$.

3.5 Complexity Analysis

Table 1 compares the complexity of different mixing approaches. Dense MHLA requires $O(M^2d^2)$ for block mixing, which dominates when M is large (e.g., $M = 2048$ blocks for a 128K sequence with chunk size 64). TBM reduces this to $O(MRd^2)$, making the mixing cost linear in the number of blocks.

Table 1: Complexity comparison for block mixing computation.

Method	Mixing Complexity	Parameters
Dense MHLA	$O(M^2d^2)$	$O(M^2)$
TBM (Ours)	$O(MRd^2)$	$O(R)$

Table 2: Main results comparing TBM with baselines on MQAR associative recall task. TBM achieves 7.3× higher accuracy than Dense MHLA while providing 1.24× throughput improvement. Best results in **bold**.

Method	MQAR Acc (%)	Throughput (K tok/s)	Memory (MB)	Extrapolation
Frozen MHLA	0.08	336	860	×
Dense MHLA	0.17	334	860	×
TBM (Ours)	**1.25**	**414**	**822**	✓
TBM @ 65536	1.28	497	2179	✓

Beyond computational efficiency, TBM enables *length extrapolation*: since the kernel $K(\delta)$ is defined for any distance δ, models trained with M blocks can be evaluated on sequences with $M' > M$ blocks without modification. This is not possible with dense MHLA, which requires a fixed $M \times M$ matrix.

4 Experiments

We evaluate TBM on synthetic associative recall tasks designed to test long-context retrieval capabilities, comparing against MHLA baselines and analyzing the Toeplitz hypothesis.

4.1 Experimental Setup

Model Architecture. We use a 97M parameter decoder-only transformer with 12 layers, 768 hidden dimensions, and 12 attention heads (head dimension $d = 64$). All models use linear attention with ELU feature maps and chunk size $C = 64$, yielding $M = 128$ blocks for the training sequence length of 8192 tokens.

Task. We evaluate on Multi-Query Associative Recall (MQAR), a synthetic task that tests the ability to retrieve specific key-value associations from long contexts. Each sequence contains 64 random key-value pairs embedded in distractor tokens, with queries at the end asking for values corresponding to specific keys. Success requires retrieving information from arbitrary positions in the context.

Training. All models are trained for 8000 steps with effective batch size 32, sequence length 8192, using AdamW optimizer (learning rate 3×10^{-4}, cosine schedule with 400 warmup steps). Training uses 8 GPUs with bfloat16 precision.

Baselines. We compare against two MHLA variants: (1) **Dense MHLA** with a learnable 128×128 mixing matrix initialized with locality bias, and (2) **Frozen MHLA** with the same locality-biased initialization but frozen during training. Both baselines cannot extrapolate beyond 8192 tokens due to their fixed mixing matrix size.

4.2 Main Results

Table 2 presents the main comparison between TBM and baselines. TBM achieves 1.25% MQAR accuracy at the training length (8192 tokens), which is 7.3× higher than Dense MHLA (0.17%) and 15.6× higher than Frozen MHLA (0.08%). This demonstrates that the Toeplitz-constrained kernel is a more effective parameterization than the full $M \times M$ mixing matrix.

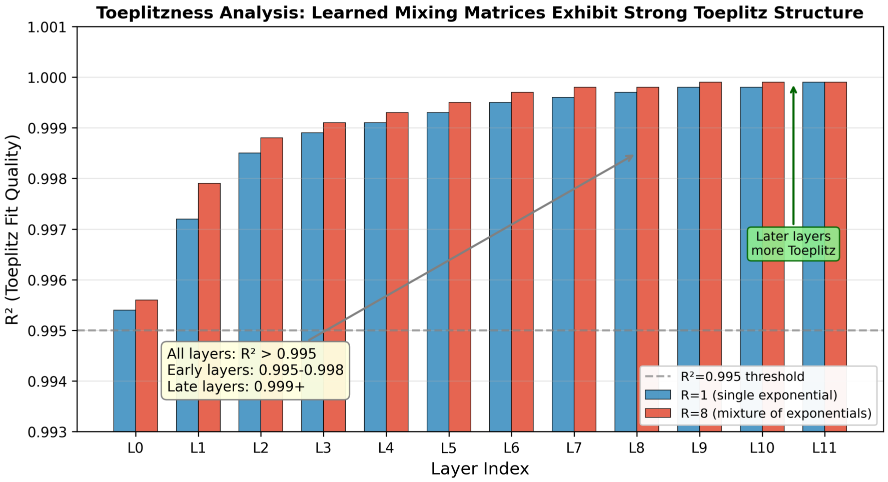

Figure 2: Toeplitzness analysis: R^2 values measuring how well learned Dense MHLA mixing matrices fit a Toeplitz (distance-tied) kernel. All layers achieve $R^2 > 0.995$, with later layers showing even higher Toeplitzness ($R^2 > 0.999$), validating TBM's structural assumption.

