OVERLAPPING SUBSPACE COMMUNICATION FOR SPIKING DECISION TRANSFORMER

Anonymous authors
Paper under double-blind review

ABSTRACT

Offline reinforcement learning can be formulated as sequence modeling, enabling policies to learn directly from fixed trajectories. While spiking decision transformers provide an efficient alternative to dense Transformers through sparse event-driven computation, the internal communication structure of sparse spiking attention remains underexplored. We observe that spike-driven communication supports vary across return-ranked trajectories, while a subset of token interactions repeatedly recurs, suggesting the coexistence of reusable and context-dependent communication patterns. Motivated by this observation, we propose the Spiking Subspace Decision Transformer (SS-DT) with Overlapping Spiking Subspace Attention (OSSA). OSSA factorizes intra-head low-dimensional spike communication into multiple overlapping subspaces, where shared components capture reusable structures and private components preserve context-specific variations. Spike-guided allocation dynamically composes these subspaces according to spiking activity, while predictive feedback further adapts communication to trajectory dynamics and decision context. Together, SS-DT introduces structured adaptive communication into sparse spiking sequence modeling. Across 21 tasks on MuJoCo and Adroit, SS-DT outperforms the strongest spiking baseline by 3.7 normalized-return points on average, with gains of 1.3 and 5.4 points on MuJoCo and Adroit, respectively, establishing state-of-the-art average performance among spiking offline RL methods.

1 INTRODUCTION

Offline reinforcement learning (RL) aims to learn effective policies from fixed datasets without additional environment interaction. A promising line of work formulates offline RL as sequence modeling, where Decision Transformer (DT) predicts actions conditioned on historical states, actions, and returns-to-go Chen et al. (2021). Recent work further improves sequence modeling by adaptively balancing long-term and local dependencies Zheng et al. (2025). This formulation avoids explicit value estimation and Bellman backup during policy learning, casting policy learning as supervised return-conditioned sequence prediction. However, most existing methods rely on dense ANN Transformers, whose continuous activations and attention operations incur substantial computation and energy costs, particularly for long-horizon embodied decision making.

Spiking neural networks (SNNs) provide an efficient alternative through sparse event-driven computation and are well suited to resource-constrained embodied agents Van den Berghe et al. (2025). Recent work has therefore explored spike-driven Transformers for offline decision making. In particular, Decision SpikeFormer Huang et al. (2025) adapts the DT framework to SNNs and develops temporal and positional spiking self-attention mechanisms for trajectory modeling. These advances demonstrate the potential of spiking Transformers for efficient offline decision making. Nevertheless, existing methods primarily focus on how temporal dependencies are computed, while how sparse spike communication is structured and adapted remains less explored.

Unlike dense representations, spiking attention communicates through sparse token-pair interactions, whose supports can vary across offline trajectories. As illustrated in Fig. 1, although communication supports differ across return-ranked trajectories, a subset of interactions repeatedly recurs. This reveals that sparse spike communication contains both reusable structures and trajectory-dependent variations. However, existing spiking attention models represent these interactions within

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.

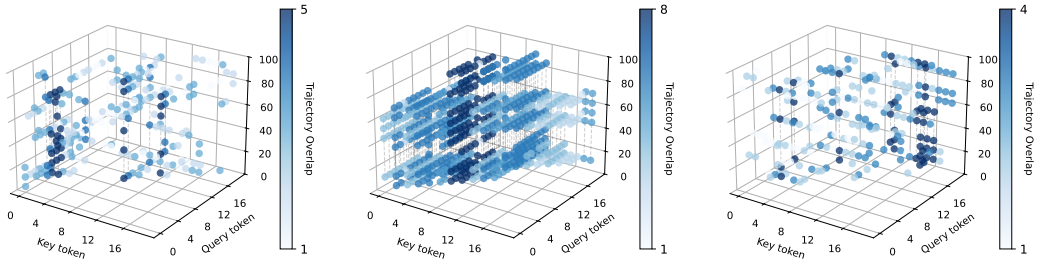

Figure 1: **Recurring and trajectory-dependent sparse spike communication.** Representative trajectories are sampled across the empirical return spectrum on three offline RL tasks. Each point denotes an active spike-driven token-pair communication support. Recurring supports indicate repeatedly activated interactions across trajectories, while other supports exhibit trajectory-dependent variations.

a single intra-head communication space, making it difficult to explicitly capture both properties. This motivates structuring sparse spike communication with adaptive and overlapping representations.

Motivated by this observation, we propose *Overlapping Spiking Subspace Attention* (OSSA), which factorizes intra-head spike communication into multiple low-dimensional overlapping subspaces. Each subspace combines shared and private projections of query, key, and value representations, allowing shared components to capture recurring communication structures while private components model subspace-specific variations. This shared-private factorization provides a structured representation of sparse spike communication beyond independent attention heads.

