remote sensing

MDPI

Communication

An Improved Hybrid Beamforming Algorithm for Fast Target Tracking in Satellite and V2X Communication

Aral Ertug Zorkun *, Miguel A. Salas-Natera and Ramón Martínez Rodríguez-Osorio

Centro de Procesamiento de Información y Telecomunicaciones, Escuela Técnica Superior de Ingenieros de Telecomunicación, Universidad Politécnica de Madrid, 28040 Madrid, Spain; miguel.salas@upm.es (M.A.S.-N.); ramon.martinez@upm.es (R.M.R.-O.)

* Correspondence: ar.zorkun@upm.es

Abstract: Autonomous remote sensing systems establish communication links between nodes. Ensuring coverage and seamless communication in highly dense environments is not a trivial task as localization, separation, and tracking of targets, as well as interference suppression, are challenging. Therefore, smart antenna systems fulfill these requirements by employing beamforming algorithms and are considered a key technology for autonomous remote sensing applications. Among many beamforming algorithms, the recursive least square (RLS) algorithm has proven superior convergence and convergence rate performances. However, the tracking performance of RLS degrades in the case of dynamic targets. The forgetting factor in RLS needs to be updated constantly for fast target tracking. Additionally, multiple beamforming algorithms can be combined to increase tracking performance. An improved hybrid constant modulus RLS beamforming algorithm with an adaptive forgetting factor and a variable regularization factor is proposed. The forgetting factor is updated using the low-complexity yet robust adaptive moment estimation method (ADAM). The sliding-window technique is applied to the proposed algorithm to mitigate the steady-state noise. The proposed algorithm is compared with existing RLS-based algorithms in terms of convergence, convergence rate, and computational complexity. Based on the results, the proposed algorithm has at least 10 times better convergence (accuracy) and a convergence rate two times faster than the compared RLS-based algorithms.

Keywords: adaptive beamforming; hybrid beamforming algorithm; RLS; smart antenna system; target tracking

check for updates

Citation: Zorkun, A.E.; Salas-Natera, M.A.; Rodríguez-Osorio, R.M. An Improved Hybrid Beamforming Algorithm for Fast Target Tracking in Satellite and V2X Communication. *Remote Sens.* **2024**, *16*, 13. https://doi.org/10.3390/rs16010013

Academic Editors: Teresa Pamuła, Wiesław Pamuła, Zhenwei Shi and Andrzej Stateczny

Received: 3 October 2023
Revised: 1 December 2023
Accepted: 15 December 2023
Published: 19 December 2023

Copyright: © 2023 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

Autonomous remote sensing systems are finding their place in an increasing number of applications with a high number of users today, thanks to developing technology. Autonomous driving and advanced driver assistance systems [1,2], intelligent traffic control systems [3], accurate mapping with unmanned aerial vehicles (UAV) [4], and non-terrestrial networks (NTN) [5] can be given as examples of fifth-generation (5G) remote sensing applications. In Figure 1, the NTN and 5G communication scenario in a highly dense environment is given. In highly dense communication environments, where there are many vehicles, estimating the directions of the vehicles with high accuracy is important for traffic control. Therefore, remote sensing systems are required to suppress radio frequency (RF) interference signals. This enables the system to distinguish and track desired targets by processing received signals and ensures communication by transmitting signals at high speed. In such systems, seamless communication under rough environmental factors is required. These demands can only be fulfilled by employing smart antennas systems (SASs).

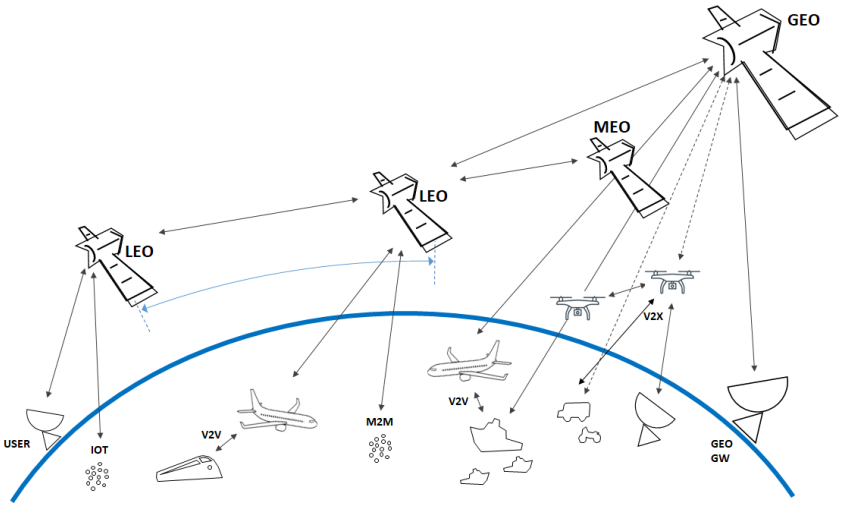


Figure 1. NTN and 5G communication scenario.

SASs are still considered one of the main technologies for modern wireless communication systems, optimizing coverage and mobility. The SAS consists of digital phased arrays to perform beamforming in order to accomplish challenging tasks such as interference suppression and fast tracking [6–8]. A large number of antenna elements and low-complexity adaptive beamforming algorithms fulfill these tasks [6].

In a SAS, adaptive beamforming algorithms generate weights for each antenna element in the arrays to direct the array radiation pattern to desired directions [6,9]. By applying beamforming, the output signals from an antenna array are summed and the beam is directed precisely to the target. In the meantime, deep nulls are directed to undesired signal directions, and thus, interference suppression is performed. The performance of the systems highly depends on the beamforming and calibration algorithms used [10]. Moreover, fast target tracking is a highly challenging task for SASs. Most likely, optimum results may not be achieved by using only one type of algorithm. Therefore, hybrid adaptive algorithms are proposed to achieve a higher convergence rate, better interference suppression, low steady-state noise, and fast target tracking. However, the performance metrics of algorithms depend not only on the adaptability of the algorithm but also on the adaptive parameters in the algorithm.

In this study, we propose a hybrid adaptive beamforming algorithm based on the constant modulus algorithm (CMA) and the recursive least squares (RLS) algorithm. The CMA has the ability to track without any training sequence, is simple, and has robust convergence [11]. However, it has a slower convergence rate, and its steady-state error is large. In addition, blind algorithms including CMA are not capable of suppressing interference. On the other hand, the RLS algorithm has a faster convergence rate and is more robust at steady state [12]. Unlike other algorithms, RLS has superior convergence performance, as past examples are taken into account when calculating the weights [13]. The performance of the RLS algorithm depends on two parameters: the forgetting factor and the regularization factor. In the case of fast target tracking, maintaining a fixed and high value for the forgetting factor makes the past samples more effective in calculating the weights, and this causes convergence rate degradation [12,14]. Conversely, convergence also degrades if the forgetting factor is fixed and has a low value since the error contribution becomes more pronounced. In addition, the regularization factor reduces variance without causing important data loss. However, if the regularization parameter is fixed, after a certain number of samples, important data loss exists, and convergence degradation occurs [15]. For the proposed algorithm, a low-complexity adaptive moment estimation method (ADAM)-based adaptive forgetting factor and non-closed form variable regularization factors are adopted in the RLS algorithm in order to achieve faster tracking

ability and better convergence. The sliding window technique is also implemented to reduce steady state noise [16].

