BEYOND FORECASTING: WORLD MODELS FOR COUPLED RADIO NETWORK CONTROL

Anonymous authors
Paper under double-blind review

ABSTRACT

Model-based RAN control requires predicting how future resource allocations reshape network dynamics, including effects that persist after an intervention has passed. We propose TRACTOR+, an action-conditioned world model that separates recurrent network-state evolution from persistent action-response dynamics. TRACTOR+ combines multi-scale recurrent states, an instantaneous action-conditioned predictor, and adaptive fast/medium/slow response memories for free-running multi-horizon rollout. Experiments on three-slice RAN traces show that TRACTOR+ achieves the lowest standardized RMSE across all evaluated natural and allocation-event horizons with only 0.193M parameters. Removing future-action inputs degrades event prediction, while mechanism ablations show that persistent-response memory strengthens action-path discrimination. The learned dynamics also support short-horizon offline planning within the evaluated Near-RT budget. All alternative-action results are interpreted as predictive diagnostics under observational support rather than causal or closed-loop effects.

1 INTRODUCTION

Radio access networks increasingly operate under heterogeneous traffic demands, service requirements, and dynamically changing resource configurations. Accurate prediction is therefore important for proactive network management, allowing operators to anticipate traffic and key performance indicator (KPI) evolution before congestion or service degradation occurs Zhang et al. (2019); Li et al. (2017); Yu et al. (2018). Recent forecasting models have improved long-range and multi-scale temporal modeling through Transformer and structured state-space architectures Liu et al. (2024); Rezazadeh et al. (2025b). These advances provide strong predictive models of network evolution from historical observations.

Prediction for control, however, requires a different capability. A Near-RT controller must reason about how the network may evolve under different future resource-allocation trajectories rather than extrapolating a single future from the observed history. World models provide a natural framework for this purpose by learning action-conditioned dynamics that can be rolled forward under candidate controls Ha & Schmidhuber (2018); Hafner et al. (2019a); Hansen et al. (2024); Zhang et al. (2023). This perspective has recently been introduced into wireless-network intelligence, including mobile traffic extrapolation under network parameter adjustments Qi et al. (2026), action-conditioned O-RAN state-space modeling and planning Rezazadeh et al. (2025a), and learned state–action models for mobile-network control Bouton et al. (2026).

Yet a less explored question remains: *should the predictive response to a past control action be represented only through the recurrent network state?* Consider a one-step resource transfer between network slices followed by several HOLD actions. Although the allocation change occurs at one step, the subsequent throughput and buffer trajectory may continue to reflect the network response over multiple later steps. Queue drainage, scheduling adaptation, and traffic redistribution can all evolve more slowly than the control input itself. A conventional recurrent world model can retain this information implicitly in its hidden state, but the remaining response is then entangled with ordinary state evolution. This challenge is particularly relevant for offline network traces, where limited exploration can already make learned dynamics insensitive to control inputs Bouton et al. (2026).

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 illustrates this distinction. A recurrent predictor must implicitly preserve both the evolving network condition and the remaining response to previous allocation changes in a shared hidden representation. We instead argue that persistent action response is a distinct predictive object and should be modeled explicitly alongside ordinary network-state dynamics.

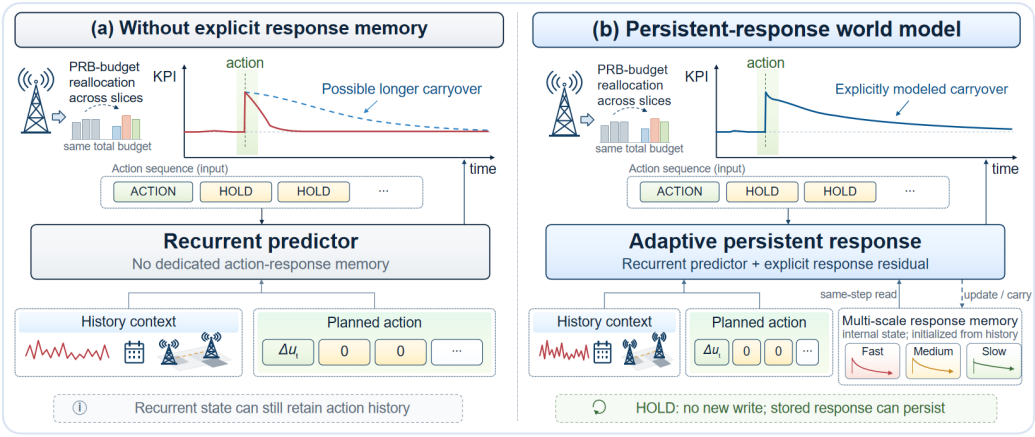


Figure 1: Motivation for persistent-response modeling. A conventional recurrent predictor stores the remaining response to a previous allocation action implicitly in its hidden state. TRACTOR+ introduces dedicated multi-timescale response memories, allowing predictive carryover to remain available during subsequent HOLD steps.

