

Recursive State Estimation for Short-Term V2I Cluster-Delay Prediction

Wanji Hou ¹, Gang Lv ¹ and Shuang Liu ^{2,*}

¹ Changchun Automobile Test Center Co., Ltd., Changchun, Jilin, 130011, China

² Institute of Science and Technology Information of Jilin, Changchun, Jilin, 130033, China

* Corresponding author: Shuang Liu

Abstract

Short-horizon cluster-delay prediction must balance fluctuation suppression against responsiveness while accommodating newly observed clusters. We investigate a scalar local-level Kalman predictor that summarizes past observations recursively instead of requiring a fixed-length input window. Training-only trajectory validation selects its process-to-residual variance ratio; prediction precedes each observation update, and evaluation includes segment startup. Existing NYUSIM trajectories support direction holdouts, speed transfer, and a supplementary configuration-transfer evaluation with frozen parameters. At 25 m/s, the predictor reduces RMSE from 5.625 ns for persistence to 4.354 ns, closely matching a validated moving mean at 4.355 ns. The supplementary errors are 4.577 and 4.580 ns, respectively. Trend and innovation-adaptive extensions provide no consistent benefit. Startup errors and a segment-level counterexample delimit the results. These results characterize the accuracy and startup behavior of a compact recursive predictor within the evaluated stable segments.

Keywords

V2I Channel; Cluster Delay; Recursive Prediction; Kalman Filter; Evaluation.

1. Introduction

Vehicle-to-infrastructure (V2I) channels evolve with vehicle motion, propagation geometry, and blockage, making timely channel representations important for adaptive communication [1, 2]. Cluster-based descriptions summarize related multipath components through parameters such as delay, power, and angle [3]. Predicting an associated cluster's next delay provides a focused way to examine temporal evolution without reconstructing the entire channel. At short horizons, the predictor must balance suppression of local fluctuations against responsiveness to changes. It must also accommodate the beginning of a tracked segment, when only a few observations are available. These requirements motivate a sequential prediction interface that can operate from the first available observation. The scope here is the accuracy of extracted cluster-delay forecasts; improvements in beam selection or communication performance require additional downstream evaluation.

Existing channel-prediction studies offer complementary approaches. Deep learning and Transformer models learn temporal relationships for channel-state prediction [4, 5], while measurement-based vehicular research also investigates forecasts of channel state and received signal levels [6]. Parameter-oriented work uses per-cluster angular information and angular-delay tracking to support prediction in high-mobility environments [7]. Statistical recursion is another established route: neural predictors have already been compared with Kalman-based wireless channel predictors [8]. These studies concern different targets and information inputs, so their reported gains cannot be transferred directly to scalar cluster-delay prediction. Moreover, a fixed-length history representation introduces a window-

selection decision and requires a policy for incomplete startup histories. Padding, shorter-length windows, or recurrent state representations can mitigate this issue; requiring a complete observation window is not an inherent limitation of existing prediction frameworks. The relevant question is which representation is adequate for the present target and evaluation setting.

A local-level state model offers a compact alternative by carrying historical information through a recursively updated estimate and its covariance. Standard Kalman prediction and correction provide the causal computation [9, 10]: the current state predicts the next observation, which is assimilated only after that forecast has been scored. This removes the requirement for a fixed-length input vector, while the process-to-residual variance ratio still controls adaptation and effective memory. Training-only parameter selection determines this effective memory. Abrupt changes and uncertain initialization can still produce lag or large errors. Evaluation must consequently distinguish the benefit of using history from the benefit of the particular recursive model. Persistence addresses the former, while a moving mean with training-only window selection provides a direct smoothing control. Simple baselines are also informative in broader time-series forecasting comparisons [11]. Startup errors and segment-level failures are needed alongside aggregate accuracy to assess the scope of any observed benefit.