The proposed algorithm is compared with revised state-of-the-art algorithms under various performance metrics in two different use cases: (a) Non-Terrestrial-Network (NTN) connecting a ground station of a Low-Earth-Orbit satellite (LEO) and (b) Vehicular-to-Thing communication (V2X). Both scenarios can be adapted to provide position data for ships and vessels in the port of Barcelona for docking traffic management. Similarly, they can be adapted to provide location data for high-speed trains at Madrid Atocha railway station for the control of arrival and departure traffic. Figures 2 and 3 illustrate the brief vehicle traffic in the port of Barcelona and at Madrid Atocha railway station, respectively. First, the performance characteristics of the proposed methods were investigated in single-user scenarios and then, the performance of the proposed algorithm was analyzed in the multi-user case [1] since in highly dense environments there might be tens of users.

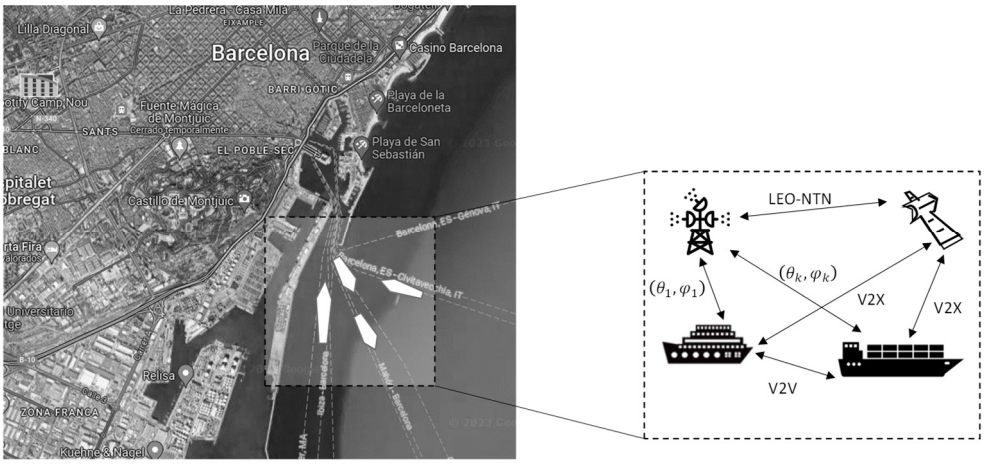


Figure 2. Communication links between maritime radar tower, satellite, and marine vessels for docking traffic management in the port of Barcelona.

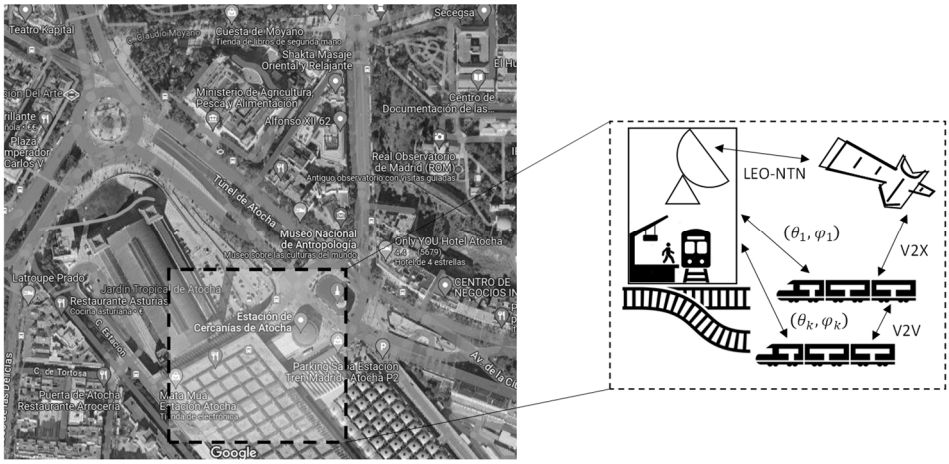


Figure 3. Communication links between the railroad controller, satellite, and high-speed arriving trains for the control of arrival and departure traffic at Madrid Atocha railway station.

The remainder of this paper is organized as follows: In Section 2, the system model and a brief explanation of beamforming algorithms are provided. In Section 3, the proposed algorithm is explained. In Section 4, simulations and discussions are presented. Finally, Section 5 concludes the paper.

Notations: Upper-case and lower-case boldface letters denote the matrices and vectors, respectively. Upper-case lower-case lightface letters denote scalars. The symbols $(.)^T$, $(.)^H$, $(.)^{-1}$, $\mathrm{Re}\{.\}$, $\mathrm{tr}[.]$, $|.|$, $\|.\|^2$ represent the transpose, conjugate transpose (Hermitian),

inversion, real part selection operator, matrix trace operator, absolute value notation, and norm-squared notation, respectively.

2. Background

2.1. The System Model

This section explains the smart antenna system model that is used in this study. We consider a receiver structure consisting of N antennas serving K different signals. In our model, each antenna element receives k different user signals as $[s_1, s_2, \ldots, s_k]$ and $K-k$ different interference signals as $[s_{k+1}, s_{k+2}, \ldots, s_K]$ at the same time. The beamforming is applied in the receiver part. The user signals are digitally modulated and multiplied by the same radio frequency (RF) carrier. In Figure 4, a simple illustration of a smart antenna system is given.

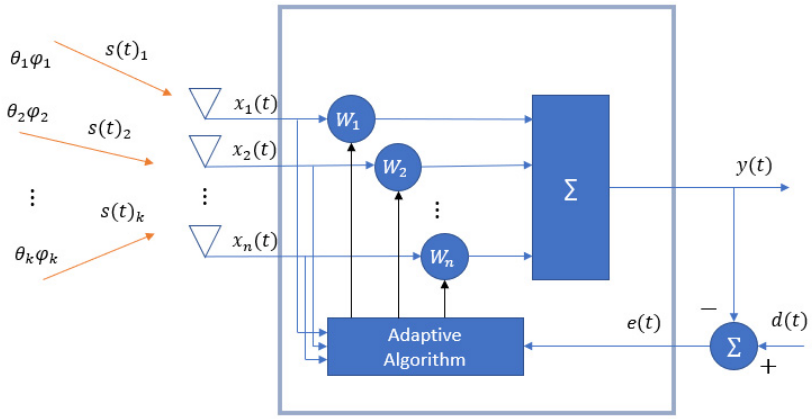


Figure 4. Smart antenna system.