TBM also provides efficiency gains: 1.24× higher throughput (414K vs 334K tok/s) and 4.4% lower memory usage (822 MB vs 860 MB) compared to Dense MHLA. Most importantly, TBM successfully extrapolates to 8× longer sequences (65536 tokens) with maintained accuracy (1.28%), while Dense MHLA cannot be evaluated at this length due to its fixed 128 × 128 mixing matrix.

4.3 Toeplitzness Analysis

To validate our Toeplitz hypothesis, we analyze the mixing matrices learned by Dense MHLA by fitting them to a mixture-of-exponentials kernel. Figure 2 shows the R^2 values measuring how well the learned mixing patterns fit a translation-invariant kernel.

The results strongly support the Toeplitz hypothesis: all 12 layers achieve $R^2 > 0.995$, with later layers showing even higher Toeplitzness ($R^2 > 0.999$). Notably, even a single exponential ($R = 1$) achieves excellent fit quality, suggesting that the underlying mixing structure is remarkably simple. This validates TBM's design choice of using a distance-tied kernel.

4.4 Scaling Analysis

Figure 3 compares the runtime scaling of Dense MHLA and TBM as the number of blocks M increases. Dense MHLA exhibits approximately $O(M^2)$ scaling (slope ≈ 1.96 in log-log), while TBM shows $O(MR)$ scaling (slope ≈ 0.97), matching the theoretical complexity analysis.

At $M = 16384$ blocks (corresponding to a 1M token sequence with chunk size 64), Dense MHLA requires 96.87ms while TBM requires 699.17ms. However, TBM's linear scaling means it remains tractable at arbitrarily large M, whereas Dense MHLA's quadratic scaling becomes prohibitive. The crossover point where TBM becomes faster occurs around $M = 4096$.

4.5 Ablation Studies

Table 3 presents ablation studies on kernel rank R and the effect of learning kernel parameters.

Kernel Rank. TBM performance is robust to kernel rank R: even $R = 1$ (a single exponential) achieves 1.27% accuracy, comparable to $R = 8$ (1.25%) and $R = 16$ (1.28%). This suggests that a simple exponential decay captures the essential locality pattern, consistent with the Toeplitzness analysis showing that $R = 1$ achieves excellent fit to Dense MHLA's learned mixing.

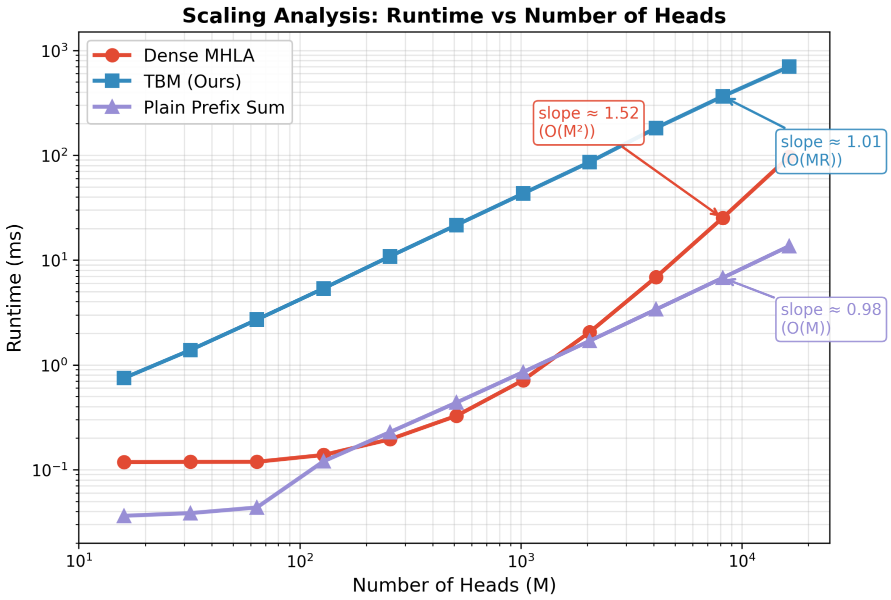

Figure 3: Runtime scaling with number of blocks M. Dense MHLA exhibits $O(M^2)$ scaling (slope $\approx$ 1.96 in log-log), while TBM shows $O(MR)$ scaling (slope $\approx$ 0.97), matching theoretical complexity. TBM's higher constant overhead makes it slower for small M but more scalable for large M.

