An Algorithm for Online Inertia Identification and Load Torque Observation via Adaptive Kalman Observer-Recursive Least Squares

Authors:

Ming Yang, Zirui Liu, Jiang Long, Wanying Qu, Dianguo Xu

Date Submitted: 2020-06-23

Keywords: motor control, parameter identification, full-order observer

Abstract:

> In this paper, an on-line parameter identification algorithm to iteratively compute the numerical values of inertia and load torque is proposed. Since inertia and load torque are strongly coupled variables due to the degenerate-rank problem, it is hard to estimate relatively accurate values for them in the cases such as when load torque variation presents or one cannot obtain a relatively accurate priori knowledge of inertia. This paper eliminates this problem and realizes ideal online inertia identification regardless of load condition and initial error. The algorithm in this paper integrates a full-order Kalman Observer and Recursive Least Squares, and introduces adaptive controllers to enhance the robustness. It has a better performance when iteratively computing load torque and moment of inertia. Theoretical sensitivity analysis of the proposed algorithm is conducted. Compared to traditional methods, the validity of the proposed algorithm is proved by simulation and experiment results.

DOI of Published Version: https://doi.org/10.3390/en11040778 energies

Article

An Algorithm for Online Inertia Identification and Load Torque Observation via Adaptive Kalman Observer-Recursive Least Squares

Ming Yang *, Zirui Liu, Jiang Long, Wanying Qu and Dianguo Xu

Institute of Power Electronics and Electrical Drives, Harbin Institute of Technology, Harbin 150001, China; tradigrada@163.com (Z.L.); volkswage.n@163.com (J.L.); 18091720657@163.com (W.Q.); xudiang@hit.edu.cn (D.X.)

* Correspondence: yangming@hit.edu.cn; Tel.: +86-133-9460-4501

Received: 29 January 2018; Accepted: 26 March 2018; Published: 28 March 2018

Abstract: In this paper, an on-line parameter identification algorithm to iteratively compute the numerical values of inertia and load torque is proposed. Since inertia and load torque are strongly coupled variables due to the degenerate-rank problem, it is hard to estimate relatively accurate values for them in the cases such as when load torque variation presents or one cannot obtain a relatively accurate priori knowledge of inertia. This paper eliminates this problem and realizes ideal online inertia identification regardless of load condition and initial error. The algorithm in this paper integrates a full-order Kalman Observer and Recursive Least Squares, and introduces adaptive controllers to enhance the robustness. It has a better performance when iteratively computing load torque and moment of inertia. Theoretical sensitivity analysis of the proposed algorithm is conducted. Compared to traditional methods, the validity of the proposed algorithm is proved by simulation and experiment results.

Keywords: full-order observer; parameter identification; motor control

1. Introduction

The permanent magnet synchronous motor (PMSM) has several advantages such as high torque density, high precision, and high efficiency. Thus, it has been widely applied in modern industrial fields such as industrial robotics and electric vehicles. Much research has been conducted to improve PMSM control performance [1–5]. Nevertheless, precise PMSM control requires accurate knowledge of the system parameters which must be measured or identified.

Among these system parameters, moment of inertia is an essential parameter which can influence the design of speed loop controllers [6,7]. Also, load torque information is necessary to cope with disturbances [8,9]. Thus, their knowledge is of significance to the desired system performance.

However, moment of inertia and load torque cannot be obtained without identification methods. Therefore, many studies have researched this field. The inertia identification methods in [10–12] estimate offline and a specific speed command, such as a sine wave, is implemented in such identification methods. For load torque estimation, an observer is mostly widely used [13–15]. One assumption in conventional observer design is that the moment of inertia in the system model is known. However, due to the complex working environments, in real applications it is difficult to obtain a relatively accurate a priori value to implement the referenced algorithms. Moreover, for industrial robotics cases, the load torque may vary from time to time. Since inertia and load torque are variables which cannot be precisely decoupled, if inertia is estimated by the observer and the identified inertia is one of the parameters in the identification algorithm, the observer will be severely affected by the initial value of the moment of inertia, and in the worst case, the system will be unstable. The common way is solve this issue is to combine two methods together to solve this problem [16–24].

Several studies have been done to overcome this coupling problem. The authors in [16] proposed a digital disturbance observer integrated with an inertia self-turning function. In [17], a full-order Luenberger Observer is used to obtain the knowledge of load torque, velocity and rotor position. The rotor position is the only measurement signal which is used to correct the knowledge of the machine load torque. However, the observer is severely affected by the initial value of the inertia. Moreover, the system will be unstable in the worse cases. Like [17], in [18] an Extended State Observer is proposed to obtain load torque and velocity, then the knowledge of load torque is used to identify the inertia. Regardless of the accuracy of the identified inertia, there is still an estimated error of the observer in this approach. The authors in [19] proposed an inertia identification algorithm based on a complex frequency-domain method. A Gopinath load torque observer is also used in [19] to obtain the knowledge of the load torque. The main disadvantage is that it requires massive computations and is quite time consuming. Due to the fact that the Kalman filter algorithm has low dependence on the model [20–22], thus it is a better choice for velocity and load torque observation. In [23], an algorithm which integrates a Kalman Observer (KO) and recursive least squares (RLS) was proposed, where the RLS estimator is used for inertia identification and the KO is mainly used to obtain the knowledge of load torque, but the authors in [23] mentioned that it's hard to choose appropriate values of the system noise matrix and measurement noise matrix in the KO. Like [23], the authors in [24] also used KO and RLS to obtain knowledge of both load torque and inertia. The main disadvantage of the algorithms in [23,24] is that the values of the system noise matrix and measurement noise matrix are hard to choose and difficult to transport to other platforms. Also, all the aforementioned researches do not concern time-varying and load torque conditions.

In this paper, the RLS method is used to estimate the moment of inertia meanwhile the KO is also adopted to observe load rotor position, velocity and load torque. Based on the original KO and RLS, the Adaptive Kalman Observer-Recursive Least Square (AKO-RLS) is proposed by adding adaptive algorithm to both the KO and the RLS. The main contribution is that the AKO-RLS algorithm can realize ideal inertia identification regardless of the load torque condition. Compared with the aforementioned existing methods, there are two main advantages to the AKO-RLS algorithm. Firstly, AKO-RLS improves the robustness in that it can achieve a better performance under the time-varying load condition. Secondly, AKO-RLS is more accurate when implemented with an unprecise pre-set inertia. With the aforementioned advantages, the contribution of this research is that while nowadays the control performance of most most servo products, such as Yaskawa, Fuji and Panasonic, is very good (the bandwidth of their speed loop and current loop is getting higher and higher), however, their auto-tuning function cannot operate with a sine-wave load, which is common for the robotic manipulator arms: they cannot identify inertia under a time-varying load-torque, which is exactly what is needed in the field of robotics. Optimization of the speed loop controller requires real-time knowledge of the inertia and load torque, which is main concern of this paper. With the proposed AKO-RLS algorithm, one can eliminate the impact of load torque variation below a certain frequency and let the inertia identification adapt to the most industrial field cases.

This paper is organized as follows: first, the PMSM discrete-time model is shown. Based on this, the structure of Kalman Observer is also presented in Section 2. Then, the RLS algorithm is explained in Section 3 and the original KO-RLS is also derived. In Section 4, the details and design of AKO-RLS are shown. Section 5 demonstrates the sensitivity analysis of the proposed algorithm. Then, simulations and experiments results are presented to verify the effectiveness of proposed algorithm in Section 6. Finally, Section 7 concludes the paper.

