

Article

A Modified Kalman Filter Based on Radial Basis Function Neural Networks for the Improvement of Numerical Weather Prediction Models

Athanasios Donas ¹, George Galanis ², Ioannis Pytharoulis ³  and Ioannis Th. Famelis ^{1,*} 

¹ microSENSES Laboratory, Department of Electrical and Electronics Engineering, University of West Attica, Ancient Olive Grove Campus, 250, Thivon Ave., Egaleo, 12241 Athens, Greece; adonas@uniwa.gr

² Mathematical Modeling and Applications Laboratory, Hellenic Naval Academy, Hatzikiriakion, 18539 Piraeus, Greece; ggalanis@hna.gr

³ Department of Meteorology and Climatology, School of Geology, Aristotle University of Thessaloniki, 54124 Thessaloniki, Greece; pyth@geo.auth.gr

* Correspondence: ifamelis@uniwa.gr

Abstract: This study introduces a novel enhancement to the Kalman filter algorithm by integrating it with Radial Basis Function neural networks to improve numerical weather prediction models. Traditional Kalman filters frequently underperform when used by dynamical systems due to their reliance on fixed covariance matrices, resulting in inaccuracies and forecast uncertainty. The proposed modified Kalman filter utilizes Radial Basis Function neural networks to estimate covariance matrices adaptively during the filtering process. This self-adaptive computational system enables the simultaneous targeting of the systematic and the remaining non-systematic parts of the forecast error, producing an innovative and efficient post-process strategy. The suggested methodology is evaluated on predictions of 10-meter wind speed and 2-meter air temperature obtained from the Weather Research and Forecasting model for observation stations in northern Greece. The derived results demonstrate a significant reduction in systematic error, as the bias decreased by up to 88% for 10-meter wind speed and 58% for 2-meter air temperature. Additionally, the forecast variability was successfully mitigated, with the RMSE reduced by 39% and 40%, respectively. Compared to the traditional Kalman filter, which exhibited increased RMSE in several cases and failed to control forecast uncertainty, the proposed approach consistently outperformed by providing stable and reliable predictions across all examined scenarios. These improvements validate the robustness of the method in comparison to conventional techniques, highlighting its potential to produce reliable and stable predictions for environmental applications.

Keywords: air temperature; post-process algorithms; Radial Basis Function neural networks; Kalman filters; Weather Research and Forecasting model; wind speed



Academic Editor: Stephan Havemann

Received: 29 December 2024

Revised: 17 February 2025

Accepted: 19 February 2025

Published: 22 February 2025

Citation: Donas, A.; Galanis, G.; Pytharoulis, I.; Famelis, I.T. A Modified Kalman Filter Based on Radial Basis Function Neural Networks for the Improvement of Numerical Weather Prediction Models.

Atmosphere **2025**, *16*, 248. <https://doi.org/10.3390/atmos16030248>

Copyright: © 2025 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The accurate and stable forecasts produced by numerical weather prediction (NWP) models are frequently employed worldwide to simulate global environmental conditions and support various vital applications, including atmospheric dispersion analysis and wildfire modeling. However, when the forecast pertains to a specific local area, NWP models fail to produce high-quality predictions mainly due to the multiparametric feature of the problem solved. Among others, the problem is significantly influenced by the initial and lateral boundary conditions [1].

One possible solution to tackle these issues might be model resolution enhancement, which would rapidly increase the computational cost of the approach but would not guarantee its success. An alternative strategy would be to utilize post-processing algorithms and assimilation systems to enhance the model's direct output or its initial conditions, respectively. Toward this, Dong et al. [2] introduced a hybrid assimilation system that leverages machine learning to boost numerical weather prediction models, effectively addressing the shortcomings of classic methodologies. In the same direction, Rojas-Campos et al. [3] employed deep learning techniques to post-process NWP precipitation forecasts, achieving substantial improvements in predictive reliability. Krasnopolsky [4] offered a detailed analysis of machine learning applications in data assimilation, model physics, and the post-processing of model outputs, stressing their revolutionary potential for improving weather prediction and atmospheric science. Ladino-Moreno et al. [5] demonstrated the efficacy of combining Monte Carlo Dropout models with deep learning architectures for complex systems like urban hydraulics, while Yanfatriani et al. [6] emphasized the need for robust forecasting systems to tackle extreme rainfall trends and hydrometeorological disasters in tropical regions. Consequently, these studies highlight the growing importance of computational intelligence algorithms in enhancing the NWP model's accuracy and addressing complex environmental challenges.

The proposed methodology is a novel post-processing strategy. More specifically, it aims to improve the predictions of an NWP model for the weather parameters of wind speed and air temperature by addressing both the systematic and the remaining non-systematic components of the forecast error. Systematic errors, or biases, represent consistent and predictable discrepancies that stem from inherent limitations in the model, such as inaccurate parameterizations or incomplete representation of physical processes. These errors are stable under specific conditions and can often be identified and reduced using techniques like bias correction or model recalibration.

On the contrary, non-systematic errors are unpredictable, particularly affecting complex parameters like wind speed and air temperature. For wind speed, such errors may arise from the interplay of meteorological factors, including turbulence, convergence zones, and rapid atmospheric fluctuations that are challenging to capture in numerical models. Similarly, for air temperature, non-systematic errors frequently result from unresolved small-scale phenomena, such as local land-use changes, urban heat effects, or sharp temperature gradients near coastal regions. Their random and chaotic nature presents a significant challenge to mitigation. Effectively addressing both error types is critical to achieving more reliable and accurate predictions of these key atmospheric variables.

The initial objective has been thoroughly explored and addressed using various methods like Artificial Neural Networks (ANNs) [7,8] or advanced statistical models [9,10]. However, in this study, a non-linear Kalman filter is applied. Kalman filters (KFs) are commonly used to minimize systematic errors [11–13] as they are considered a statistically optimal sequential technique that reduces the associated biases by integrating current predictions with recursive observations through weights [14]. Additionally, another key advantage of KF algorithms is their minimal CPU memory requirements, which offer significant benefits in many applications.

In many circumstances though, Kalman filters fail to detect and thereby control the forecast error variability [15], resulting in unstable final predictions due to the remaining uncertainty. Partly, this is a result of the filter's preliminary hypotheses, which frequently require constant covariance matrices that are determined prior to using the KF algorithm. This methodology, however, is unsuccessful when the study concerns dynamical systems as in such cases, the error in the measurements varies because of external factors.

To circumvent this challenge and restrain the remaining white noise, the proposed methodology modifies the non-linear Kalman filter by incorporating Radial Basis Function (RBF) neural networks to update the covariance matrices during the filtering process. This modified Kalman filter integrates RBF neural networks to create a novel self-adaptive framework that continuously adjusts covariance matrices based on real-time input, enabling the algorithm to account for the evolving nature of forecast errors. Unlike existing methodologies—which often target only systematic or non-systematic errors—this approach simultaneously addresses both components, ensuring a holistic improvement in model accuracy. Furthermore, the proposed method leverages a hybrid computational intelligence framework, bridging the adaptability of machine learning with the statistical robustness of Kalman filtering. This dual-targeting capability sets the technique apart as it not only reduces biases but also mitigates the variability and uncertainty associated with random, non-systematic errors.

The suggested system was assessed via an innovative time-window process. In particular, the modified Kalman filter is applied for 10-meter wind speed and 2-meter air temperature forecasts obtained from two different domains of the Weather Research and Forecasting (WRF) model and various observation stations located in the Thessaloniki region (northern Greece). The extracted results are compared with those derived from the non-linear Kalman filter utilizing fixed covariance matrices, through a series of statistical indicators, to measure the degree of improvement offered by the proposed strategy. The suggested approach demonstrates a significant reduction in forecast error metrics, including bias and Root Mean Square Error, showcasing its ability to produce more accurate and stable predictions in complex and dynamic environmental scenarios.

The structure of this study can be summarized as follows: In Section 2, the Weather Research and Forecasting model is presented along with a thorough analysis of the suggested methodology. Section 3 provides the main features of the Kalman filter algorithm alongside a detailed analysis of a non-linear Kalman filter. Section 4 introduces the modified Kalman filter based on RBF NNs, whereas the time-window process is presented in Section 5. The obtained results for the various case studies are discussed extensively in Section 6, while in Section 7, the drawn conclusions are recapped. Finally, Section 8 highlights the key elements and achievements of this study.

2. Materials and Methods

The weather numerical model WRF is presented in this section along with a detailed analysis of the proposed methodology.

2.1. Weather Numerical Model WRF

The Weather Research and Forecasting model is a numerical non-hydrostatic system that generates high-quality atmospheric predictions used in a variety of atmospheric applications [16–18]. More thoroughly, the WRF [19] employs the Advanced Research dynamic solver and is run by the Department of Meteorology and Climatology at Aristotle University of Thessaloniki to generate operational numerical weather forecasts. This modeling system was initially developed as part of the Wave4Us [20] project and effectively applied in several studies on extreme weather incidents in the Mediterranean Sea [21–23] and the hydrography of the Thermaikos Gulf in northern Greece [24].

The forecasts are generated from three one-way nested domains that encompass the following: (a) Europe, the Mediterranean Sea Basin, and Northern Africa (Do1); (b) a significant portion of the central and eastern Mediterranean, including all of Greece (Do2); and (c) northern Greece and adjacent regions (Do3). These domains have horizontal grid spacings of 15 km × 15 km, 5 km × 5 km, and 1.667 km × 1.667 km (longitude–latitude),

respectively. Moreover, high-resolution data from the US Geological Survey (USGS) at 30×30 arcseconds are used to define the land–sea mask, land use, and topography.

The initial and boundary conditions for the outer model domain (Do1) are derived from the operational analysis and predictions of the GFS/NCEP for the 12:00 UTC forecast cycle at a resolution of $0.25^\circ \times 0.25^\circ$ (longitude–latitude). The boundary conditions for Do2 are provided by the forecasts of Do1, while those for Do3 are supplied by Do2. Sea surface temperatures are constrained by the daily global analyses of NCEP at a resolution of $1/12^\circ \times 1/12^\circ$ and remain constant throughout each WRF forecast cycle. The model utilizes 39 sigma levels with improved vertical resolution in the lower troposphere, extending to a top pressure of 50 hPa.

The physical parameterizations incorporate the Eta Ferrier [25] and the Betts–Miller–Janjic [26,27] schemes for the microphysics and cumulus convection (limited to Do1 and Do2). The Mellor–Yamada–Janjic [26,28,29] parameterization is employed for the boundary layer processes, while the Monin–Obukhov (Eta) [26,30] scheme represents the surface layer. Longwave and shortwave radiations are parameterized using the RRTMG [31] scheme, while soil processes are represented by the Unified NOAH land surface model [32], which utilizes four layers to a depth of 2 meters.

The Weather Research and Forecasting simulation system produces results for various environmental parameters, such as air temperature, mean sea level pressure, wind speed, and azimuth, as well as relative humidity. However, this study focuses on the 10-meter wind speed and 2-meter air temperature, which are predicted by the domains Do2 and Do3 (Figure 1a). These weather parameters were selected due to their pivotal role in various environmental and engineering applications, including renewable energy forecasting, maritime operations, and urban climatology. Furthermore, unlike precipitation-related variables such as accumulated rainfall or rain rate, which are heavily influenced by convective processes and microphysical uncertainties, wind speed and air temperature primarily depend on large-scale atmospheric dynamics and surface interactions. Therefore, they are more suitable for assessing systematic and non-systematic forecast errors in NWP models.

The notable spatiotemporal variability and rapid fluctuations in wind speed—caused by the physiographic characteristics of the region of interest as well as by meteorological factors like fronts, convergence lines, and thunderstorms—particularly near the land/sea surface, make accurate predictions challenging, even with high-resolution numerical modeling. Inadequate representation of the land surface characteristics, whether due to horizontal resolution limitations or recent changes in land use, can introduce systematic biases in the prediction of 2-meter air temperature, especially in areas where these features vary significantly.

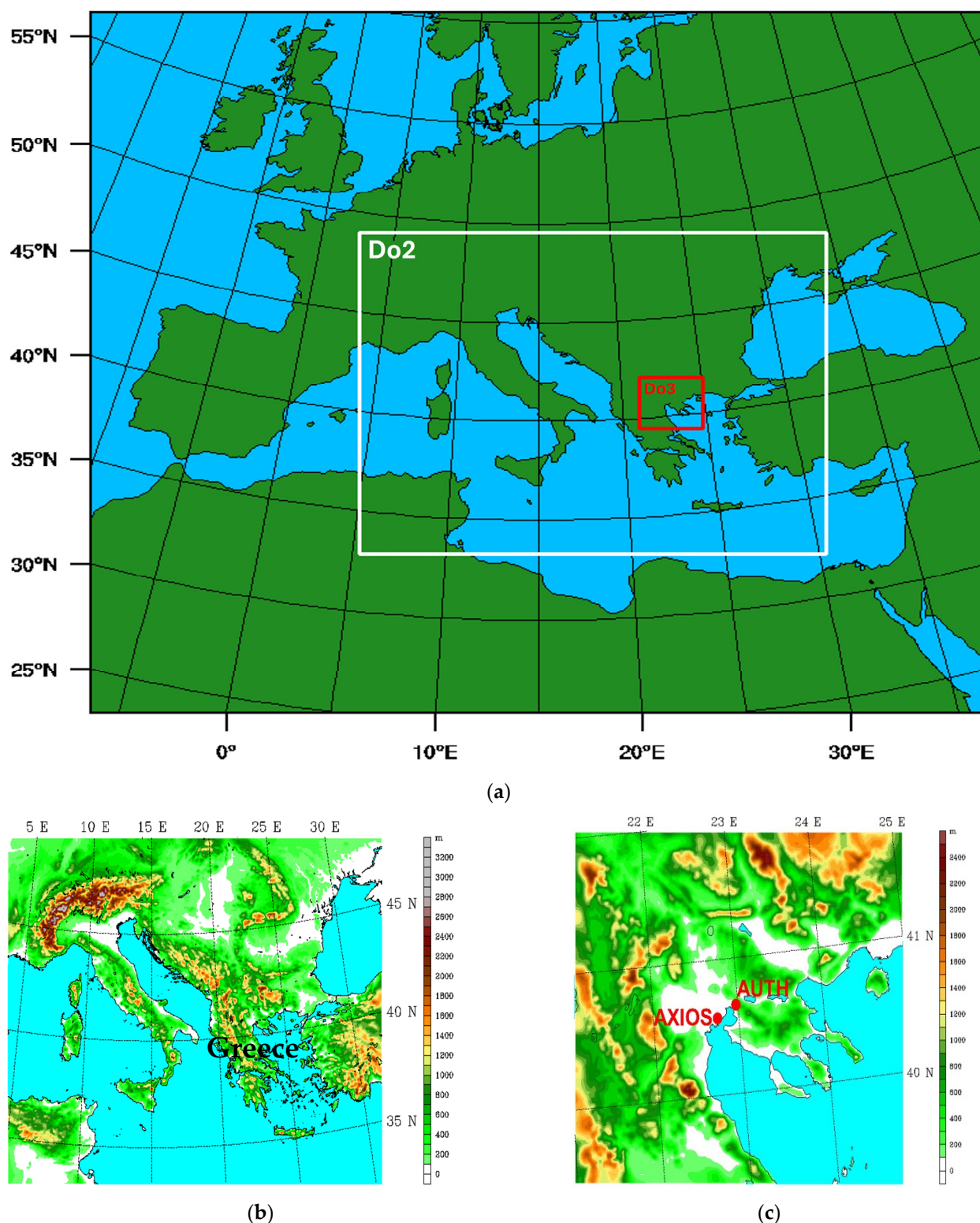


Figure 1. (a) The different domains of the WRF model. The images below show the topography (m) of the domains (b) Do2 and (c) Do3, along with the observation stations (c).

