

Estimação da Covariância do Ruído de Medição em Filtros de Kalman Utilizando Modelos de Aprendizado de Máquina

Arthur Medeiros Macêdo¹ e Karl Heinz Kienitz²

¹ ² Instituto Tecnológico de Aeronáutica, São José dos Campos / SP - Brasil

Resumo— Este artigo propõe um algoritmo que incorpora um modelo de aprendizado de máquina para estimar, em tempo real, a matriz de covariância do ruído de medição a cada iteração do filtro de Kalman. Essa estimação dinâmica da matriz R visa aprimorar a precisão das estimativas fornecidas pelo filtro.

Palavras-Chave— Filtro de Kalman, aprendizado de máquina, LSTM

I. INTRODUÇÃO

A precisão na estimativa da trajetória de aeronaves é essencial para garantir a segurança e a eficiência da navegação aérea. O filtro de Kalman é amplamente utilizado para estimar os estados de sistemas dinâmicos, como a posição e a velocidade de aeronaves, a partir de medições ruidosas. Trata-se de um algoritmo recursivo que combina o modelo dinâmico do sistema com medições observadas para fornecer estimativas confiáveis, mesmo na presença de incertezas.

Diversas variantes do filtro de Kalman têm sido empregadas na estimação de trajetórias e no rastreamento de alvos [1]. Em [2], é proposto o uso de um filtro de Kalman com equações de estado convertidas para lidar com dados de sensores como radar e sonar, que operam em coordenadas polares. Já [3] exploram o filtro de Kalman estendido invariante no contexto do controle de tráfego aéreo, utilizando a estrutura *Frenet-Serret* para uma modelagem mais precisa da trajetória. Combinando filtragem clássica com técnicas de aprendizado de máquina, [4] propõem o modelo *Interaction-aware Kalman Neural Networks (IaKNN)*, no qual redes neurais são integradas ao filtro de Kalman para prever trajetórias de veículos com base em suas interações dinâmicas com o ambiente.

Um dos principais desafios na aplicação do filtro de Kalman padrão é a definição adequada das matrizes de covariância dos ruídos de processo (Q) e de medição (R). Esses parâmetros são cruciais para o desempenho do filtro, pois afetam diretamente a qualidade das estimativas. No entanto, na prática, seus valores são difíceis de determinar, uma vez que podem variar ao longo do tempo e conforme as condições operacionais, ou ainda porque a modelagem estocástica utilizada pelo filtro de Kalman é, geralmente, uma aproximação relativamente simples (mas muito tratável e portanto conveniente) da realidade. A escolha

inadequada de Q ou R pode comprometer a confiabilidade das estimativas.

Este trabalho propõe um método para ajustar, em tempo real, a matriz de covariância do ruído de medição (R) por meio de aprendizado de máquina, utilizando um modelo do tipo LSTM (*Long Short-Term Memory*). A proposta consiste em treinar a LSTM com dados observacionais para prever R dinamicamente, melhorando o desempenho do filtro de Kalman em cenários com características variáveis. O algoritmo desenvolvido realiza essa atualização a cada iteração do filtro, resultando em estimativas mais precisas e robustas. Os resultados obtidos demonstram que a abordagem proposta contribui para a melhoria da acurácia na estimação de trajetórias.

II. METODOLOGIA

Este trabalho foca na estimação da posição de um objeto por meio do Filtro de Kalman Padrão. Para modelar as equações dinâmicas do sistema, é adotado o modelo *Constant Velocity Model* [5].

A. Equações do Filtro de Kalman

O filtro de Kalman é um algoritmo recursivo amplamente utilizado para estimar os estados de um sistema dinâmico linear a partir de medições afetadas por ruído. Baseia-se em um modelo de sistema conhecido e opera em duas etapas principais: predição e atualização, que são executadas iterativamente a cada instante de tempo.