2. Design of the Kalman Observer

The PMSM mechanical equation can be expressed as:

$$\frac{d\omega}{dt} = -\frac{B}{J}\omega - \frac{1}{J}T_L + \frac{K_T}{J}i_q \tag{1}$$

$$\frac{d\theta}{dt} = \omega \tag{2}$$

where θ is the rotor position, ω is the motor mechanical angular velocity, T_L is the external force and disturbance torque, K_T is the motor torque constant, i_q is the q-axis current, J is the equivalent motor and load inertia and B is the friction viscous coefficient

In most cases T_L can be considered as a slow time-variant. Therefore, its differential can be described as:

$$\frac{dT_L}{dt} = 0 \tag{3}$$

Equations (1)–(3) can be concluded in state space form:

$$\begin{bmatrix} \dot{\theta} \\ \dot{\omega} \\ \dot{T}_L \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -\frac{B}{J} & -\frac{1}{J} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} \theta \\ \omega \\ T_L \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{K_T}{J} \\ 0 \end{bmatrix} i_q \tag{4}$$

Discretizing (4) by the Euler method, (5) can be obtained:

$$\begin{cases} \boldsymbol{x}(k+1) = \boldsymbol{Ax}(k) + \boldsymbol{Bu}(k) + \boldsymbol{w}(k) \\ \boldsymbol{y}(k) = \boldsymbol{Hx}(k) + \boldsymbol{v}(k) \end{cases} \tag{5}$$

where k is the sampling number, $\boldsymbol{X} = [\theta\omega T_L]^T$ is the state variable, $\boldsymbol{u} = [i_q]$ is the control signal, $\boldsymbol{Y} = [\theta]$ is the observed signal, and T_S is the sampling time interval:

$$\boldsymbol{A} = \begin{bmatrix} 1 & T_S & 0 \\ 0 & 1-\frac{B}{J}T_s & -\frac{T_S}{J} \\ 0 & 0 & 1 \end{bmatrix}; \boldsymbol{B} = \begin{bmatrix} 0 \\ \frac{T_s K_T}{J} \\ 0 \end{bmatrix}; \boldsymbol{H} = \begin{bmatrix} 1 & 0 & 0 \end{bmatrix}$$

where $\boldsymbol{w} = [w_\theta \; w_\omega \; w_{T_L}]^T$ and $\boldsymbol{v} = [v_\theta]^T$ denote the system and measurement Gaussian noises, respectively. w_θ and v_θ are variables that represent the system and measurement Gaussian noises of the rotor position signal θ, respectively. Similarly, w_ω and w_{T_L} denote the system noises of the angular velocity w and the load torque T_L, respectively. v is a first-order matrix because θ is the only observed signal. w_θ, w_w and w_{T_L} are not related to each other and other variables. Their covariance is equal to:

$$\begin{cases} \boldsymbol{Q} = \mathrm{cov}(w) = \begin{bmatrix} q_\theta & 0 & 0 \\ 0 & q_\omega & 0 \\ 0 & 0 & q_{T_L} \end{bmatrix} \\ \boldsymbol{R} = \mathrm{cov}(v) \end{cases} \tag{6}$$

q_θ is the variance of w_θ and q_w and q_{T_L} are the variances of w_w and w_{T_L}, respectively. From (5), the recursive form of the Kalman Observer can be obtained. It includes two distinct phases, the prediction phase and the updating phase:

- The prediction phase:

$$\begin{cases} \hat{\boldsymbol{X}}^-(k) = \boldsymbol{A}\hat{\boldsymbol{X}}(k-1) + \boldsymbol{Bu}(k) \\ \boldsymbol{P}^-(k) = \boldsymbol{AP}(k-1)\boldsymbol{A}^T + \boldsymbol{Q}(k) \end{cases} \tag{7}$$

- The updating phase:

$$\begin{cases} \hat{\boldsymbol{X}}(k) = \hat{\boldsymbol{X}}^{-}(k) + \boldsymbol{K}(k)\boldsymbol{V}(k) \\ \boldsymbol{P}(k) = (\boldsymbol{IK}(k) - \boldsymbol{H}) + \boldsymbol{P}^{-}(k) \end{cases} \tag{8}$$

where:

$$\begin{array}{l} \hat{\boldsymbol{M}}^{-}(k) = \boldsymbol{H}\hat{\boldsymbol{X}}^{-}(k) \\ \boldsymbol{V}(k) = \boldsymbol{M}(k) - \hat{\boldsymbol{M}}^{-}(k) \\ \boldsymbol{K}(k) = \boldsymbol{P}^{-}(k)\boldsymbol{H}^T\boldsymbol{S}^{-1}(k) \\ \boldsymbol{G}(k) = \left(\boldsymbol{HP}^{-}(k)\boldsymbol{H}^T + \boldsymbol{R}(k)\right)^{-1} \end{array} \tag{9}$$

$\boldsymbol{V}(k)$ is the innovation. $\boldsymbol{S}(k)$ represents the covariance of $\boldsymbol{V}(k)$. $\boldsymbol{I}$ denotes the unit matrix. $\hat{\boldsymbol{X}}^{-}(k)$ and $\hat{\boldsymbol{X}}(k)$ represent the a priori estimated state variable and a posteriori estimated state variable in the prediction phase, respectively. $\boldsymbol{P}^{-}(k)$ and $\boldsymbol{P}(k)$ are the a priori estimated error covariance and the a posteriori estimated error covariance after the updating phase.

3. Inertia Identification by RLS Estimator and Original KO-RLS

3.1. Inertia Identification by RLS

Equation (1) can be transformed to the z plane by applying the Z-transformation with the zero-order hold:

$$\begin{aligned} \omega(z) \quad &= Z\{\tfrac{1-e^{T_1 s}}{s}\tfrac{1}{B+sJ}\}(K_T i_q(z) - T_L(z)) \\ &= \tfrac{b_1 z^{-1}}{1+a_1 z^{-1}}(K_T i_q(z) - T_L(z)) \end{aligned} \tag{10}$$

where T_1 denotes the sampling time interval of the RLS algorithm and:

$$a_1 = -\exp\left(-\frac{B}{J}T_1\right), b_1 = \frac{1}{B}\left(1 - \exp\left(-\frac{B}{J}T_1\right)\right) \tag{11}$$

Then the RLS estimator for estimating a_1 and b_1 can be written as:

$$\begin{cases} \eta(n) = \frac{\boldsymbol{\psi}(n-1)\tau(n)}{\lambda+\tau^T(n)\boldsymbol{\psi}(n-1)\tau(n)} \\ \boldsymbol{e}(n) = \boldsymbol{d}(n) - \boldsymbol{\sigma}^T(n-1)\tau(n) \\ \boldsymbol{\sigma}(n) = \boldsymbol{\sigma}(n-1) + \eta(n)\boldsymbol{e}(n) \\ \boldsymbol{\psi}(n) = \frac{1}{\lambda}\boldsymbol{\psi}(n-1) - \frac{1}{\lambda}\eta(n)\tau^T(n)\boldsymbol{\psi}(n-1) \end{cases} \tag{12}$$

in which:

$$\begin{cases} \boldsymbol{\sigma}(n) = \begin{bmatrix} a_1 & b_1 \end{bmatrix}^T \\ \boldsymbol{d}(n) = \omega(n) \\ \tau(n) = \begin{bmatrix} -\omega(n-1) & K_T i_q(n-1) - T_L(n-1) \end{bmatrix}^T \end{cases} \tag{13}$$

This RLS algorithm can be obtained by minimizing the least squares cost function [25]:

$$\varsigma(n) = \sum_{i=0}^{n} \lambda^{n-i}\left[\boldsymbol{d}(i) - \boldsymbol{\sigma}^T(n)\tau(i)\right]^2 \tag{14}$$

λ is the forgetting factor, set in the range of 0.95–1 to ensure the rapidity of RLS. According to (11), J and B can be obtained by solving the following equation:

$$\begin{cases} \hat{B} = \frac{1+a_1}{b_1} \\ \hat{J} = -\frac{\hat{B}T_1}{\ln(-a_1)} \end{cases}$$

3.2. Details about KO-RLS

The KO-RLS can be obtained through connecting them by updating parameters to each other. T_L and w in (10) can be substituted by T_L and $\hat{\omega}$ provided by the KO. Moment of inertia J in $\boldsymbol{A}$ in (5) of KO can also be replaced by the identified inertia $\hat{J}$ provided by the RLS estimator. Since the impact of B can be ignored compared to J [19], B in (5) is treated as a constant in this paper.

To avoid divergence problem, the observed values $\hat{T}_L$ and $\hat{\omega}$ provided by the KO should be updated to RLS estimator during the steady-state of KO, which can be identified by $\boldsymbol{V}(k)\boldsymbol{V}^T(k)$.

From above, through judging the innovation $\boldsymbol{V}(k)$, which situation should KO and RLS update their parameters to each other is determined. As this is summed up in Table 1.

Table 1. The KO-RLS Algorithm.