2.2. Methodology

The main objective of this study is to introduce a modified Kalman filter based on Artificial Neural Networks. More specifically, a non-linear Kalman filter is applied in conjunction with Radial Basis Function neural networks to improve the forecasting capacities

of the WRF simulation system regarding the weather parameters of 10-meter wind speed and 2-meter air temperature.

Initially, the non-linear Kalman implementation targets the simulation's systematic error, expressed as a quadratic polynomial of the model's previous prediction, aiming to decode and thus eliminate that bias. During that process, the covariance matrices are not fixed but updated via RBF structures. More precisely, two distinct RBF architectures are utilized to construct a non-linear adaptation rule that correlates the previous and current state of the covariance matrices. The first RBF concerns the evolution of the measurement covariance matrix, whereas the second concerns the evolution of the state covariance matrix. Here, the utilized RBF topologies express non-linear functions that need to be minimized to achieve an accurate approximation of the covariance matrices in every step of Kalman filtering. Through this process, the proposed technique aims to detect the remaining non-systematic part of the prediction error to constrain its variability and the associated forecast uncertainty.

When the filtering process, based on a specified Training data set, is finalized, the systematic error of the simulation is obtained and added to the model's prediction to produce the improved forecasts from the modified Kalman filter, based on a Testing data set. Finally, the extracted results are contrasted with those obtained from the non-linear Kalman filter with stable covariance matrices to measure the efficiency of the suggested approach. The key steps of the proposed strategy are summarized in the diagram of the method (Figure 2).

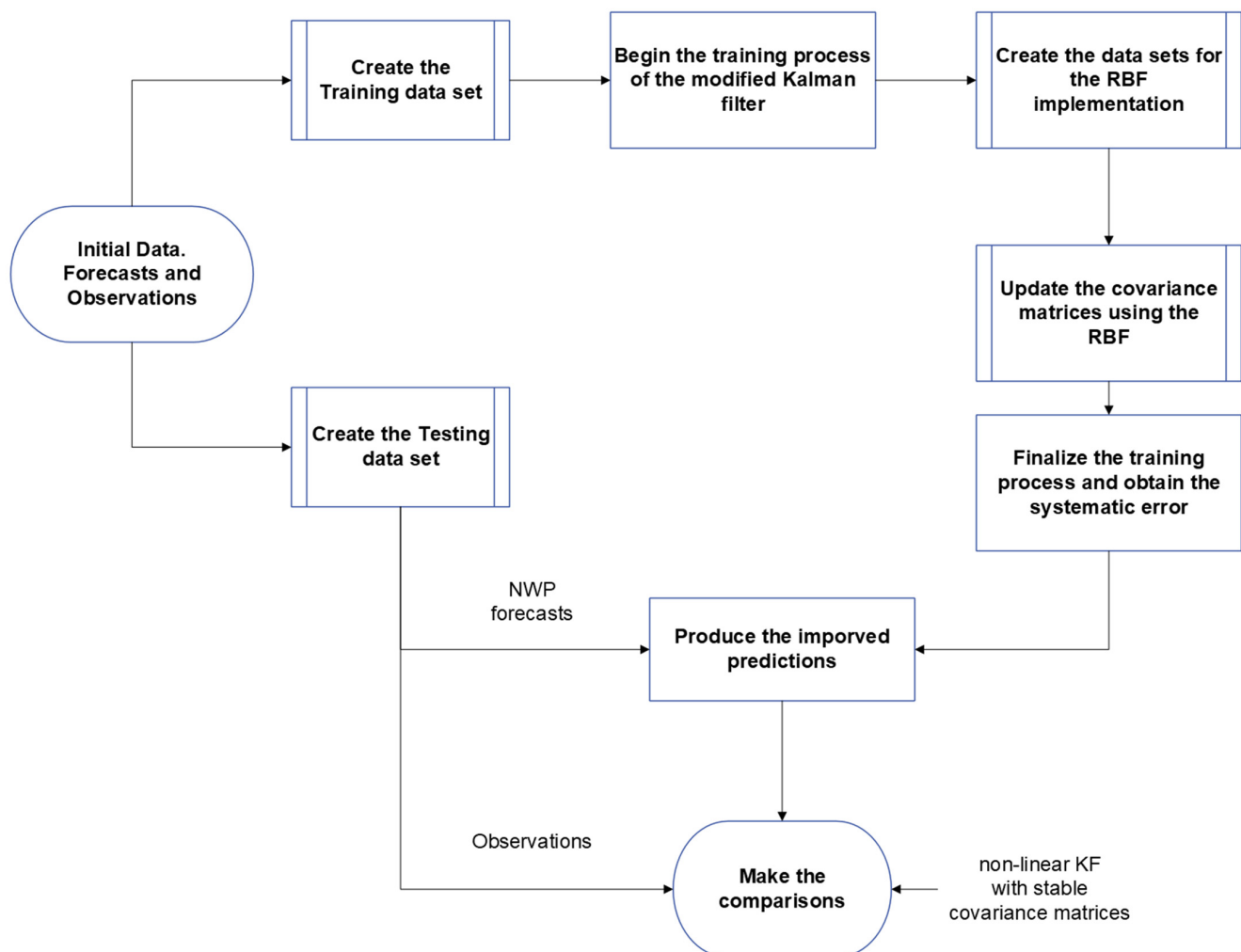


Figure 2. Diagram of the method.

2.3. Study Area and Observation Stations

To implement the suggested algorithm, the novel optimization system requires both the (recorded) observations and the corresponding model predictions. The forecasts are obtained from the numerical weather model WRF, whereas the required observations are from different stations located in the Thessaloniki region in northern Greece (Figure 1c). To be exact, the environmental parameters under study are derived from the AUTH (Aristotle University of Thessaloniki) and the AXIOS (Axios River Delta) stations of the Department of Meteorology and Climatology of AUTH (Figure 1c). The main characteristics of the observation stations are presented in Table 1.

Table 1. Key features of the meteorological stations.

Stations	Location	Latitude (°N)	Longitude (°E)	Time Interval
AUTH	Thessaloniki–Campus of AUTH	40.63177	22.95755	1 h
AXIOS	Thessaloniki–Delta of Axios River	40.55157	22.74187	1 h

The Thessaloniki region was selected as the study area due to its diverse meteorological and topographical characteristics, which make it an ideal evaluation site for assessing post-processing techniques in numerical weather prediction models. Thessaloniki is a coastal city influenced by both continental and marine atmospheric dynamics, resulting in significant spatial and temporal variability in wind speed and air temperature. These variations are driven by complex interactions between synoptic-scale weather patterns, local topography, and land–sea contrasts.

The AUTH and AXIOS stations were selected for their high-quality, uninterrupted observations, which are essential for effectively training and validating the proposed modified Kalman filter. The forecasts at these locations exhibit significant variability in simulation errors, making the two stations particularly well-suited for evaluating the effectiveness of the suggested optimization strategy.

Furthermore, the selection of these two stations, representing distinct environmental settings—an urban location (AUTH) and a coastal plain (AXIOS)—enables a broader evaluation of the method’s adaptability under different local meteorological conditions. Finally, this selection also ensures a robust assessment of the methodology’s ability to generate reliable and accurate final predictions in complex environments, overcoming key limitations of NWP models and existing forecasting techniques, like the Kalman filter.

3. Kalman Filter

This section presents the fundamentals of Kalman’s theory along with a thorough analysis of a non-linear Kalman filter with stable covariance matrices.

3.1. Fundamentals of the Kalman Filter Theory

Let x_t be a vector (the state vector) that describes an unknown process at time t and y_t a known pertinent vector at the same time. It is assumed that the evolution of the state vector x from time $t - 1$ to t is given by the following system equation:

$$x_t = F_t * x_{t-1} + w_t, \quad (1)$$

whereas its correlation with the observable vector y_t is given by the measurement equation (or observation equation):

$$y_t = H_t * x_t + v_t. \quad (2)$$

where F_t expresses the system transition matrix, and H_t is the corresponding measurement transition matrix. Both of them must be determined before the application of the filter, and the same goes for the covariances matrices Q_t and R_t of the random vectors w_t and v_t , respectively. The variables w_t and v_t are non-systematic parts of the model error that follow the Gaussian distribution with a zero mean. Furthermore, they are independent, which denotes that $E(w_q \cdot v_k) = 0$ for any $q, k \in N$, and also time-independent, which implies that $E(w_q \cdot w_k) = 0$ and $E(v_q \cdot v_k) = 0$ for all $q \neq k$.

Kalman's theory [33–35] provides a recursive method for estimating the state vector x_t , utilizing the available (recorded) observation(s) stored in vector y_t . Roughly speaking, the Kalman filter algorithm can be summarized in three stages:

1. Initialization stage: here, initial values for the state vector x and its error covariance matrix P at time $t - 1$ are determined;
2. Prediction stage: based on the defined values, an optimal estimation for the variables x and P at time t is given by the following:

$$\hat{x}_{t/t-1} = F_t * \hat{x}_{t-1} \quad (3)$$

$$P_{t/t-1} = F_t * P_{t-1} * F_t^T + Q_t \quad (4)$$

where Equations (3) and (4) are called prediction equations and $\hat{x}_{t-1} = E[x_{t-1}]$;

3. Correction stage: when vector y_t is available, the corrected values for the state vector and its error covariance matrix, at time t , are outlined below:

$$\hat{x}_{t/t} = \hat{x}_{t/t-1} + K_t * (y_t - H_t * \hat{x}_{t/t-1}), \quad (5)$$

$$P_t = (I - K_t * H_t) * P_{t/t-1} \quad (6)$$

where

$$K_t = P_{t/t-1} * H_t^T * (H_t * P_{t/t-1} * H_t^T + R_t)^{-1}. \quad (7)$$

Equations (5) and (6) are known as correction (update) equations and conduct the third and final stage of the Kalman filter process. Equation (7) presents the Kalman gain, which is the most vital parameter of the filtering procedure as it indicates the amount of influence the available observation will exert on the state vector prediction. More thoroughly, a small value for the Kalman gain denotes a high level of uncertainty in the measurements.

To utilize the KF algorithm, initial values must be defined for the state vector x_t , its error covariance matrix P_t , and additionally for the covariance matrices Q_t and R_t . However, their effects on filter performance have different magnitudes. Both x_t and P_t converge to their actual values eventually, thus, their effect is not significant. In contrast, that is not the case with the covariance matrices R_t and Q_t as their impact on the Kalman gain will determine the degree of filter adjustment to any possible new conditions [36]. Consequently, the chosen adaptation rule is of utmost importance since it will have a severe effect on the filter's successful implementation.

Various approaches have been developed to update these quantities. Some researchers utilize predefined and fixed covariance matrices before initiating the filtering process [34], while others dynamically update them during the procedure using the most recent seven values of $w_t = x_t - x_{t-1}$ and $v_t = y_t - x_t$ [11]. Here, for the non-linear Kalman filter implementation, the initial strategy is employed.

3.2. A Non-Linear Kalman Filter with Stable Covariance Matrices

To decode the standard error in the direct outputs of the WRF model, a non-linear Kalman filter is utilized. More specifically, that error (y_t) is expressed as a quadratic

polynomial [37] of the model's previous prediction (10-meter wind speed or 2-meter air temperature) $pred_{t-1}$ [38]:

$$y_t = x_{0,t} + x_{1,t} * pred_{t-1} + x_{2,t} * pred_{t-1}^2 + v(t) \quad (8)$$

where y_t represents the disparity between the observed measurement (recorObservation) and the associated prediction generated by the simulation system:

$$y_t = \text{recorObservation}_t - pred_t$$

Equation (8) expresses the measurement equation with the state vector $x_t = [x_{0,t} \ x_{1,t} \ x_{2,t}]$ and transition matrix $H_t = [1 \ pred_{t-1} \ pred_{t-1}^2]$. On the other hand, the system transition matrix is set to be $F_t = 1$ since the progression of the state vector over time is uncertain, caused by the lack of pertinent knowledge.