1) *Etapas de Predição*: Na etapa de predição, o filtro estima o estado futuro do sistema com base no modelo dinâmico, além de propagar a incerteza associada à estimativa anterior [6]. As equações de predição são dadas por:

$$\hat{X}_{k|k-1} = F\hat{X}_{k-1|k-1} \quad (1)$$

$$P_{k|k-1} = FP_{k-1|k-1}F^T + GQG^T \quad (2)$$

onde:

- $\hat{X}_{k|k-1} \in \mathbb{R}^{n \times 1}$ representa a predição do vetor de estado no instante k .
- $P_{k|k-1} \in \mathbb{R}^{n \times n}$ é a matriz de covariância do erro de predição.
- $F \in \mathbb{R}^{n \times n}$ é a matriz de transição de estados.
- $G \in \mathbb{R}^{n \times l}$ é a matriz de entrada do ruído de processo.
- $Q \in \mathbb{R}^{l \times l}$ é a matriz de covariância do ruído de processo, assumido como ruído branco gaussiano.

A. Macêdo, macedoamm@fab.mil.br; K. Kienitz, kienitz@ieee.org. O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001. O segundo autor agradece pelo apoio à pesquisa do CNPq (processo 304557/2021-8).

2) *Etapa de Atualização*: Na etapa de atualização, as estimativas obtidas na predição são corrigidas com base nas novas medições adquiridas. As equações correspondentes são:

$$S_k = HP_{k|k-1}H^\top + R \quad (3)$$

$$K_k = P_{k|k-1}H^\top S_k^{-1} \quad (4)$$

$$\hat{X}_{k|k} = \hat{X}_{k|k-1} + K_k(Y_k - H\hat{X}_{k|k-1}) \quad (5)$$

$$P_{k|k} = (I - K_kH)P_{k|k-1} \quad (6)$$

onde:

- $S_k \in \mathbb{R}^{m \times m}$ é a matriz de covariância da inovação.
- $K_k \in \mathbb{R}^{n \times m}$ é o ganho de Kalman.
- $\hat{X}_{k|k} \in \mathbb{R}^{n \times 1}$ é a estimativa corrigida do vetor de estado.
- $P_{k|k} \in \mathbb{R}^{n \times n}$ é a matriz de covariância do erro da estimativa corrigida.
- $H \in \mathbb{R}^{m \times n}$ é a matriz de observação.
- $R \in \mathbb{R}^{m \times m}$ é a matriz de covariância do ruído de medição.
- $Y_k \in \mathbb{R}^{m \times 1}$ é o vetor de medições no instante k .
- $I \in \mathbb{R}^{n \times n}$ é a matriz identidade.

B. Constant Velocity Model (CVM)

O modelo de velocidade constante (*Constant Velocity Model* — CVM) [5] é amplamente utilizado na modelagem de sistemas dinâmicos quando se assume que a velocidade do alvo permanece aproximadamente constante entre amostras, sendo sujeita a pequenas variações representadas por ruído branco gaussiano. Esse modelo é adequado para alvos com movimento retilíneo uniforme, com acelerações não medidas tratadas como perturbações estocásticas.

No domínio contínuo, o sistema pode ser representado pela seguinte equação diferencial de primeira ordem:

$$\dot{x}(t) = Fx(t) + Gw(t), \quad (7)$$

onde:

- $x(t) = \begin{pmatrix} p(t) \\ v(t) \end{pmatrix} \in \mathbb{R}^{2 \times 1}$ é o vetor de estado, com posição $p(t)$ e velocidade $v(t)$,
- $F = \begin{pmatrix} 0 & 1 \\ 0 & 0 \end{pmatrix} \in \mathbb{R}^{2 \times 2}$ é a matriz dinâmica do sistema,
- $G = \begin{pmatrix} 0 \\ 1 \end{pmatrix} \in \mathbb{R}^{2 \times 1}$ é a matriz de entrada do ruído de processo,
- $w(t) \in \mathbb{R}$ representa o ruído branco com média zero e variância constante.

Após discretização com período de amostragem T , o modelo é representado na forma discreta por:

$$x_{k+1} = Fx_k + Gw_k, \quad (8)$$

com:

$$F = \begin{pmatrix} 1 & T \\ 0 & 1 \end{pmatrix}, \quad G = \begin{pmatrix} \frac{T^2}{2} \\ T \end{pmatrix}, \quad (9)$$

onde:

- $x_k = \begin{pmatrix} p_k \\ v_k \end{pmatrix}$ é o vetor de estado no instante k ,
- $w_k \in \mathbb{R}$ é o ruído de processo discreto, assumido como ruído branco gaussiano com covariância Q .