To address this problem, we propose TRACTOR+, an action-conditioned world model that separates recurrent network-state evolution from persistent action-response dynamics. TRACTOR+ maintains fast, medium, and slow recurrent states to capture multi-scale network dynamics, together with an instantaneous action-conditioned pathway for the current allocation step. A dedicated persistent-response pathway writes allocation changes into adaptive fast, medium, and slow memories, whose retention depends on the evolving network state and supplied action. The two pathways are combined during free-running rollout, allowing a previous allocation response to remain explicitly represented even after later controls return to HOLD.

We evaluate TRACTOR+ on offline three-slice RAN traces using factual multi-horizon rollout, future-action diagnostics, predictive planning, and controlled mechanism ablations. TRACTOR+ achieves the lowest standardized RMSE across all evaluated natural and strict allocation-event horizons while using only 0.193M parameters. On strict events, it reduces RMSE relative to TD-MPC2-WM by 3.6%, 6.8%, and 4.7% at $H = 2$, 4, and 8, respectively. Removing future-action information degrades event prediction, while removing persistent-response memory weakens action-path discrimination. The frozen model also supports short-horizon offline planning within the evaluated Near-RT latency budget. All alternative-action results are interpreted as predictive diagnostics under observational support rather than identified causal or measured closed-loop effects.

Our contributions are summarized as follows:

- We identify persistent action response as a distinct modeling problem in action-conditioned RAN prediction, where the network response following an allocation change may remain predictive after the action step itself has passed.
- We propose TRACTOR+, which separates multi-scale recurrent state dynamics from instantaneous and adaptive multi-timescale persistent-response dynamics for free-running prediction under future allocation trajectories.
- We evaluate TRACTOR+ through multi-horizon prediction, future-action sensitivity, offline predictive planning, and mechanism ablations, demonstrating improved predictive accuracy and stronger action-path discrimination around allocation events.

2 Related Work

Network Forecasting. Data-driven network prediction has been studied with recurrent, spatio-temporal, Transformer, and state-space models Li et al. (2017); Yu et al. (2018); Liu et al. (2024);

Rezazadeh et al. (2025b). These approaches are effective for extrapolating future traffic or KPIs from historical observations, while recent cellular models such as MS3M further emphasize multi-scale dynamics and Near-RT efficiency Rezazadeh et al. (2025b). Their primary objective, however, is forecasting rather than predicting how the network evolves under an explicitly specified future control trajectory.

World Models and Predictive Control. World models learn action-conditioned dynamics that can be rolled forward before interacting with the real environment Ha & Schmidhuber (2018); Hafner et al. (2019a). Recent architectures improve latent imagination and predictive control through compact action-conditioned dynamics or stochastic Transformer-based transitions Hansen et al. (2024); Zhang et al. (2023). Long-horizon consistency has also motivated mechanisms such as DMWM Wang et al. (2025). TRACTOR+ focuses on a different aspect of rollout dynamics: the response to an action may persist after the action step itself has passed, and we explicitly represent this carryover rather than relying only on a generic recurrent latent state.

World Models for Wireless Networks. Action-conditioned world modeling has recently been explored for wireless prediction and control. MobiWM models network-parameter adjustments as actions for mobile traffic rollout Qi et al. (2026), while WM–MS3M combines action-conditioned state-space dynamics with short-horizon planning Rezazadeh et al. (2025a). NetWorld studies diffusion-based world models for wireless multi-agent resource allocation Meng et al. (2026), and Bouton *et al.* learn reusable mobile-network state–action dynamics from historical operational data Bouton et al. (2026). These works establish the importance of explicit action conditioning. TRACTOR+ instead concentrates on the temporal structure of the action response, separating immediate effects from adaptive multi-timescale persistent carryover.

3 METHODOLOGY

We present **TRACTOR+**, an action-conditioned world model for multi-horizon RAN dynamics. Its key design is to distinguish ordinary network-state evolution from the response induced by resource-allocation actions. The former is represented by multi-scale recurrent states, while the latter is modeled through an instantaneous prediction path and dedicated persistent-response memories.

Figure 2 illustrates the overall architecture. Given a historical network window, TRACTOR+ first warms up the recurrent state and response memories using the observed KPI and allocation history. During future rollout, each planned allocation is supplied to the same world-model cell. The current action updates the network state, produces an immediate predictive response, and is written into multi-timescale response memory. The resulting KPI prediction is fed back to the next rollout step, allowing the model to generate a free-running trajectory under an arbitrary feasible future action path.