Therefore, the system and the measurement equations for the quadratic variation of the non-linear Kalman filter are defined as follows:

$$x_t = [x_{0,t-1} \ x_{1,t-1} \ x_{2,t-1}]^T + w_t \quad (9)$$

and

$$y_t = [1 \ pred_{t-1} \ pred_{t-1}^2] x_t + v_t. \quad (10)$$

The next step concerns the determination of the initial values. To begin with, as there is insufficient information about the previous condition of the state vector, it is assumed that its value is zero, $x_0 = 0$. Furthermore, regarding its error covariance matrix P at the same time, this study suggests a diagonal matrix with relatively large values, which denotes poor initial guesses. More specifically, the following is proposed:

$$P_0 = \begin{bmatrix} 4 & 0 & 0 \\ 0 & 4 & 0 \\ 0 & 0 & 4 \end{bmatrix}.$$

The selection of the covariance matrices, though, is not that clear and thus requires a more careful analysis. Here, their values are considered stable during the filtering procedure and defined before using the Kalman algorithm [34,39]. In particular, a series of sensitivity tests are conducted with different combinations of Q and R to determine the optimal one for the various weather parameters. The extracted results reveal that for the 10-meter wind speed and the 2-meter air temperature, the optimal values were $R_t = 4$ and $Q_t = I_3$, respectively, where I_3 expresses the identity matrix.

4. The Proposed Methodology

The strategy of fixed covariance matrices, as used by the quadratic Kalman filter presented in Section 3, is inappropriate for dynamic systems because the error in the measurements diverges due to a variety of external factors [40]. Thus, an alternative approach is necessary. More precisely, to update the variables R and Q during the filtering process, this study modifies an estimation rule suggested by Akhlaghi et al. [41].

The initial methodology provides the following adaptation formula:

$$R_t = \alpha R_{t-1} + (1 - \alpha) (\varepsilon_t \varepsilon_t^T + H_{t/t-1} P_{t/t-1} H_{t/t-1}^T) \quad (11)$$

$$Q_t = \alpha Q_{t-1} + (1 - \alpha) (K_t d_t d_t^T K_t^T) \quad (12)$$

where the quantity a is a fixed positive (memory) parameter, $0 < a \leq 1$, which influences the effect of the preceding and subsequent states on the current outcome; and the quantities $\varepsilon_t = y_t - H_t \hat{x}_{t/t}$ and $d_t = y_t - H_t \hat{x}_{t/t-1}$ are the residual and the innovation, respectively. The residual is defined as the difference between the recorded observation and its projected value based on $\hat{x}_{t/t}$, while the innovation presents the corresponding discrepancy between the observed value and its forecasted value derived from $\hat{x}_{t/t-1}$. More information about the estimation rule can be found in Akhlaghi et al. [41].

Before this study, Donas et al. [42] proposed a hybrid Extended Kalman Filter that utilizes Equations (9) and (10) to update the covariance matrices. The main novelty of that research was the determination of the memory parameter, α , through various ANN architectures and in particular via the Radial Basis Function and FeedForward neural networks. These parametrized ANNs were trained for multiple values of that parameter to identify the optimal one, which afterward was applied by an Extended Kalman Filter to estimate the covariance matrices in every step of the filtering process. The results revealed two key findings: Firstly, the value of the memory parameter α and the linear correlation that creates significantly affect the efficiency of the hybrid system. Secondly, the RBF structure emerged as the most effective ANN topology, primarily due to its low computational cost.

To eliminate dependence on the memory parameter and the linear assumptions it imposes, this study introduces an innovative estimation rule for the covariance matrices R and Q by utilizing Radial Basis Function neural networks. Specifically, RBFs are integrated into the filtering procedure, enabling a non-linear association between the previous and current states of the covariance matrices that replace the parameter-dependent approach. This modification exploits the advantages of machine learning and statistical filtering, providing a robust, adaptive, and computationally efficient post-processing algorithm.

4.1. Radial Basis Function Neural Networks

Radial Basis Function neural networks are a unique and widely used type [43–45] of ANN that are characterized by their simple structure and various efficient training algorithms [46]. A typical RBF structure consists of one input layer, one hidden layer with radial (basis) functions (ϕ) as activation functions, and one linear output layer, as illustrated in Figure 3.

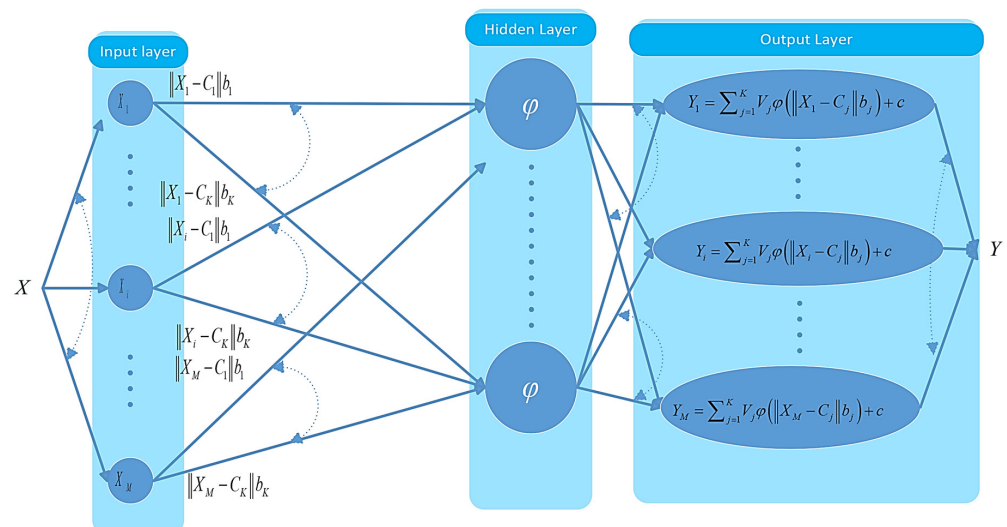


Figure 3. A three-layer Radial Basis Function neural network.

At the outset, the RBF network calculates the distance between the t th input vector (X_t) and each centroid (C_j) in the hidden layer. This outcome is then multiplied by the so-called width (b_j) parameter, which defines the stretching degree of the transfer function (stretch or compress). Consequently, the net input for the j th neuron is computed by the following:

$$netinput_j = ||X_t - C_j|| b_j, \quad (13)$$

where $||\circ||$ represents the Euclidean distance.

Through the activation function, each net input is transformed and generates the output of its corresponding neuron, which is multiplied by the respective external weight (V_j) that connects the j neuron of the hidden layer with the output layer. Afterward, the produced results sum up along with a scalar parameter c (bias of the output layer) and form the direct output of the Radial Basis Function neural network:

$$Y_t = \sum_{j=1}^K V_j \varphi(||X_t - C_j|| b_j) + c, \quad (14)$$

where K is the total number of centroids.

Based on the above, Equations (11) and (12) create the proposed modified estimation rule for the covariance matrices R_t and Q_t by utilizing the following transformation:

$$R_t = R_{t-1} + Y_t \quad (15)$$

$$Q_t = Q_{t-1} + \hat{Y}_t \quad (16)$$

replacing Equation (14) with Equations (15) and (16), the suggested adaptive rule becomes the following:

$$R_t = R_{t-1} + \sum_{j=1}^K V_j \varphi(||X_t - C_j|| b_j) + c \quad (17)$$

$$Q_t = Q_{t-1} + \sum_{j=1}^K \hat{V}_j \varphi(||\hat{X}_t - \hat{C}_j|| \hat{b}_j) + \hat{c}, \quad (18)$$

where $X_t = [\varepsilon_t \ H_{t/t-1} \ P_{t/t-1}^T]^T$ and $\hat{X}_t = [K_t^T \ d_t]^T$ are the t th input (column) vectors for the Radial Basis Function neural networks. In particular, the vector X_t concerns the evolution of the measurement covariance matrix R_t , while $\hat{X}_t$ concerns the corresponding evolution of the state covariance matrix Q_t . Note here that, since we work in a three-dimensional state space (quadratic KF), the variable $P_{t/t-1}$ is a 3×3 matrix and, thus, it must be converted to a column vector before it is utilized as a component of the X_t input vector. Additionally, here, K_t is a 3×1 column vector.

Following the discussed procedure, the next step is to define the size of the RBF network, i.e., the number of centroids, and the corresponding activation function. Regarding the activation function, the Gaussian, $\varphi(x) = e^{-x^2}$, is employed because of its widespread preference in RBF modeling [47]. However, selecting the centroids is less straightforward. To address this, a series of experiments were conducted, varying the number of centroids from 5 to 20. The optimal configuration was the one that produced the lowest training error.

The results indicated that ten centroids provided the best balance for the weather parameters under study, achieving a trade-off between computational efficiency and the complexity of forecast error patterns. While increasing the number of centroids beyond ten yielded slight improvements in accuracy, the gains were marginal and came at a significantly higher computational cost, particularly considering the iterative nature of the Kalman filtering process. Conversely, using fewer than ten centroids led to a marked reduction in performance as the network struggled to capture the non-linear relationships between forecast errors and evolving covariance matrices. Hence, choosing ten centroids ensured sufficient generalization of the RBF structure while maintaining computational feasibility.

Once the optimal RBF topology is identified, the final stage of the suggested approach involves the determination of the network parameters through a training algorithm. The selected learning rule aims to minimize the quantities Y_t and $\hat{Y}_t$ to achieve an accurate approximation for the covariances matrices in time t based on Equations (17) and (18). Typically, two main categories exist for training a Radial Basis Function neural network: the first procedure is frequently faster and divides the training process into two stages. Initially, the number and the locations of the hidden layer's centroids are defined; then, the synaptic weights between the hidden and the output layer are specified via linear regression. In contrast, the second approach aims to determine all the network parameters simultaneously using non-linear optimization algorithms [48–50]. Here, the second strategy is applied.

More thoroughly, the Levenberg–Marquardt (LM) is implemented as a training algorithm. LM can be considered as a combination of the steepest descent method and the Gauss–Newton algorithm, inheriting the positive traits of both. To be more precise, Levenberg–Marquardt carries out a combined training process: in high-complexity regions, it transitions to the steepest descent algorithm until the local curvature is suitable for a quadratic approximation. At this point, it roughly transforms into the Gauss–Newton algorithm, which can accelerate convergence considerably [51].

The key features of the Levenberg–Marquardt training algorithm are summarized in the following diagram (Figure 4).

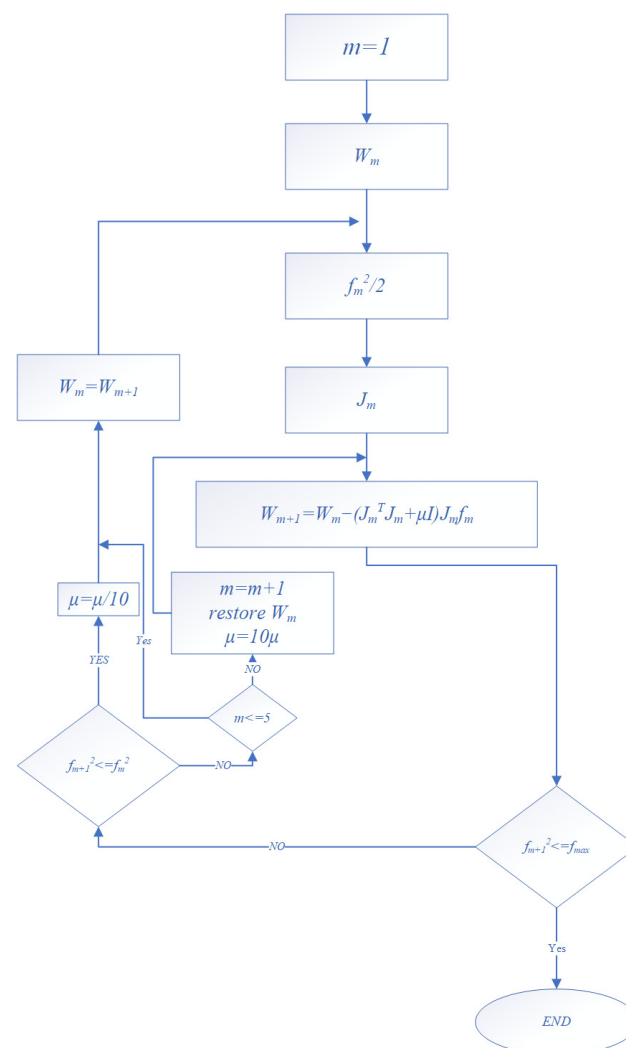


Figure 4. Levenberg–Marquardt training algorithm.

In the figure, the following notation applies:

- W is a column vector that contains the parameters of the RBF neural network, $W = [C, b, V, c]$ or $W = [\hat{C}, \hat{b}, \hat{V}, \hat{c}]$. Note here that the locations of the estimated centroids must be within the range of the input vector and also that the values of the widths must be positive;
- f is a non-linear function expressed as a direct output of the RBF, $f = \sum_{j=1}^K V_j \varphi(\|X_t - C_j\| b_j) + c$ or $f = \sum_{j=1}^K \hat{V}_j \varphi(\|\hat{X}_t - \hat{C}_j\| \hat{b}_j) + \hat{c}$;
- J is the Jacobian of the non-linear function f ;
- μ is a positive variable that ensures the reversibility of the $J^T J$ matrix. Here, the initial value for the combination parameter was equal to $\mu = 0.001$.

Another aspect that needs to be clarified before the presentation of the modified Kalman filter is the phenomenon of over-fitting. Here, the RBF networks are employed exclusively as non-linear function approximators to update the covariance matrices of the Kalman filter and not as predictive models for the environmental parameters under study. Therefore, issues such as over-fitting, which typically arise when a model is over-trained on specific data, are inherently avoided.