KO-RLS Algorithm

```
$k \leftarrow 0$
Set original states for KO: KO $\leftarrow P(0), X(0), \hat{J}(0), B, H, Q(0), R$
Set original states for RLS: RLS $\leftarrow \sigma(0), \psi(0), \lambda(0), T_L(0)$
While (1)
    Read inputs: $\theta(k), i_q(k)$
    calculate $\hat{X}_{k+1}, P_{k+1}, V_{k+1}$ by KO in (7), (8), (9)
    calculate $\hat{J}(k+1), \sigma(k+1), \psi(k+1)$ by RLS in (12), (13)
    if $V_{k+1}V_{k+1}^T \leq e_{threshold}$ then
    $A(k+1) \leftarrow \hat{J}(k+1)$
    $\tau(k+1) \leftarrow \hat{\omega}(k+1), \hat{T}_L(k+1)$
output: $\hat{J}(k+1), \hat{T}_L(k+1), \hat{\omega}(k+1), \hat{\theta}(k+1)$
$k \leftarrow k$ +1
```

4. Proposed AKO-RLS Algorithm

4.1. Design of the Adaptive Algorithm in KO

In the traditional Kalman Observer, the covariance matrix of the system noise $\boldsymbol{Q}$ and covariance matrix of the measurement noise $\boldsymbol{R}$ are both treated as a constant matrix. A larger $\boldsymbol{Q}$ means that the prediction generated by the system model is less credible and the observer attains a faster transient response, but during steady-state, this results in ripples [23]. Similarly, a larger $\boldsymbol{R}$ means that the measurement results are relatively less credible.

In the past work, the moment of inertia J can be precisely achieved, also, the load torque doesn't vary with time, so an accurate system parameter of $\boldsymbol{A}$ in (5) can be achieved. The common way of choosing $\boldsymbol{Q}$ and $\boldsymbol{R}$ is that q_θ, q_ω and q_{T_L} in $\boldsymbol{Q}$ are set relatively small and value in $\boldsymbol{R}$ is set relatively large because the system model is more creditable. i.e., $\boldsymbol{Q}$ = *diag*(0.001, 0.001, 0.1) and $\boldsymbol{R}$ = [10].

However, it is difficult to obtain a relatively accurate priori value of J to implement the algorithm in some industrial applications. In this case, obviously, the observed load torque $\hat{T}_L$ provided by KO which is used in RLS is inaccurate, and the identified inertia $\hat{J}$ has a large error deviating from true value since they are variables that cannot be decoupled. Moreover, the load torque may vary in some applications. In these situations, a larger $\boldsymbol{Q}$ should be chosen because the system model is unprecise, and it can make convergence process faster. Along with the iterative computations of KO and RLS, a smaller $\boldsymbol{Q}$ should be chosen during steady-state to ensure the stability of the algorithm.

From above, it is inappropriate to treat $\boldsymbol{Q}$ as constant matrix when implement with inaccurate initial value or load torque variation exists.

Since the innovation $\boldsymbol{V}(k)$ in (9) contains the information of error between system model and measurement model, the adaptive algorithm can be designed as:

$$\begin{cases} \boldsymbol{V}(k)\boldsymbol{V}^T(k) \geq e_{threshold}, & \boldsymbol{Q}(k+1) = (1+\rho)\boldsymbol{Q}(k) \\ \boldsymbol{V}(k)\boldsymbol{V}^T(k) \leq e_{threshold}, & \boldsymbol{Q}(k+1) = (1-\rho)\boldsymbol{Q}(k) \end{cases} \tag{15}$$

where ρ is a scalar ranges that from 0 to 1.

In Equation (9), it can be seen that elements in the gain matrix $\boldsymbol{K}$ are directly proportional to the corresponding elements in $\boldsymbol{Q}$ and inversely proportional to $\boldsymbol{R}$. When the system is under dynamics, ρ can increase the values in $\boldsymbol{Q}$ to enhance the tracking ability. When the system is under steady-state, ρ can guarantee steady-state performance by decreasing the values in $\boldsymbol{Q}$.

4.2. Design of the Adaptive Algorithm in RLS

The inertia identification requires accurate information of load torque. However, the load torque provided by KO may be unprecise because of the the inaccuracy of the pre-set inertia in KO. Moreover, from (3), the load torque is treated as a constant, so if the load torque variation exists, the results of RLS estimator will deteriorate.

From (14), λ is the forgetting factor which gives exponentially less weight to older error samples. λ should become smaller to weaken the effect of unprecise results such as inaccuracy information of load torque, preventing them from contributing to the algorithm, so it is inappropriate to set λ as a constant

$\boldsymbol{e}(n)$ in (12) is the a priori error and computed at time $n-1$. Then, the a posteriori error can be expressed as:

$$\boldsymbol{\xi}(n) = \boldsymbol{d}(n) - \boldsymbol{\sigma}^T(n)\tau(n) = \boldsymbol{e}(n)\left[1 - \tau^T(n)\eta(n)\right] \tag{16}$$

The forgetting factor λ can be modified according to [26]:

$$E\{\boldsymbol{\xi}(n)\boldsymbol{e}(n)\} = E\left\{\boldsymbol{e}^2(n)\left[1 - \frac{\tau^T(n)\boldsymbol{\sigma}^T(n)\tau(n)}{\lambda(n) + \tau^T(n)\boldsymbol{\sigma}^T(n)\tau(n)}\right]\right\} \tag{17}$$

where $E\{\cdot\}$ denotes the expectations operator. $E\{\boldsymbol{\xi}(n)\boldsymbol{e}(n)\} = \tau_v^2$ is the power of distortion in the system.

The expression of the variable forgetting factor can obtained by using (17) as:

$$\lambda(n) = \frac{\chi(n)\tau_v^2(n)}{\tau_e^2(n) - \tau_v^2(n)} \tag{18}$$

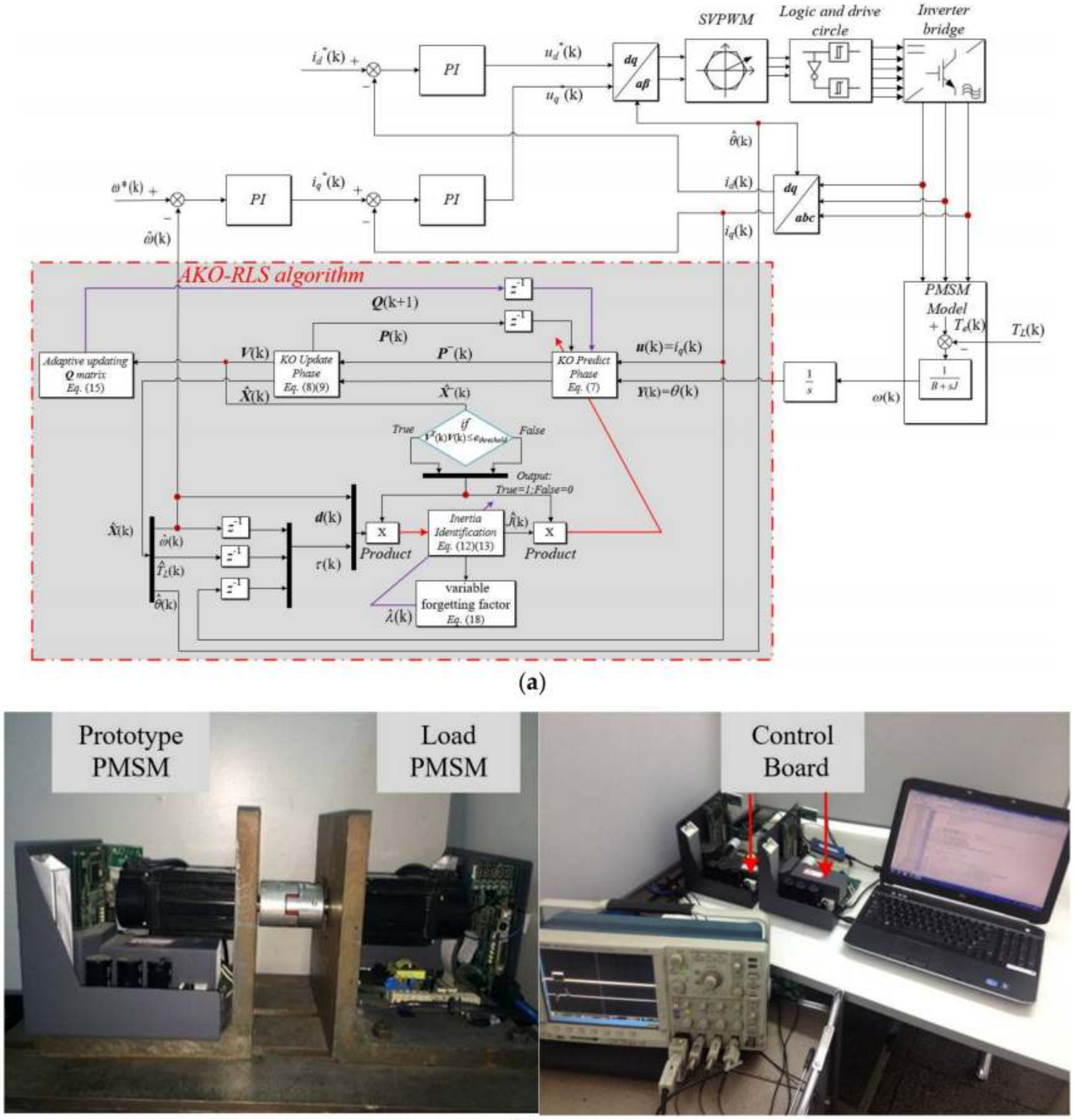
where $\chi(n) = \tau^T(n)\boldsymbol{\sigma}^T(n)\tau(n)$, $E\{\boldsymbol{e}^2(n)\} = \tau_e^2(n)$.