The contribution of these communication structures should further adapt to the instantaneous spiking and decision context. We therefore introduce spike-guided communication allocation, which dynamically composes multiple subspace messages using spike-derived query features, firing rates, and temporal spike statistics. Predictive feedback derived from latent prediction discrepancy further modulates communication according to latent consistency. Together, these mechanisms enable context-dependent composition of the learned overlapping communication structures. Based on these designs, we propose the **S**piking **S**ubspace **D**ecision **T**ransformer (**SS-DT**), which integrates structured and adaptive sparse communication into return-conditioned spiking sequence modeling. Our contributions are summarized as follows:

- We reveal that sparse spike communication exhibits both reusable and trajectory-dependent patterns across offline trajectories. Based on this observation, we propose Overlapping Spiking Subspace Attention (OSSA), which factorizes intra-head communication into overlapping shared-private subspaces to capture recurring structures and context-dependent variations.
- We introduce adaptive communication allocation for OSSA, which dynamically composes overlapping subspace messages using spike-derived activity and predictive feedback, enabling flexible sparse communication across different decision contexts.
- Extensive experiments on 21 MuJoCo and Adroit tasks demonstrate that SS-DT achieves state-of-the-art average performance among spiking offline RL methods, improving over the strongest spiking baseline by 3.7 normalized-return points on average. Further analyses show that the gains come from structured communication organization rather than increased model capacity.

2 Related Work

Offline Reinforcement Learning. Offline reinforcement learning aims to learn effective policies from fixed datasets without further environment interaction. Classical value-based methods, such as BCQ Fujimoto et al. (2019), BEAR Kumar et al. (2019), CQL Kumar et al. (2020), and IQL Kostrikov et al. (2022), address distribution shift by constraining policy updates or regularizing

value estimates. Another line of work formulates offline RL as sequence modeling, where Decision Transformer Chen et al. (2021) and Trajectory Transformer Janner et al. (2021) model trajectories with Transformer architectures conditioned on returns or future outcomes. Although trajectory-level sequence modeling provides a natural framework for return-conditioned decision making, existing approaches predominantly rely on dense ANN Transformers. Recent spiking decision models provide an alternative through sparse event-driven computation. Our work follows this direction but focuses on the internal structure and adaptive composition of sparse spike communication in offline trajectory modeling.

Structured Attention and Adaptive Communication. Multi-head attention learns multiple representation subspaces through parallel query, key, and value projections Vaswani et al. (2017). Shared-private representations have also been explored to capture common and task- or modality-specific information Liu et al. (2017); Hazarika et al. (2020), while soft mixture-of-experts methods adaptively combine multiple expert representations through differentiable allocation Puigcerver et al. (2024). These approaches demonstrate the benefits of structured parameter sharing and adaptive composition in dense neural networks. However, they mainly operate at the level of independent representation spaces or expert selection. In contrast, OSSA focuses on the communication structure within each existing spiking attention head. It explicitly factorizes intra-head spike communication into low-dimensional overlapping subspaces, where shared components preserve reusable communication patterns and private components capture subspace-specific communication variations.

Spiking Transformers for Representation and Decision Making. Spiking Transformers redesign attention around sparse spike representations. Spikformer Zhou et al. (2023), Spike-driven Transformer Yao et al. (2023), and QKFormer Zhou et al. (2024a) develop spike-compatible attention mechanisms for efficient representation learning. Subsequent studies further improve spiking attention through addition-only computation Guo et al. (2025), spike-specific similarity measures Xiao et al. (2025), spatial-temporal interaction Lee et al. (2025), multi-delay temporal modeling Shi et al. (2026), and relative positional encoding Lv et al. (2025). SEMM Zhou et al. (2024b) further introduces expert routing to integrate multiple spiking attention experts. These methods mainly investigate how spike interactions are computed, enhanced, or routed, while the organization of sparse communication within an individual attention head remains less explored.

For offline decision making, DSFormer Huang et al. (2025) extends spiking Transformers to trajectory modeling by introducing TSSA and PSSA for temporal and positional dependencies. Our work complements this direction by addressing a different aspect of spiking decision modeling: rather than only improving sparse dependency computation, we explicitly structure and adapt the intra-head communication patterns underlying these dependencies.

3 METHOD

We consider offline decision making with spiking sequence models. Given a trajectory segment $\tau = \{(s_t, a_t, R_t)\}_{t=1}^{T} \sim \mathcal{D}$, where s_t, a_t, and R_t denote the state, action, and return-to-go (RTG), respectively, each decision token is constructed as

$$x_t^0 = \mathrm{LN}(E[s_t; a_{t-1}; R_t]), \qquad X^0 = [x_1^0, \ldots, x_T^0]. \tag{1}$$

The token sequence is processed by L decoder-style spiking Transformer blocks with H attention heads. Different from conventional spiking attention that directly learns sparse communication through a single parameterization, we aim to explicitly structure the internal communication patterns of spike-driven attention. To this end, we introduce *Overlapping Spiking Subspace Attention* (OSSA), which factorizes intra-head spike communication into multiple overlapping shared-private subspaces and adaptively composes their messages according to the current spiking and predictive context.