This paper investigates scalar local-level recursion for 20-ms-ahead V2I cluster-delay prediction as an application and validation study. We specify initialization at supplied segment boundaries, training-only trajectory validation, and prediction-before-update scoring on all eligible targets, including startup. We compare the recursive predictor with persistence and a window-selected moving mean, then examine whether adding a trend state or innovation-based adaptation improves prediction. Direction holdouts, speed transfer, and supplementary transfer with frozen parameters assess behavior across the available evaluation settings. The results establish performance close to validated averaging rather than a demonstrated accuracy advantage over it, while startup and segment-level counterexamples delimit applicability. The contribution is a reproducible specification and evaluation of a compact predictor; cluster extraction and association remain upstream operations, and the underlying filtering algorithm is established.

2. Prediction Problem

Let $z_{s,t}$ be the extracted delay of a tracked cluster in contiguous segment-cluster series s , sampled every $\Delta t = 20$ ms. At time $t - 1$, the predictor produces $\hat{z}_{s,t|t-1}$ using only $z_{s,0}, \dots, z_{s,t-1}$. The target $z_{s,t}$ is scored before it enters the state update. Each series initializes from its first observation and contributes forecasts from its second observation onward.

Cluster extraction, association, and segment identification are existing upstream operations, outside the predictor. The evaluation is conditional on their supplied identities and boundaries. A segment boundary or missing-observation gap resets the state; the predictor neither detects boundaries nor links identities across them. Because stable segments are identified offline, causal forecasting within a segment must not be confused with an entirely online channel-processing pipeline.

3. Recursive Predictor

3.1. Local-level Model and Causal Update

Suppressing s , use the working state model

$$d_t = d_{t-1} + w_t, \quad z_t = d_t + \epsilon_t \quad (1)$$

where independent zero-mean Gaussian terms have variances $Q = qR$ and R , respectively. These are modeling assumptions for extracted delays, not identified physical noise mechanisms. Initialize $m_0 = z_0$ and $P_0 = R$. For each subsequent target,

$$\hat{z}_{t|t-1} = m_{t-1}, \quad P_t^- = P_{t-1} + qR \quad (2)$$

$$S_t = P_t^- + R, \quad K_t = P_t^- / S_t \quad (3)$$

$$m_t = m_{t-1} + K_t(z_t - m_{t-1}) \quad (4)$$

$$P_t = (1 - K_t)^2 P_t^- + K_t^2 R \quad (5)$$

The last equation is the scalar Joseph covariance update. The forecast is recorded before applying (4); Fig. 1 shows this ordering. The innovation variance S_t determines the gain used to incorporate the new observation.

Only (m_t, P_t) must persist per cluster, with constant arithmetic per update. Historical information is retained through the recursive state, with q controlling adaptation and effective memory. An optimized rolling mean also has constant update time and ordinarily maintains a finite-window buffer.

3.2. Training-Only Scale and Parameter Selection

For training series of length T_s , define $\Delta z_{s,t} = z_{s,t} - z_{s,t-1}$ and its within-series mean $\overline{\Delta z_s}$. The residual scale is

$$R = \max\left(10^{-8}, \frac{\sum_s \sum_{t=1}^{T_s-1} (\Delta z_{s,t} - \overline{\Delta z_s})^2}{2 \sum_s (T_s - 1)}\right) \quad (6)$$

This difference-based estimate is a working approximation under locally stable levels. It does not separately identify process and observation noise. With $P_0 = R$ and $Q = qR$, the scalar gain depends on q , not the absolute scale of R .