The AKO-RLS algorithm is summarized in Table 2 and the overall control scheme implemented with the proposed algorithm is shown in Figure 1.

Table 2. The AKO-RLS Algorithm.

```
AKO-RLS Integration Algorithm
$k \leftarrow 0$
Set original states for KO: KO $\leftarrow P(0), X(0), \hat{J}(0), B, H, Q(0), R$
Set original states for RLS: RLS $\leftarrow \boldsymbol{\sigma}(0), \boldsymbol{\psi}(0), \lambda(0), T_L(0)$
While (1)
    Read inputs: $\theta(k)$, $i_q(k)$
    calculate $\hat{X}_{k+1}$, $P_{k+1}$, $V_{k+1}$ by AKO in (7), (8), (9)
    calculate $\hat{J}(k+1)$, $\boldsymbol{\sigma}(k+1)$, $\boldsymbol{\psi}(k+1)$ by RLS in (12), (13)
    calculate $\chi(n+1)$, $\tau_v^2(n+1)$, $\tau_e^2(n+1)$, $\lambda(n+1)$ in (18)
    if $V_{k+1}V_{k+1}^T \le e_{threshold}$ then
    $Q_{k+1} = (1-\rho)Q_k$
    $A(k+1) \leftarrow \hat{J}(k+1)$
    $\tau(k+1) \leftarrow \hat{\omega}(k+1)$, $\hat{T}_L(k+1)$
else
$Q_{k+1} = (1+\rho)Q_k$
output: $\hat{J}(k+1)$, $\hat{T}_L(k+1)$, $\hat{\omega}(k+1)$, $\hat{\theta}(k+1)$
$k \leftarrow k$ +1
```

In Figure 1a, the rotor position signal θ provided by the encoders is not implemented in the control scheme. It just serves as the input of the proposed AKO to generate $\hat{\theta}$ and $\hat{\omega}$. They are more precise than measurements and they can improve the performance of the servo system, which is demonstrated in [13]. However, this only works well during steady-state. $\hat{\theta}$ and $\hat{\omega}$ cannot be implemented during transient conditions because they are unprecise. From above, in this paper, θ and ω should be implemented during transient, $\hat{\theta}$ and $\hat{\omega}$ should be implemented during steady-state.



(a)

(b)

Figure 1. Overall Schematic diagrams and photographs of hardware platform for system parameter estimation. (**a**) Structure diagram of the integral PMSM drive system with AKO-RLS algorithm; (**b**) photographs of experiment platform.

5. Sensitivity Analysis of the Proposed Algorithm

From (7) and (8), the description of $\hat{T}_L$ can be obtained:

$$\hat{T}_L(n) = \hat{T}_L(n-1) + K(n) \times \left(\theta(n) - \hat{\theta}(n)\right) \tag{19}$$

where $K(n)$ denotes the gain of $\hat{T}_L$ in $K(n)$.

Similar to (10), applying a Z transformation to (1) and (2), θ can be described in the z plane with zero-order hold:

$$\begin{aligned}\theta(z) &= Z\left\{\frac{1-e^{sT_S}}{s}\frac{1}{Bs+s^2J}\right\}\left(K_T i_q(z)-T_L(z)\right)\\ &= \left(\frac{1}{B}\frac{T_s z^{-1}}{1-z^{-1}}-\frac{J}{B}\frac{b_1 z^{-1}}{1+a_1 z^{-1}}\right)\left(K_T i_q(z)-T_L(z)\right)\end{aligned} \tag{20}$$

where a_1 and b_1 are identical to those in (10).

Applying a Z transformation to (19), and substituting $\theta(z)$ with (20) into (19), yields:

$$\frac{\hat{T}_L(z)}{T_L(z)} = \frac{K(z)z\left(\frac{T_s}{B(z-1)}-\frac{Jb_1}{B(a_1+z)}\right)}{K(z)z\left(\frac{T_s}{B(z-1)}-\frac{Jb_1}{B(a_1+z)}\right)-z+1} \tag{21}$$

When T is sufficiently small, $z \approx 1 + sT$. Then the continuous transfer function is obtained as follows:

$$\frac{\hat{T}_L(s)}{T_L(s)} = \frac{-K(s)\left(\frac{1}{Bs}-\frac{Jb_1}{B(a_1+1+sT_S)}\right)(1+sT_S)}{sT_S-K(s)\left(\frac{1}{Bs}-\frac{Jb_1}{B(a_1+1+sT_S)}\right)(1+sT_S)} \tag{22}$$

All motor parameters agree with those in Table 3. All parameters of the AKO are identical to those in the simulation section.

Table 3. PMSM Specifications.

Parameter	Quantity
Rated Power	750 W
Rated Torque	2.39 Nm
Rated Speed	3000 r/min
Rated current	4.8 A
Pole-pairs number	4
Moment of inertia J with Loading motor	5.2×10^{-4} kg·m^2

The amplitude characteristics with different K values are plotted in the Bode diagram in Figure 2, according to (28).