3.1 SPIKE-GUIDED OVERLAPPING SUBSPACE COMMUNICATION

At layer l, spike-driven query, key, and value representations are generated from X^{l-1} through Linear–Norm–LIF projections:

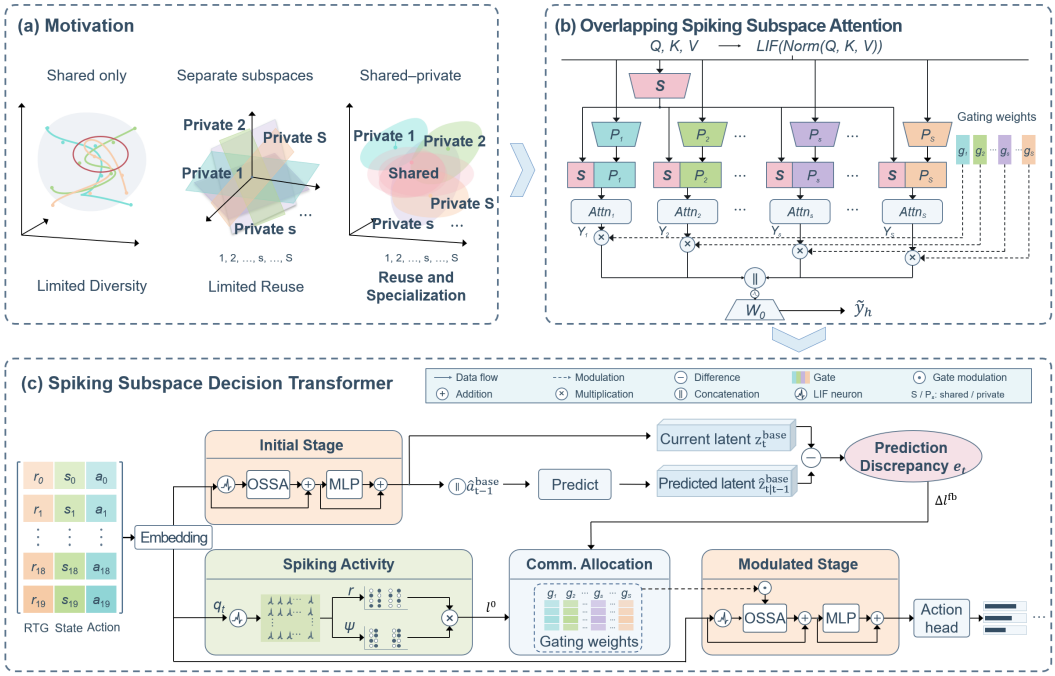

Figure 2: **Overview of the proposed Spiking Subspace Decision Transformer (SS-DT).** (a) Motivation for shared-private communication. (b) OSSA factorizes intra-head spike communication into multiple shared-private subspaces and composes their messages through communication weights g_s. (c) SS-DT first obtains an initial latent representation, from which the predictive discrepancy is derived. Spiking activity and predictive feedback jointly determine the communication allocation, which modulates OSSA in the subsequent stage to produce the final action.

$$Q^l = \mathrm{LIF}(\mathrm{Norm}(X^{l-1}W_Q^l)), \quad K^l = \mathrm{LIF}(\mathrm{Norm}(X^{l-1}W_K^l)), \quad V^l = \mathrm{LIF}(\mathrm{Norm}(X^{l-1}W_V^l)). \tag{2}$$

After multi-head reshaping, the representations of head h are denoted by $Q_h^l, K_h^l, V_h^l \in \mathbb{R}^{T\times d'_h}$, where $d'_h = T_s d_h$ and the spiking simulation dimension is folded into the head feature dimension. OSSA structures sparse communication within each attention head into S overlapping subspaces through shared-private projections. For $U \in \{Q, K, V\}$,

$$U_{h,s}^l = \mathrm{Concat}\left(U_h^l W_{U,\mathrm{sh}}^l,\, U_h^l W_{U,\mathrm{pr},s}^l\right), \qquad s = 1, \ldots, S, \tag{3}$$

where $W_{U,\mathrm{sh}}^l$ is shared across all subspaces within the same head, while $W_{U,\mathrm{pr},s}^l$ captures subspace-specific interaction patterns. The shared projection provides reusable communication components, whereas the private projections preserve diverse interaction structures. Because shared and private components are concatenated, the query-key correlation in each subspace can be decomposed as

$$Q_{h,s}^l (K_{h,s}^l)^\top = \underbrace{Q_{h,\mathrm{sh}}^l (K_{h,\mathrm{sh}}^l)^\top}_{C_{h,\mathrm{sh}}^l} + \underbrace{Q_{h,\mathrm{pr},s}^l (K_{h,\mathrm{pr},s}^l)^\top}_{C_{h,\mathrm{pr},s}^l}. \tag{4}$$

Therefore, all subspaces inherit a common communication structure $C_{h,\mathrm{sh}}^l$ while maintaining subspace-specific variations through $C_{h,\mathrm{pr},s}^l$. This overlapping parameterization enables sparse communication to simultaneously preserve reusable interaction patterns and specialized decision dependencies. Within each subspace, spike communication is computed as

$$A^l_{h,s} = Q^l_{h,s}(K^l_{h,s})^\top \odot M_{\text{causal}} \odot M_{\text{local}} \odot P^l_s,$$
$$Y^l_{h,s} = A^l_{h,s} V^l_{h,s}, \tag{5}$$

where M_{causal} enforces autoregressive dependency and M_{local} restricts communication to a local temporal neighborhood. The subspace-specific positional modulation is

$$P^l_s = 1 + \sigma(\alpha^l_{\text{pos}}) \tanh(B^l_s), \tag{6}$$

where B^l_s is a learnable positional bias. Given the token-wise communication weights $g^l_{t,h,s}$ introduced in Sec. 3.2, the subspace messages are composed as

$$Y^l_{\text{sub},h} = W^l_O \,\text{Concat}\left(g^l_{t,h,1} y^l_{t,h,1}, \ldots, g^l_{t,h,S} y^l_{t,h,S}\right)_{t=1}^{T}. \tag{7}$$

Since the communication weights softly modulate subspace messages rather than selecting a single subspace, multiple communication patterns can contribute simultaneously to the same decision token.