3.1 ACTION-CONDITIONED MULTI-SCALE DYNAMICS

Network KPIs contain dynamics at different temporal scales. TRACTOR+ therefore maintains fast, medium, and slow recurrent states for each slice. During history encoding, the observed network features and allocation history update these states and provide the initial context for future rollout.

At rollout step k, the planned allocation transition is encoded together with the current network context. The action representation, the previous prediction, and the recurrent state are used to update the three state pathways, whose fused representation is denoted by $\mathbf{z}_k$. This makes the state transition explicitly dependent on the supplied future allocation rather than extrapolating only from the observed history.

The updated state is decoded by an instantaneous branch, $\Delta\widehat{\mathbf{y}}_k^{\mathrm{inst}}$, which captures the response associated with the current allocation step. This branch is intended for effects that appear immediately after a transfer. Effects that remain after the action itself has passed are handled separately by the persistent-response pathway described below.

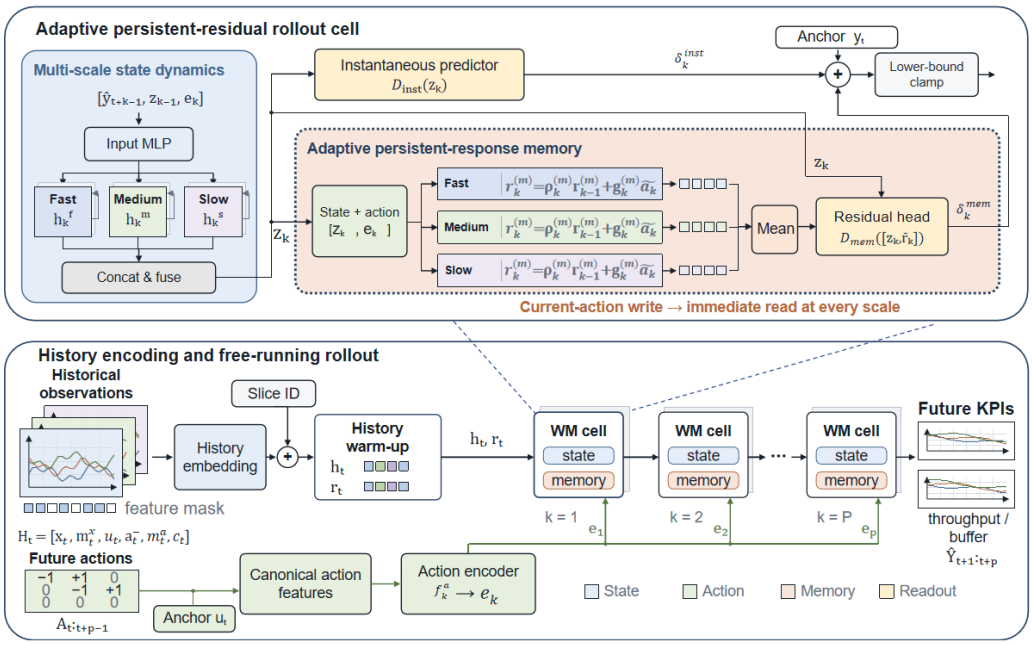


Figure 2: Overview of TRACTOR+. Historical observations initialize multi-scale recurrent states and persistent-response memories. Each future allocation action updates the world-model cell, and predicted throughput and buffer states are fed back autoregressively during rollout.

3.2 Adaptive Persistent-Response Memory

A resource transfer may influence throughput and buffer dynamics for several subsequent steps even when later actions are HOLD. Encoding this effect only inside the recurrent state requires the model to implicitly preserve both ordinary network dynamics and the remaining action response in the same representation.

TRACTOR+ instead maintains fast, medium, and slow response memories. For response scale m, the current state $\mathbf{z}_k$ and action embedding $\mathbf{e}_k$ adapt both the retention of the previous response and the write induced by the current signed allocation change $\widetilde{\mathbf{a}}_k$:

$$\begin{aligned} \rho_k^{(m)} &= \mathcal{B}\big(\beta_m + \alpha \tanh\left(\mathcal{D}_m([\mathbf{z}_k, \mathbf{e}_k])\right)\big), \\ \mathbf{r}_k^{(m)} &= \rho_k^{(m)} \odot \mathbf{r}_{k-1}^{(m)} + \mathcal{G}_m([\mathbf{z}_k, \mathbf{e}_k]) \odot \widetilde{\mathbf{a}}_k, \end{aligned} \tag{1}$$

where β_m is a learned scale-specific base retention, $\mathcal{B}(\cdot)$ bounds the retention factor, and $\mathcal{G}_m$ controls the current action write.