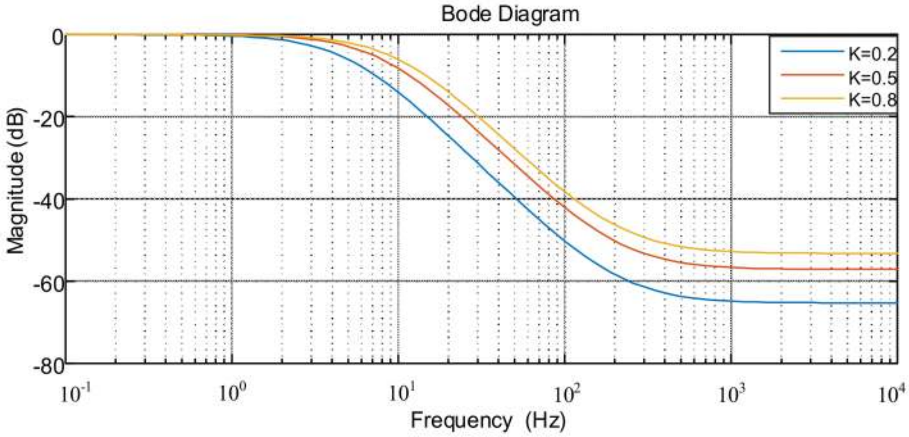
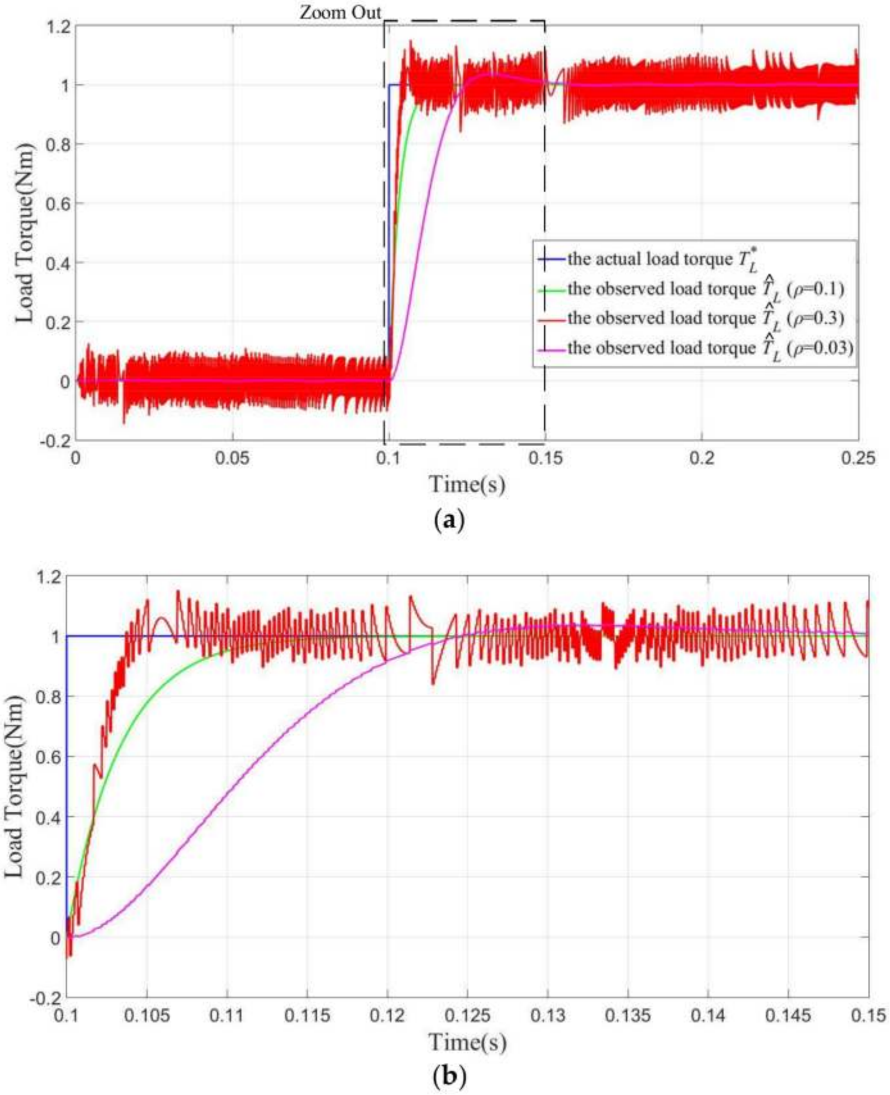


Figure 2. Frequency characteristic with different K (input: T_L, output: $\hat{T}_L$).

The ideal transfer function is $\frac{\hat{T}_L(s)}{T_L(s)} = 1$. Figure 2 shows that since the load disturbance is low-frequency or constant in most situations, the tracking ability can be enhanced when the K can make the low-frequency gain bigger. Since K is a variable of $\boldsymbol{K}(n)$ which is related to $\boldsymbol{Q}$ and $\boldsymbol{R}$. From (7) and (9), a larger $\boldsymbol{Q}$ can result in a larger K. When the load torque varies, from the adaptive algorithm in Table 2, $\boldsymbol{Q}$ is increased and results in an increased K to trace the load torque and attain better transient performance.

Figure 3 shows the observed load torque $\hat{T}_L$ implemented with different ρ. A 1.2 Nm load torque suddenly increased at 0.1 s. All other parameters of AKO-RLS are set the same as those in Section 4. $e_{threshold}$ and ρ are hyper-parameters of the adaptive algorithm. A larger ρ in (15) can enhance the tracking ability of the KO, which are in accordance with the results of Figure 2. However, if ρ is set too large, it will result in shaking during steady-state. This can be explained by (15); if ρ is very large, $\boldsymbol{Q}$ will change dramatically, it will fail to obtain the optimal values of $\boldsymbol{Q}$. $e_{threshold}$ is less important, it can be regarded as an empirical value. It's appropriate to set it within 1×10^{-4}–1×10^{-7}.



(**a**)

(**b**)

Figure 3. Simulation Comparison of $\hat{T}_L$ implemented with different values ρ (**a**) Comparison of $\hat{T}_L$ with different values ρ (**b**) Large chart of (**a**).

According to [26], the variable forgetting factor is obtained by minimizing $E\{e^2(n)\} = \tau_e^2(n)$ using the gradient-based method. When $\boldsymbol{V}(k)\boldsymbol{V}^T(k)$ in Table 2 are large, $\hat{T}_L$ and $\hat{\omega}$ provided by KO, which are used as the inputs of RLS, subsequently causing a larger $e(n)$ in (12), and in this situation, from (18), λ becomes smaller to give less weight to these unprecise historic data, which can speed up the process of the inertia identification and adapt to the load torque variation.

6. Simulation and Experiment

6.1. Simulation Analysis

To verify the validation of the AKO-RLS algorithm, simulation results are presented in this section. The main parameters of PMSM used in this paper are shown in Table 3. The PI gains of the speed loop are designed just follow Equations (40) and (41) in [19]. PI gains of the current loop are designed similar to those of the speed loop according to pole zero cancellation. From Sections 2–5, it can be seen that all parameters and equations in the paper are mechanical parameters and equations. We assume the bandwidth of the current loop is infinite. It cannot be true in the industrial fields. However, as shown in Section 6.2, the bandwidth of the current loop of our experiment platform is 500 Hz. In robotic fields, the load torque changes much slower than 500 Hz, i_q^* can easily follow the true time-varying load torque. Even with the delay of the current loop, it has little influence on the proposed algorithm.

The covariance of system noise $\boldsymbol{Q}$ and the covariance of measurement noise $\boldsymbol{R}$ are the weight factors of the prediction model and measurement model, respectively. According to the method to select weight factors in [27], since the rotor position θ is the only measured signal and relatively high uncertainty of T_L, in this paper, parameters of the AKO-RLS are chosen as $\boldsymbol{Q}(0) = diag(0.001, 0.01, 0.1)$, $\boldsymbol{R} = [0.001]$, $e_{threshold} = 1 \times 10^{-4}$, $\boldsymbol{P}(0)$ is set as a unit matrix. From Section 5 and Figure 3, ρ in (15) is set as 0.1. For simplicity, all the initial values in (12) are set as 0, $\boldsymbol{\psi}(n)$ is set as a unit matrix. The initial value of forgetting factor $\lambda(0)$ is chosen as 0.99.

In traditional KO-RLS methods, its main parameters are chosen as $\boldsymbol{Q} = diag(0.001, 0.01, 1)$, $\boldsymbol{R} = [1]$, $\lambda = 0.99$. All other parameters are identical to the AKO-RLS. Computational period of AKO-RLS and KO-RLS are both 0.1 ms.

The actual inertia of the drive system is not time-varying during the simulation and experiment. Also, because the inaccurate information of the inertia and load torque present (which is necessary to verify the validation of the proposed algorithm), so under such condition, the observed $\hat{\theta}$ and $\hat{\omega}$ cannot be implemented during transient conditions. All speed response in the simulation and experiment results during transient are the ω calculated by the θ.

In Figure 4, simulation contrastive results of AKO-RLS and KO-RLS are illustrated. The steady state estimation errors of in AKO-RLS method and KO-RLS method are 1.2% and 5.6%, respectively. $\hat{J}(0)$ in $\hat{\boldsymbol{A}}$ in Table 2 is set as $5J$ (J denotes the actual total inertia of the interconnect loading system). From (1), the inertia identification requires changing speed. Since the speed command is 0–1000 rpm repeated step command. Thus, the identified inertia resembles a staircase wave. The results of the identified inertia $\hat{J}$ converge toward the true value when the speed changes. The identified inertia $\hat{J}$ estimated by AKO-RLS converges faster than KO-RLS and during steady-state, it has a smaller error. Also, the load torque observed by AKO-RLS shows no fluctuation during steady-state conditions.

In Figure 5, contrastive results of AKO-RLS and KO-RLS are illustrated. Load torque in sine form (the value of peak-peak is 0.6 Nm and the period is 2 s, with a 0.2 Nm offset) is being added. Besides, speed reference in triangle-wave (the range is from 300 rpm to 2800 rpm, the period is 0.599 s approximately) is chosen. A repeated speed step command is not appropriate because the load torque varies from time to time.