4.2. The Modified Kalman Filter

Applying the above strategy to the quadratic Kalman filter, presented in Section 3, the modified Kalman filter is constructed. More specifically, the suggested optimization system performs the following steps:

1. Initialization. For $t = 0$, set

$$\begin{aligned}\hat{x}_0 &= E[x_0], \\ P_0 &= E[(x_0 - E[x_0])(x_0 - E[x_0])^T], \\ R_0, Q_0\end{aligned}$$

2. Iteration. For $t = 1, 2, \dots$, Training data compute:

Prediction Step.

$$\hat{x}_{t/t-1} = F_t * \hat{x}_{t-1} \quad (19)$$

$$P_{t/t-1} = F_t * P_{t-1} * F_{t-1}^T + Q_{t-1} \quad (20)$$

Correction Step.

$$d_t = y_t - H_t * \hat{x}_{t/t-1} \quad (21)$$

$$K_t = P_{t/t-1} * H_{t/t-1}^T * [H_{t/t-1} * P_{t/t-1} * H_{t/t-1}^T + R_t]^{-1} \quad (22)$$

$$\hat{x}_{t/t} = \hat{x}_{t/t-1} + K_t * [y_t - H_t \hat{x}_{t/t-1}] \quad (23)$$

$$\varepsilon_t = y_t - H_t * \hat{x}_{t/t} \quad (24)$$

Apply LM after converting the $P_{t/t-1}$ to a column vector, and obtain the minimized Y_t ($\min Y_t$) based on $X_t = [\varepsilon_t \ H_{t/t-1} \ P_{t/t-1}^T]^T$.

Update the measurement noise covariance matrix and recalculate the Kalman gain:

$$R_t = |R_{t-1} + \min Y_t| \quad (25)$$

$$K_t = P_{t/t-1} * H_{t/t-1}^T * [H_{t/t-1} * P_{t/t-1} * H_{t/t-1}^T + R_t]^{-1} \quad (26)$$

Apply LM and obtain the minimized $\hat{Y}_t$ ($\min \hat{Y}_t$) based on $\hat{X}_t = [K_t^T \ d_t]^T$.

Update the state covariance matrix and make the second state correction:

$$Q_t = |Q_{t-1} + \min \hat{Y}_t| \quad (27)$$

$$\hat{x}_{t/t} = \hat{x}_{t/t-1} + K_t * [y_t - H_t \hat{x}_{t/t-1}] \quad (28)$$

Update the error covariance matrix:

$$P_{t/t} = [I_3 - K_t * H_{t/t-1}] * P_{t/t-1} \quad (29)$$

Following the application of the modified Kalman Filter, the systematic error of the WRF model at time t , represented as $\hat{x}_t$, is obtained. This component is added to the model's prediction at time $t + 1$, resulting in an enhanced forecast for the various weather parameters.

$$\text{Improvedpred}_{t+1} = \text{pred}_{t+1} + H_{t+1} * \hat{x}_t$$

The adaptive nature of the proposed method allows the creation of a vigorous, efficient filter that effectively addresses both systematic and non-systematic forecast errors by continuously adjusting to the evolving uncertainties inherent in numerical weather prediction models. While Kalman filters and Radial Basis Function neural networks are well-established techniques, this combined approach provides an innovative contribution to the current landscape of weather forecasting methodologies. Specifically, it forms a conceptual bridge between traditional data assimilation methods and contemporary deep learning tools [52–54], like Transformer-based models and Graph Neural Nets, leveraging their strengths to deliver a robust and complementary solution for advanced environmental prediction systems.

The proposed modified Kalman filter is also designed to be a time-efficient computational system. The Kalman filter is inherently efficient, with a computational complexity of $O(\text{dimstate}^3)$, with dimstate being the dimension of the state vector (here $\text{dimstate} = 3$). This complexity arises due to the matrix inversion required in the update state. However, its sequential nature makes it well-suited for real-time applications as it processes data iteratively rather than in large batches.

Moreover, the RBF architecture, trained by the Levenberg–Marquardt algorithm, is employed exclusively for covariance adaptation. This introduces an additional complexity $O(\text{Iter} * (C * \text{dimstate} + C^3))$, with Iter being the number of LM's iterations (in this case $\text{Iter} = 1$), and C the number of centroids. Nonetheless, since training is performed periodically rather than at every filtering step, its impact on computational cost is substantially reduced. Therefore, by strategically integrating these elements, the proposed system achieves an optimal balance between predictive accuracy and computational efficiency, making it scalable and applicable to large-scale data sets without excessive processing demands.

5. Time-Window Process

To evaluate the efficiency of the proposed methodology, a time-window process is utilized for various weather parameters derived from two different domains (Do2 and Do3) of the WRF model and at two stations located in the Thessaloniki region, northern Greece. More precisely, the suggested modified Kalman filter was applied to 10-meter wind speed and 2-meter air temperature forecasts obtained at the locations of AUTH and AXIOS stations for 2020. During this process, the stability of the proposed method is assessed along with the degree of improvement it provides over classical forecasting methodologies, based on a variety of statistical measures.

Each time window applies Algorithm 1 using predetermined Training data sets and forecasting intervals (Testing data). These hyper-parameters are not selected at random but rather following a series of tests, the results of which are summarized in Table 2.

Algorithm 1: Time-Window Process

Initial data:	{Inputs, Targets} $\rightarrow$ {Model's Forecast, Observations}
Normalize the data:	$[-1, 1]$
History and the Forecast data set definition	
Maximum number of time windows:	$Max_tw = Time\ Windows - 1$
Necessary matrix and vectors to store the results	
Define the RBF NN neurons:	Here, the number of neurons ranges from 5 to 20.
for $qq = 0, \dots, Max_tw$	
$Tr \rightarrow (1 + qq) : Training\ data + qq$	Data set for the modified Kalman filter training.
Apply the modified Kalman filter	Obtain the systematic error as $\hat{x}_t$.
$Tr_1 \rightarrow Tr(end) + 1 : Tr(end) + Testing\ data$	Data set for the method's assessment.
Denormalize the data	Model predictions, recorded observations, and produced predictions via modified filter.
Evaluate the method based on Tr_1 and store the result	Bias, RMSE, and Ns indices.
End	
Store the aggregate results from every time window	

Table 2. Main characteristics of the time-window process.

Temperature	Time Windows	Training Data Set	Testing Data Set
AUTH Do2	3	375	18
AXIOS Do2	2	375	18
AXIOS Do3	3	470	18
Wind Speed	Time Windows	Training Data Set	Testing Data Set
AUTH Do2	3	400	18
AUTH Do3	3	400	18
AXIOS Do2	3	600	18
AXIOS Do3	3	600	18

Analyzing the hyper-parameters (Table 2) of the time-window process, it is observed that a fixed Training data set that covers all stations and weather parameters does not exist. Therefore, the data necessary for the modified Kalman filter's training procedure vary. However, focusing on the Testing data sets and time windows, a different conclusion is extracted. The executed experiments revealed that the ideal value for the forecasting interval was stable and equal to eighteen observations, while, in most cases (except for AXIOS Do2—Table 2), the corresponding ideal value for the time windows was three. The selection of the time-window size plays a crucial role in the performance of the modified Kalman filter as it determines the balance between the amount of training data available for adaptation and the model's responsiveness to new conditions. To assess its impact, a sensitivity analysis was conducted by varying the time-window size from 1 to 5 while keeping all other hyper-parameters constant.

The results indicate that a time window of 3 provides the optimal trade-off between stability and forecast accuracy. When the time-window parameter was reduced to 1 or 2, the filter exhibited increased sensitivity to short-term variations, which led to unstable covariance matrix updates and higher RMSE values. Conversely, increasing the time window beyond 3 (e.g., 4 or 5) resulted in a loss of adaptability as the filter became less responsive to evolving forecast errors.

Furthermore, the value of the time window determines the total number of improved predictions that the proposed system will produce, based on a certain amount of Training data. For instance, the hyper-parameters for the AUTH Do2 are *Time Windows* = 3, *Traning data set* = 375, and *Testing data set* = 18, which implies fifty-four final forecasts utilizing one thousand, one hundred, and twenty-five previous (training) data points

(375 data points for each time window). For more clarity, the suggested procedure is outlined in Algorithm 1 and in Appendix C.

Each Testing set consists of 18 consecutive prediction samples, representing an 18-hour forecast horizon—a period that is operationally relevant in numerical weather prediction. Although 18 samples may appear limited at first glance, the rolling nature of the time-window approach ensures robust validation by capturing the temporal variability inherent in weather data. Furthermore, the variation in Training set sizes (ranging from 375 to 600 samples) reflects the differing data availability and characteristics across the various observation stations and domains. These sizes were optimized through extensive sensitivity analyses to best suit the individual case studies. Given the sequential and time-dependent nature of the utilized data, standard cross-validation techniques are less applicable; however, the suggested time-window strategy effectively serves as a tailored form of cross-validation, ensuring that the validation process remains rigorous and reflective of real-world forecasting conditions.

Validation Procedure

A rigorous validation procedure was employed to ensure the efficiency of the proposed modified Kalman filter. The assessment was conducted using statistical measures that quantify the degree of improvement provided by the method. The key statistical indicators that were utilized are as follows:

- Bias of the forecasting values:

$$\text{Bias} = \frac{\sum_{t=1}^N (\text{obs}(t) - \text{for}(t))}{N}$$

where $\text{obs}(t)$ presents the recorded observation at time t ; $\text{for}(t)$ is the respective forecast (improved forecast through the modified filter or model's direct output); and N the size of the forecasting data set. Bias is a crucial statistical measure for every filtering process since we seek to eliminate the systematic error;

- Root Mean Square Error:

$$\text{RMSE} = \sqrt{\frac{\sum_{t=1}^N (\text{obs}(t) - \text{for}(t))^2}{N}}$$

This crucial indicator quantifies the model's ability to estimate the general functioning of the system, with a specific focus on the non-systematic part of the forecast error;

- Nash–Sutcliffe model efficiency coefficient:

$$\text{Ns} = 1 - \frac{\sum_{t=1}^N (\text{obs}(t) - \text{for}(t))^2}{\sum_{t=1}^N (\text{obs}(t) - \overline{\text{obs}})^2}$$

where $\overline{\text{obs}}$ expresses the mean value of the (recorded) observations. The Nash–Sutcliffe is a normalized statistic that fluctuates between $-\infty$ and 1 and dictates the level of concordance between observations and forecasts.

It should be noted that Earth Control Points (ECPs) were not utilized in this study. Instead, the validation relied on direct comparisons between the observed data from the AUTH and AXIOS stations and the corresponding forecasts generated by the WRF model. To achieve this direct comparison and further ensure the consistency of the validation process, the WRF model outputs were interpolated to the exact locations of the meteorological stations. This approach minimizes discrepancies arising from differences in spatial scales

and allows for a more accurate evaluation of the modified Kalman filter's effectiveness in improving local predictions.

6. Results

This section illustrates the results obtained from the proposed modified Kalman filter for the parameters of 2-meter air temperature (Tem) and 10-meter wind speed (WS). In particular, the following Figures 5–11 present the forecasting time series diagrams (based on Testing data sets) for each time window, whereas Tables 3–6 below contain the corresponding (average) values of the various evaluation indicators. It is crucial to note at this point that the case study AUTH Do3 for the 2-meter air temperature was excluded from this research due to the low non-systematic inaccuracy that was observed, implying that using an ANN is not highly beneficial.

A thorough analysis for each time window is available in Appendices A and B, respectively.

Table 3. Evaluation indicators. Average results from AUTH station, Do2 domain, for 2-meter air temperature.

Average Results–AUTH Do2	Bias	RMSE	Ns
Model	−2.9420	3.1923	−1.8749
Kalman	1.1296	3.6239	−2.8826
Modified Kalman	0.6833	1.5844	0.4141

Table 4. Evaluation indicators. Average results from AXIOS station, Do2 and Do3 domains, for 2-meter air temperature.

Average Results–AXIOS Do2	Bias	RMSE	Ns
Model	−3.7458	4.0487	−2.2314
Kalman	9.1426	9.2855	−16.0506
Modified Kalman	2.5373	3.0586	−0.8557
Average Results–AXIOS Do3	Bias	RMSE	Ns
Model	−4.4672	4.8905	−5.8089
Kalman	4.8820	5.2990	−8.4395
Modified Kalman	1.6307	2.6914	−0.8372

Table 5. Evaluation indicators. Average results from AUTH station, Do2 and Do3 domains, for 10-meter wind speed.

Average Results–AUTH Do2	Bias	RMSE	Ns
Model	−1.7126	2.4256	−23.6023
Kalman	0.7395	1.6540	−10.2769
Modified Kalman	0.3858	1.4421	−7.5849
Average Results–AUTH Do3	Bias	RMSE	Ns
Model	−4.9319	6.3636	−9.4715
Kalman	−1.1546	3.5911	−2.3361
Modified Kalman	−0.3595	3.0696	−1.4395

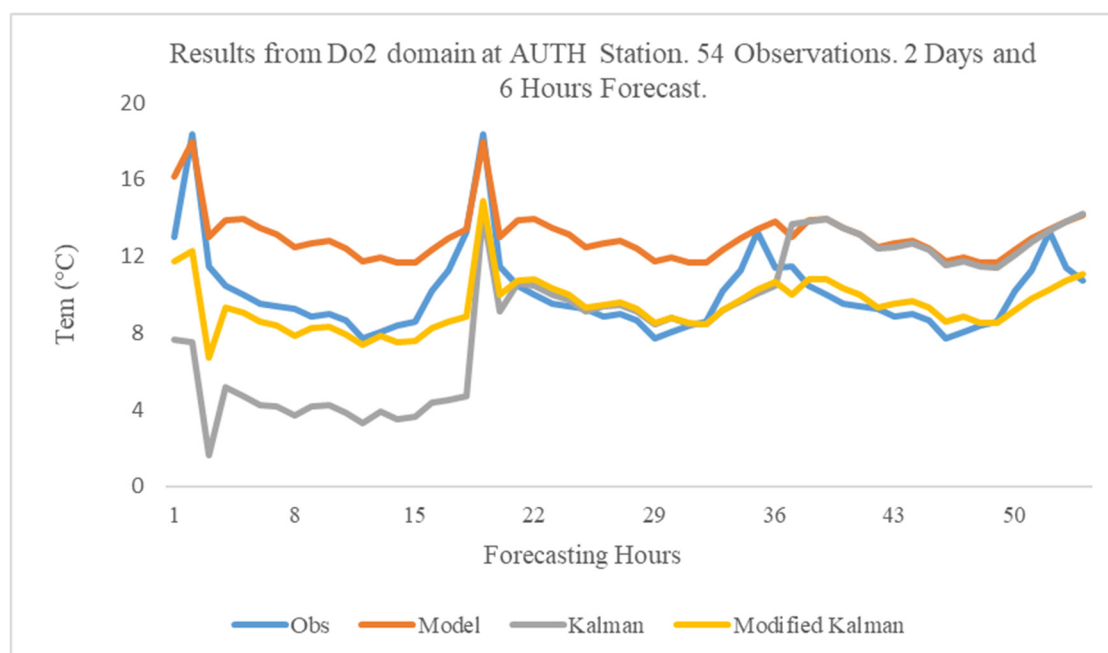
Table 6. Evaluation indicators. Average results from AXIOS station, Do2 and Do3 domains, for 10-meter wind speed.