Esse modelo é frequentemente utilizado em filtros de Kalman para rastreamento de alvos, pois oferece um bom equilíbrio entre simplicidade computacional e fidelidade na modelagem de movimentos suaves e contínuos.

C. Proposta do Trabalho

Este trabalho propõe um algoritmo para melhorar a estimação de estados por meio do filtro de Kalman, utilizando aprendizado de máquina para atualização dinâmica da matriz de covariância do ruído de medição (R). Especificamente, emprega-se um modelo do tipo LSTM (*Long Short-Term Memory*), treinado para prever em tempo real o valor de R com base em dados observacionais do sistema.

A cada iteração do filtro de Kalman, o valor de R é ajustado com base na predição da rede LSTM e em uma taxa de adaptação controlada por um parâmetro λ . A atualização é dada por:

$$R_k = (1 - \lambda)R_{k-1} + \lambda \cdot \hat{R}, \quad (10)$$

onde:

- $\hat{R} \in \mathbb{R}^{m \times m}$ é a matriz de covariância do ruído de medição predita pela rede LSTM no instante k ,
- $R_k \in \mathbb{R}^{m \times m}$ é o valor atualizado da matriz de covariância do ruído de medição,
- $\lambda \in [0, 1]$ é um fator de suavização que define a taxa de adaptação entre o valor anterior e o valor predito,
- k é o índice temporal da iteração corrente do filtro.

A utilização dessa abordagem permite adaptar R ao longo do tempo, considerando possíveis variações nas condições do sistema e nas características do ruído de medição. Isso contribui para o aumento da robustez e da acurácia do filtro de Kalman em ambientes dinâmicos e não estacionários.

D. Implementação em Python

A implementação do algoritmo proposto será realizada na linguagem Python, utilizando bibliotecas como NumPy, SciPy, scikit-learn e TensorFlow/Keras. Os experimentos incluirão a simulação de trajetórias sob diferentes níveis de ruído para avaliação do desempenho do filtro de Kalman com e sem adaptação da matriz R .

Para o treinamento e validação do modelo baseado em LSTM, serão gerados dados sintéticos que reproduzem medições ruidosas de um sistema com comportamento descrito pelo modelo de velocidade constante. Serão considerados diferentes cenários com variações no nível de ruído de medição, a fim de testar a capacidade de generalização e robustez do modelo em condições diversas.

E. Modelo LSTM

O modelo LSTM (*Long Short-Term Memory*) é uma arquitetura especializada de redes neurais recorrentes (RNNs), desenvolvida para lidar com o aprendizado de padrões em sequências temporais. Ao contrário das RNNs tradicionais, as LSTMs mitigam os problemas de explosão

e desaparecimento do gradiente por meio de uma estrutura composta por células de memória e mecanismos de controle: portas de entrada, saída e esquecimento.

Essa arquitetura permite que o modelo retenha e manipule informações relevantes ao longo de longas janelas temporais, sendo particularmente eficaz na modelagem de séries temporais com dependências de longo prazo, como resíduos e variâncias associadas ao filtro de Kalman.

Neste estudo, o modelo LSTM será treinado para prever dinamicamente a matriz de covariância do ruído de medição R , com base nas variâncias dos resíduos provenientes das etapas de predição e atualização do filtro de Kalman. A função de ativação adotada será a ReLU (*Rectified Linear Unit*), enquanto a função de perda será o erro quadrático médio (MSE — *Mean Squared Error*), com otimização realizada por meio do algoritmo Adam.

F. Geração do Dataset

A geração do *dataset* será realizada por meio de múltiplas simulações numéricas de um sistema dinâmico sob diferentes condições de ruído. O resultado de cada simulação é uma sequência temporal de amostras do sistema. As primeiras amostras dessa sequência são desconsideradas a fim de garantir que o filtro de Kalman atinja um regime de convergência satisfatório.