In this study, users are considered as moving signal sources and interference is considered as static signal sources. The displacement of a moving source, with different angular speeds in both elevation and azimuth, can be written as [17]:

$$\theta_k(t) = \theta_{0,\mathrm{k}} + c_k(t) \tag{1}$$

$$\varphi_k(t) = \varphi_{0,\mathrm{k}} + r_k(t) \tag{2}$$

where $\theta_k(t) \in \left[0, \frac{\pi}{2}\right)$ is the elevation angle of the source at time t, $\varphi_k(t) \in [0, 2\pi)$ is the azimuth angle of the source at time t, $\theta_{0,\mathrm{k}}$ (rad) is the initial elevation angle of the source, and $\varphi_{0,\mathrm{k}}$ (rad) is the initial azimuth angle of the k-th source. Assuming the source has a linear motion model, then $c_k(t) = \omega_{1,\mathrm{k}} t$ and $r_k(t) = \omega_{2,\mathrm{k}} t$, where $\omega_{1,\mathrm{k}}$ (rad/s) and $\omega_{2,\mathrm{k}}$ (rad/s) are angular velocities in elevation and azimuth angles of the k-th source, respectively. The system involves a uniform rectangular array (URA) of N elements receiving K signals in total from both users and interference. We assume that the desired and interference signals are multipath signals with L propagation paths. For brevity, L is the same for all signals. The received signal at the n-th element with Cartesian coordinates $(\hat{x}_n, \hat{y}_n)$ at a time t can be formulated as:

$$u_n(t) = \sum_{k=1}^{K} \sum_{l=1}^{L} \alpha_{k,l}(t) s_k(t) e^{-j\frac{2\pi f}{c} \hat{r}_{k,l} \bar{r}_n} + z_n(t) \tag{3}$$

where $\alpha_{k,l}(t) \in \mathbb{C}$ is the complex coefficient (amplitude) of the l-th path of the signal. The complex coefficients follow i.i.d. normal distribution $\mathcal{CN}(0,1)$, the phase of the n-th element is $\hat{r}_{k,l}\bar{r}_n = (\hat{x}_n \sin\theta_{k,l} \cos\varphi_{k,l} + \hat{y}_n \sin\theta_{k,l} \sin\varphi_{k,l})$, c is the speed of light, f is the operating frequency, and $z_n(t)$ is the additive white Gaussian noise.

The impinging signal s_k in time t can be formulated as follows:

$$s_k(t) = a_k(t) b_k(t) \cos(2\pi f t) \tag{4}$$

where $b_k(t) \in \{-1, +1\}$ is the BPSK modulated signal and $a_k(t) \in \mathbb{C}$ is the complex amplitude of the carrier signal.

Then, the output signal $y(t)$ can be expressed as:

$$y(t) = \sum_{n=1}^{N} w_n^H(t) u_n(t) \tag{5}$$

where $w_n(t) \in \mathbb{C}$ is the beamformer weighting coefficients at time t.

2.2. Constant Modulus Algorithm

The CMA exploits the constant modulus (CM) characteristic of the signal. The CMA seeks signals with constant magnitude and only detects the signal with the greatest power. The CMA is given by the equations below [18]:

$$y(i) = \boldsymbol{w}^H(i-1)\boldsymbol{u}(i) \tag{6}$$

$$e(i) = \frac{y(i)}{|y(i)|} - y(i) \tag{7}$$

$$\boldsymbol{w}(i) = \boldsymbol{w}(i-1) + \frac{\mu}{\|\boldsymbol{u}(i)\|^2}\boldsymbol{u}(i)e*(i) \tag{8}$$

where i denotes the iteration, μ is the step-size, and $e(i)$ is the error at i-th iteration.

2.3. Recursive Least Squares Algorithm

The RLS algorithm updates the weight vectors by taking into account the sum of the past least square errors. The RLS algorithm has high convergence rate since past samples effects are stored. The forgetting factor λ and the regularization factor Δ are the key parameters for the RLS algorithm. The RLS algorithm is given by the equations below [18]:

$$\mathbf{P}(0) = \Delta^{-1}\begin{bmatrix} 1 & \cdots & 0 \\ 0 & \ddots & 0 \\ 0 & 0 & 1 \end{bmatrix}_{(N)\times(N)} \qquad \mathbf{w}(0) = 0 \tag{9}$$

$$\boldsymbol{g}(i) = \frac{\lambda^{-1}\boldsymbol{P}(i-1)\boldsymbol{u}(i)}{1 + \lambda^{-1}\boldsymbol{u}^H(i)\boldsymbol{P}(i-1)\boldsymbol{u}(i)} \tag{10}$$

$$e(i) = d(i) - \boldsymbol{w}^H(i-1)\boldsymbol{u}(i) \tag{11}$$

$$\boldsymbol{w}(i) = \boldsymbol{w}(i-1) + \boldsymbol{g}(i)e*(i) \tag{12}$$

$$\mathbf{P}(i) = \lambda^{-1}\Big(\mathbf{P}(i-1) - \boldsymbol{g}(i)\boldsymbol{u}^H(i)\boldsymbol{P}(i-1)\Big) \tag{13}$$

where N is the number of receiving antennas, $\mathbf{P}(i)$ is autocorrelation matrix of received signal samples, and $\boldsymbol{g}(i)$ is the Kalman gain at i-th iteration.

2.4. Sliding Window

The sliding window technique is applied to adaptive beamforming algorithms in two steps. The first step is the update part and the second step is the downdate part. In the update step, a new input sample $\boldsymbol{u}(i+1)$ is added, using $Z+1$ samples from $\boldsymbol{u}(i-Z+1)$ to $\boldsymbol{u}(i+1)$, where Z is the window size. In the downdate step, $\boldsymbol{u}(i-Z+1)$ is discarded to conserve the Z subsequent samples from $\boldsymbol{u}(i-Z+2)$ to $\boldsymbol{u}(i+1)$ [13]. As a simple example, the sliding window technique for CMA is given by the equations below:

$$y_{ud}(i) = \boldsymbol{w}_{dd}^H(i-1)\boldsymbol{u}(i) \tag{14}$$

$$e_{ud}(i) = \frac{y_{ud}(i)}{|y_{ud}(i)|} - y_{ud}(i) \tag{15}$$

$$\boldsymbol{w}_{ud}(i) = \boldsymbol{w}_{dd}(i-1) - \frac{\mu}{\|\boldsymbol{u}(i)\|^2}\boldsymbol{u}(i)e_{ud}(i)* \tag{16}$$

$$y_{dd}(i) = \boldsymbol{w}_{ud}^H(i)\boldsymbol{u}(i-Z) \tag{17}$$

$$e_{dd}(i) = \frac{y_{dd}(i)}{\|y_{dd}(i)\|} - y_{dd}(i) \tag{18}$$

$$\boldsymbol{w}_{dd}(i) = \boldsymbol{w}_{ud}(i) + \frac{\mu}{\|\boldsymbol{u}(i)\|^2}\boldsymbol{u}(m-Z)e_{dd}^*(i) \tag{19}$$

where $y_{ud}(i)$ is the update step output, $e_{ud}(i)$ is the update step error, $y_{dd}(i)$ is the downdate step output, and $e_{dd}(i)$ is the downdate step at i-th iteration.

3. Proposed Algorithm

The proposed adaptive beamforming is the combination of the CMA and RLS algorithms, with an adaptive forgetting factor and a variable regularization factor implemented with the sliding window technique.