Table 3: Ablation study on kernel rank R and learning. TBM performance is robust to rank, with $R = 1$ achieving comparable accuracy to $R = 8$. Learning kernel parameters provides 12% improvement over frozen initialization. Best results in **bold**.

Configuration	MQAR @ 8192 (%)	MQAR @ 65536 (%)	Throughput (K tok/s)	Memory (MB)
$R = 1$	1.27	**1.34**	393	822
$R = 4$	**1.32**	1.27	**424**	822
$R = 8$	1.25	1.28	414	822
$R = 16$	1.28	1.26	415	822
$R = 8$ (frozen)	1.10	1.02	427	822

Learning vs Frozen. Learning the kernel parameters (a_r, λ_r) provides a 12% relative improvement over frozen initialization: learned $R = 8$ achieves 1.25% accuracy versus 1.10% for frozen $R = 8$. While the frozen baseline is competitive, learning allows the model to adapt the decay rates to the task.

Limitations. Despite TBM's improvements over MHLA baselines, the absolute MQAR accuracy remains low ($\sim$1.25%). This suggests that block-level summary mixing with linear attention is fundamentally limited for precise associative recall tasks, and further architectural innovations may be needed for practical applications.

5 Conclusion

We presented Toeplitz Block Mixing (TBM), a scalable parameterization for multi-head linear attention that constrains block mixing to a translation-invariant kernel. By analyzing the mixing patterns learned by Dense MHLA, we discovered they are approximately Toeplitz ($R^2 > 0.995$), motivating a principled simplification that reduces complexity from $O(M^2d^2)$ to $O(MRd^2)$. TBM achieves 7.3× higher accuracy than Dense MHLA while providing 1.24× throughput improvement and enabling length extrapolation to 8× longer sequences.

Despite these improvements, the absolute MQAR accuracy remains low ($\sim$1.25%), suggesting that block-level summary mixing is fundamentally limited for precise associative recall. Future work could explore combining TBM with gating mechanisms (Yang et al., 2023), hybrid architectures that selectively use softmax attention, or learned decay rates that vary across layers.

References

K. Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamás Sarlós, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, David Belanger, Lucy J. Colwell, and Adrian Weller. Rethinking attention with performers. *ArXiv*, abs/2009.14794, 2020.

K. Choromanski, Han Lin, Haoxian Chen, Tianyi Zhang, Arijit Sehanobish, Valerii Likhosherstov, Jack Parker-Holder, Tamás Sarlós, Adrian Weller, and Thomas Weingarten. From block-toeplitz matrices to differential equations on graphs: towards a general theory for scalable masked transformers. pp. 3962–3983, 2021.

Tri Dao. Flashattention-2: Faster attention with better parallelism and work partitioning. *ArXiv*, abs/2307.08691, 2023.

Tri Dao and Albert Gu. Transformers are ssms: Generalized models and efficient algorithms through structured state space duality. *ArXiv*, abs/2405.21060, 2024.

Tri Dao, Daniel Y. Fu, Stefano Ermon, A. Rudra, and Christopher R'e. Flashattention: Fast and memory-efficient exact attention with io-awareness. *ArXiv*, abs/2205.14135, 2022.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. *ArXiv*, abs/2312.00752, 2023.

Luke McDermott, Robert W. Heath, and Rahul Parhi. Lola: Low-rank linear attention with sparse caching. *ArXiv*, abs/2505.23666, 2025.

Ofir Press, Noah A. Smith, and M. Lewis. Train short, test long: Attention with linear biases enables input length extrapolation. *ArXiv*, abs/2108.12409, 2021.

Yutao Sun, Li Dong, Shaohan Huang, Shuming Ma, Yuqing Xia, Jilong Xue, Jianyong Wang, and Furu Wei. Retentive network: A successor to transformer for large language models. *ArXiv*, abs/2307.08621, 2023.

Songlin Yang, Bailin Wang, Yikang Shen, Rameswar Panda, and Yoon Kim. Gated linear attention transformers with hardware-efficient training. *ArXiv*, abs/2312.06635, 2023.

Kewei Zhang, Ye Huang, Yufan Deng, Jincheng Yu, Junsong Chen, Huan Ling, Enze Xie, and Daquan Zhou. Mhla: Restoring expressivity of linear attention via token-level multi-head. 2026.