Average Results—AXIOS Do2	Bias	RMSE	Ns
Model	−1.0767	1.6384	−5.3589
Kalman	−1.2850	2.2946	−13.6531
Modified Kalman	−0.1265	1.0561	−1.6278
Average Results—AXIOS Do3	Bias	RMSE	Ns
Model	−1.1446	1.4911	−2.1064
Kalman	−0.7893	2.8227	−0.3158
Modified Kalman	−0.0856	1.0898	−0.2628

6.1. Evaluation of 2-Meter Air Temperature Forecasts from WRF Do2 and Do3 Domains at AUTH and AXIOS Stations

Analyzing the results obtained at the AUTH station, it is observed that the modified Kalman filter significantly enhances the Do2 forecasts of the weather numerical model WRF. Every evaluation indicator (Table 3) was noticeably improved. More specifically, the average reduction in the bias index exceeds 75% (changing sign), which implies that the suggested approach successfully reduces the standard error of the simulation. Furthermore, the modified filter was able to detect and, thus, constrain the non-systematic part of the forecast error as the RMSE assessor decreased by 50%. On the other hand, the corresponding value of RMSE from the non-linear Kalman filter implementation increased by almost 14%, which dictates poor final predictions.

The above can also be concluded by studying the time series diagram (Figure 5). The predictions produced by the modified Kalman filter describe the morphology of the recorded observations (Figure 5—Obs) more accurately in comparison to those derived from the WRF numerical system. Hence, the proposed methodology not only improves the forecasting capabilities of the NWP model in use but also avoids the limitations of classic Kalman filters.

**Figure 5.** Time series diagram. Results from AUTH station at Do2 domain for 2-meter air temperature.

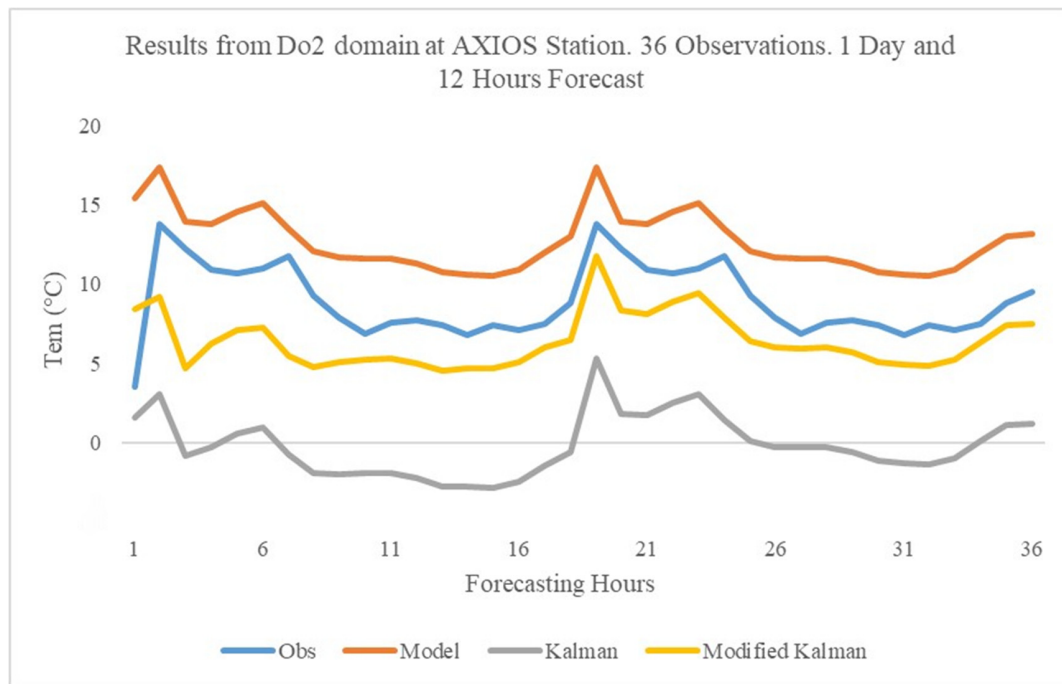


Figure 6. Time series diagram. Results from AXIOS station at Do2 domain for 2-meter air temperature.

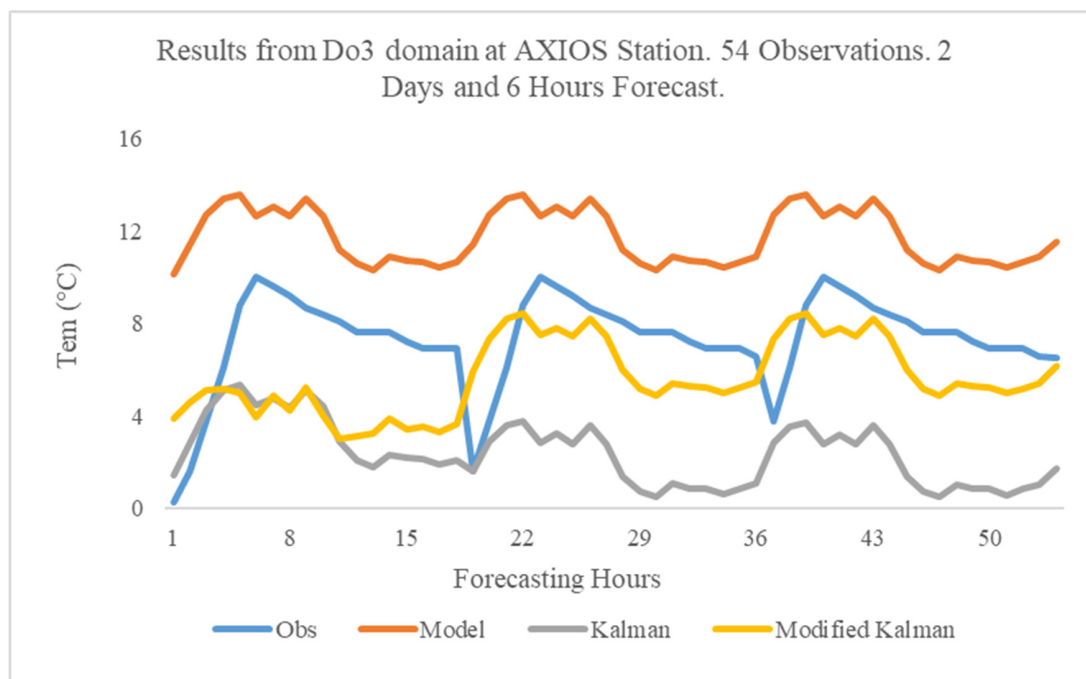


Figure 7. Time series diagram. Results from AXIOS station at Do3 domain for 2-meter air temperature.

Moving to the AXIOS station, the proposed filter exhibits similar behavior. Focusing on the Do2 (Table 4—Do2), it is revealed that Kalman's initial implementation fails to improve the forecasting capacity of the WRF model as there is a huge increase in every evaluation indicator. In contrast, the enhancement offered by the modified Kalman filter is considerable. More accurately, the largest improvement exceeded 60% and concerns the Ns index (from -2.2314 to -0.8557), while the minimum improvement concerns the RMSE assessor (from 4.0487 to 3.0586) and is about 24%.

The proposed filter contributes to a greater extent in Do3 (Table 4—Do3). In this case study, the bias and RMSE indices decreased by 63% (changing sign) and 45%, respectively,

while the corresponding values from KF's application deteriorated by 9% and 8%. This denotes that the modified Kalman filter overcomes the drawbacks of classic KFs as it reduces the standard error and additionally constrains the remaining forecast uncertainty, providing stable and secure forecasts.

The time series diagrams (Figures 6 and 7) demonstrate the same results. In both cases, the distribution of the modified filter's predictions is nearly identical to the corresponding distribution of the recorded observations. However, it is worth noting that during the first 11 h (Figure 7) of the forecasting period, the improved predictions from the proposed filter are similar to those produced by the non-linear KF when the WRF forecasts of Do3 are used.

6.2. Evaluation of 10-Meter Wind Speed Forecasts from WRF Do2 and Do3 Domains at AUTH and AXIOS Stations

Regarding the environmental parameter of 10-meter wind speed, the proposed methodology once again improves the simulation's performance. In Do2 at AUTH (Table 5—Do2) the bias index reduced by 77% (changing sign), whereas the corresponding value from the KF's implementation decreased by 57% (changing sign). Furthermore, the RMSE assessor through the modified filter enhanced by 41%, while via the standard Kalman filter, this improvement was 32%.

Even greater is the enhancement offered by the suggested approach in Do3 at AUTH (Table 5—Do3). More specifically, the bias was almost eliminated (from -4.9319 to -0.3595) as the average reduction surpassed 90%, and the RMSE index showed a substantial improvement close to 52%. Similar results can be extracted by using the Kalman filter. More thoroughly, the bias and RMSE indices decreased by 77% and 44%, respectively.

Focusing on the time series diagrams (Figures 8 and 9), it is observed that both strategies boost the forecasting capabilities of the numerical model for the 10-meter wind speed parameter, as also revealed by the analysis of the evaluation indicators; nevertheless, the predictions produced by the suggested filter are once again superior to those obtained from the traditional methodology.

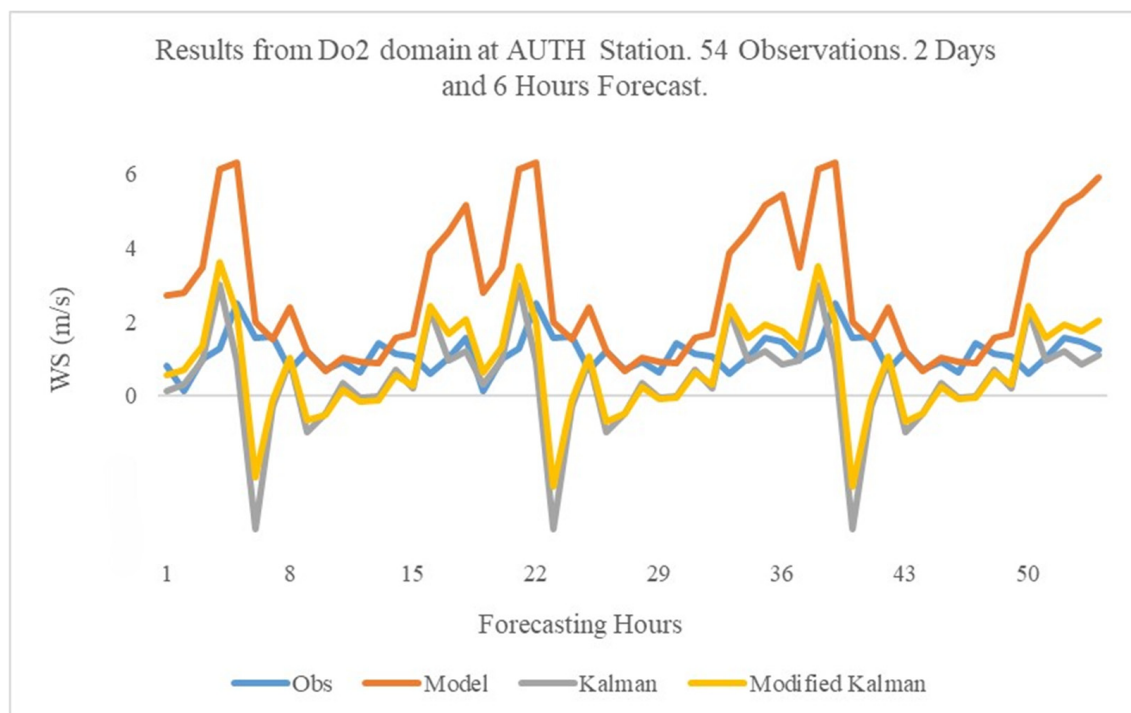


Figure 8. Time series diagram. Results from AUTH station at Do2 domain for 10-meter wind speed.