In RLS, the forgetting factor takes into account both previous data and new data. The error may be increased and passed to the next iterations when the forgetting factor is constant, resulting in performance degradation, especially in fast target tracking. This problem is solved here by adapting the forgetting factor gradually. In this sense, we adapted the adaptive moment estimation method (ADAM) [19] to update the forgetting factor at each iteration. This adaptation involves running averages of both the gradient and the second moment of the gradient according to the cost function $j(i) = \sum_{l=0}^{i} \lambda^{i-l}|e(i)|^2$. The adaptive forgetting factor based on ADAM can be expressed through the following equations:

$$p(i) = \beta_1 p(i-1) + (1-\beta_1)\nabla_\lambda(i) \tag{20}$$

$$v(i) = \beta_2 v(i-1) + (1-\beta_2)\nabla_\lambda(i)^2 \tag{21}$$

$$\lambda(i) = \lambda(i-1) - \mathrm{Re}\left\{\eta\frac{p(i)}{\sqrt{v(i)}+\xi}\right\} \tag{22}$$

where β_1, $\beta_2 \in [0,1)$, ξ is a small scalar to prevent division by 0, η is the step size, and $\nabla_\lambda(i)$ is the gradient of the forgetting factor in each iteration, calculated by taking the partial derivative of the $j(i)$ with respect to λ. The gradient of the forgetting factor, $\nabla_\lambda(i)$, can be expressed as [20,21]:

$$\nabla_\lambda(i) = \frac{dj'(i)}{d\lambda} = -\frac{1}{2}\mathrm{E}\left[\boldsymbol{\psi}^H(i-1)\boldsymbol{u}(i)e*(i) + \boldsymbol{u}^H(i)\boldsymbol{\psi}(i-1)e(i)\right] \tag{23}$$

where $\boldsymbol{\psi}(i) = \frac{d\boldsymbol{w}(i)}{d\lambda}$ and can be expressed as:

$$\boldsymbol{\psi}(i) = \left[\mathbf{I}_N - \boldsymbol{g}(i)\boldsymbol{u}^H(i)\right]\boldsymbol{\psi}(i-1) + \mathbf{S}(i)\boldsymbol{u}(i)e*(i) \tag{24}$$

where $\mathbf{S}(i) = \frac{d\mathbf{P}(m)}{d\lambda}$ and can be expressed as:

$$\mathbf{S}(i) = \lambda(i-1)^{-1}\left[\mathbf{I}_N - \boldsymbol{g}(i)\boldsymbol{u}^H(i)\right]\mathbf{S}(i-1) + \left[\mathbf{I}_N - \boldsymbol{u}(i)\boldsymbol{g}^H(i)\right] + \lambda(i-1)^{-1}\boldsymbol{g}(i)\boldsymbol{g}^H(i) - \lambda(m-1)^{-1}\mathbf{P}(i) \tag{25}$$

The regularization parameter is important since noise variations can be very high when fast target tracking is required. Practically, the optimum regularization factor has no closed-form solution. However, the optimal regularization factor can be approximated as follows [17]:

$$\Delta_{opt} \approx \frac{N\sigma_z^2}{\mathrm{tr}[\mathbf{R}_{\mathrm{uu}}]} \tag{26}$$

where σ_z^2 is noise variance, N is number of antennas, and $\mathbf{R}_{\mathrm{uu}} \triangleq \boldsymbol{u}\boldsymbol{u}^H$.

Finally, the sliding window technique is adapted in our proposed method, which contains both downdating and updating procedures. Our proposed adaptive beamforming algorithm is hereinafter referred to as SW-AFVF-CMARLS to improve traceability. The proposed algorithm is given in Algorithm 1 as follows:

Algorithm 1 Proposed Algorithm (SW-AFVF-CMARLS)

Initialize: $\Delta^{-1}(0) = 0.5, \lambda(0) = 0.97, p(0) = v(0) = 0, \mu = \eta = 0.001, \xi = 10^{-6}$, $\beta_1 = 0.9$, $\beta_2 = 0.99$, $\mathbf{P}(0) = \Delta^{-1}(0)\mathbf{I}_N, \boldsymbol{w}_{RLS-dd}(0) = \boldsymbol{w}_{CMA-dd}(0) = \mathbf{0}$.
Input: **u**, **d**, I, N, Z.
Output: **w.**

for $i = 1$ *to* I **do**
//updating weights
$y_{ud}(i) = \boldsymbol{w}^H_{CMA-dd}(i-1)\boldsymbol{u}(i)$
$e_{CMA-ud}(i) = \frac{y_{ud}(i)}{|y_{ud}(i)|} - y_{ud}(i)$
$\boldsymbol{w}_{CMA-ud}(i) = \boldsymbol{w}_{CMA-dd}(i-1) - \frac{\mu}{\|\boldsymbol{u}(i)\|^2}\boldsymbol{u}(i)e^*_{CMA-ud}(i)$
$\boldsymbol{g}_{ud}(i) = \frac{\mathbf{P}_{dd}(i-1)\boldsymbol{u}(i)}{1+\lambda^{-1}\boldsymbol{u}(i)^H\mathbf{P}_{dd}(i-1)\boldsymbol{u}(i)}$
$e_{RLS-ud}(i) = d(i) - \boldsymbol{w}^H_{CMA-ud}(i)\boldsymbol{u}(i)$
$\boldsymbol{w}_{RLS-ud}(i) = \boldsymbol{w}_{RLS-dd}(i-1) - \boldsymbol{g}_{ud}(\mathrm{i})e^*_{RLS-ud}(i)$
$\mathbf{P}_{ud}(i) = \lambda(i-1)^{-1}\Big(\mathbf{P}_{dd}(i-1) - \boldsymbol{g}_{ud}(\mathrm{i})\boldsymbol{u}(i)^H\mathbf{P}_{dd}(i-1)\Big)$
*//*updating forgetting factor
$\mathbf{S}(i) = \lambda(i-1)^{-1}\left[\mathbf{I}_N - \boldsymbol{g}_{ud}(i)\boldsymbol{u}^H(i)\right]\mathbf{S}(i-1) + \left[\mathbf{I}_N - \boldsymbol{u}(i)\boldsymbol{g}^H_{ud}(i)\right] + \lambda(i-1)^{-1}\boldsymbol{g}_{ud}(i)\boldsymbol{g}^H_{ud}(i) - \lambda(m-1)^{-1}\mathbf{P}_{ud}(i)$
$\boldsymbol{\psi}(i) = \left[\mathbf{I}_N - \boldsymbol{g}_{ud}(i)\boldsymbol{u}^H(i)\right]\boldsymbol{\psi}(i-1) + \mathbf{S}(i)\boldsymbol{u}(i)e^*_{RLS-ud}(i)$,
$\nabla_\lambda(i) = -\frac{1}{2}\mathrm{E}\Big[\boldsymbol{\psi}^H(i-1)\boldsymbol{u}(i)e^*_{RLS-ud}(i) + \boldsymbol{u}^H(i)\boldsymbol{\psi}(i-1)e_{RLS-ud}(i)\Big]$
//apply ADAM algorithm
$p(i) = \beta_1 p(i-1) + (1-\beta_1)\nabla_\lambda(i)$
$v(i) = \beta_2 v(i-1) + (1-\beta_2)\nabla_\lambda(i)^2$
$\lambda(i) = \lambda(i-1) - \mathrm{Re}\left\{\eta\frac{p(i)}{\sqrt{v(i)+\xi}}\right\}$
//downdating weights
$y_{dd}(i) = \boldsymbol{w}^H_{CMA-ud}(i)\boldsymbol{u}(i-\mathrm{Z})$
$e_{CMA-dd}(i) = \frac{y_{dd}(i)}{\|y_{dd}(i)\|} - y_{dd}(i)$
$\boldsymbol{w}_{CMA-dd}(i) = \boldsymbol{w}_{CMA-ud}(i) + \frac{\mu}{\|\boldsymbol{u}(i)\|^2}\boldsymbol{u}(m-\mathrm{Z})e^*_{CMA-dd}(i)$
$\boldsymbol{g}_{dd}(i) = \frac{\mathbf{P}_{ud}(i)\boldsymbol{u}(i-\mathrm{Z})}{1-\lambda(i-1)^{-1}\boldsymbol{u}(i-\mathrm{Z})^H\mathbf{P}_{ud}(i)\boldsymbol{u}(i-\mathrm{Z})}$
$e_{RLS-dd}(i) = d(i-\mathrm{Z}) - \boldsymbol{w}^H_{CMA-ud}(i)\boldsymbol{u}(i-\mathrm{Z})$
$\boldsymbol{w}_{RLS-dd}(i) = \boldsymbol{w}_{RLS-ud}(i) + \boldsymbol{g}_{dd}(\mathrm{i})e^*_{RLS-dd}(i)$
$\mathbf{P}_{dd}(i) = \lambda(i-1)^{-1}\Big(\mathbf{P}_{ud}(i) + \boldsymbol{g}_{dd}(\mathrm{i})\boldsymbol{u}(i-\mathrm{Z})^H\mathbf{P}_{ud}(i)\Big)$
//update regularization factor
$\Delta(i) = \frac{N\sigma^2_z(i)}{\mathrm{tr}\left[\left(\boldsymbol{u}(i)\boldsymbol{u}(i)^H\right)\right]} - \Delta(i-1)$
$\mathbf{P}_{dd}(i) = \mathbf{P}_{dd}(i-1) - \Delta(i)\mathbf{P}_{dd}(i)$
//final weights
$\boldsymbol{w} = \boldsymbol{w}_{RLS-ud}(i)$
end for