This update provides the behavior illustrated in Figure 3. A transfer is written into response memory at the step when it occurs. If the following action is HOLD, there is no new signed-action write, while the previously stored response can remain according to its adaptive retention. The three memory scales allow short- and longer-lived responses to coexist, rather than enforcing a single decay pattern.

The three response memories are fused and decoded together with the current recurrent state to produce a persistent residual $\Delta\widehat{\mathbf{y}}_k^{\mathrm{mem}}$. The same decay-and-write mechanism is also applied during history warm-up, so future rollout starts from a response state consistent with the observed allocation history.

3.3 Persistent-Residual Rollout

TRACTOR+ combines the instantaneous and persistent responses relative to the last observed KPI state:

$$\widehat{\mathbf{y}}_{t+k} = \mathbf{y}_t^{\mathrm{anchor}} + \Delta\widehat{\mathbf{y}}_k^{\mathrm{inst}} + \Delta\widehat{\mathbf{y}}_k^{\mathrm{mem}}. \tag{2}$$

A physical lower bound is applied to the predicted throughput and buffer values in the standardized space.

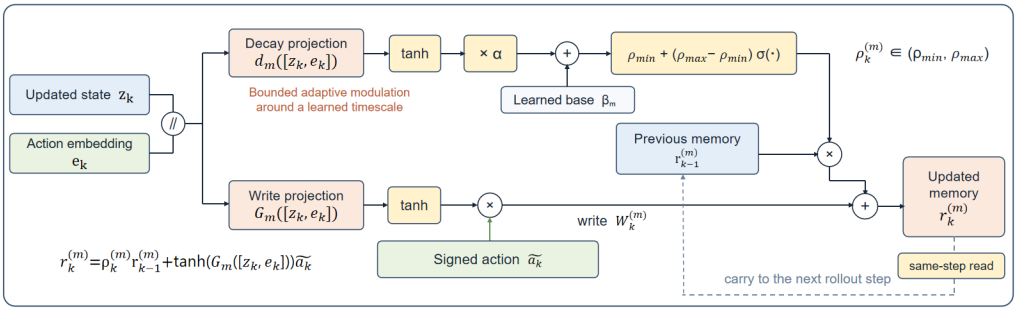


Figure 3: Adaptive persistent-response memory. The current network state and action determine the retention of the previous response and the write generated by the current allocation change. Stored responses can remain active during subsequent HOLD steps.

The prediction is then used as the state input for the next world-model cell. Future ground-truth KPIs are therefore unavailable after the historical context, and the complete horizon is generated autoregressively. This free-running design allows the same frozen model to evaluate different future allocation paths from a common observed network state.

3.4 Progressive Training and Predictive Planning

The instantaneous and persistent pathways are optimized progressively to keep their roles separated during learning. We train the instantaneous component for 25 epochs, introduce persistent-response learning for another 15 epochs, and jointly refine the complete model for 10 epochs with a reduced learning rate. Model selection uses only natural factual free-running validation performance; alternative-action and TEST-event evaluations do not participate in checkpoint selection.

The trained world model can also be used for short-horizon predictive planning. For each observed network state, we roll out candidate paths formed by HOLD and the six feasible one-unit transfers and score their predicted throughput–buffer trade-off under the frozen planning objective. Only feasible, TRAIN-supported, and physically valid paths are eligible for selection. Section 4.4 evaluates whether increasing the imagination horizon changes the resulting decision. The planner is used as an offline diagnostic of the learned dynamics, rather than as evidence of causal or measured closed-loop improvement.

4 Experiments

We evaluate TRACTOR+ through four questions that progressively examine predictive accuracy, action conditioning, model-based planning, and the mechanisms responsible for persistent action responses.

RQ1 Can TRACTOR+ accurately roll out network dynamics over multiple horizons, particularly around resource-allocation events?

RQ2 Does TRACTOR+ make predictive use of the supplied future action trajectory?

RQ3 Does longer model imagination lead to different and higher-valued predictive planning decisions while remaining compatible with Near-RT operation?

RQ4 Which components of the persistent-response design are responsible for its action-sensitive rollout behavior?

4.1 Experimental Setup

Dataset and prediction protocol. We evaluate on an offline three-slice RAN dataset sampled every 250 ms. The network state contains slice-level downlink throughput and downlink-buffer measurements together with historical network context. Control actions reallocate a fixed resource budget across the three slices. The action space contains HOLD and six one-unit transfers between slice pairs. Feasible actions preserve the total allocation budget of 17 units, keep every allocation nonnegative, and retain at least one unit for the second slice.

All models observe a history of $L = 16$ steps and generate a free-running rollout of up to $P = 8$ future steps. We report $H \in \{2, 4, 8\}$ to expose how predictive behavior changes with rollout depth. Training, validation, and test trials are disjoint, and normalization statistics are estimated from the training split only. The natural test set contains 16,093 windows from 43 trials. We additionally construct a strict allocation-event cohort containing a single resource transfer followed by HOLD for the remainder of the $P = 8$ rollout. This cohort contains 91 windows from 20 trials and isolates the transient and persistent dynamics following a clearly localized resource-allocation event.