3.2 Context-Dependent Communication Allocation

While shared-private factorization defines a set of reusable and specialized communication patterns, their contributions should adapt to each decision token. As illustrated in Fig. 2(c), we determine the communication allocation from two complementary sources: the current spiking activity and predictive transition context.

Spike-guided allocation. The initial communication allocation is derived from the current query representation and spike statistics. Specifically, the allocation logit is computed as

$$\ell^{l,0}_{t,h,s} = [W^l_{\text{sub}} q^l_{t,h}]_s + \sigma(\alpha^l_{\text{sp}}) \left([W^l_{\text{rate}} r^l_t]_s + [W^l_{\text{time}} \psi^l_t]_s\right), \tag{8}$$

where $q^l_{t,h}$ denotes the query feature of token t in head h, while r^l_t and ψ^l_t summarize the firing rate and temporal spike distribution, respectively. This formulation allows the communication allocation to jointly consider semantic content and intrinsic spiking dynamics.

Predictive feedback modulation. Beyond instantaneous spike activity, decision making also depends on whether the current latent transition is consistent with the learned trajectory dynamics. Therefore, we introduce a lightweight predictive feedback signal to adapt communication allocation. In the initial stage, the spiking Transformer produces latent representations

$$Z^{\text{base}} = [z^{\text{base}}_1, \ldots, z^{\text{base}}_T], \tag{9}$$

together with the corresponding action predictions

$$\hat{a}^{\text{base}}_t = \pi_\theta(z^{\text{base}}_t). \tag{10}$$

A lightweight latent predictor estimates the current representation from the previous latent-action pair:

$$\hat{z}^{\text{base}}_{t|t-1} = F_{\text{pred}}([z^{\text{base}}_{t-1}; \hat{a}^{\text{base}}_{t-1}]),$$
$$e_t = \text{sg}(z^{\text{base}}_t) - \hat{z}^{\text{base}}_{t|t-1}, \qquad c_t = \tanh(W_c e_t). \tag{11}$$

Here, e_t measures the discrepancy between the observed latent transition and its prediction, while c_t provides predictive context for communication allocation. This predictive signal is only used to

modulate sparse communication and does not perform trajectory rollout or future planning. The predictive context produces a feedback modulation term:

$$\Delta\ell^{\mathrm{fb},l}_{t,h,s} = \sigma(\alpha^l_{\mathrm{fb}})\rho^l_{t,s}\kappa^l_t[W^l_{\mathrm{fb}}c_t]_s, \tag{12}$$

where

$$\rho^l_{t,s} = \sigma([W^l_\rho c_t]_s), \qquad \kappa^l_t = \exp\left(-\frac{\|c_t\|_2^2/d}{\mathrm{softplus}(\eta^l)+\epsilon}\right). \tag{13}$$

The term $\rho^l_{t,s}$ controls subspace-specific sensitivity to predictive feedback, while κ^l_t regulates the modulation strength according to the reliability of the predictive context. The final communication logits combine spike-guided allocation and predictive feedback:

$$\ell^l_{t,h,s} = \ell^{l,0}_{t,h,s} + \Delta\ell^{\mathrm{fb},l}_{t,h,s}. \tag{14}$$

The communication weights are obtained through mean-normalized sigmoid activation:

$$g^l_{t,h,s} = \frac{\sigma(\ell^l_{t,h,s}/\tau^l_g)}{\frac{1}{S}\sum_{j=1}^{S}\sigma(\ell^l_{t,h,j}/\tau^l_g)+\epsilon}. \tag{15}$$

Unlike mutually exclusive routing, this formulation allows multiple overlapping subspaces to contribute simultaneously to the same decision token. The initial stage provides predictive context c_t, which together with spike statistics determines the communication weights. In the subsequent modulated stage, the learned allocation adaptively composes OSSA subspace messages and produces the final latent sequence:

$$Z = [z_1, \ldots, z_T]. \tag{16}$$

3.3 Policy Learning

The final latent sequence is mapped to actions through

$$\hat{a}_t = \pi_\theta(z_t). \tag{17}$$

Following return-conditioned sequence modeling, SS-DT is optimized with a behavior cloning objective together with an auxiliary latent prediction objective:

$$\begin{aligned}\mathcal{L}_{\mathrm{BC}} &= \frac{\sum_{t=1}^{T} m_t\|\hat{a}_t - a_t\|_2^2}{\sum_{t=1}^{T} m_t + \epsilon},\\ \mathcal{L}_{\mathrm{pred}} &= \frac{\sum_{t=2}^{T} m_t\|\hat{z}^{\mathrm{base}}_{t|t-1} - \mathrm{sg}(z^{\mathrm{base}}_t)\|_2^2}{d\sum_{t=2}^{T} m_t + \epsilon}.\end{aligned} \tag{18}$$

where m_t masks padded trajectory positions. The overall training objective is

$$\mathcal{L} = \mathcal{L}_{\mathrm{BC}} + \beta_{\mathrm{pred}}\mathcal{L}_{\mathrm{pred}} + \mathcal{L}_{\mathrm{gate}}, \tag{19}$$

where $\mathcal{L}_{\mathrm{gate}}$ regularizes subspace activation, usage balance, and correlations among private projections.