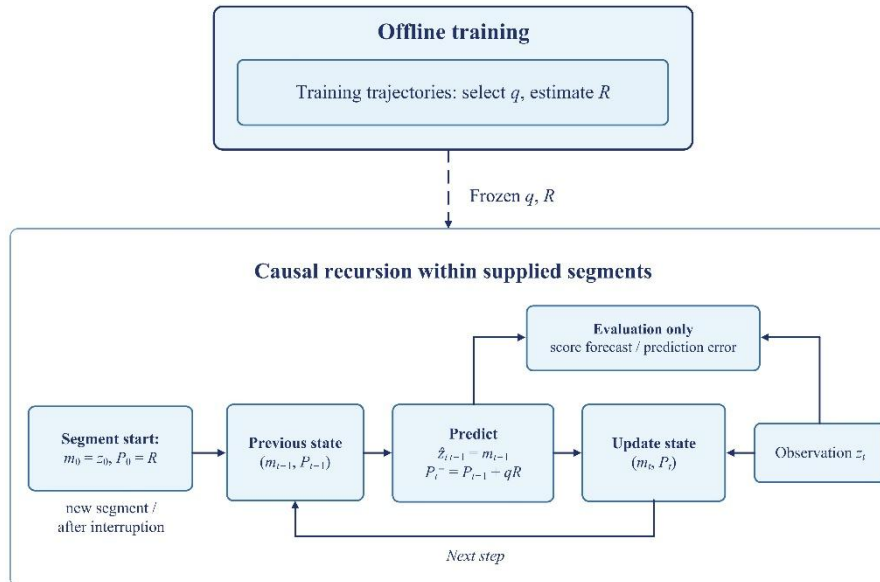


Fig 1. Causal recursion within an upstream-defined segment. A target is scored before assimilation. Training settings remain frozen throughout evaluation.

The candidates are $q = 0$ and $q = 10^k$ for integer k from -4 to 0 . Each outer training set uses validation that holds out one complete trajectory at a time, minimizing the equally weighted mean of trajectory RMSEs. Each inner fit recomputes R using only its training trajectories. The selected ratio and the outer-training scale are then frozen for evaluation. Separating complete trajectories avoids selecting settings on the targets subsequently reported as held out.

4. Experiments and Discussion

4.1. Data and Evaluation Protocol

The study uses existing NYUSIM urban-microcell line-of-sight simulations at 5.9 GHz [13, 14]. The main collection combines three directions (30° , 45° , 60°) and three speeds (5, 15, 25m/s). A supplementary previously inspected trajectory has a different source configuration (Table 1). Original simulator seeds are unavailable, so different files do not establish independent channel realizations.

Table 1. Existing simulation sources and eligible data. Both sources use 5.9 GHz UMi LOS and 20-ms prediction steps.

Property	Main	Supplement
Bandwidth (MHz)	10	20
Travel distance (m)	50	100
Native spatial step (m)	0.1	0.01
Trajectories	9	1
Stable segments	45	10
Segment-cluster series	179	37
Cluster observations	8929	1258
One-step targets	8750	1221

Existing preprocessing groups components using a 20ns delay gap, associates clusters using Hungarian matching with delay/power weights 0.75/0.25, and supplies stable segments sampled at 20ms. The selected samples coincide with native spatial-grid points. Stable segments have constant multipath counts and at least 20 native snapshots; excluded transitions are not prediction targets. Association checks support reproducibility under tested settings, but agreement with an angle-based diagnostic is imperfect. Thus tracked indices are operational identities, not verified persistent physical scatterers. Targets are extracted simulation delays, not measured receiver delays or demonstrated receiver resolution.

Direction evaluations hold out one direction at 5/15 m/s, yielding 2908, 3036, and 1674 targets. Speed transfer trains on all six 5/15 m/s trajectories and evaluates 1132 targets at 25m/s. The supplementary evaluation freezes this speed-transfer fit, including R , and scores 1221 targets without refitting. It tests transfer to one changed configuration, not an isolated speed effect or a new blind test. All data have previously been inspected; these are exploratory reuse evaluations.

4.2. Controls and Scoring

Persistence predicts the last observation. Moving averaging selects $H \in \{2, 4, 8, 16, 32\}$ by the same training-only trajectory criterion, using $\min(H, t)$ available observations at startup. A two-state local-linear control adds the delay increment per frame, with

$$F = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix}, Q = qR \begin{bmatrix} \frac{1}{4} & \frac{1}{2} \\ \frac{1}{2} & 1 \end{bmatrix}, P_0 = R \text{diag}(1, v_0) \quad (7)$$

Its initial increment is zero; q uses the same grid and $v_0 \in \{0, 0.01, 0.1, 1\}$. An adaptive control smooths clipped normalized squared innovations e_t^2/S_t :

$$a_t = (1 - \rho)a_{t-1} + \rho \min(e_t^2/S_t, 25) \quad (8)$$

$$q_{t+1} = \min(10, q_0 + g \max(a_t - 1, 0)) \quad (9)$$

with $a_0 = 1$, $g \in \{0, 0.001, 0.01, 0.1\}$ and $\rho \in \{0.05, 0.2\}$. Base (q_0, v_0) selection is additionally nested within each inner training partition. These controls test whether a trend state and innovation-driven adaptation improve prediction.