Baselines. We compare TRACTOR+ against forecasting architectures and native world models under the same predictive interface. DLinear-WM Zeng et al. (2023), iTransformer-WM Liu et al. (2024), TimeKAN-WM Huang et al. (2025), and MS3M-WM Rezazadeh et al. (2025b) reuse the corresponding forecasting backbone and are equipped with the same future-action conditioning interface used by the forecasting baselines. RSSM-WM Hafner et al. (2019b), TD-MPC2-WM Hansen et al. (2024), and STORM-WM Zhang et al. (2023) retain their native action-conditioned world-model dynamics while excluding policy- or reward-specific components that are not required for the predictive comparison. Every method receives the same historical inputs, future action paths, target variables, train/validation/test splits, and validation-only checkpoint selection protocol.

Metrics. Our primary metric is joint standardized RMSE over the six predicted KPI dimensions. We also report MSE skill relative to a last-observation persistence predictor,

$$\mathrm{Skill}_{\mathrm{MSE}} = 1 - \frac{\mathrm{MSE}_{\mathrm{model}}}{\mathrm{MSE}_{\mathrm{persistence}}}, \tag{3}$$

where positive values indicate improvement over persistence.

To measure whether a model actually uses the supplied future action path, we define

$$\Delta_{\mathrm{alt}}(H) = \mathrm{RMSE}_{\mathrm{alt}}(H) - \mathrm{RMSE}_{\mathrm{factual}}(H), \tag{4}$$

where the alternative trajectory can be HOLD, shuffled actions, or a same-family model without future-action inputs. A positive value means that the logged factual action trajectory predicts the logged future more accurately. Confidence intervals are computed by trial-cluster bootstrap so that windows from the same trial are not treated as independent samples.

Training and model selection. TRACTOR+ is trained with the frozen progressive schedule used throughout the study: 25 epochs for the instantaneous branch, 15 epochs for persistent-response learning, and 10 epochs of lower-learning-rate joint optimization. The initial learning rate is 10^{-3} and is reduced to 5×10^{-4} during joint training. The checkpoint is selected solely according to natural factual free-running $P = 8$ validation error. The random seed is fixed to 11. TEST event labels, HOLD/shuffled trajectories, ablation outcomes, and planning objectives are never used for checkpoint selection.

4.2 RQ1: Multi-Horizon Predictive Dynamics

Table 1 summarizes predictive performance on natural network evolution and strict allocation-event windows. TRACTOR+ achieves the lowest joint standardized RMSE at all six reported endpoints.

On natural dynamics, the improvement is consistent but moderate. Relative to the strongest competing model at each horizon, TRACTOR+ reduces RMSE by approximately 3.3% at $H = 2$, 3.9% at $H = 4$, and 1.4% at $H = 8$. This result is important because TRACTOR+ is not specialized only to event windows; its explicit action-response mechanism does not sacrifice ordinary free-running state prediction.

The separation becomes clearer after resource reallocation. TD-MPC2-WM is the strongest competing model on all three strict-event horizons, yet TRACTOR+ reduces its RMSE from 0.7954 to 0.7669 at $H = 2$, from 0.9722 to 0.9061 at $H = 4$, and from 1.4172 to 1.3512 at $H = 8$, corresponding to relative reductions of approximately 3.6%, 6.8%, and 4.7%. The larger advantage around allocation events is consistent with the design of TRACTOR+, which explicitly separates instantaneous action-conditioned prediction from persistent response memory.

TRACTOR+ also obtains a natural persistence skill of 0.1846 at $H = 8$, corresponding to an 18.5% MSE reduction relative to a last-observation predictor. Its parameter count remains 0.193M, substantially below the native TD-MPC2-WM and STORM-WM baselines.

Table 1: Multi-horizon predictive dynamics performance. We report joint standardized RMSE on natural test windows and strict allocation-event windows. Persistence Skill@H8 measures the MSE improvement over persistence on natural dynamics. The best RMSE is shown in bold and the second-best is underlined.