Table 1: Results on MuJoCo and Adroit.

Tasks	BC	CQL	DT	FCNet	SpikeGPT	SpikeBert	TSSA	PSSA	SSDT
MuJoCo									
halfcheetah-m-e	35.8	62.4	86.8±1.3	91.2±0.3	23.6±4.5	24.3±6.0	91.3±0.2	91.5±0.3	**93.09±0.33**
walker2d-m-e	6.4	98.7	108.1±0.2	108.8±0.1	22.6±4.8	92.5±22.4	108.6±0.2	108.9±0.1	**109.99±0.27**
hopper-m-e	**111.9**	111.0	107.6±1.8	110.5±0.5	32.7±5.4	84.1±8.8	111.0±0.7	110.9±0.2	111.30±0.32
halfcheetah-m	36.1	**44.4**	42.6±0.1	42.9±0.4	26.9±0.8	20.0±3.5	42.5±0.4	42.8±0.3	43.10±0.15
walker2d-m	6.6	**79.2**	74.0±1.4	75.2±0.5	16.4±10.2	22.9±10.4	72.4±4.5	75.2±1.4	77.59±0.51
hopper-m	29.0	58.0	67.6±1.0	57.8±6.0	25.1±6.4	31.4±4.9	64.6±2.1	74.1±4.3	**77.28±3.18**
halfcheetah-m-r	38.4	**46.2**	36.6±0.8	39.8±0.8	21.8±2.0	32.2±8.0	38.7±1.1	38.8±0.7	40.25±0.10
walker2d-m-r	11.3	26.7	66.6±3.0	63.5±7.5	16.7±3.3	21.2±6.4	66.0±3.3	71.0±3.6	**74.63±3.55**
hopper-m-r	11.8	48.6	82.7±7.0	85.8±1.7	51.5±7.1	30.1±8.6	85.8±3.6	**96.3±1.3**	95.41±2.58
MuJoCo Mean ↑	31.9	63.9	74.7	75.1	26.4	39.9	75.7	78.8	**80.1**
Adroit									
pen-e	85.1	107.0	110.4±20.9	108.0±11.3	30.5±10.3	46.2±19.5	104.6±13.2	122.0±17.8	**129.87±3.02**
door-e	34.9	101.5	95.5±5.7	102.9±2.9	65.3±16.9	96.4±4.6	105.0±0.3	105.2±0.1	**105.49±0.23**
hammer-e	125.6	86.7	89.7±24.6	121.1±6.1	51.1±18.7	71.3±16.5	126.4±0.4	127.2±0.3	**127.64±0.38**
relocate-e	101.3	95.0	15.3±3.6	50.0±6.0	0.7±0.9	0.3±0.5	106.3±2.6	**108.4±2.2**	107.08±1.90
pen-h	34.4	37.5	-0.2±1.8	57.7±11.1	29.8±11.7	20.0±16.7	**89.7±10.0**	75.7±25.1	87.37±5.38
door-h	0.5	**9.9**	0.1±0.0	0.4±0.5	0.1±0.0	0.2±0.0	0.4±0.1	0.2±0.0	2.41±1.84
hammer-h	1.5	4.4	0.3±0.0	1.2±0.0	0.3±0.0	0.3±0.0	0.4±0.1	0.2±0.0	**4.88±0.58**
relocate-h	0.0	0.2	0.2±0.2	0.0±0.0	0.1±0.0	0.0±0.0	0.0±0.0	0.0±0.0	**0.54±0.16**
pen-c	56.9	39.2	22.7±17.1	50.4±24.1	17.0±22.0	17.6±29.0	41.1±19.7	44.8±14.7	**80.05±7.90**
door-c	-0.1	0.4	0.1±0.0	-0.2±0.0	0.2±0.0	0.2±0.0	0.4±0.8	0.0±0.0	**1.17±0.70**
hammer-c	0.8	**2.1**	0.3±0.0	0.2±0.0	0.3±0.0	0.3±0.5	0.2±0.0	0.2±0.0	2.00±0.62
relocate-c	-0.1	-0.1	-0.3±0.0	-0.2±0.0	**0.1±0.0**	0.0±0.5	-0.2±0.0	-0.2±0.0	0.03±0.05
Adroit Mean ↑	36.7	40.3	27.8	41.0	16.3	21.1	47.9	48.6	**54.0**

4 EXPERIMENTS

4.1 COMPARISON RESULTS

Benchmarks and Baselines. We evaluate SS-DT on the D4RL offline reinforcement learning benchmark, covering continuous-control tasks from MuJoCo locomotion and Adroit dexterous manipulation. All models are trained purely offline with the return-conditioned behavior cloning objective. Unless otherwise specified, we use trajectory segments of length $K = 20$ and four spiking Transformer blocks. Evaluation follows the standard D4RL protocol and reports normalized scores, where 0 and 100 correspond to random and expert-level performance, respectively.