G. Importância das variáveis de entrada (features)

Para avaliar a relevância das variáveis de entrada utilizadas no modelo LSTM, será empregado o método de importância de *features* por permutação. Essa técnica consiste em medir o impacto individual de cada *feature* sobre o desempenho do modelo ao perturbar seu valor.

O procedimento será conduzido da seguinte forma: inicialmente, calcula-se a performance de referência do modelo sobre o conjunto de validação. Em seguida, os valores de uma *feature* são aleatoriamente permutados, preservando as demais, e a performance é novamente avaliada. A queda de desempenho resultante da permutação reflete a importância daquela variável.

A importância de cada *feature* será quantificada pela diferença entre a performance média dos modelos com permutação e a performance de referência. Esse método permite identificar quais características têm maior influência na previsão da matriz R e, consequentemente, no desempenho global do filtro adaptativo.

A análise de importância será realizada após o treinamento do modelo, utilizando o conjunto de validação, como forma de validar a escolha e a contribuição das variáveis extraídas durante a geração do *dataset*.

H. Divisão do Dataset

O *dataset* gerado será particionado em três subconjuntos: treinamento, validação e teste. O conjunto de treinamento será utilizado para o ajuste dos pesos da rede neural LSTM por meio do algoritmo de otimização. O conjunto de validação será empregado para monitorar o desempenho do modelo durante o treinamento e para a escolha de hiperparâmetros, como o número de unidades ocultas, taxa de aprendizado e número de épocas. Por fim,

o conjunto de teste será reservado exclusivamente para a avaliação final do modelo, com o objetivo de verificar sua capacidade de generalização em dados não vistos.

I. Avaliação do Modelo

A avaliação do modelo será realizada com base na análise das curvas de perda (*loss*) ao longo das épocas, tanto para o conjunto de treinamento quanto para o de validação. Essa análise visa identificar indícios de *underfitting* (modelo incapaz de aprender o padrão dos dados) ou *overfitting* (modelo excessivamente ajustado aos dados de treinamento) [7].

Além disso, a perda será computada sobre o conjunto de teste ao final do treinamento, a fim de validar a estabilidade e a capacidade de generalização do modelo. O desempenho será avaliado por meio da função de perda MSE (*Mean Squared Error*), permitindo verificar se o modelo treinado é capaz de estimar corretamente a matriz R a partir das variáveis de entrada.

J. Simulações de Desempenho do Filtro de Kalman

Serão conduzidas simulações adicionais com o objetivo de comparar o desempenho do filtro de Kalman utilizando duas abordagens distintas para a matriz de covariância do ruído de medição R : (i) o valor predito pelo modelo LSTM treinado e (ii) um valor aleatório de R atribuído em cada execução.

As simulações abrangerão diferentes valores verdadeiros de R , representando cenários com variadas intensidades de ruído de medição. O desempenho será quantificado por meio da métrica RMSE (*Root Mean Squared Error*), calculada entre a trajetória estimada e a trajetória real do sistema.

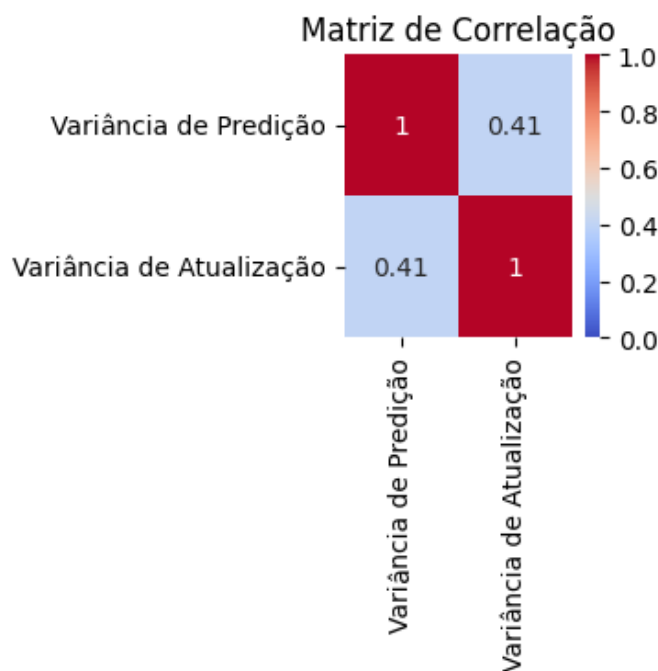
A comparação entre as abordagens será complementada por análises estatísticas, visando avaliar a significância dos resultados e confirmar se a estimativa de R via aprendizado de máquina contribui, de fato, para a melhoria da acurácia do filtro de Kalman em relação a alternativas não informadas.