Model	Natural Dynamics			Strict Allocation-Event Dynamics			Persistence Skill@H8	Params (M)
	H2	H4	H8	H2	H4	H8		
DLinear-WM	0.9410	1.1896	1.3282	1.0044	1.1863	1.6396	-0.0175	0.0045
iTransformer-WM	0.9649	1.1418	1.3363	1.0217	1.1162	1.6327	-0.0310	0.2726
TimeKAN-WM	0.8009	<u>0.9931</u>	<u>1.2090</u>	0.8522	0.9873	1.5334	<u>0.1602</u>	0.0138
MS3M-WM	0.8061	1.0085	1.2401	0.8814	1.0055	1.4700	0.1154	2.5389
RSSM-WM	0.9072	1.0566	1.2529	0.8926	1.0208	1.5216	0.0961	0.8239
TD-MPC2-WM	<u>0.7823</u>	0.9954	1.2376	<u>0.7954</u>	<u>0.9722</u>	<u>1.4172</u>	0.1227	1.7551
STORM-WM	0.8368	1.0379	1.2696	0.8408	1.0161	1.4621	0.0754	10.3496
TRACTOR+	**0.7561**	**0.9549**	**1.1926**	**0.7669**	**0.9061**	**1.3512**	**0.1846**	0.1925

4.3 RQ2: Predictive Use of Future Actions

Accurate factual rollout alone does not establish that a world model uses the supplied future action trajectory. A model could obtain low error primarily from temporal persistence while largely ignoring its action input. We therefore evaluate action conditioning independently of the main accuracy comparison.

Figure 4 compares TRACTOR+ with a same-family *NoFutureAction* variant that preserves the model capacity, optimization schedule, and evaluation protocol while removing access to future intervention inputs. The plotted quantity is the paired increase in RMSE caused by removing the future-action channel.

The difference is small on ordinary natural windows but becomes substantial around resource-allocation events. On the strict cohort, removing future actions increases standardized RMSE by 0.0302 at $H = 2$, 0.0698 at $H = 4$, and 0.0626 at $H = 8$. Their 95% confidence intervals are $[0.0129, 0.0442]$, $[0.0263, 0.1010]$, and $[0.0011, 0.1234]$, respectively. All three intervals remain above zero. Natural windows exhibit much smaller changes, indicating that the action channel is most useful precisely when an allocation decision changes the subsequent network trajectory.

The frozen full model yields the same qualitative conclusion when the supplied action trajectory is modified only at evaluation time. On strict-event windows, factual trajectories outperform both HOLD and shuffled alternatives at $H = 2$, $H = 4$, and $H = 8$, with positive bootstrap lower bounds throughout. At $H = 4$, for example, the factual-versus-HOLD and factual-versus-shuffled RMSE margins are 0.0762 and 0.0821, respectively.

These results establish predictive dependence on the supplied action path. They should not be interpreted as identified causal effects: all evaluations remain offline and are supported by observational network trajectories rather than randomized interventions.

4.4 RQ3: Multi-Horizon Offline Predictive Planning

The preceding results show that TRACTOR+ predicts different futures under different action paths. We next examine whether this distinction is large enough to affect model-based decision making.

The planner evaluates HOLD together with six feasible one-unit resource transfers. Greedy-$H1$ evaluates seven one-step choices, MPC-$H2$ evaluates 7^2 two-step sequences, and MPC-$H4$ evaluates 7^4 four-step sequences. Every candidate is evaluated in batch by the same frozen TRACTOR+ model and the same objective. Logged TEST future targets are not used during action selection.

Figure 5a shows that longer imagination substantially increases the predicted separation between the selected sequence and HOLD. On natural anchors, the mean predicted gain grows from 0.024 at $H = 1$ to 0.139 at $H = 2$ and 0.854 at $H = 4$. On strict-event anchors, the corresponding values are 0.065, 0.394, and 2.672. The growth is not simply a consequence of accumulating the same one-step preference. The selected first action itself changes as the planning horizon expands.

Figure 5b quantifies this behavior. On natural anchors, $H = 1$ and $H = 4$ select different first actions on 44.3% of evaluated anchors, while $H = 1$ and $H = 2$ disagree on 27.3%. The disagreement