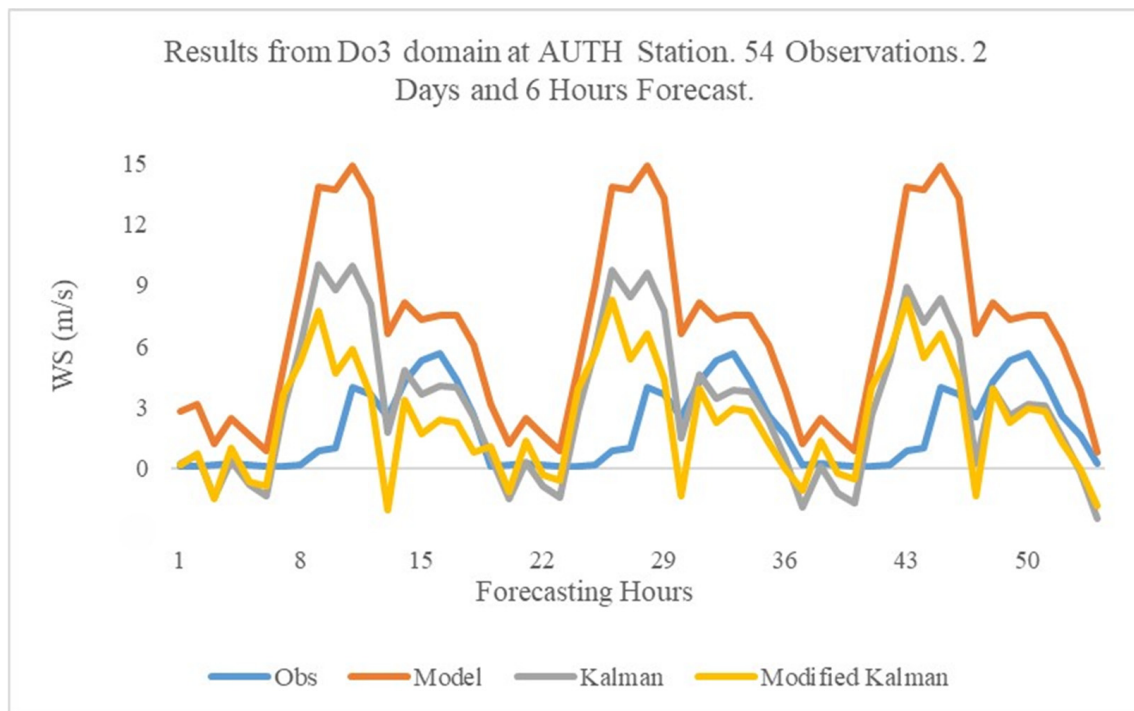


Figure 9. Time series diagram. Results from AUTH station at Do3 domain for 10-meter wind speed.

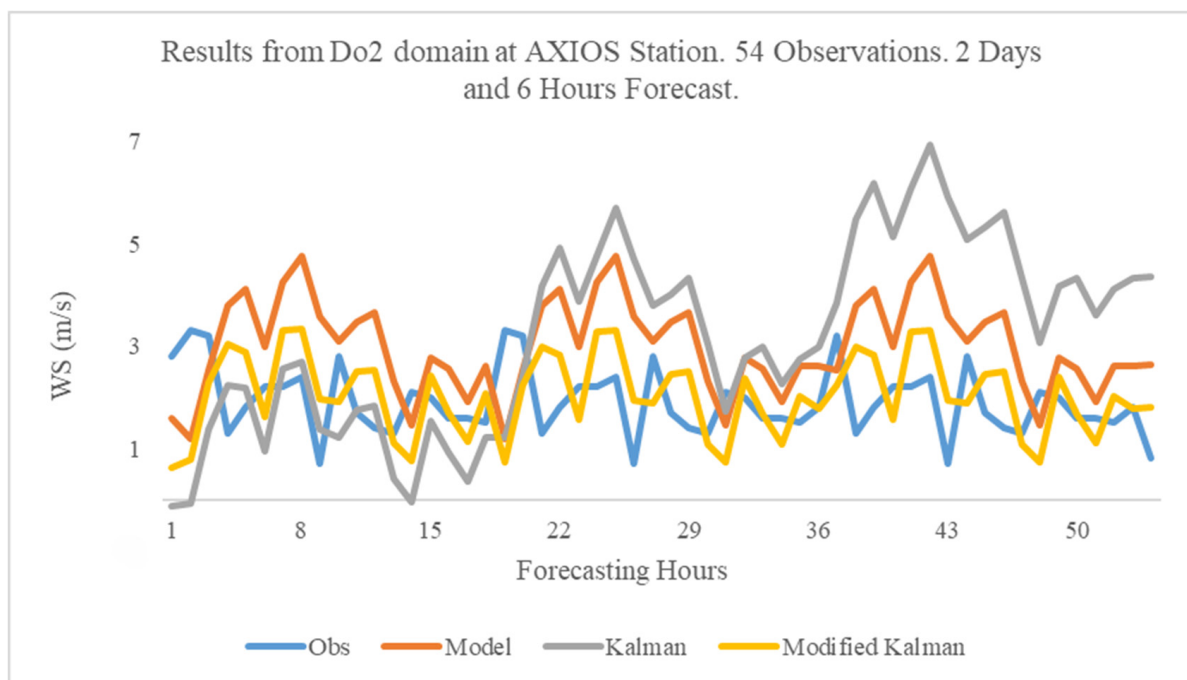


Figure 10. Time series diagram. Results from AXIOS station at Do2 domain for 10-meter wind speed.

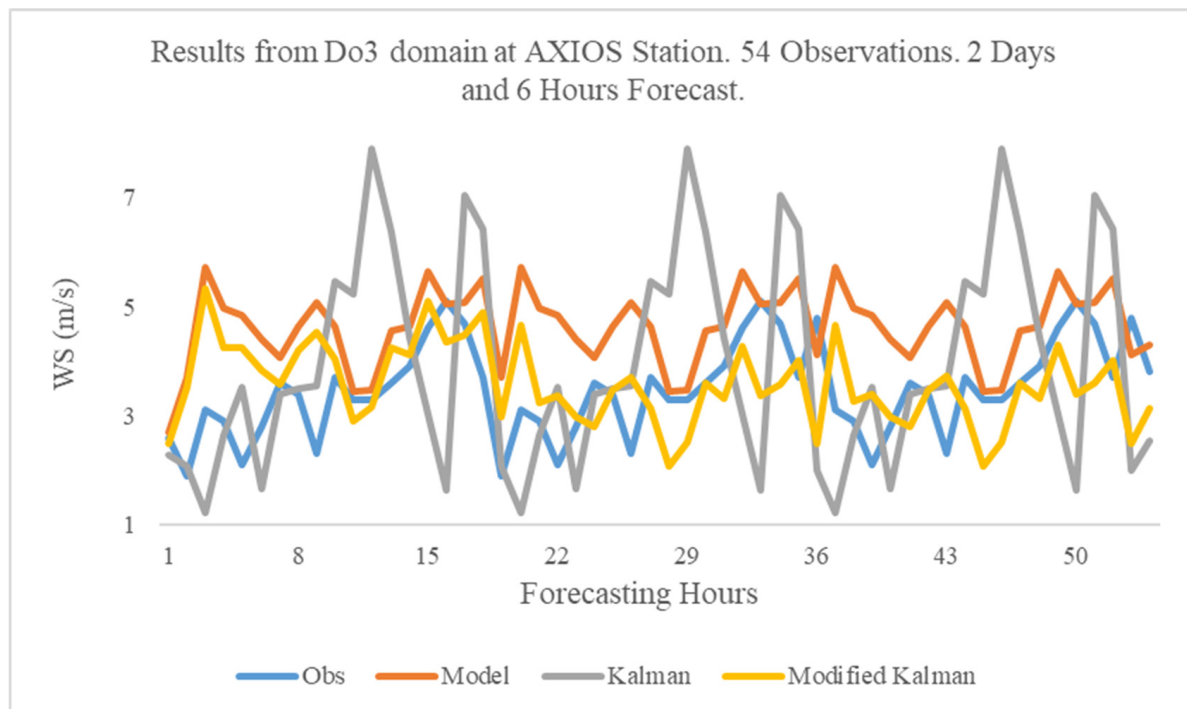


Figure 11. Time series diagram. Results from AXIOS station at Do3 domain for 10-meter wind speed.

The proposed filter based on Radial Basis Function neural networks is the suitable choice for the AXIOS station. According to the derived results (Table 6—Do2), the standard KF fails to boost the forecasting capacity of the NWP model since each evaluation indicator has worsened. The bias and the RMSE assessors increased by 19% and 40%, respectively, while the Ns indicator demonstrated the highest increase. On the other hand, the corresponding values through the modified Kalman filter were reduced by 88%, 36%, and 70%, respectively.

Moving on to the Do3 forecasts at AXIOS, the proposed methodology is still the superior option as it successfully improves the predictions of the WRF model. Here, the bias, the RMSE, and the Ns indices were reduced from -1.1446 to -0.0856 (dropped by 93%), from 1.4911 to 1.0898 (dropped by 27%), and from -2.1064 to -0.2628 (dropped by 87%), respectively (Table 6—Do3). On the contrary, the Kalman filter was once again unable to improve every evaluation indicator as the RMSE assessor deteriorated by 90%.

These conclusions can also be obtained from the time series graphs (Figures 10 and 11). In every case, the modified Kalman filter based on RBF NNs generates more qualitative forecasts that lead to an adequate convergence of the modeled PDFs to the recorded observations. As a result, the integration of Radial Basis Function neural networks into Kalman's process improves significantly the forecasts of the NWP model.

7. Discussion

The obtained results emphasize the limitations of traditional Kalman filters and their non-linear extensions in mitigating systematic and non-systematic forecast errors within complex and dynamic systems like weather prediction. While conventional Kalman filters excel in reducing biases by integrating observations with model forecasts, their reliance on fixed covariance matrices hinders their adaptability to fluctuating forecast uncertainties, often leading to unstable predictions. In contrast, the proposed modified Kalman filter, incorporating RBF topologies, provides a substantial enhancement in forecast accuracy.

Furthermore, the proposed modified Kalman filter successfully reduces forecast uncertainty as the RMSE indicator decreases significantly. The observed reduction in Root

Mean Square Error across multiple meteorological stations highlights further the improved stability and precision of the modified strategy. A lower RMSE signifies that the suggested filtering approach effectively minimizes deviations between predictions and observations, enhancing both short-term responsiveness and long-term trend accuracy.

Another critical aspect that warrants discussion is the occurrence of negative values in the Nash–Sutcliffe efficiency coefficient. The non-linear variation of the Kalman filter, constrained by fixed covariance matrices, produced highly negative N_s values, indicating that it failed to sufficiently reduce forecast errors and, in some cases, even worsened the predictions. Negative N_s values typically arise when a predictive model introduces additional errors rather than mitigating them, resulting in worse performance than simply using the mean of the observed data.

Several factors can contribute to this outcome, including incorrect assumptions about the statistical properties of the errors, suboptimal tuning of the covariance matrices, and an inability to account for dynamically evolving uncertainties in the forecast. Specifically, utilized fixed covariance matrices in the standard Kalman filter can lead to an inaccurate representation of the error structure, preventing the filter from properly weighting new observations and adjusting predictions accordingly; on the contrary, the adaptive nature of the modified Kalman filter enables a more accurate representation of both systematic and non-systematic forecast errors, leading to enhanced stability and accuracy.

Improvements in wind speed and air temperature predictions using the adaptive approach stem from the dynamic updating of covariance matrices through RBF neural networks. For wind speed, the modified Kalman filter accounts for the effects of localized and transient meteorological phenomena, such as boundary layer processes and rapid atmospheric fluctuations, which are difficult to model accurately. For air temperature, the adaptive process reduces errors from unresolved small-scale processes like land–sea interactions, urban heat effects, and abrupt topographical variations. This self-adaptive mechanism ensures more accurate representation of evolving forecast uncertainties, leading to significant reductions in both bias and RMSE compared to traditional Kalman filtering methods.

Overall, these findings demonstrate the ability of the proposed methodology to simultaneously address both types of errors, ultimately enhancing prediction stability, reliability, and robustness in handling forecast uncertainty in numerical weather prediction models. Future research could extend this study by incorporating direct benchmarking with state-of-the-art methods and performing ablation studies to quantify the individual contribution of each methodological component, further solidifying the effectiveness and generalizability of the approach.

8. Conclusions

This study introduces a modified Kalman filter algorithm, innovatively integrating Radial Basis Function neural networks to enhance the predictive performance of the Weather Research and Forecasting model for 10-meter wind speed and 2-meter air temperature forecasts. The proposed approach addresses the critical limitations of standard Kalman filters, particularly their dependence on fixed covariance matrices, which struggle to adapt to the dynamic nature of environmental systems.

The novelty of the proposed method lies in three key aspects: (i) the use of a non-linear Kalman filter with a quadratic polynomial bias correction to effectively capture systematic forecast errors; (ii) the adaptive estimation of covariance matrices, allowing them to evolve constantly based on real-time forecast uncertainty; and (iii) the integration of RBF NNs to model this evolution in a non-linear, data-driven manner, replacing traditional heuristic

adaptation rules. This third component targets the remaining non-systematic part of the forecast error.

The suggested modified filter was assessed through a time-window process involving various stations located in the Thessaloniki region. The extracted results were compared to those obtained by the same non-linear Kalman filter but with stable covariance matrices and revealed the following:

- The recorded standard errors decreased significantly: The bias index for the 2-meter air temperature reduced by 58%, while the corresponding reduction for the 10-meter wind speed was 88%. In contrast, the values from the non-linear Kalman implementation for the 2-meter air temperature increased by 31%, whereas for the 10-meter wind speed, it decreased by 36%;
- The error variability was successfully restrained: The RMSE assessor reduced by 40% for the 2-meter air temperature and 39% for the 10-meter wind speed. On the other hand, the corresponding values from using the non-linear KF were intensified by 50% and 13%, respectively.

Through the above analysis, it is clear that the suggested methodology manages to boost the forecasting capabilities of the numerical weather model WRF regardless of the observation station or the environmental parameter, providing a reliable and stable prediction tool that avoids the limitations of standard Kalman filters.

The authors believe that the proposed modified Kalman filter constitutes a comprehensive computational system for environmental applications. While it has been applied to wind speed and air temperature forecasts, its adaptability offers expansion to other atmospheric parameters derived from the Weather Research and Forecasting model, such as mean sea-level pressure, relative humidity, and specific humidity. Future research could further assess the method's effectiveness in these contexts, examining its ability to reduce systematic biases and mitigate uncertainties associated with cloud microphysics and turbulent interactions in numerical weather prediction models. In this regard, a comparative analysis with other techniques, such as Transformer-based models and Graph Neural Networks, will be conducted to evaluate its relative strengths and limitations.