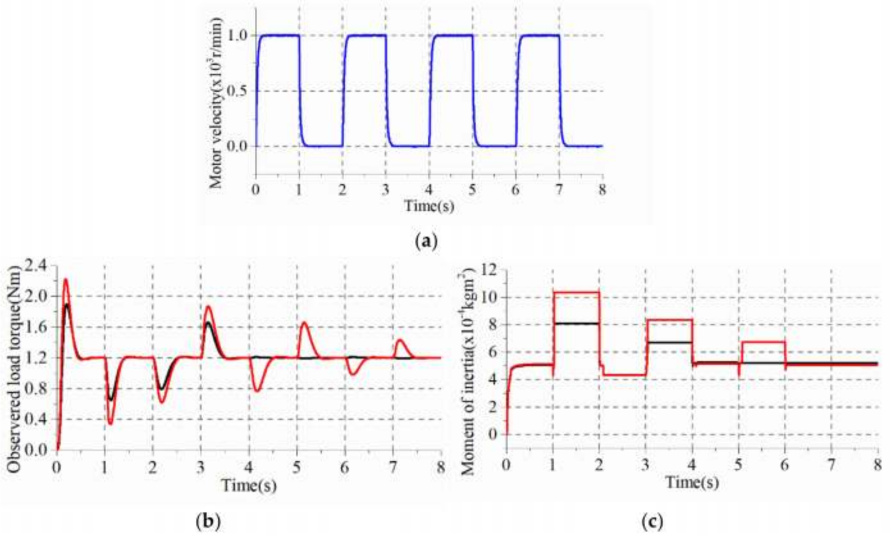


Figure 4. Simulation Comparison of the two algorithms with initial value $\hat{J}(0) = 5J$ under 1.2 Nm load torque and repeated speed command. (**a**) Speed response (blue); (**b**) Observed load torque $\hat{T}_L$ (black line denotes AKO-RLS result, red line denotes KO-RLS result); (**c**) identified inertia $\hat{J}$ (black line denotes AKO-RLS result, red line denotes KO-RLS result).

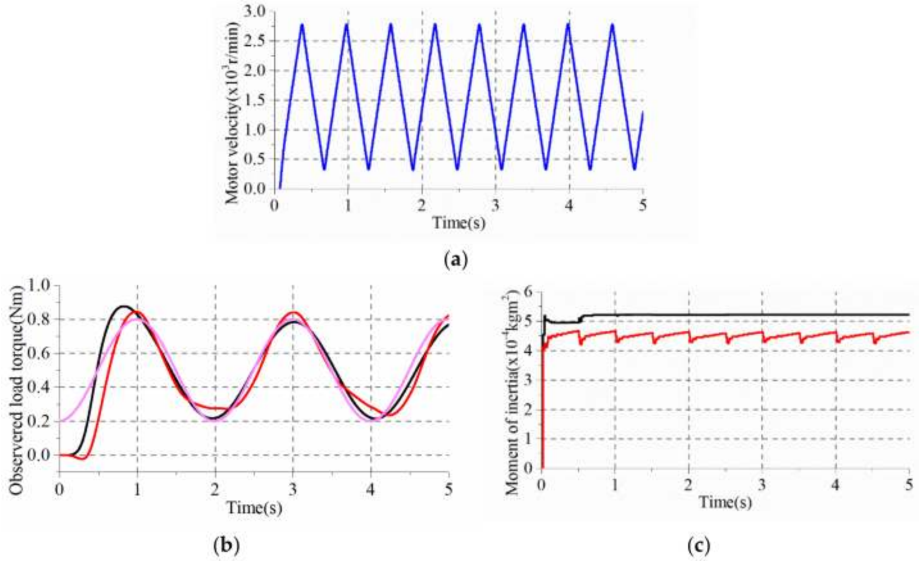


Figure 5. Simulation Comparison of the two algorithms under sine-wave load torque and triangle-wave speed command. (**a**) Speed response (blue); (**b**) Observed load torque $\hat{T}_L$ (pink line denotes the true load torque, black line denotes AKO-RLS result, red line denotes KO-RLS result); (**c**) identified inertia $\hat{J}$ (black line denotes AKO-RLS result, red line denotes KO-RLS result).

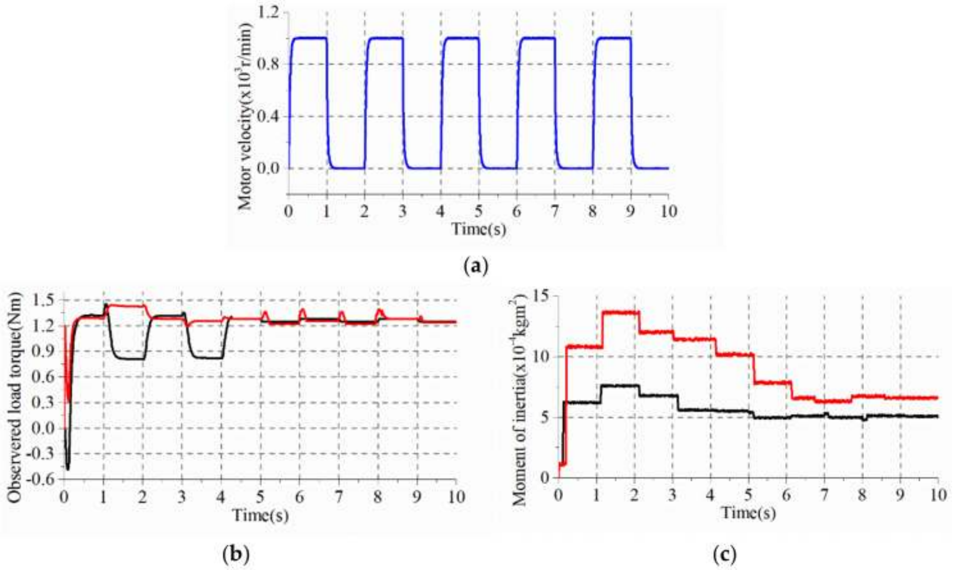
A time-varied speed command is necessary to ensure the identified inertia converges faster. AKO-RLS's identified inertia needs about 0.5 s to converge to the steady-state. However, KO-RLS's identified inertia cannot converge under sine-wave load torque. Steady state estimation errors of in AKO-RLS method and KO-RLS method are 3.8% and 11.6%, respectively. The load torque provided by the KO-RLS method has big ripples, besides, the inertia provided by KO-RLS method has a relatively large error than that of AKO-RLS and it fluctuates during speed dynamic processes, because the load torque and the moment of inertia are strongly coupled variables. Both the observed load torque shows a little lag compared to the true load torque. However, it has little impact on the results of the identified inertia.

6.2. Experimental Analysis

Figure 1b shows the experiment platform adopted in this paper. All control algorithms were implemented in a TI DSP TMS320F28335 microcontroller. The current loop bandwidth of the experiment servo motor is 500 Hz. The current loop bandwidth of the experiment servo motor is relatively low, but it does not deteriorate to the validation of the algorithm.

The parameters of the KO-RLS and AKO-RLS algorithm are same as those in the simulation. The actual inertia of the drive system is not time-varying during the experiment.

The initial value of identified inertia $\hat{J}(0)$ in $\hat{\boldsymbol{A}}$ in Table 2 is set as $0.2J$ or $5J$ (J is the actual total inertia of the interconnect loading system). Since the inertia of the experiment motor system is relatively small, the choices of initial values $\hat{J}(0) = 0.2J$ and $\hat{J}(0) = 5J$ will greatly affect the control performance, thus it is adopted to prove the validity of the proposed algorithm. Figures 6 and 7 show the experimental comparison between the AKO-RLS and KO-RLS with initial values $\hat{J}(0) = 0.2J$ and $\hat{J}(0) = 5J$ under 1.2 Nm load torque, respectively. The speed command is a repeated 0–1000 r/min change. In Figures 6 and 7, it can be seen that both KO-RLS and AKO-RLS's identified inertia need six speed change instances to converge to the steady-state. The inertia identification needs time-varying speed. The identified inertia converges faster during the speed variation. Thus, the converge time is mainly determined by the rise time and fall time of the speed. The identified inertia in Figures 5 and 8 converges faster than it in Figures 6 and 7, because the speed command in Figures 5 and 8 is a triangle wave and speed changes all the time. The results are summarized in Table 4. There are patent differences between two algorithms. During steady-state, the load torque observed by AKO-RLS shows no fluctuation when speed changes and identified inertia $\hat{J}$ provided by AKO-RLS is more precise. The observed load torque $\hat{T}_L$ of two algorithms all have relatively larger errors, which might be caused by the fact that knowledge of the viscous coefficient $\boldsymbol{B}$ in real systems cannot be accurately obtained and it deteriorates the performance of the observer.