RMSE and MAE pool eligible targets within each evaluation group. All methods use identical targets, including startup; the first four predictions are also reported separately as a diagnostic, not an input requirement. Segments and targets are dependent, and no target-level significance claim is made. Saved CSV forecasts, split/selection records, source hashes, and causal-prefix checks support reproducibility.

4.3. Accuracy and Extension Ablations

Every outer fit selects $q = 0.001$ for the scalar predictor and $H = 32$ for averaging. Fig. 2 shows lower aggregate RMSE than persistence across all five evaluation groups. At 25m/s, the reduction is 22.6%, but the advantage over validated averaging is only 0.00154 ns (0.035%). Table 2 therefore supports competitive performance, not a demonstrated accuracy improvement over smoothing.

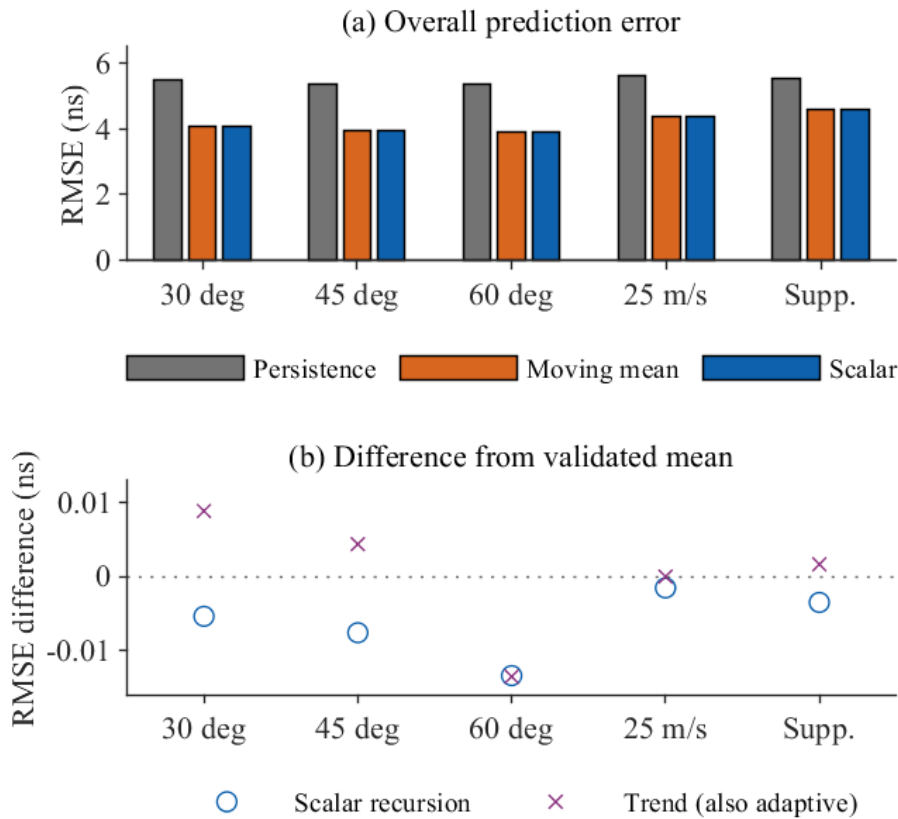


Fig 2. (a) Pooled RMSE for direction holdouts, speed transfer, and supplementary transfer. (b) RMSE difference from validated moving averaging; negative values favor recursion. Groups are discrete evaluation settings.

The local-linear fits select $q_0 = v_0 = 0$, and adaptive selection chooses $g = 0$ in all four outer runs. Both consequently reduce to cumulative averaging within each segment, agreeing numerically to rounding precision. At 60° , this control has marginally lower RMSE than the scalar predictor (3.88941 versus 3.88954 ns). The scalar predictor therefore does not attain the lowest RMSE in every evaluation group. Speed-transfer segments contain fewer than 32 observations, so the selected moving mean also becomes cumulative averaging there. The selected upper-grid window is not evidence that 32 is universally optimal.