4. Results and Discussions

The SINR is considered as one of the performance metrics, which can be calculated as follows [22]:

$$\mathrm{SINR}_i = \frac{\sigma^2_k(i)\left|\boldsymbol{w}^H_k(i)\mathbf{a}_k(i)\right|^2}{\boldsymbol{w}^H_k(i)\mathbf{R}_{int+n}(i)\boldsymbol{w}_k(i)} \tag{27}$$

where $\sigma^2_k(i)$ is the source signal power in iteration i, $\mathbf{a}_k(i) = [\hat{r}_k\bar{r}_1, \ldots, \hat{r}_k\bar{r}_n, \ldots, \hat{r}_k\bar{r}_N]^T$ is the actual steering vector of the desired signal in iteration i, $\boldsymbol{w}_i$ is the estimated weights in iteration i, and $\boldsymbol{R}_{int+n}[t]$ is the interference-plus-noise covariance matrix in iteration i.

The SINR performance of the proposed algorithm SW-AFVF-CMARLS is compared with three different RLS algorithms under a single-user case. The SW-RLS is the conventional sliding window RLS algorithm with a fixed forgetting factor, SW-CMARLS is the sliding window hybrid CMA and RLS algorithms with a fixed forgetting factor, and SW-VRF-CMARLS is the proposed sliding window variable regularization factor with a fixed forgetting factor, as presented in [17] and extended with CMA. All algorithms are compared in two use cases: (a) NTN connecting a ground station of LEO, and (b) V2X. In both use cases, there are two static interference signals located at $\theta_{int1} = 40^{\circ}$, $\theta_{int2} = 60^{\circ}$, and $\varphi_{int1} = 30^{\circ}$, $\varphi_{int2} = 75^{\circ}$ in spherical coordinates, and $L = 3$. The interference-to-noise ratios (INR) of interference signals are 10 dB. The step sizes in all related algorithms are, $\mu = \eta = 0.001$. Moreover, the forgetting factor, λ, for the SW-RLS, SW-CMARLS, SW-VRF-RLS is fixed and equals 0.99, and the regularization parameter, Δ, equals 0.5. Additional parameters used in the simulations for the presented algorithm SW-AFF-CMARLS (in both use cases) are given in Table 1.

Table 1. Simulation parameters for SW-AFF-CMARLS.

Parameters	Value
β_1, β_2	0.9, 0.99
Initial λ, Δ	0.97, 0.5
Initial p, v	0, 0
ξ	10^{-6}

In the case of NTN-LEO, the realistic parameters in [5] are used. In addition, the samples are considered to be collected per minute and the window length, Z, is set to 20 samples. In this case. the target (satellite) moves in elevation. The scenario lasts for 15 min, during which a scanning of 44 degrees in elevation is performed. Figure 5a shows the simulation results according to the parameters in Table 2 for the NTN-LEO use case. First of all, the SW-CMARLS has the worst performance and cannot converge. It can be seen that the conventional SW-RLS algorithm has a lower convergence rate, with an almost 3 dB lower steady-state gain than the proposed SW-AFVF-CMARLS and SW-VRF-CMARLS. In addition, the SW-RLS algorithm has the highest steady-state noise since the regularization parameter is fixed. On the other hand, according to Figure 5a, our proposed SW-AFVF-CMARLS algorithm outperforms the SW-RLS algorithm in terms of convergence rate, steady-state gain and noise. In the case of V2X, the realistic parameters in [2] are used. In addition, the samples are considered to be collected per second and the window length, Z, is set to 20 samples. In this case, the target (vehicle) moves in elevation. The scenario lasts for 12 s and scanning of 60 degrees in elevation is performed. Figure 5b shows the simulation results according to the parameters in Table 3 for the V2X use case. Similar results to the previous example are observed. Figure 5b shows that the SW-VFF-RLSCMA algorithm outperforms the compared algorithms in terms of convergence rate, steady-state noise, and SINR gain.

Table 2. Parameters for the NTN-LEO simulation.

Parameters	Value
Satellite height	600 km.
Minimum elevation angle	40 deg.
Field of view	44 deg.
No. of Rx antennas, x-dim.	4
No. of Rx antennas, y-dim.	4
Simulation time	15 min.
Displacement of satellite	≈2.9 deg./min.
Samples per minute	400

Table 3. Parameters for the V2X simulation.