(**a**)

(**b**) (**c**)

Figure 6. Comparison of two algorithms with initial value $\hat{J}(0) = 0.2J$ and 1.2 Nm load torque, 0–1000 rpm repeated step speed command. (**a**) Speed response (blue); (**b**) Observed load torque $\hat{T}_L$ (black line denotes AKO-RLS result, red line denotes KO-RLS result); (**c**) identified inertia $\hat{J}$ (black line denotes AKO-RLS result, red line denotes KO-RLS result).

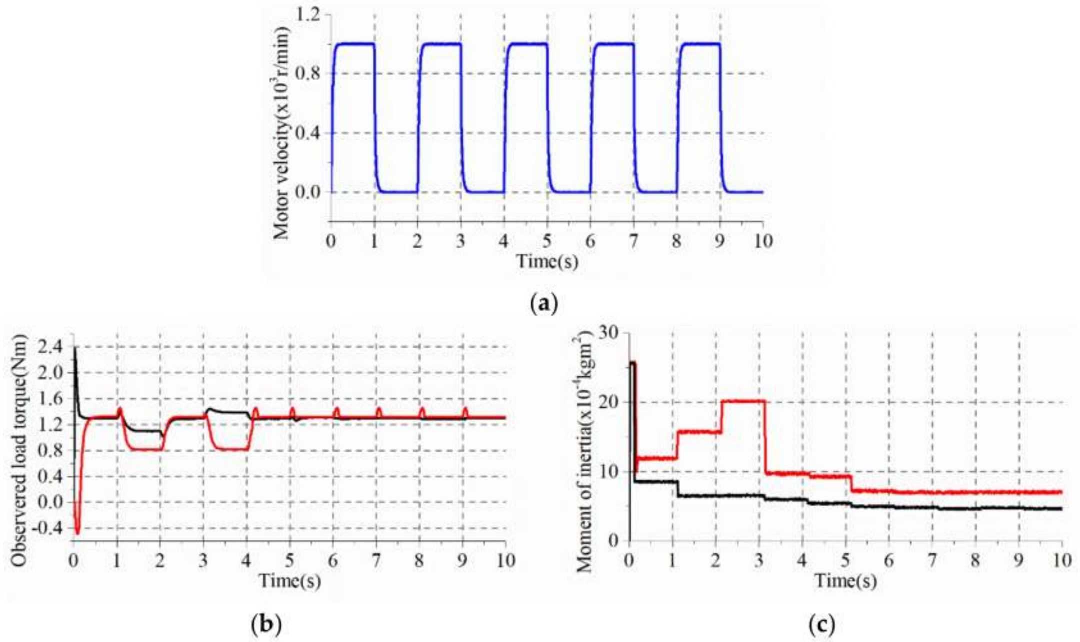


Figure 7. Comparison of two algorithms with initial value $\hat{J}(0) = 5J$ and 1.2 Nm load torque, 0–1000 rpm repeated step speed command. (**a**) Speed response (blue); (**b**) Observed load torque $\hat{T}_L$ (black line denotes AKO-RLS result, red line denotes KO-RLS result); (**c**) identified inertia $\hat{J}$ (black line denotes AKO-RLS result, red line denotes KO-RLS result).

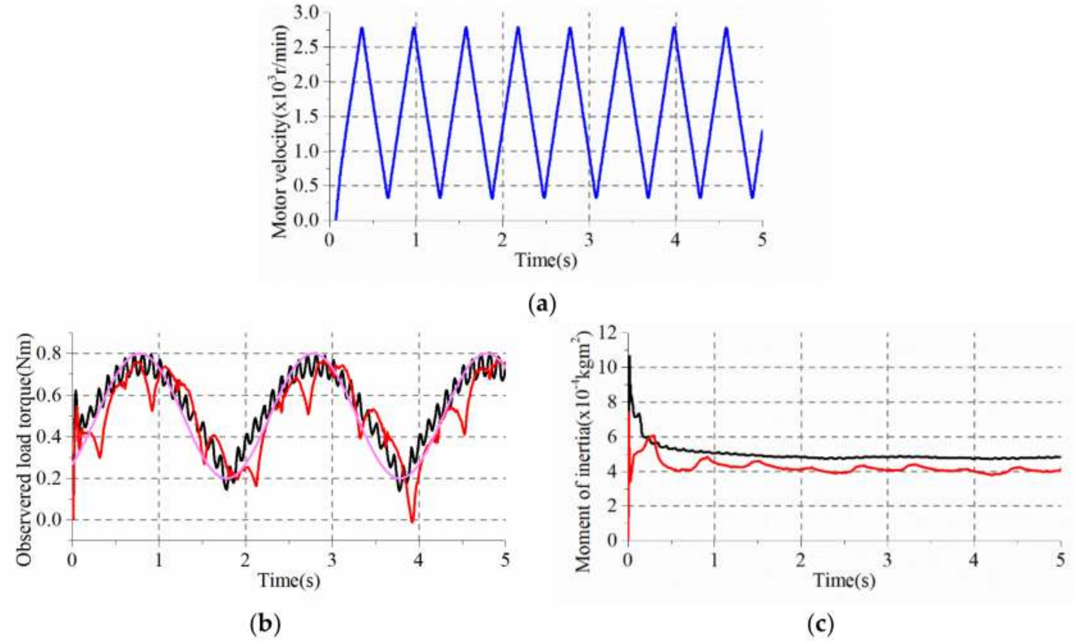
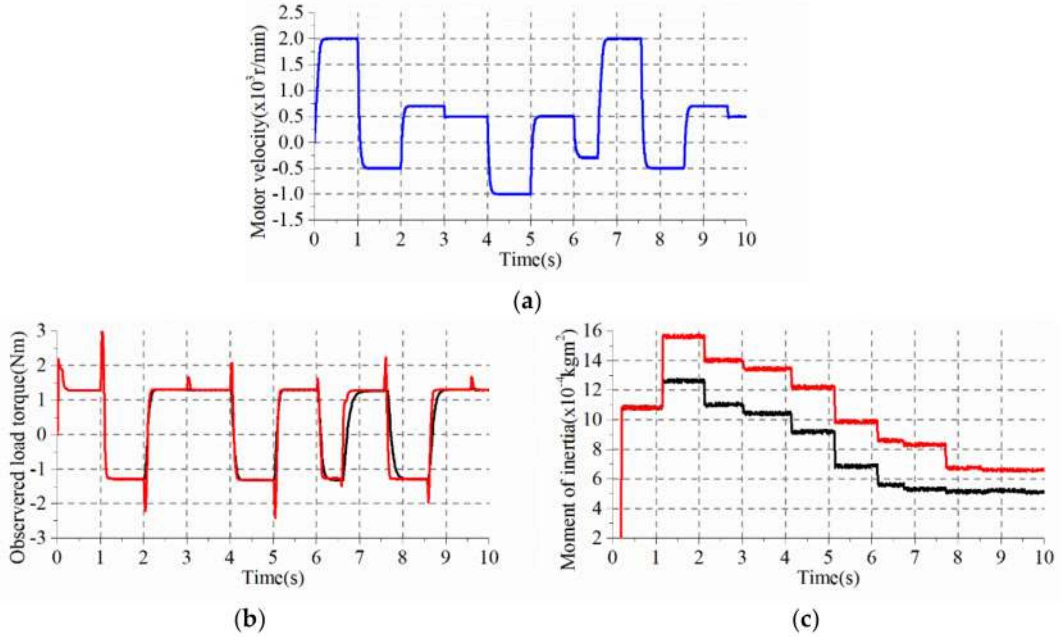


Figure 8. Comparison of two algorithms under sine-wave load torque and triangle-wave speed command. (**a**) Speed response (blue); (**b**) Observed load torque $\hat{T}_L$ (pink line denotes the true load torque, black line denotes AKO-RLS result, red line denotes KO-RLS result); (**c**) identified inertia $\hat{J}$ (black line denotes AKO-RLS result, red line denotes KO-RLS result).