Training and evaluation. All models are trained purely offline using the fixed D4RL datasets. The policy is optimized with a behavior-cloning-style action prediction objective under the return-conditioned sequence modeling framework. Unless otherwise specified, we use trajectory segments of length $K = 20$, four spiking transformer blocks. For MuJoCo experiments, we train for 50 iterations with 1000 gradient updates per iteration. Evaluation is performed periodically by rolling out the learned policy in the corresponding simulator. We report D4RL normalized scores, where a score of 0 corresponds to random policy performance and 100 corresponds to expert-level performance as defined by D4RL.

Overall Performance. Table 1 compares SS-DT with representative offline RL methods and recent spiking sequence models on MuJoCo and Adroit benchmarks. SS-DT achieves the best average performance among spiking offline RL methods, obtaining normalized returns of 80.1 on MuJoCo and 54.0 on Adroit. Compared with the strongest spiking baseline PSSA, SS-DT improves the average return by 1.3 and 5.4 points on MuJoCo and Adroit, respectively. On MuJoCo, SS-DT outperforms PSSA (80.1 vs. 78.8) and TSSA (80.1 vs. 75.7), with notable gains on diverse trajectory datasets such as Walker2d-m (77.59 vs. 75.2) and Walker2d-m-r (74.63 vs. 71.0). On Adroit, SS-DT achieves an average return of 54.0, improving over PSSA (48.6) and TSSA (47.9), with substantial gains on challenging manipulation tasks such as pen-c (80.05 vs. 44.8 of PSSA). These improvements indicate that adaptive overlapping communication provides additional flexibility for modeling heterogeneous decision dependencies. Overall, SS-DT extends spiking sequence modeling beyond efficient attention computation by enabling adaptive internal communication, leading to consistent improvements over existing spiking offline RL methods.

Table 2: Ablation study of SS-DT components on representative MuJoCo tasks.

Model	Shared-Private	Spike Allocation	Predictive Feedback	Capacity	Hopper-m-e	HC-m-r	Walker-m-e	Walker-m	Walker-m-r
PSSA	×	×	×	×	110.9	38.8	108.9	75.2	71.0
+ Shared-Private	✓	×	×	×	108.8	39.9	108.4	75.5	73.8
+ Spike Allocation	✓	✓	×	×	110.9	40.1	109.7	76.2	74.4
PSSA + Heads	×	×	×	✓	111.0	39.1	109.2	75.8	73.0
SS-DT	✓	✓	✓	✓	**111.3**	**40.3**	**110.0**	**77.6**	**74.6**

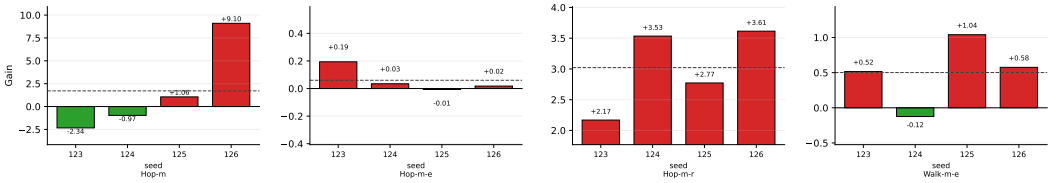

Figure 3: **Performance gains across random seeds.** We report the normalized-return improvement over the corresponding spiking baseline for random seeds (123–126) on representative MuJoCo tasks. Positive values indicate improvements over the baseline, while negative values indicate performance decreases. Dashed lines denote the average gain across the four seeds.

4.2 Ablation Results

Ablation of communication components. Table 2 evaluates the contribution of each SS-DT component. Adding shared-private decomposition improves performance by introducing structured communication subspaces, while spike-guided allocation further improves results through adaptive message composition. The capacity-control variant with additional attention heads does not consistently improve over PSSA, indicating that the gains come from communication organization rather than simply increasing model capacity. The complete SS-DT achieves the best performance across all tasks.

Robustness across random seeds. Figure 3 evaluates SS-DT across four random seeds on representative MuJoCo tasks. Although the improvement magnitude varies across different trajectory distributions, SS-DT consistently achieves positive gains on most tasks, particularly on challenging medium-replay datasets. The variance across seeds remains limited compared with the overall performance improvement, indicating that the learned adaptive communication patterns are not sensitive to initialization. These results demonstrate the robustness of SS-DT under different training stochasticity.

4.3 Analysis of Adaptive Subspace Communication

Context-dependent communication diversity. To investigate how OSSA organizes sparse spike communication, we analyze subspace allocation and communication-support similarity across different return-ranked trajectory contexts. As shown in Fig. 4, OSSA consistently activates multiple overlapping subspaces rather than relying on a single communication component. HalfCheetah-m-r exhibits balanced subspace contributions, while Walker2d-m-r shows larger variations across layers, indicating flexible composition of communication patterns. We further measure top-support Jaccard similarity across trajectory contexts. HalfCheetah-m-r maintains high support overlap (0.68–0.76), while Walker2d-m-r shows larger variations between distant contexts. These results suggest that sparse spike communication contains both reusable structures and context-dependent variations, supporting the shared-private factorization and adaptive composition design of OSSA.