Parameters	Value
V2I Euclidean distance (initial)	100 m.
Infrastructure to road distance	≈70.71 m.
Vehicle velocity	60 km/h (≈16.6 m/s)
Minimum elevation angle	30 deg.
Field of view	60 deg.
No. of Rx antennas, x-dim.	4
No. of Rx antennas, y-dim.	4
Simulation time	12 s
Displacement of vehicle	5 deg./s
Samples per second	500

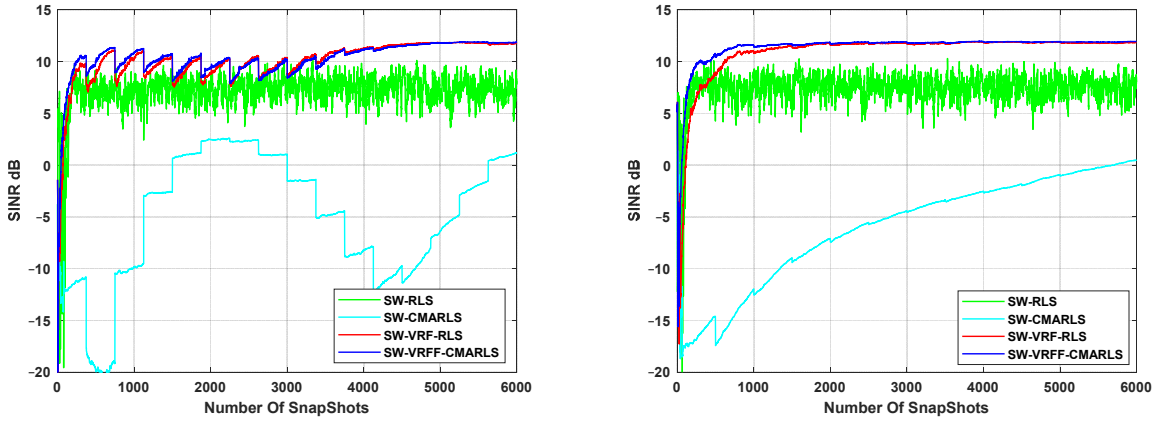
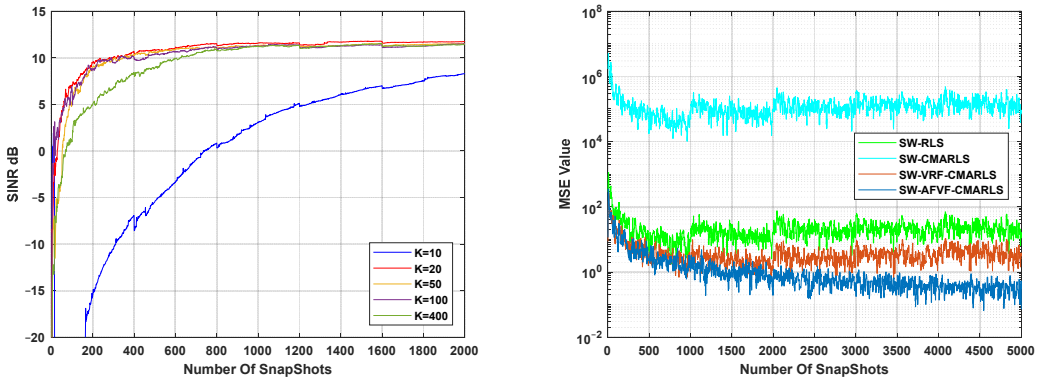


Figure 5. (**a**) NTN-LEO use case: SINR versus number of snapshots comparison (**b**) V2X use case: SINR versus number of snapshots comparison.

Figure 6a demonstrates the effect of window size on the convergence performance and convergence rate of the proposed algorithm. Iterations up to 2000 are given because it is sufficient for the demonstration of the desired effect. I can be seen that $Z = 20$ is sufficient for a proper application. The optimal size of Z was found using the trial-and-error method since there is no closed expression for the optimal window size. Figure 6b shows the mean-square-error comparison of the proposed and state-of-the art algorithms revised. In MSE analysis, we consider the squared mean error between the ideal output signal, whose weights are calculated using the Wiener optimal weight calculation, and the estimated output signal, whose weights are calculated using the proposed algorithm. The MSE between the ideal output signal and the estimated output signal is MSE $= \frac{1}{n}\sum_{i=1}^{n}(\hat{y}_i - y_i)^2$. It is clear that the proposed algorithms outperform all compared algorithms in terms of MSE at steady state by at least 10 times. The initial parameters of the ADAM algorithm are given in Table 1. The initial forgetting factor, λ, and the initial regularization factor, Δ, are 0.96 and 0.5, respectively. The related parameters for the Figure 6 are given in Table 4.

Table 4. Parameter for the multi-user case simulations.

Parameters	Value
No. of Rx antennas, x-dim.	4
No. of Rx antennas, y-dim.	4
Simulation time	5 s
Displacement of target	1 deg./s (θ), 2 deg./s (φ)
Samples per second	1000



(**a**) (**b**)

Figure 6. (**a**) Window size, Z, effect on convergence and convergence rate, (**b**) MSE comparison.

The computational complexity analysis of the algorithms is given in Table 5. In Figure 7, the complexity comparison of the algorithms is given with respect to the number of antennas in the antenna array. The computational complexity is calculated according to complex multiplications involved per iteration in the algorithms. The MUSIC [23] and Wiener optimal weights [24] are given for a better perception. The MUSIC algorithm involves high computational cost due to covariance matrix and eigen value decomposition, while the Wiener optimal algorithm involves high computational cost due to inverse matrix computation (which is often inaccurate in hardware). The window size is set to $Z = 20$ for both MUSIC and Wiener optimal algorithms. The proposed algorithm has more complexity than the revised adaptive algorithms since more computations are involved to calculate the adaptive forgetting factor and the variable regularization factor, but it results in better performance metrics. The related parameters for Figure 7 are given in Table 4.

Table 5. Complexity analysis.

Algorithm	Complex Multiplications
SW-RLS	$5N^2 + 8N$
SW-CMARLS	$5N^2 + 14N$
SW-VRF-CMARLS	$8N^2 + 16N$
SW-AFF-CMARLS	$13N^2 + 18N$
MUSIC	$N^3 + NZ^3(N - Z) + N^2Z$
Wiener-Optimal	$N^3 + N^2Z + N^2$

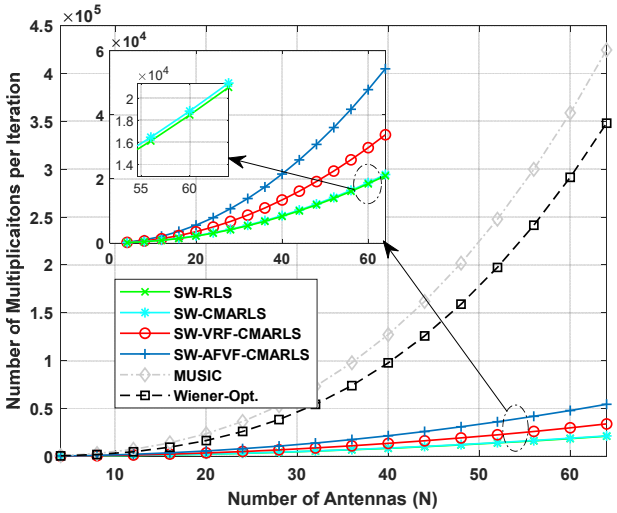
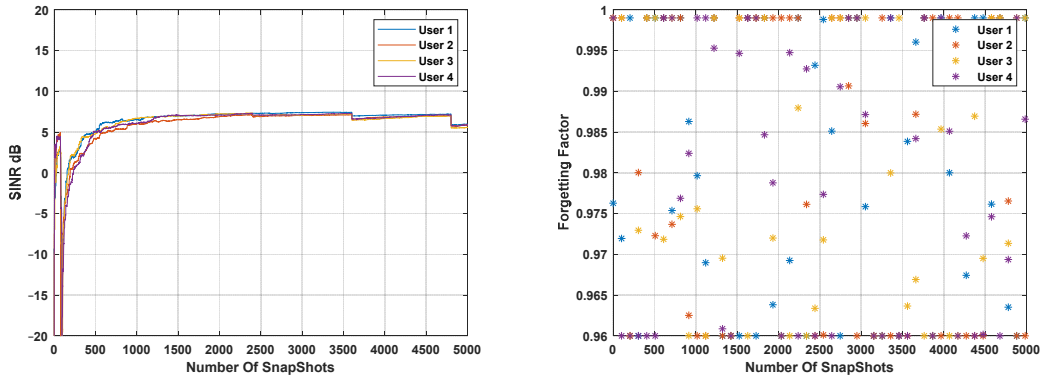


Figure 7. Number of complex multiplications per iteration with respect to number of antennas.