Author Contributions: Conceptualization, A.D., G.G. and I.T.F.; methodology, A.D., G.G. and I.T.F.; software, A.D. and I.T.F.; validation, A.D., G.G. and I.T.F.; formal analysis, A.D., G.G. and I.T.F.; investigation, A.D., G.G. and I.T.F.; resources, G.G. and I.P.; data curation, A.D., G.G. and I.P.; writing—original draft preparation, A.D. and I.T.F.; writing—review and editing, A.D., G.G., I.P. and I.T.F.; visualization, A.D., G.G. and I.T.F.; supervision, G.G. and I.T.F.; project administration, G.G. and I.T.F. All authors have read and agreed to the published version of the manuscript.

Funding: This work has been funded by the Unit of Economic and Administrative Support (E.L.K.E.) of the University of West Attica.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Due to privacy, the data presented in this study are available on request from the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest.

Appendix A. Time-Window Process—Results from AUTH and AXIOS Stations—Do2 and Do3—Temperature

Table A1. Evaluation indicators for the time-window process. AUTH station—Do2.

Do2–AUTH		Time Window 1	
	Bias	RMSE	Ns
Model	−2.8967	3.1822	−0.6260
Kalman	5.9146	6.1970	−5.1661
Modified Kalman	1.7076	2.3672	0.1002
Do2–AUTH		Time Window 2	
	Bias	RMSE	Ns
Model	−2.8572	3.1474	−0.6865
Kalman	0.4873	1.4922	0.6209
Modified Kalman	0.2972	1.3381	0.6952
Do2–AUTH		Time Window 3	
	Bias	RMSE	Ns
Model	−3.0722	3.2473	−4.3121
Kalman	−3.0130	3.1826	−4.1026
Modified Kalman	0.0450	1.0478	0.4469

Table A2. Evaluation indicators for the time-window process. AXIOS station—Do2 and Do3.

Do2–AXIOS				Do3–AXIOS			
	Bias	Time Window 1 RMSE	Ns		Bias	Time Window 1 RMSE	Ns
Model	−3.9761	4.4915	−2.4151	Model	−4.7617	5.3160	−3.3805
Kalman	9.8326	10.0761	−16.1872	Kalman	3.6442	4.3173	−1.8892
Modified Kalman	2.9382	3.8391	−1.4951	Modified Kalman	2.8887	3.9502	−1.4187
Do2–AXIOS				Do3–AXIOS			
	Bias	Time Window 2 RMSE	Ns		Bias	Time Window 2 RMSE	Ns
Model	−3.5156	3.6059	−2.0476	Model	−4.4522	4.8881	−5.1725
Kalman	8.4526	8.4949	−15.9140	Kalman	5.3557	5.7236	−7.4628
Modified Kalman	2.1363	2.2780	−0.2164	Modified Kalman	0.8675	2.1857	−0.2342
				Do3–AXIOS			
					Bias	Time Window 3 RMSE	Ns
				Model	−4.1878	4.4675	−8.8738
				Kalman	5.6460	5.8563	−15.9664
				Modified Kalman	1.1359	1.9383	−0.8587

Appendix B. Time-Window Process—Results from AUTH and AXIOS Stations—Do2 and Do3—Wind Speed

Table A3. Evaluation indicators of the time-window process. AUTH station—Do2 and Do3.

Do2–AUTH				Do3–AUTH			
	Bias	Time Window 1 RMSE	Ns		Bias	Time Window 1 RMSE	Ns
Model	−1.6000	2.2766	−18.4533	Model	−4.9950	6.3834	−9.1758
Kalman	0.7350	1.6547	−9.2762	Kalman	−1.6060	3.7548	−2.5208
Modified Kalman	0.3880	1.4266	−6.6382	Modified Kalman	−0.1498	2.9985	−1.2452
Do2–AUTH				Do3–AUTH			
	Bias	Time Window 2 RMSE	Ns		Bias	Time Window 2 RMSE	Ns
Model	−1.7133	2.4190	−20.9448	Model	−4.9700	6.3737	−9.6456
Kalman	0.7323	1.6537	−9.2552	Kalman	−1.3590	3.6504	−2.4919
Modified Kalman	0.3934	1.4471	−6.8527	Modified Kalman	−0.5439	3.0889	−1.5003

Table A3. Cont.

Do2–AUTH	Bias	Time Window 3 RMSE	Ns	Do3–AUTH	Bias	Time Window 3 RMSE	Ns
Model	−1.8244	2.5812	−31.4088	Model	−4.8306	6.3336	−9.5930
Kalman	0.7512	1.6535	−12.2994	Kalman	−0.4990	3.3681	−1.9956
Modified Kalman	0.3761	1.4526	−9.2638	Modified Kalman	−0.3849	3.1216	−1.5732

Table A4. Evaluation indicators of the time-window process. AUTH station–Do2 and Do3.

Do2–AXIOS	Bias	Time Window 1 RMSE	Ns	Do3–AXIOS	Bias	Time Window 1 RMSE	Ns
Model	−0.9283	1.6538	−4.8686	Model	−1.1972	1.5045	−2.1293
Kalman	0.7528	1.4332	−3.4070	Kalman	−0.9428	2.8035	−0.2550
Modified Kalman	−0.0061	1.1874	−2.0252	Modified Kalman	−0.7037	1.1510	−0.3316
Do2–AXIOS	Bias	Time Window 2 RMSE	Ns	Do3–AXIOS	Bias	Time Window 2 RMSE	Ns
Model	−1.0411	1.6398	−5.2674	Model	−1.1544	1.5126	−1.8966
Kalman	−1.5269	2.1462	−9.7362	Kalman	−0.7606	2.8392	−0.3155
Modified Kalman	−0.0853	1.0678	−1.6577	Modified Kalman	0.1773	1.0699	−0.2492
Do2–AXIOS	Bias	Time Window 3 RMSE	Ns	Do3–AXIOS	Bias	Time Window 3 RMSE	Ns
Model	−1.2606	1.6216	−5.9406	Model	−1.0822	1.4562	−2.2932
Kalman	−3.0808	3.3043	−27.8161	Kalman	−0.6646	2.8255	−0.3770
Modified Kalman	−0.2881	0.9131	−1.2004	Modified Kalman	0.2697	1.0485	−0.2075

Appendix C. Time-Window Process—Matlab Code—General Instructions

```
% The indication Value depends on the user and the application. Is not the same
% for every variable.
% The variable WH presents the recorded observations, while SWH the forecasts
% from the simulation model
format long e
load('') % Load the available data
% Initial parameters for the Time Window Process.
timewindow= Value;
BiasModel=zeros(1,timewindow+1);nsModel=BiasModel;RmseModel=BiasModel;
BiasRBFNN=BiasModel;RmseRBFNN=BiasModel;nsRBFNN=BiasModel;
% Training and Testing data sets
Training data=Value; Testing data =Value;
FORECASTS=zeros(Testing data,timewindow+1);
TrueObs=FORECASTS; ModelsObs=FORECASTS; Est=zeros(1,Training data);
% Training Parameters for the LM
maxtrain=Value;minNumberOfNeurons=Value;maxNumberOfNeurons=Value;
Aa=minNumberOfNeurons:maxNumberOfNeurons; rr=1;error=Value;ww=1;www=1;
bestNeuron=zeros(1,timewindow+1);
% Find the best topology
trainMSE1=zeros(1,size(minNofN:maxNofN,2));
TotalMSE=zeros(timewindow+1,size(minNofN:maxNofN,2));
% Initial parameters for the Kalman Filter
dimState=Value;dimMes=Value;
F=Value;I=eye(dimState); % Here Value=1
P=Value*eye(dimState,dimState); % Here Value=4
ybest=zeros(dimState,size(minNofN:maxNofN,2));
%Time Window Process
```

```

        for tttt=0:timewindow
            histdataaa=tttt+1:histdata+tttt;
            N=size(data(histdataaa),1);
            %systematic error of the simulation
            obs=WH(histdataaa) '-SWH(histdataaa)';
            % Initial values for the covariance matrices
            R=4;Q=eye(dimState,dimState);
            y=zeros(dimState,N);
            par=SWH(histdataaa)';ESTer=zeros(dimMes,N);
            for ii=minNofN:maxNofN
                l=ii; % Number of neurons
                while rr<=maxtrain
                    y(:,1)=rand(dimState,1);
                    % Kalman filter algorithm. Quadratic variation
                    for k=2:N
                        H=[1 par(:,k-1) par(:,k-1).^2];
                        % estimate state
                        y(:,k)=F*y(:,k-1);
                        % error covariance
                        P=F*P*F'+Q;
                        %Innovation Covariance
                        S=H*P*H'+R;
                        %Kalman gain
                        K=P*H'/S;
                        %Innovation
                        d=obs(:,k)-H*y(:,k);
                        % Update the state vector
                        y(:,k)=y(:,k)+K*(obs(:,k)-H*y(:,k));
                        %Update the R
                        yy=(obs(:,k)-y(:,k))'*(obs(:,k)-y(:,k));% residual
                        y1=R;y2=H*P*H';Inputt=[yy;y2];
                        [MIN,~,~]=Levenberg-Marquardt(Inputt,1);
                        Rnew=abs(y1+MIN);R=Rnew;
                        % New Kalman Gain
                        K=P*H'/(H*P*H'+R);
                        %Update the error covariance
                        P=(I-K*H)*P;
                        %Update the Q
                        Inputtt=[d;K];
                        MIN1=Levenberg-Marquardt(Inputtt,1);
                        Q=abs(Q+MIN1*eye(dimState,dimState));
                        %Update the state
                        y(:,k)=y(:,k)+K*(obs(:,k)-H*y(:,k));
                    end
                    % Corrected data from modified Kalman filter.
                    for i=2:N
                        Est(:,i)=[1 par(:,i-1) par(:,i-1).^2]*y(:,end);
                    end
                    z1=SWH(histdataaa)'+Est;
                    %Calculate the training MSE
                    trainMSE=mean((WH(histdataaa)-z1').^2);
                    if trainMSE<=error
                        error=trainMSE;
                        ybest(:,ww)=y(:,N);
                        trainMSE1(1,ww)=error;
                    end
                    rr=rr+1;
                end
                error=10^10;ww=ww+1;rr=1;
            end
            %Find the abs(error)
            ff1=find(trainMSE1==min(trainMSE1));
            if size(ff1,2)==1
                ff=ff1;
            end
        end
    end
end

```

```

        y(:,N)=ybest(:,ff);bestNeuron(1,www)=Aa(ff);
        WW=WH(histdataa(end)+1:histdataa(end)+Testing data);nn=size(WW,1);
        par=SWH(histdataa(end)+1:histdataa(end)+Testing data)';
        par2=SWH(histdataa(end):histdataa(end)+Testing data-1)';
        ESTer1=zeros(1,nn);z2=zeros(1,nn);
        for ww=1:nn
            H=[1 par2(:,ww) par2(:,ww).^2];
            ESTer1(:,ww)=H*y(:,N);
            z2(:,ww)=par(:,ww)+ESTer1(:,ww);
        end
        % Evaluation of the method
        t=1:nn;
        BiasModel(1,tttt+1)=sum(WW(t)-par(t)')/size(t,2);
        RmseModel(1,tttt+1)=sqrt(mean((WW(t)-par(t)').^2));
        BiasRBFNN(1,tttt+1)=sum(WW(t)-z2(t)')/size(t,2);
        RmseRBFNN(1,tttt+1)=sqrt(mean((WW(t)-z2(t)').^2));
        av=sum(WW(t))/size(t,2);
        nsModel(1,tttt+1)=1-(sum((WW(t)-par(t)').^2)/sum((WW(t)-av).^2));
        nsRBFNN(1,tttt+1)=1-(sum((WW(t)-z2(t)').^2)/sum((WW(t)-av).^2));
        TrueObs(:,tttt+1)=WW(t);ModelsObs(:,tttt+1)=par(t);
        FORECASTS(:,tttt+1)=z2(t)';TotalMSE(tttt+1,:)=trainMSE1;
        www=www+1;
    end
    % Store the results
    tt=1;qq=1;TimeSeries=zeros((timewindow+1)*forecast,3);
    for tttt=0:timewindow
        TimeSeries(qq:tt*forecast,1:3)=[TrueObs(:,tttt+1) ModelsObs(:,tttt+1)
            FORECASTS(:,tttt+1)];
        qq=tt*forecast+1;
        tt=tt+1;
    end
    EvalInd=[BiasModel;RmseModel;nsModel;BiasRBFNN;RmseRBFNN;nsRBFNN]';