Shared-private communication decomposition. Figure 5 visualizes the shared-private communication components learned by OSSA. The shared components preserve consistent local interaction structures across subspaces, while the private components capture subspace-specific variations. The learned gates further compose these components flexibly; for example, HC-m-e activates all four subspaces with gate values of 1.94, 0.68, 0.69, and 0.69, whereas other tasks exhibit different subspace preferences. These results show that OSSA combines reusable communication structures with adaptive subspace specialization.

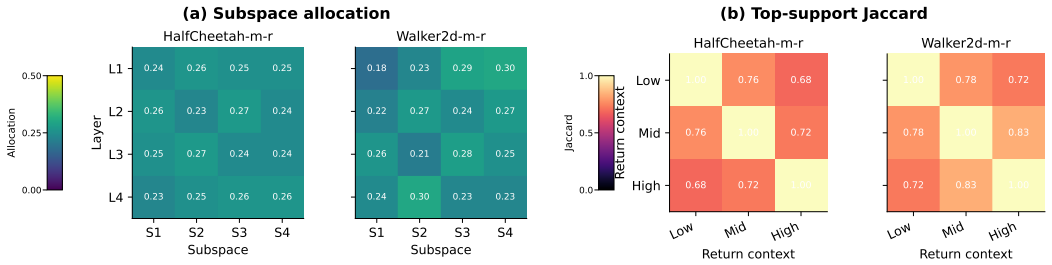

Figure 4: **Analysis of subspace allocation and context-dependent communication.** The left part shows the learned subspace allocation across four OSSA layers and four overlapping subspaces, where each value denotes the normalized communication contribution of a subspace. The right part presents the top-support Jaccard similarity between trajectories with different return-ranked contexts.

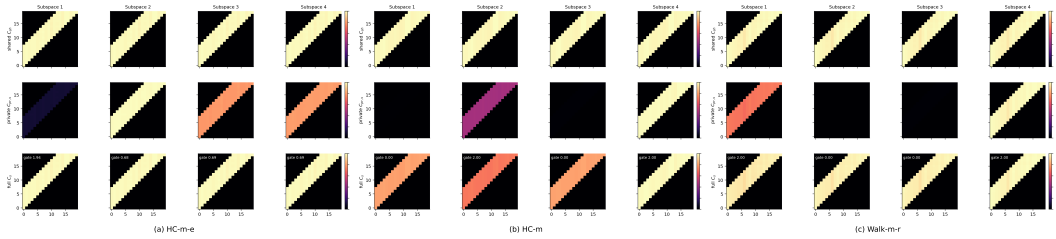

Figure 5: **Visualization of shared-private spike communication and subspace composition.** Four OSSA communication subspaces are visualized on HC-m-e, HC-m, and Walker2d-m-r. From top to bottom, each group shows the shared component, subspace-specific private component, and the resulting gated communication pattern, with learned gate values annotated for each subspace. Shared components capture common local interaction structures, while private components introduce subspace-specific variations. The learned gates compose multiple communication patterns without exclusive routing.

Communication diversity analysis. We further quantify the diversity of subspace communication in Appendix. The analysis shows that OSSA does not collapse into either a fully shared communication pattern or exclusive subspace selection. Instead, different tasks exhibit different degrees of multi-subspace participation: some tasks maintain broad activation across multiple subspaces, while others favor more specialized communication compositions. This behavior demonstrates that the overlapping shared-private design provides a flexible communication spectrum between reuse and specialization, allowing spiking representations to adapt their internal communication structure according to task-dependent decision requirements.

5 Conclusion

We presented Spiking Subspace Decision Transformer (SS-DT), a spiking sequence model for offline reinforcement learning that enables structured and adaptive sparse communication through Overlapping Spiking Subspace Attention (OSSA). Beyond improving performance, our analyses reveal that spike communication patterns are neither fully shared nor completely independent: different trajectory contexts exhibit partially overlapping communication structures with reusable and specialized components. Moreover, capacity-control experiments show that improving spiking decision models requires better communication organization rather than simply increasing model capacity. Across 21 MuJoCo and Adroit tasks, SS-DT achieves state-of-the-art average performance among spiking offline RL methods, improving over the strongest spiking baseline by 3.7 normalized-return points on average. These results highlight the potential of biologically inspired communication structures for scaling spiking neural networks and developing more capable embodied decision agents.

AI USE STATEMENT

In this work, we used generative AI tools for language refinement and proofreading of the abstract and introduction to improve clarity and readability. We have not used generative AI tools for generating research ideas, designing methods, conducting experiments, analyzing results, or producing scientific claims. We have carefully reviewed and revised all AI-assisted content and take full responsibility for the final content of this work, including all text, claims, and artifacts.

REPRODUCIBILITY STATEMENT

Detailed implementation details, experimental settings, and training configurations are provided in the Appendix. We include the source code in the supplementary materials to facilitate reproducibility. Upon publication, we will further release the code in a public GitHub repository. The released implementation will contain the necessary components, training scripts, and evaluation protocols to reproduce the reported results.