In the second part of the analysis, after concluding that the proposed algorithm has better performance than its counterparts, the proposed SW-AFF-CMARLS algorithm was tested in the multi-user case. In order to evaluate our proposed algorithm's robustness, we tested it in more detail in the multi-user case.

In multi-user case analysis, we considered four spatially uncorrelated different users. The users were also considered as moving sources, as in the previous examples, but with different angular speeds and with two static interference signals. To enhance later analysis of the transmitted signal, the users are initially located at $\varphi_1 = 45^\circ$, $\varphi_2 = 135^\circ$, $\varphi_3 = 225^\circ$, $\varphi_4 = 315^\circ$, and $\theta_1 = \theta_2 = \theta_3 = \theta_4 = 45^\circ$, and $L = 3$. The static interference signals are located at $\varphi_5 = 30^\circ$, $\varphi_6 = 60^\circ$, and $\theta_5 = 90^\circ$, $\theta_6 = 180^\circ$ in spherical coordinates, and $L = 3$. The interference-to-noise ratios (INR) of interference signals are 10 dB. For the multi-user case analysis, the step sizes are $\mu = \eta = 0.01$. Lastly, the window length, Z, is set to 75 samples. The initial parameters of the ADAM algorithm are given in Table 1. The initial forgetting factor, λ, and the initial regularization factor, Δ, are 0.96 and 0.5, respectively. Each user has the same angular speed and moves in azimuth coordinate only. Figure 8a shows the SINR performance of the proposed algorithm with four users. The related parameters for the multi-user case are given in Table 4. Figure 8a shows that there is a 6 dB loss at the steady-state gain, according to the previous SINR analysis; however, this outcome is expected since the same number of antennas are used for more users. Apart from the loss, performance degradation related to the convergence rate can be observed. The reason is that the algorithm needs to extract four different users' information from the same signal. Despite slight performance degradation, the proposed SW-AFF-CMARLS algorithm still has acceptable convergence and convergence rate. We can conclude that the proposed algorithm can be used in a multi-user case. Figure 8b shows the variations of the forgetting factor according to the number of iterations for each user. It should be noted that the forgetting factor value is given in Figure 8b for every 100 iterations. This is because, given the forgetting factor values for all iterations, the graph becomes very complex and hard to comprehend.



(**a**) (**b**)

Figure 8. (**a**) SINR with respect to the number of iterations, (**b**) forgetting factor variation of each user with respect to the iteration number.

In Figure 9, the normalized radiation pattern of the estimated transmitter signals is given. The purpose here is to demonstrate if the antenna array beams are correctly directed to each desired user direction, as well as with canceled interference signals, after 5 s (as specified in the simulation time) using our proposed SW-AFVF-CMARLS algorithm. In Figure 9a, users and interference signals are located at their initial directions. In Figure 9b, after the proposed algorithm was applied, the directions of the users are given with suppressed interference. The interference-free displacements of the users are clearly seen.

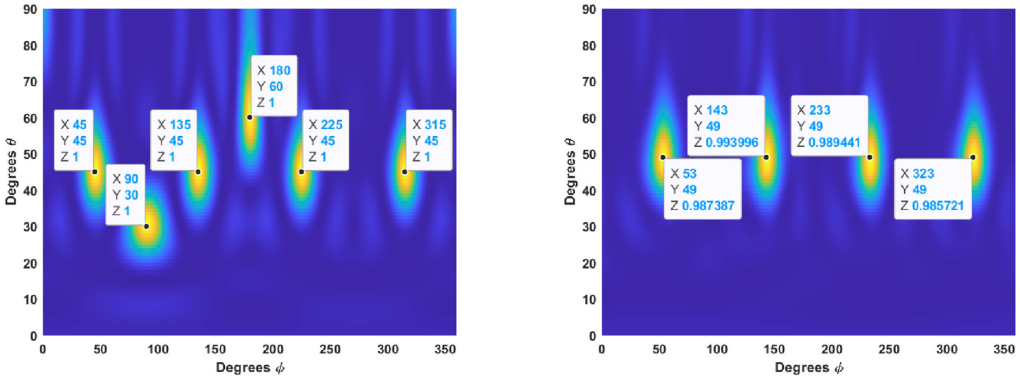


Figure 9. Users and interference directions (**a**) at the initial stage, (**b**) estimated after applying the proposed algorithm.

As a remark, assuming that we have a satellite communication on the move (SOTM) terminal in the transmitting time-of-movement area with a 30 GHz carrier and 30 MHz sampling frequency, with a modulation coding (exp. DVB-S2), the proposed algorithm is able to obtain the relative position and angular velocity of the trains in the same service area in $\cong$ 0.083 milliseconds. Regarding Figures 2 and 3, it is worth mentioning that by using the proposed algorithm, θ_k and φ_k can be obtained via the satellite communication system (NTN-LEO) or V2X system. In addition, the proposed algorithm also reduces the estimation error of the relative position and angular velocity of trains in the service area by $\cong 10^5$ times.

5. Conclusions

In this study, we proposed an adaptive forgetting factor and variable regularization factor hybrid CMA-RLS algorithm with a sliding window technique. The proposed algorithm exploits the fast convergence rate of RLS and the high convergence of CMA. Finite memory is also taken into account to reduce the steady-state noise. In addition, we introduced the ADAM method to adaptively update the forgetting factor. The proposed algorithm was evaluated under two use cases, NTN-LEO and V2X, respectively, and compared with the conventional RLS and other two different RLS-based algorithms. The results demonstrate that our proposed algorithm outperforms its counterparts in terms of convergence rate, steady-state gain, and noise when the variable forgetting factor is used. In addition, the performance of the proposed algorithm was analyzed in a multi-user tracking scenario. It can be concluded that the proposed algorithm can be used to track both fast moving single and multiple targets in interference environments. Based on the results, the proposed method has a 3 dB SINR gain and far less steady-state noise than the conventional sliding window RLS (the ripple is 5 dB for sliding window RLS and almost 0.1 dB for sliding window adaptive forgetting factor variable regularization factor CMA-RLS). However, it has the same performance as sliding window variable regularization factor CMA-RLS. On the other hand, the proposed algorithm has at least two times faster convergence than the revised algorithms and outperforms all compared algorithms in terms of MSE at steady state by at least 10 times, with the cost of only $8N^2 + 10N$ more complex multiplication. The next step for future work will be to implement the proposed algorithm on a real-time processor and compare the measured results with the simulation results.