```

References

1. Nordström, J. Uncertain Data in Initial Boundary Value Problems: Impact on Short and Long Time Predictions. *arXiv* **2023**, arXiv:2306.16486. [\[CrossRef\]](#)
2. Dong, R.; Leng, H.; Zhao, C.; Song, J.; Zhao, J.; Cao, X. A Hybrid Data Assimilation System Based on Machine Learning. *Front. Earth Sci.* **2023**, *10*, 1012165. [\[CrossRef\]](#)
3. Rojas-Campos, A.; Wittenbrink, M.; Nieters, P.; Schaffernicht, E.J.; Keller, J.D.; Pipa, G. Postprocessing of NWP Precipitation Forecasts Using Deep Learning. *Weather Forecast.* **2023**, *38*, 487–497. [\[CrossRef\]](#)
4. Krasnopolsky, V. *Review: Using Machine Learning for Data Assimilation, Model Physics, and Post-Processing Model Outputs*; National Centers for Environmental Prediction (U.S.): College Park, MD, USA, 2023. [\[CrossRef\]](#)
5. Ladino-Moreno, E.O.; García-Ubaque, C.A. Leak Detection in Urban Hydraulic Systems Using the K-BiLSTM-Monte Carlo Dropout Model. *Civ. Eng. J.* **2024**, *10*, 2066–2087. [\[CrossRef\]](#)
6. Yanfatriani, E.; Marzuki, M.; Vonnisa, M.; Razi, P.; Hapsoro, C.A.; Ramadhan, R.; Yusnaini, H. Extreme Rainfall Trends and Hydrometeorological Disasters in Tropical Regions: Implications for Climate Resilience. *Emerg. Sci. J.* **2024**, *8*, 1860–1874. [\[CrossRef\]](#)
7. Kariniotakis, G.N.; Pinson, P. Evaluation of the MORE-CARE Wind Power Prediction Platform. Performance of the Fuzzy Logic Based Models. In Proceedings of the EWEC 2003—European Wind Energy Conference, Madrid, Spain, 16–19 June 2003.
8. Kariniotakis, G.; Martí, I.; Casas, D.; Pinson, P.; Nielsen, T.S.; Madsen, H.; Giebel, G.; Usaola, J.; Sanchez, I. What Performance Can Be Expected by Short-Term Wind Power Prediction Models Depending on Site Characteristics? In Proceedings of the EWC 2004 Conference, Tokyo, Japan, 2–4 August 2004; pp. 22–25.
9. Vanem, E. Long-term time-dependent stochastic modelling of extreme waves. *Stoch. Environ. Res. Risk Assess.* **2011**, *25*, 185–209. [\[CrossRef\]](#)
10. Resconi, G. Geometry of risk analysis (morphogenetic system). *Stoch. Environ. Res. Risk Assess.* **2009**, *23*, 425–432. [\[CrossRef\]](#)
11. Delle Monache, L.; Eckel, F.A.; Rife, D.L.; Nagarajan, B.; Searight, K. Probabilistic Weather Prediction with an Analog Ensemble. *Mon. Weather Rev.* **2013**, *141*, 3498–3516. [\[CrossRef\]](#)
12. Kalnay, E. *Atmospheric Modeling, Data Assimilation and Predictability*, 1st ed.; Cambridge University Press: Cambridge, UK, 2002. [\[CrossRef\]](#)

13. Jetli, H.V.; Salicone, S. A Possibilistic Kalman Filter for the Reduction of the Final Measurement Uncertainty, in Presence of Unknown Systematic Errors. *Metrology* **2021**, *1*, 39–51. [\[CrossRef\]](#)
14. Lio, W.H.; Meng, F. Kalman-Based Interacting Multiple-Model Wind Speed Estimator for Wind Turbines. *IFAC-Pap.* **2020**, *53*, 12644–12649. [\[CrossRef\]](#)
15. Kolei, S.E. Propagation of Initial Errors on the Parameters for Linear and Gaussian State Space Models. *arXiv* **2013**, arXiv:1303.3518. [\[CrossRef\]](#)
16. Davis, C.A.; Jones, S.C.; Riemer, M. Hurricane Vortex Dynamics during Atlantic Extratropical Transition. *J. Atmos. Sci.* **2008**, *65*, 714–736. [\[CrossRef\]](#)
17. Lu, Y.; Deng, Y. Initial Transient Response of an Intensifying Baroclinic Wave to Increases in Cloud Droplet Number Concentration. *J. Clim.* **2015**, *28*, 9669–9677. [\[CrossRef\]](#)
18. Shi, J.J.; Tao, W.-K.; Matsui, T.; Cifelli, R.; Hou, A.; Lang, S.; Tokay, A.; Wang, N.-Y.; Peters-Lidard, C.; Skofronick-Jackson, G.; et al. WRF Simulations of the 20–22 January 2007 Snow Events over Eastern Canada: Comparison with In Situ and Satellite Observations. *J. Appl. Meteorol. Climatol.* **2010**, *49*, 2246–2266. [\[CrossRef\]](#)
19. Skamarock, W.C.; Klemp, J.B.; Dudhia, J.; Gill, D.O.; Liu, Z.; Berner, J.; Wang, W.; Powers, J.G.; Duda, M.G.; Barker, D.M.; et al. *A Description of the Advanced Research WRF Model Version 4*; UCAR/NCAR: Boulder, CO, USA, 2019. [\[CrossRef\]](#)
20. Makris, C.V.; Krestenitis, Y.N.; Androulidakis, Y.; Baltikas, V. Storm Surge Simulations in the Greek Coastal Zone under Climate Change. In Proceedings of the 3rd Joint Conference: 13th Hellenic Hydrotechnical Association (PSO), 9th Hellenic Committee for Water Resources Management (EEDYP), 1st Hellenic Water Association. «INTEGRATED WATER RESOURCES MANAGEMENT IN THE NEW ERA», Athens, Greece, 12 December 2015.
21. Androulidakis, Y.; Makris, C.; Kolovoyiannis, V.; Krestenitis, Y.; Baltikas, V.; Mallios, Z.; Pytharoulis, I.; Topouzelis, K.; Spondylidis, S.; Tegoulis, I.; et al. Hydrography of Northern Thermaikos Gulf based on an integrated observational-modeling approach. *Cont. Shelf Res.* **2023**, *269*, 105141. [\[CrossRef\]](#)
22. Krestenitis, Y.; Pytharoulis, I.; Karacostas, T.S.; Androulidakis, Y.; Makris, C.; Kombiadou, K.; Tegoulis, I.; Baltikas, V.; Kotsopoulos, S.; Kartsios, S. Severe Weather Events and Sea Level Variability Over the Mediterranean Sea: The WaveForUs Operational Platform. In *Perspectives on Atmospheric Sciences*; Karacostas, T., Bais, A., Nastos, P.T., Eds.; Springer International Publishing: Berlin/Heidelberg, Germany, 2017; pp. 63–68. [\[CrossRef\]](#)
23. Pytharoulis, I.; Karacostas, T.; Christodoulou, M.; Matsangouras, I. The July 10, 2019 Catastrophic Supercell over Northern Greece. Part II: Numerical Modelling. In Proceedings of the 15th International Conference on Meteorology, Climatology and Atmospheric Physics (COMCAP2021), Ioannina, Greece, 26–29 September 2021; pp. 885–890.
24. Androulidakis, Y.; Makris, C.; Mallios, Z.; Pytharoulis, I.; Baltikas, V.; Krestenitis, Y. Storm surges and coastal inundation during extreme events in the Mediterranean Sea: The IANOS Medican. *Nat. Hazards* **2023**, *117*, 939–978. [\[CrossRef\]](#)
25. Eta-12 TPB. Available online: <https://www.emc.ncep.noaa.gov/users/mesoimplydocs/mesoimpl/eta12tpb/> (accessed on 19 June 2024).
26. Janjić, Z.I. The Step-Mountain Eta Coordinate Model: Further Developments of the Convection, Viscous Sublayer, and Turbulence Closure Schemes. *Mon. Weather Rev.* **1994**, *122*, 927–945. [\[CrossRef\]](#)
27. Janjić, Z.I. Comments on “Development and Evaluation of a Convection Scheme for Use in Climate Models”. *J. Atmos. Sci.* **2000**, *57*, 3686. [\[CrossRef\]](#)
28. Mellor, G.L.; Yamada, T. Development of a turbulence closure model for geophysical fluid problems. *Rev. Geophys.* **1982**, *20*, 851–875. [\[CrossRef\]](#)
29. Janjic, Z.I. *Nonsingular Implementation of the Mellor–Yamada Level 2.5 Scheme in the NCEP Meso Model*; NCEP Office Note, No. 437; National Centers for Environmental Prediction (U.S.): College Park, MD, USA, 2002; 61p.
30. Janjic, Z.I. The surface layer in the NCEP Eta Model. In Proceedings of the 11th Conference on Numerical Weather Prediction, Norfolk, VA, USA, 19–23 August 1996; American Meteorological Society: Norfolk, VA, USA, 1996; pp. 354–355.
31. Iacono, M.J.; Delamere, J.S.; Mlawer, E.J.; Shephard, M.W.; Clough, S.A.; Collins, W.D. Radiative forcing by long-lived greenhouse gases: Calculations with the AER radiative transfer models. *J. Geophys. Res. Atmos.* **2008**, *113*, D13103. [\[CrossRef\]](#)
32. Chen, F.; Dudhia, J. Coupling an Advanced Land Surface–Hydrology Model with the Penn State–NCAR MM5 Modeling System. Part I: Model Implementation and Sensitivity. *Mon. Weather Rev.* **2001**, *129*, 569–585. [\[CrossRef\]](#)
33. Brockwell, P.; Davis, R. *Time Series: Theory and Methods*; Springer: New York, NY, USA, 1987.
34. Homleid, M. Diurnal Corrections of Short-Term Surface Temperature Forecasts Using the Kalman Filter. *Weather Forecast.* **1995**, *10*, 689–707. [\[CrossRef\]](#)
35. Kalman, R.E.; Bucy, R.S. New Results in Linear Filtering and Prediction Theory. *J. Basic Eng.* **1961**, *83*, 95–108. [\[CrossRef\]](#)
36. Delle Monache, L.; Wilczak, J.; McKeen, S.; Grell, G.; Pagowski, M.; Peckham, S.; Stull, R.; McHenry, J.; McQueen, J. A Kalman-Filter Bias Correction Method Applied to Deterministic, Ensemble Averaged and Probabilistic Forecasts of Surface Ozone. *Tellus B Chem. Phys. Meteorol.* **2008**, *60*, 238–249. [\[CrossRef\]](#)

37. Bogdanovs, N.; Belinskis, R.; Bistrovs, V.; Petersons, E.; Ipatovs, A. Forecasting Algorithm Based on Temperature Error Prediction Using Kalman Filter for Management System Development. *Latv. J. Phys. Tech. Sci.* **2021**, *58*, 38–49. [[CrossRef](#)]
38. Perera, D.; Skeie, N.-O. Comparison of Space Heating Energy Consumption of Residential Buildings Based on Traditional and Model-Based Techniques. *Buildings* **2017**, *7*, 27. [[CrossRef](#)]
39. Libonati, R.; Trigo, I.; DaCamara, C.C. Correction of 2 m-temperature forecasts using Kalman Filtering technique. *Atmos. Res.* **2008**, *87*, 183–197. [[CrossRef](#)]
40. Ullah, I.; Fayaz, M.; Kim, D. Improving Accuracy of the Kalman Filter Algorithm in Dynamic Conditions Using ANN-Based Learning Module. *Symmetry* **2019**, *11*, 94. [[CrossRef](#)]
41. Akhlaghi, S.; Zhou, N.; Huang, Z. Adaptive adjustment of noise covariance in Kalman filter for dynamic state estimation. In Proceedings of the IEEE Power & Energy Society General Meeting, Chicago, IL, USA, 16–20 July 2017; pp. 1–5. [[CrossRef](#)]
42. Donas, A.; Galanis, G.; Pytharoulis, I.; Famelis, I. A Hybrid Extended Kalman Filter Based on Parametrized ANNs for the Improvement of the Forecasts of Numerical Weather and Wave Prediction Models. *Atmosphere* **2024**, *15*, 828. [[CrossRef](#)]
43. Dey, P.; Gopal, M.; Pradhan, P.; Pal, T. On robustness of radial basis function network with input perturbation. *Neural Comput. Appl.* **2019**, *31*, 523–537. [[CrossRef](#)]
44. Que, Q.; Belkin, M. Back to the Future: Radial Basis Function Network Revisited. *IEEE Trans. Pattern Anal. Mach. Intell.* **2020**, *42*, 1856–1867. [[CrossRef](#)] [[PubMed](#)]
45. Teng, P. Machine-learning quantum mechanics: Solving quantum mechanics problems using radial basis function networks. *Phys. Rev. E* **2018**, *98*, 033305. [[CrossRef](#)]
46. Alexandridis, A.; Chondrodima, E.; Sarimveis, H. Radial Basis Function Network Training Using a Nonsymmetric Partition of the Input Space and Particle Swarm Optimization. *IEEE Trans. Neural Netw. Learn. Syst.* **2013**, *24*, 219–230. [[CrossRef](#)] [[PubMed](#)]
47. Gyamfi, K.S.; Brusey, J.; Gaura, E. Differential radial basis function network for sequence modelling. *Expert Syst. Appl.* **2022**, *189*, 115982. [[CrossRef](#)]
48. Sarimveis, H.; Alexandridis, A.; Mazarakis, S.; Bafas, G. A New Algorithm for developing Dynamic Radial Basis Function Neural Network Models based on Genetic Algorithms. In *Computer Aided Chemical Engineering*; Elsevier: Amsterdam, The Netherlands, 2002; Volume 10, pp. 949–954. [[CrossRef](#)]
49. Hefny, H.A.; Bahnasawi, A.A.; Abdel Wahab, A.H.; Shaheen, S.I. Logical radial basis function networks a hybrid intelligent model for function approximation. *Adv. Eng. Softw.* **1999**, *30*, 407–417. [[CrossRef](#)]
50. Peng, H.; Ozaki, T.; Haggan-Ozaki, V.; Toyoda, Y. A parameter optimization method for radial basis function type models. *IEEE Trans. Neural Netw.* **2003**, *14*, 432–438. [[CrossRef](#)]
51. Yu, H.; Wilamowski, B. Levenberg–Marquardt Training. In *Intelligent Systems*; Irwin, J., Ed.; CRC Press: Boca Raton, FL, USA, 2011; pp. 1–16. [[CrossRef](#)]
52. Ren, X.; Li, X.; Ren, K.; Song, J.; Xu, Z.; Deng, K.; Wang, X. Deep Learning-Based Weather Prediction: A Survey. *Big Data Res.* **2021**, *23*, 100178. [[CrossRef](#)]
53. Schultz, M.G.; Betancourt, C.; Gong, B.; Kleinert, F.; Langguth, M.; Leufen, L.H.; Mozaffari, A.; Stadler, S. Can Deep Learning Beat Numerical Weather Prediction? *Phil. Trans. R. Soc. A* **2021**, *379*, 20200097. [[CrossRef](#)]
54. Bochenek, B.; Ustrnul, Z. Machine Learning in Weather Prediction and Climate Analyses—Applications and Perspectives. *Atmosphere* **2022**, *13*, 180. [[CrossRef](#)]

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.