III. RESULTADOS E DISCUSSÕES

Na etapa de geração do conjunto de dados, foram realizadas 1000 simulações contendo 250 instantes de amostragem cada. Os primeiros 50 instantes foram desconsiderados com o objetivo de permitir a convergência inicial do filtro de Kalman. Em cada simulação, foram extraídas duas *features*: a média da variância dos resíduos da etapa de predição e a média da variância dos resíduos da etapa de atualização. O valor verdadeiro da variância do ruído de medição R foi utilizado como variável de saída supervisionada.

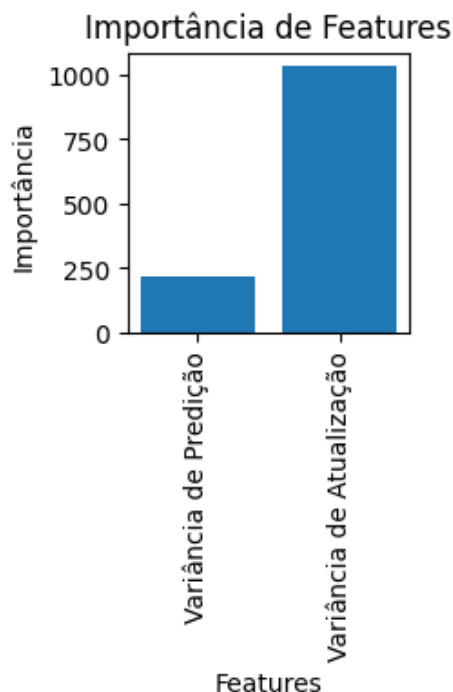
A Figura 1 apresenta a matriz de correlação entre as *features* consideradas. Observa-se que não há correlação elevada entre elas, o que indica que ambas fornecem informações complementares ao modelo de aprendizado de máquina, justificando sua inclusão no processo de treinamento.

O *dataset* foi dividido aleatoriamente em três subconjuntos: 60% para treinamento (600 amostras), 20% para validação (200 amostras) e 20% para teste (200 amostras).


Fig. 1. Matriz de correlação entre *features*

Para validar a relevância das variáveis de entrada, foi treinado um modelo LSTM com função de ativação *ReLU* e função de perda *MSE* durante 30 épocas. Em seguida, foi aplicada a análise de importância por permutação.

Conforme mostrado na Figura 2, ambas as *features* apresentaram valores de importância superiores a 100, indicando contribuição significativa para o desempenho do modelo.


Fig. 2. Importância das *features* via permutação

A Figura 3 apresenta a evolução das perdas de treinamento e validação ao longo das épocas.