Author Contributions: Conceptualization, A.E.Z., M.A.S.-N. and R.M.R.-O.; methodology, A.E.Z., M.A.S.-N. and R.M.R.-O.; validation, A.E.Z.; writing—original draft preparation, A.E.Z.; writing—review and editing, A.E.Z. and M.A.S.-N. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the Spanish Government, the Ministry of Economy, and the National Program of Research, Development, and Innovation under the project New Array Antenna Tech. and digital processing for the FUTURE integrated terrestrial and space-based millimeter-wave radio systems, UPM-InTerSpaCE (PID2020-112545RB-C51).

Data Availability Statement: Data is available upon request. Please contact the corresponding author by email to obtain the data used in this study.

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

References

1. Reis, N.; da Silva, J.M.; Correia, M.V. An Introduction to the Evaluation of Perception Algorithms and LiDAR Point Clouds Using a Copula-Based Outlier Detector. *Remote Sens.* **2023**, *15*, 4570. [CrossRef]
2. Montero, L.; Ballesteros, C.; de Marco, C.; Jofre, L. Beam management for vehicle-to-vehicle (V2V) communications in millimeter wave 5G. *Veh. Commun.* **2022**, *34*, 100424. [CrossRef]
3. Gouran, P.; Nadimi-Shahraki, M.H.; Rahmani, A.M.; Mirjalili, S. An Effective Imputation Method Using Data Enrichment for Missing Data of Loop Detectors in Intelligent Traffic Control Systems. *Remote Sens.* **2023**, *15*, 3374. [CrossRef]
4. Noor, N.M.; Abdullah, A.; Hashim, M. Remote sensing UAV/drones and its applications for urban areas: A review. *IOP Conf. Ser. Earth Environ. Sci.* **2018**, *169*, 012003. [CrossRef]
5. Caus, M.; Perez-Neira, A.; Mendez, E. Smart Beamforming for Direct LEO Satellite Access of Future IoT. *Sensors* **2021**, *21*, 4877. [CrossRef] [PubMed]
6. Kaur, S.; Kumar, N.; Dubey, S. Investigation of Adaptive Beam-forming Algorithms for Smart Antennas System. *IOP Conf. Ser. Mater. Sci. Eng.* **2021**, *1033*, 012015. [CrossRef]
7. Varade, S.W.; Kulat, K.D. Robust Algorithms for DOA Estimation and Adaptive Beamforming for Smart Antenna Application. In Proceedings of the 2009 Second International Conference on Emerging Trends in Engineering & Technology, Nagpur, India, 16–18 December 2009; pp. 1195–1200. [CrossRef]
8. Jeripotula, P.R.; Naik, B.R.; Mudavath, R. A Novel Hybrid Weighted Method-Based Beamforming for Sidelobe Level Reduction in Radar Applications. *Arab. J. Sci. Eng.* **2022**, *47*, 14133–14145. [CrossRef]
9. Raj, P.; Kumar, A.; Mudavath, R.; Naik, B. A Novel Algorithm Design for Adaptive Beamforming in Uniform Linear Array Antenna. *J. Mech. Continua Math. Sci.* **2020**, *15*, 102–113. [CrossRef]
10. Salas-Natera, M.; Martínez, R.; De Haro, L.; Perez, M.S. Novel Reception and Transmission Calibration Technique for Active Antenna Array Based on Phase Center Estimation. *IEEE Trans. Antennas Propag.* **2017**, *65*, 5511–5522. [CrossRef]
11. Zhao, H.; Zhang, L.; Liu, B.; Zhang, Q.; Wang, Y.; Tian, Q.; Yin, X.; Xin, X. MIMO signal processing for mode division multiplexing with RLSCMA algorithm. In Proceedings of the 2014 13th International Conference on Optical Communications and Networks (ICOCN), Suzhou, China, 9–10 November 2014; pp. 1–3. [CrossRef]
12. Shan, L.; Chen, H.; Luan, J.; Li, J. Application of adaptive forgetting factor RLS algorithm in target tracking. In Proceedings of the 2017 Chinese Automation Congress (CAC), Jinan, China, 20–22 October 2017; pp. 1838–1843. [CrossRef]
13. Liu, H.; He, Z. A sliding-exponential window RLS adaptive filtering algorithm: Properties and applications. *Signal Process.* **1995**, *45*, 357–368. [CrossRef]
14. Chern, S.-J.; Sun, C.-H.; Chang, C.-C. Blind Adaptive DS-CDMA Receivers with Sliding Window Constant Modulus GSC-RLS Algorithm. In Proceedings of the 2006 International Symposium on Intelligent Signal Processing and Communications, Yonago, Japan, 12–15 December 2006; pp. 979–982. [CrossRef]
15. Mahadi, M.; Ballal, T.; Moinuddin, M.; Al-Naffouri, T.Y.; Al-Saggaf, U. Low-Complexity Robust Beamforming for a Moving Source. In Proceedings of the 2020 28th European Signal Processing Conference (EUSIPCO), Amsterdam, The Netherlands, 18–21 January 2021; pp. 1846–1850. [CrossRef]
16. Belge, M.; Miller, E.L. A sliding window RLS-like adaptive algorithm for filtering alpha-stable noise. *IEEE Signal Process. Lett.* **2000**, *7*, 86–89. [CrossRef]
17. Mahadi, M.; Ballal, T.; Moinuddin, M.; Al-Saggaf, U. A Recursive Least-Squares with a Time-Varying Regularization Parameter. *Appl. Sci.* **2022**, *12*, 2077. [CrossRef]
18. Xiao, Y.; Sun, J. RLS CMA blind equalization with adaptive forgetting factor controlled by energy steady state. In Proceedings of the 2016 9th International Congress on Image and Signal Processing, BioMedical Engineering and Informatics (CISP-BMEI), Datong, China, 15–17 October 2016; pp. 935–939. [CrossRef]
19. Fei, Z.; Wu, Z.; Xiao, Y.; Ma, J.; He, W. A new short-arc fitting method with high precision using Adam optimization algorithm. *Optik* **2020**, *212*, 164788. [CrossRef]
20. Lee, S.; Lim, J.; Sung, K.-M. A low-complexity AFF-RLS algorithm using a normalization technique. *IEICE Electron. Express* **2009**, *6*, 1774–1780. [CrossRef]
21. Song, S.; Lim, J.-S.; Baek, S.; Sung, K.-M. Gauss Newton variable forgetting factor recursive least squares for time varying parameter tracking. *Electron. Lett.* **2000**, *36*, 988–990. [CrossRef]
22. Chen, Y.; Le-Ngoc, T.; Champagne, B.; Changjiang, X. Recursive Least Squares Constant Modulus Algorithm for Blind Adaptive Array. *IEEE Trans. Signal Process.* **2004**, *52*, 1452–1456. [CrossRef]

23. Krim, H.; Viberg, M. Two decades of array signal processing research: The parametric approach. *IEEE Signal Process. Mag.* **1996**, *13*, 67–94. [CrossRef]
24. Jalal, B.; Elnahas, O.; Quan, Z. Efficient DOA Estimation Under Partially Impaired Antenna Array Elements. *IEEE Trans. Veh. Technol.* **2022**, *71*, 7991–7996. [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.