Table 2. Speed and frozen supplementary transfer. Errors in ns; all startup predictions are included.

Method	25 m/s		Supplement	
	RMSE	MAE	RMSE	MAE
Persistence	5.625	3.839	5.506	3.600
Validated mean	4.355	2.900	4.580	2.869
Scalar recursion	4.354	2.899	4.577	2.867
Level + trend	4.355	2.900	4.582	2.871
Adaptive trend	4.355	2.900	4.582	2.871

4.4. Startup Behavior and Applicability Boundaries

Frozen supplementary transfer gives scalar RMSE 4.577ns versus 4.580ns for averaging, a difference of 0.00350ns (0.076%). Only five of its ten segments favor the scalar method over averaging. Fig. 3 exposes variation hidden by the pooled score. In segment 3, scalar RMSE is 7.855ns, worse than persistence at 7.644ns; smoothing is not beneficial everywhere.

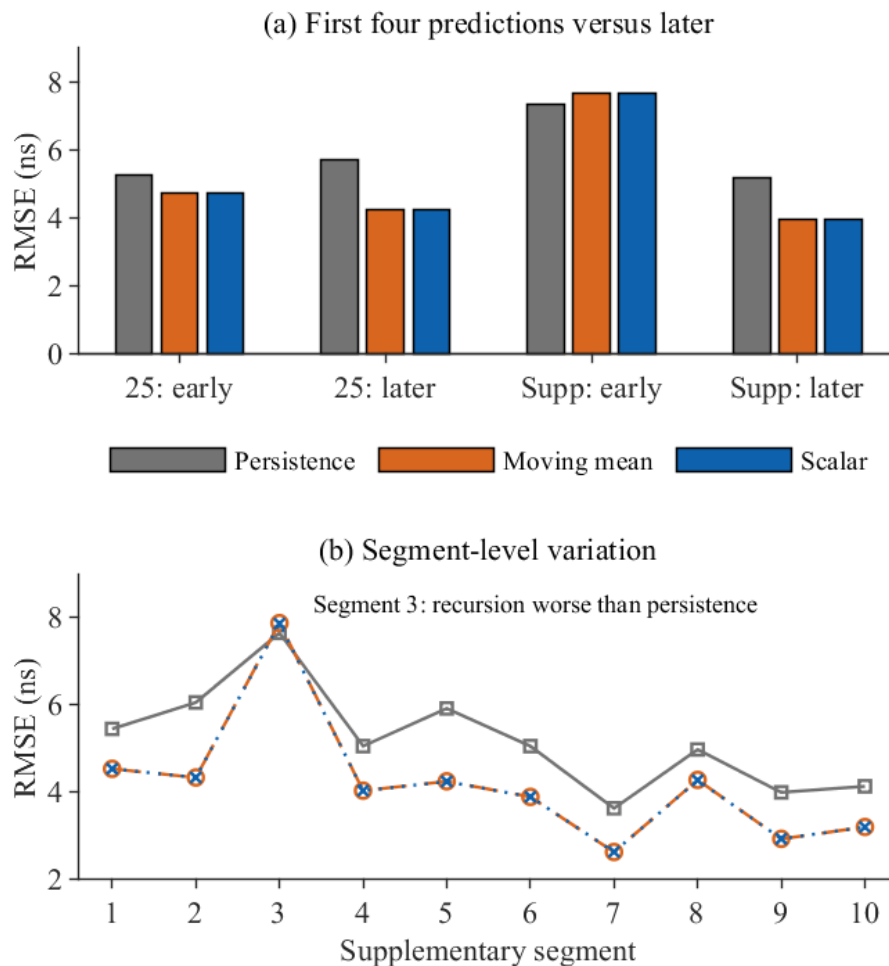


Fig 3. (a) Startup (first four predictions per series) and subsequent errors. (b) Supplementary segment errors; segment 3 is a counterexample to uniform benefit over persistence. Segment indices follow source order

Supplementary startup RMSE is 7.668ns for the scalar predictor versus 7.357ns for persistence; after startup these become 3.966 and 5.199 ns. Speed-transfer startup instead favors recursion

(4.727 versus 5.259 ns). Thus initialization sensitivity depends on the evaluated sequences, and immediate availability of a forecast does not guarantee immediate improvement.