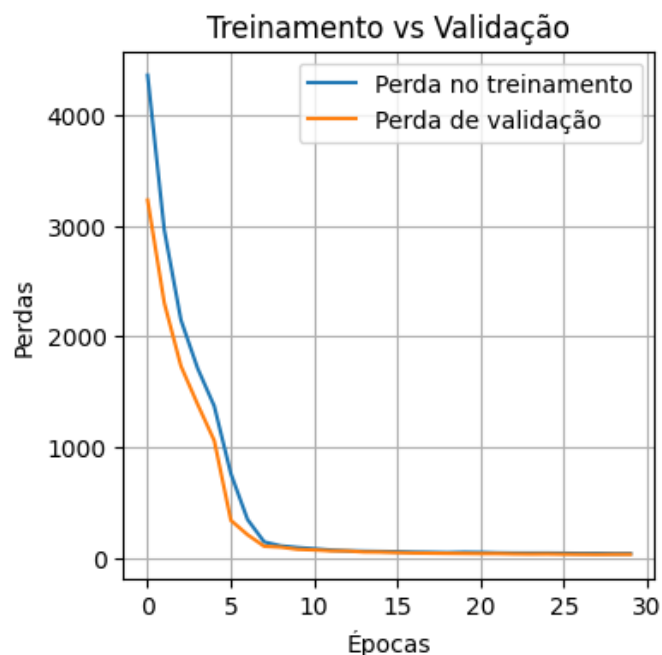


Fig. 3. Curvas de perda para os conjuntos de treinamento e validação

Observa-se que ambas as curvas convergiram de forma estável, com valores semelhantes e baixos, o que sugere ausência de *underfitting* e *overfitting*.

A Tabela I resume os valores finais da função de perda nos três subconjuntos.

Conjunto de dados	Valor da Perda
Treinamento	37,42
Validação	30,65
Teste	36,24

TABELA I

PERDA FINAL NOS CONJUNTOS DE TREINAMENTO, VALIDAÇÃO E TESTE

Com o objetivo de avaliar o impacto do valor estimado de *R* na acurácia do filtro de Kalman, foram realizadas 300 simulações adicionais. Cada uma considerou um dos 10 valores verdadeiros de *R* no intervalo [0,1, 100], com 30 repetições por valor.

Em cada simulação, foi utilizado um filtro de Kalman com matriz de covariância do ruído de processo *Q* idêntica à matriz *Q* verdadeira (diagonal com elementos aleatórios entre 0 e 100), permitindo isolar o efeito da estimação de *R*. Dois cenários foram comparados: (i) *R* aleatório entre 0 e 100, e (ii) *R* predito dinamicamente pelo modelo LSTM. Para a atualização de *R*, foi fixado o fator $\lambda = 5\%$.

A Figura 4 apresenta o boxplot das diferenças de RMSE entre os dois filtros. Valores negativos indicam vantagem da abordagem adaptativa com LSTM.

A Figura 5 detalha essa comparação para cada valor de *R* verdadeiro, evidenciando que os maiores ganhos ocorrem nos extremos do intervalo testado.

Por fim, foi realizado um teste estatístico *t*-student para comparar as duas abordagens. A hipótese nula considera que a diferença média de RMSE entre os filtros é zero; a hipótese alternativa assume que a diferença é negativa (a favor do filtro adaptativo).