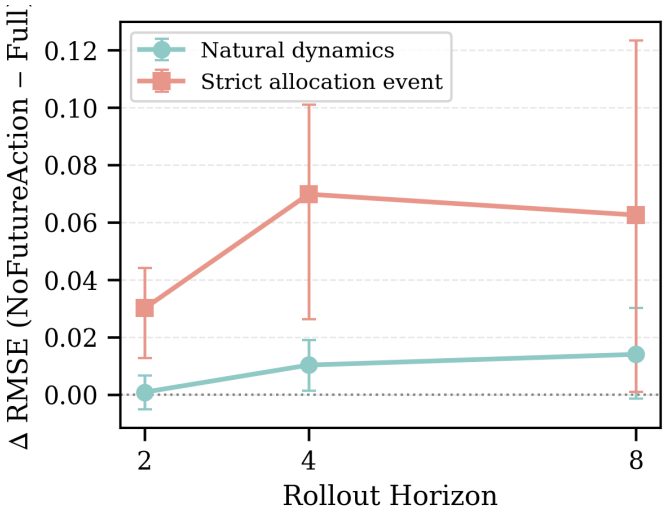
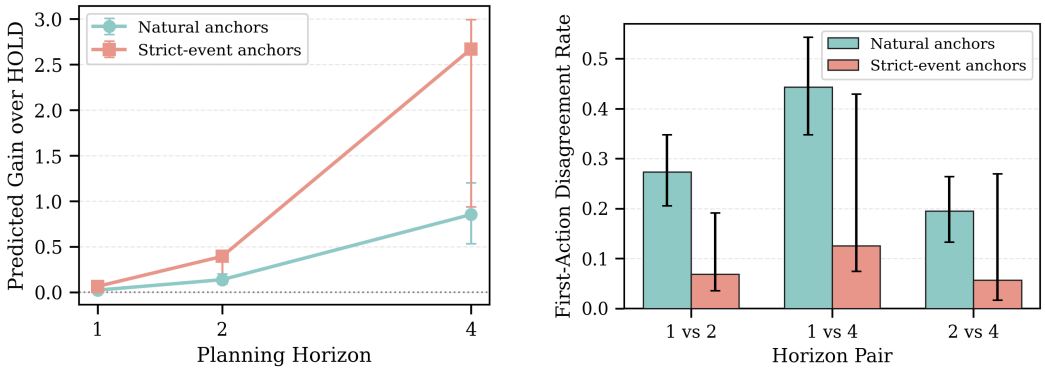


Figure 4: Effect of explicit future-action conditioning across rollout horizons. Positive values denote the increase in standardized RMSE when future-action inputs are removed from the same model family. Error bars show 95% trial-cluster bootstrap intervals.

rates on strict-event anchors are lower, which is consistent with a more constrained local decision around an already observed allocation event.

The larger search remains computationally compatible with the 250 ms Near-RT interval. At $H = 4$, p95 planner latency is 89.3 ms on natural anchors and 86.7 ms on strict-event anchors. All selected actions are physically feasible and all selected one-step transitions are observed in the training support. Exact sequence-level support naturally becomes sparser as the planning horizon increases, reaching 26.6% on natural anchors and 9.1% on strict events at $H = 4$.

The planning results therefore quantify what the frozen world model would prefer under its learned predictive objective. They are offline predictive diagnostics rather than measured closed-loop network gains.



(a) Predicted objective gain over HOLD.

(b) First-action disagreement across horizons.

Figure 5: Multi-horizon offline predictive planning. Longer imagination increases the model-predicted separation between the selected action sequence and HOLD and can change the first action selected by the planner. Predicted gains are evaluations of the frozen world model rather than measured closed-loop improvements.

4.5 RQ4: Persistent-Response Mechanism Analysis

We evaluate three variants of TRACTOR+: removing the persistent residual, retaining only the medium response timescale, and replacing adaptive decay with state-independent decay. As shown in Figure 6, the full model maintains positive factual-versus-HOLD discrimination across all horizons. Removing persistent response yields the largest degradation, with confidence intervals cross-

ing zero at $H = 2$, 4, and 8, indicating that recurrent state alone is insufficient to preserve reliable action sensitivity. The single-timescale and fixed-decay variants retain positive discrimination but show weaker long-horizon margins than the full model, supporting the roles of multi-timescale memory and adaptive retention. Factual-RMSE differences are less conclusive and are reported in the appendix.

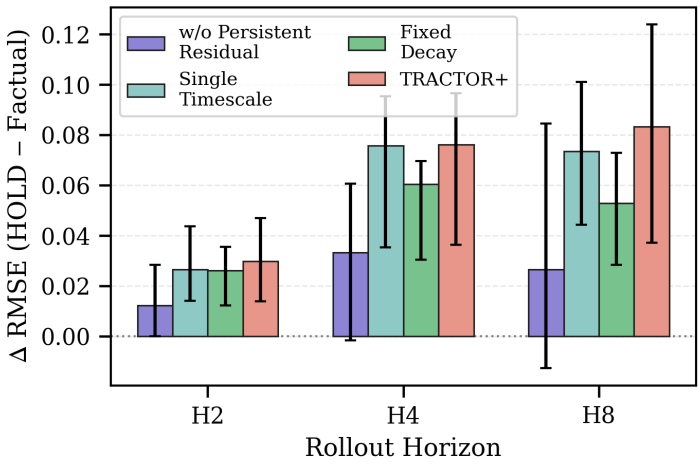


Figure 6: Persistent-response ablation on strict allocation-event windows. Positive values indicate better prediction under the factual action path than HOLD; error bars show 95% trial-cluster bootstrap intervals.