The evidence supports a compact causal implementation within supplied stable segments. It does not establish online segment detection, physical cluster identity, globally calibrated uncertainty, or improved link performance. The small number of reused trajectories, unavailable generation seeds, changed supplementary configuration, and excluded transition regions constrain generalization. Independent realizations and transition-inclusive tests are needed before extending the claim to continuous vehicle operation.

5. Summary

This paper evaluated a scalar local-level predictor for short-term V2I cluster-delay prediction without fixed-length input vectors. With training-only parameter selection, the predictor achieved accuracy comparable to validated moving averaging and lower aggregate error than persistence across the evaluated settings. Adding a trend state or innovation-based adaptation provided no consistent improvement. Startup errors and segment-level counterexamples showed that the aggregate benefit did not hold uniformly within individual sequences. These findings support compact recursive prediction within the evaluated stable segments, while further validation on independent trajectories and across segment transitions is needed to assess broader applicability.

Acknowledgments

This work was supported by the Natural Science Foundation of Jilin Province under Grant No. 20260203080SF.

References

- [1] Mecklenbrauker, C. F., Molisch, A. F., Karedal, J., et al. (2011). Vehicular channel characterization and its implications for wireless system design and performance. *Proceedings of the IEEE*, 99(7), 1189–1212.
- [2] Su, Z., Liu, L., Zhang, J., et al. (2022). Analysis of channel non-stationarity for V2V and V2I communications at 5.9GHz in urban scenarios. In *2022 IEEE International Conference on Communications Workshops (ICC Workshops)* (pp. 1130–1134). Seoul, South Korea.
- [3] Liu, L., Oestges, C., Poutanen, J., et al. (2012). The COST 2100 MIMO channel model. *IEEE Wireless Communications*, 19(6), 92–99.
- [4] Luo, C., Ji, J., Wang, Q., et al. (2020). Channel state information prediction for 5G wireless communications: A deep learning approach. *IEEE Transactions on Network Science and Engineering*, 7(1), 227–236.
- [5] Jiang, H., Cui, M., Ng, D. W. K., et al. (2022). Accurate channel prediction based on transformer: Making mobility negligible. *IEEE Journal on Selected Areas in Communications*, 40(9), 2717–2732.
- [6] Joo, J., Park, M. C., Han, D. S., et al. (2019). Deep learning-based channel prediction in realistic vehicular communications. *IEEE Access*, 7, 27846–27858.
- [7] Peng, F., Zhang, S., Chen, W., et al. (2024). Beamforming oriented angular domain channel prediction for mixed LOS and non-LOS SIMO environments with high mobility. *IEEE Transactions on Wireless Communications*, 23(5), 4315–4330.
- [8] Jiang, W., & Schotten, H. D. (2019). A comparison of wireless channel predictors: Artificial intelligence versus Kalman filter. In *2019 IEEE International Conference on Communications (ICC)* (pp. 1–6). Shanghai, China.
- [9] Kalman, R. E. (1960). A new approach to linear filtering and prediction problems. *Journal of Basic Engineering*, 82(1), 35–45.

- [10] Särkkä, S., & Svensson, L. (2023). Bayesian filtering and smoothing (2nd ed.). Cambridge University Press.
- [11] Zeng, A., Chen, M., Zhang, L., et al. (2023). Are transformers effective for time series forecasting? Proceedings of the AAAI Conference on Artificial Intelligence, 37(9), 11121–11128.
- [12] Sun, S., MacCartney, G. R., Jr., & Rappaport, T. S. (2017). A novel millimeter-wave channel simulator and applications for 5G wireless communications. In 2017 IEEE International Conference on Communications (ICC) (pp. 1–7). Paris, France.
- [13] Ju, S., Kanhere, O., Xing, Y., et al. (2019). A millimeter-wave channel simulator NYUSIM with spatial consistency and human blockage. In 2019 IEEE Global Communications Conference (GLOBECOM) (pp. 1–6). Waikoloa, HI, USA.