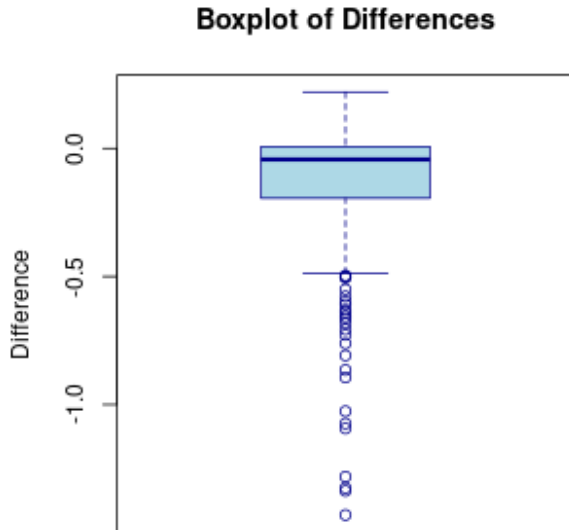


Fig. 4. Diferença de RMSE: filtro com R adaptativo vs. aleatório

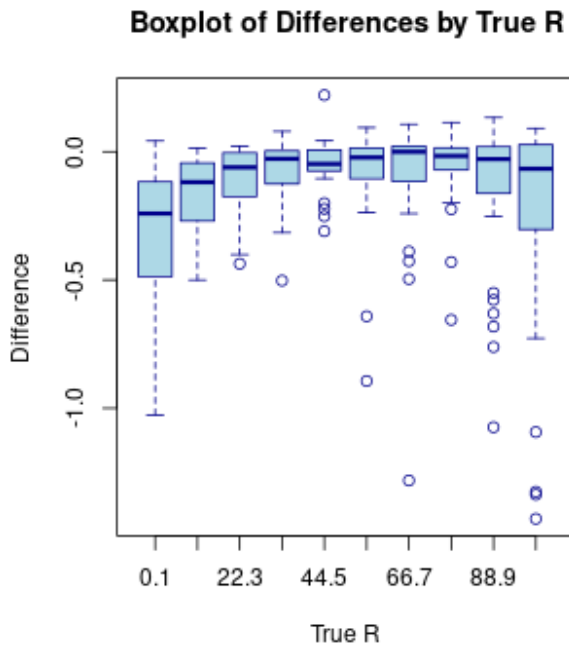


Fig. 5. Diferença de RMSE por valor verdadeiro de R

O teste resultou em um valor-p de $2,2 \times 10^{-16}$, com intervalo de confiança de 95% estimado em $-0,1134$, o que permite rejeitar a hipótese nula com alta confiança. Conclui-se, portanto, que a utilização da predição adaptativa de R via LSTM contribui significativamente para a redução do erro de estimativa de estado no filtro de Kalman.

IV. CONCLUSÕES

Este estudo demonstrou que a adaptação da matriz de covariância do ruído de medição R em filtros de Kalman, por meio de um modelo de aprendizado de máquina

baseado em LSTM (*Long Short-Term Memory*), pode melhorar significativamente a precisão na estimação de trajetórias. A abordagem proposta possibilitou a atualização dinâmica de R com base em características extraídas de resíduos do próprio filtro, resultando em estimativas mais robustas quando comparadas a abordagens tradicionais com valores fixos ou aleatórios para esse parâmetro.

Os experimentos realizados evidenciaram que a predição adaptativa de R pelo modelo LSTM contribui para a redução do erro médio quadrático (RMSE) das estimativas de estado. Essa conclusão foi corroborada por análises estatísticas, que indicaram diferença significativa de desempenho entre as abordagens comparadas.

Como perspectiva para trabalhos futuros, destaca-se a possibilidade de estender a metodologia para a adaptação conjunta das matrizes R e Q , ampliando o grau de flexibilidade do filtro. Além disso, a exploração de outras arquiteturas de aprendizado de máquina, como redes convolucionais ou modelos baseados em atenção, pode contribuir para ganhos adicionais de desempenho.

A integração de mecanismos de predição em tempo real ao processo de filtragem representa uma direção promissora para aplicações em sistemas dinâmicos, como navegação aérea, veículos autônomos, robótica e controle industrial. Este trabalho contribui para o avanço das técnicas de filtragem adaptativa, evidenciando o potencial do aprendizado de máquina como ferramenta complementar para o aprimoramento da eficácia dos filtros de Kalman em contextos reais.

REFERÊNCIAS

- [1] R. A. Zitar, A. Mohsen, A. E. F. Seghrouchni, F. Barbaresco, and N. A. Al-Dmour, "Intensive review of drones detection and tracking: Linear kalman filter versus nonlinear regression, an analysis case," pp. 2811–2830, 6 2023.
- [2] W. Zhang, X. Zhao, Z. Liu, K. Liu, and B. Chen, "Converted state equation Kalman filter for nonlinear maneuvering target tracking," *Signal Processing*, vol. 202, 1 2023.
- [3] P. Marion, J. Sami, B. Silvere, B. Frederic, F. Marc, and H. Nicolas, "Invariant extended Kalman filter applied to tracking for air traffic control," *2019 International Radar Conference, RADAR 2019*, pp. 1–6, 9 2019.
- [4] C. Ju, Z. Wang, C. Long, X. Zhang, and D. E. Chang, "Interaction-aware kalman neural networks for trajectory prediction." Institute of Electrical and Electronics Engineers Inc., 2020, pp. 1793–1800.
- [5] X. Rong Li and V. Jilkov, "Survey of maneuvering target tracking. part i. dynamic models," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333–1364, 2003.
- [6] L. Aguirre, *Introdução à Identificação de Sistemas*, 01 2015.
- [7] I. Goodfellow, Y. Bengio, and A. Courville, *Deep Learning*. MIT Press, 2016.