REFERENCES

Lili Chen, Kevin Lu, Aravind Rajeswaran, Kimin Lee, Aditya Grover, Misha Laskin, Pieter Abbeel, Aravind Srinivas, and Igor Mordatch. Decision transformer: Reinforcement learning via sequence modeling. In *Advances in neural information processing systems*, volume 34, pp. 15084–15097, 2021.

Scott Fujimoto, David Meger, and Doina Precup. Off-policy deep reinforcement learning without exploration. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pp. 2052–2062, 2019.

Yufei Guo, Xiaode Liu, Yuanpei Chen, Weihang Peng, Yuhan Zhang, and Zhe Ma. Spiking transformer: Introducing accurate addition-only spiking self-attention for transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Devamanyu Hazarika, Roger Zimmermann, and Soujanya Poria. MISA: Modality-invariant and -specific representations for multimodal sentiment analysis. In *Proceedings of the 28th ACM International Conference on Multimedia*, pp. 1122–1131, 2020.

Wei Huang, Qinying Gu, and Nanyang Ye. Decision SpikeFormer: Spike-driven transformer for decision making. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19241–19250. IEEE, 2025.

Michael Janner, Qiyang Li, and Sergey Levine. Offline reinforcement learning as one big sequence modeling problem. In *Advances in Neural Information Processing Systems*, volume 34, 2021.

Ilya Kostrikov, Ashvin Nair, and Sergey Levine. Offline reinforcement learning with implicit q-learning. In *International Conference on Learning Representations*, 2022.

Aviral Kumar, Justin Fu, Matthew Soh, George Tucker, and Sergey Levine. Stabilizing off-policy q-learning via bootstrapping error reduction. In *Advances in Neural Information Processing Systems*, volume 32, 2019.

Aviral Kumar, Aurick Zhou, George Tucker, and Sergey Levine. Conservative q-learning for offline reinforcement learning. In *Advances in Neural Information Processing Systems*, volume 33, pp. 1179–1191, 2020.

Donghyun Lee, Yuhang Li, Youngeun Kim, Shiting Xiao, and Priyadarshini Panda. Spiking transformer with spatial-temporal attention. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 13948–13958, 2025.

Pengfei Liu, Xipeng Qiu, and Xuanjing Huang. Adversarial multi-task learning for text classification. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*, pp. 1–10, 2017. doi: 10.18653/v1/P17-1001.

Changze Lv, Yansen Wang, Dongqi Han, Yifei Shen, Xiaoqing Zheng, Xuanjing Huang, and Dongsheng Li. Toward relative positional encoding in spiking transformers. In *Advances in Neural Information Processing Systems*, volume 38, 2025.

Joan Puigcerver, Carlos Riquelme Ruiz, Basil Mustafa, and Neil Houlsby. From sparse to soft mixtures of experts. In *International Conference on Learning Representations*, 2024.

Kexin Shi, Hanwen Liu, Zeyang Song, Yang Liu, Jieyuan Zhang, Shuai Wang, Jibin Wu, Malu Zhang, and Yang Yang. Temporal interaction in spiking transformers with multi-delay mixer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2026.

Korneel Van den Berghe, Stein Stroobants, Vijay Janapa Reddi, and Guido De Croon. Adaptive surrogate gradients for sequential reinforcement learning in spiking neural networks. In *Advances in Neural Information Processing Systems*, volume 38, pp. 147904–147926, 2025.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 30, pp. 5998–6008, 2017.

Yichen Xiao, Shuai Wang, Dehao Zhang, Wenjie Wei, Yimeng Shan, Xiaoli Liu, Yulin Jiang, and Malu Zhang. Rethinking spiking self-attention mechanism: Implementing α-XNOR similarity calculation in spiking transformers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2025.

Man Yao, JiaKui Hu, Zhaokun Zhou, Li Yuan, Yonghong Tian, Bo Xu, and Guoqi Li. Spike-driven transformer. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Hongling Zheng, Li Shen, Yong Luo, Deheng Ye, Bo Du, Jialie Shen, and Dacheng Tao. Decision mixer: Integrating long-term and local dependencies via dynamic token selection for decision-making. In *Forty-second International Conference on Machine Learning*, 2025.

Chenlin Zhou, Han Zhang, Zhaokun Zhou, Liutao Yu, Liwei Huang, Xiaopeng Fan, Li Yuan, Zhengyu Ma, Huihui Zhou, and Yonghong Tian. QKFormer: Hierarchical spiking transformer using Q-K attention. In *Advances in Neural Information Processing Systems*, volume 37, 2024a.

Zhaokun Zhou, Yuesheng Zhu, Chao He, Yaowei Wang, Shuicheng Yan, Yonghong Tian, and Li Yuan. Spikformer: When spiking neural network meets transformer. In *International Conference on Learning Representations*, 2023.

Zhaokun Zhou, Yijie Lu, Yanhao Jia, Kaiwei Che, Jun Niu, Liwei Huang, Xinyu Shi, Yuesheng Zhu, Guoqi Li, Zhaofei Yu, and Li Yuan. Spiking transformer with experts mixture. In *Advances in Neural Information Processing Systems*, volume 37, 2024b.