5 Conclusion

We propose TRACTOR+, an action-conditioned RAN world model that separates recurrent network-state evolution from persistent action-response dynamics. By combining multi-scale state modeling, instantaneous action conditioning, and adaptive response memory, TRACTOR+ improves multi-horizon prediction and preserves stronger action sensitivity around resource-allocation events. The learned dynamics also support short-horizon offline planning within the evaluated Near-RT budget. As the current study is based on observational offline traces, alternative-action rollouts are interpreted as predictive diagnostics rather than causal or measured closed-loop effects.

References

Maxime Bouton, Ioanna Mitsioni, Simon Lindstahl, and Jaeseong Jeong. Mobile network control with a world model. In *IEEE/IFIP Network Operations and Management Symposium*, 2026. URL https://arxiv.org/abs/2607.17747. arXiv:2607.17747.

David Ha and Jürgen Schmidhuber. Recurrent world models facilitate policy evolution. In *Advances in Neural Information Processing Systems*, pp. 2455–2467, 2018.

Danijar Hafner, Timothy Lillicrap, Jimmy Ba, and Mohammad Norouzi. Dream to control: Learning behaviors by latent imagination. *arXiv preprint arXiv:1912.01603*, 2019a.

Danijar Hafner, Timothy Lillicrap, Ian Fischer, Ruben Villegas, David Ha, Honglak Lee, and James Davidson. Learning latent dynamics for planning from pixels. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97 of *Proceedings of Machine Learning Research*, pp. 2555–2565. PMLR, 2019b. URL https://proceedings.mlr.press/v97/hafner19a.html.

Nicklas Hansen, Hao Su, and Xiaolong Wang. TD-MPC2: Scalable, robust world models for continuous control. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=Oxh5CstDJU.

Songtao Huang, Zhen Zhao, Can Li, and Lei Bai. TimeKAN: KAN-based frequency decomposition learning architecture for long-term time series forecasting. In *International Conference on Learning Representations*, 2025. URL https://openreview.net/forum?id=wTLc79YNbh.

Yaguang Li, Rose Yu, Cyrus Shahabi, and Yan Liu. Diffusion convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926*, 2017.

Yong Liu, Tengge Hu, Haoran Zhang, Haixu Wu, Shiyu Wang, Lintao Ma, and Mingsheng Long. iTransformer: Inverted transformers are effective for time series forecasting. In *International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=JePfAI8fah.

Kechen Meng, Rongpeng Li, Yansha Deng, Zhifeng Zhao, and Honggang Zhang. NetWorld: Communication-based diffusion world model for multi-agent reinforcement learning in wireless networks. *arXiv preprint arXiv:2602.00558*, 2026. URL https://arxiv.org/abs/2602.00558.

Xiaoqian Qi, Haoye Chai, Yue Wang, and Yong Li. Beyond static forecasting: Unleashing the power of world models for mobile traffic extrapolation. *arXiv preprint arXiv:2604.08199*, 2026. URL https://arxiv.org/abs/2604.08199.

Farhad Rezazadeh, Amir Ashtari Gargari, Hatim Chergui, Sandra Lagen, Merouane Debbah, Houbing Song, and Lingjia Liu. Agentic world modeling for 6g: Near-real-time generative state-space reasoning. *arXiv preprint arXiv:2511.02748*, 2025a. URL https://arxiv.org/abs/2511.02748.

Farhad Rezazadeh, Hatim Chergui, Merouane Debbah, Houbing Song, Dusit Niyato, and Lingjia Liu. Rivaling transformers: Multi-scale structured state-space mixtures for agentic 6g o-ran. *arXiv preprint arXiv:2510.05255*, 2025b. URL https://arxiv.org/abs/2510.05255.

Lingyi Wang, Rashed Shelim, Walid Saad, and Naren Ramakrishnan. DMWM: Dual-mind world model with long-term imagination. In *Advances in Neural Information Processing Systems*, 2025. arXiv:2502.07591.

Bing Yu, Haoteng Yin, and Zhanxing Zhu. Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting. In *Proceedings of the Twenty-Seventh International Joint Conference on Artificial Intelligence*, pp. 3634–3640, 2018.

Ailing Zeng, Muxi Chen, Lei Zhang, and Qiang Xu. Are transformers effective for time series forecasting? In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 11121–11128, 2023. doi: 10.1609/aaai.v37i9.26317.

Chaoyun Zhang, Paul Patras, and Hamed Haddadi. Deep learning in mobile and wireless networking: A survey. *IEEE Communications Surveys & Tutorials*, 21(3):2224–2287, 2019.

Weipu Zhang, Gang Wang, Jian Sun, Yetian Yuan, and Gao Huang. STORM: Efficient stochastic transformer based world models for reinforcement learning. In *Advances in Neural Information Processing Systems*, 2023. URL https://arxiv.org/abs/2310.09615.