Figures 9 and 10 show the contrastive results between AKO-RLS and KO-RLS with initial values $\hat{J}(0) = 0.2J$ and $\hat{J}(0) = 5J$, respectively. Under 1.2 Nm load torque condition and a random step speed command. In Figure 9, AKO-RLS's identified inertia needs six speed changes to converge to the steady-state. But KO-RLS's identified inertia needs about 10 speed changes. In Figure 10, both KO-RLS and AKO-RLS's identified inertia need six speed changes to converge to the steady-state. It can also be explained by the inertia identification needs time-varying speed. The torque limitation of the loading servo is set as ±1.2 Nm. The waveform of the observed load torque in Figures 9b and 10b is like a square wave. This is not an error. It is caused by the PI controller of the load PMSM. The speed error of the loading servo is negative when the speed of the driving motor is positive and load torque provided by the loading servo is 1.2 Nm. When speed changes from positive to negative, the accumulator of the integrator of the PI controller desaturates. The load torque provided by the loading servo changes from 1.2 Nm to −1.2 Nm. In transient process, $\hat{T}_L$ of KO-RLS fluctuates when the speed changes, while AKO-RLS 'doesn't overshoot. During steady-state, the estimated errors of $\hat{J}$ in steady state of AKO-RLS are much smaller than KO-RLS. The results are summed up in Table 5.



(a)

(b)

(c)

Figure 9. Comparison of two algorithms with initial value $\hat{J}(0) = 0.2J$ and 1.2 Nm load torque, random step speed command. (**a**) Speed response (blue); (**b**) Observed load torque $\hat{T}_L$ (black line denotes AKO-RLS result, red line denotes KO-RLS result); (**c**) identified moment of inertia $\hat{J}$ (black line denotes AKO-RLS result, red line denotes KO-RLS result).

In Figure 8, contrastive results of AKO-RLS and KO-RLS are illustrated. Load torque in sine form (the value of peak—peak is 0.6 Nm and the period is 2 s, with a 0.2 Nm offset) is added. Besides, speed reference in triangle-wave (the range is from 300 rpm to 2800 rpm, the period is 0.599 s approximately) is chosen. The frequency of the time-varying load torque is much slower than the current loop bandwidth of the experiment platform. The influence of the current loop can be ignored. The process of AKO-RLS inertia identification needs about 1 s to converge to steady-state. It's faster than the results in Figures 6, 7, 9 and 10 because the speed command in Figure 8 is a triangle wave. However, KO-RLS's identified inertia cannot converge under sine-wave load torque. Steady state estimation errors in the AKO-RLS method and KO-RLS method are 3.8% and 11.6%, respectively. Also, it can be seen that dynamic responses between the two are quite different, and share a smaller distortion when under AKO-RLS method, but it has high frequency ripple, and this is caused by the inherent system delay. The observed load torque shows a little lag compared to the true load torque, but it has little impact on the identified inertia. Considering the current loop period is 0.1 ms and the speed loop period is 1 ms, so the mechanical time constant is much larger than the electrical constant, this will lead to fluctuation during speed variation, but, it can be seen that the inertia identification is still good, even with this ripple.

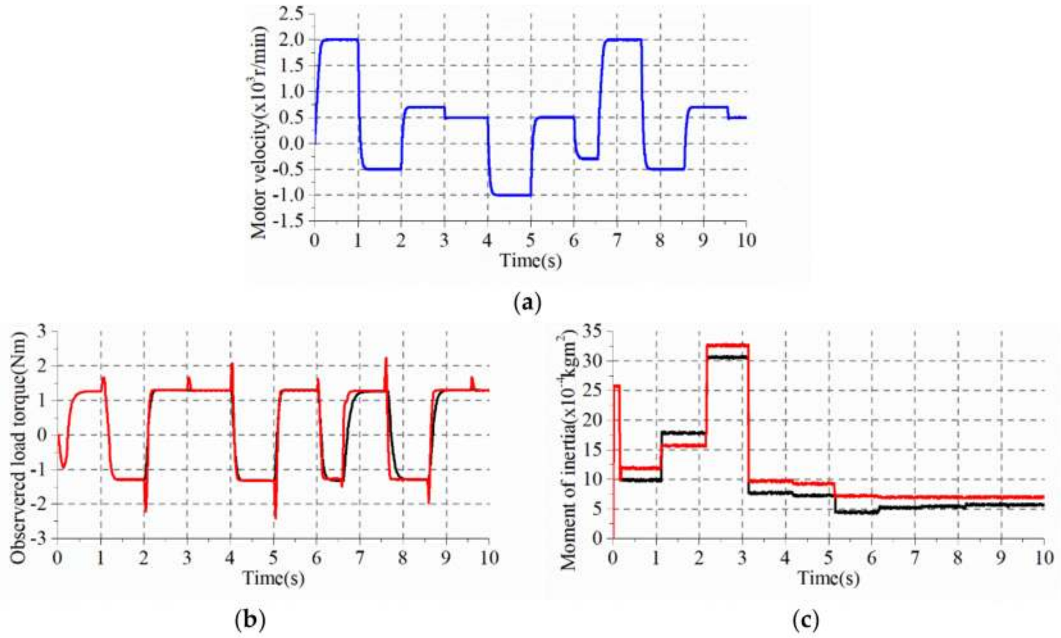


Figure 10. Comparison of two algorithms with initial value $\hat{J}(0) = 5J$ and 1.2 Nm load torque, random step speed command. (**a**) Speed response (blue); (**b**) Observed load torque $\hat{T}_L$ (black line denotes AKO-RLS result, red line denotes KO-RLS result); (**c**) identified moment of inertia $\hat{J}$ (black line denotes AKO-RLS result, red line denotes KO-RLS result).

Table 4. Experiment Identification Results (under 1.2 Nm and repeated speed step command).

Values under Different Methods		**Speed Command**	**Final Values (Steady State)**	
			$\hat{\boldsymbol{J}}(\mathbf{0}) = \mathbf{0.2}\boldsymbol{J}$	$\hat{\boldsymbol{J}}(\mathbf{0}) = \mathbf{5}\boldsymbol{J}$
AKO-RLS	$\lvert T_L - \hat{T}_L \rvert / T_L$	repeated	7.8%	8.2%
	$\lvert J - \hat{J} \rvert / J$	speed	4.5%	3.9%
KO-RLS	$\lvert T_L - \hat{T}_L \rvert / T_L$	step	8.7%	8.9%
	$\lvert J - \hat{J} \rvert / J$	command	19.2%	12.8%

Table 5. Experiment Identification Results (under 1.2 Nm and random speed step command).

Values under Different Methods		**Speed Command**	**Final Values (Steady State)**	
			$\hat{\boldsymbol{J}}(\mathbf{0}) = \mathbf{0.2}\boldsymbol{J}$	$\hat{\boldsymbol{J}}(\mathbf{0}) = \mathbf{5}\boldsymbol{J}$
AKO-RLS	$\lvert T_L - \hat{T}_L \rvert / T_L$	Random	8.5%	8.6%
	$\lvert J - \hat{J} \rvert / J$	speed	4.5%	5.4%
KO-RLS	$\lvert T_L - \hat{T}_L \rvert / T_L$	step	8.6%	8.8%
	$\lvert J - \hat{J} \rvert / J$	command	20.8%	12.8%

7. Conclusions

Load torque and moment of inertia are coupled variables, so the results of identified inertia will deteriorate when the load torque varies or implemented with unprecise initial values. In this paper, a more robust and adaptive algorithm for inertia identification called AKO-RLS is proposed. Compared with the traditional KO-RLS method, the proposed algorithm can work well even when implemented with an initial value 0.2 or 5 times the true inertia value. Also, the impact of a frequency 0.5 Hz, peak-peak value 0.6 Nm sine-wave load torque can be eliminated and the AKO-RLS can have a relatively good performance under such load torque variation. The identification results of the proposed AKO-RLS algorithm are more accurate and robust, which is validated by both simulations and experiments. The proposed algorithm can let the inertia identification adapt to the most industrial field cases. Further work will focus on industrial robotics applications, a field where moment of inertia and load torque are both time-varying [28].

Acknowledgments: This work is supported by Project of Key Scientific and Technological Achievements Commercialization of Hu Nan Province Strategic Emerging Industries (NO. 2016GK4049) and National Key Research and Development Program of China (2017YFB1300801).

Author Contributions: Zirui Liu and Ming Yang conceived and designed the experiments; Zirui Liu performed the experiments; Zirui Liu and Ming Yang analyzed the data; Ming Yang contributed materials and experiment platform; Zirui Liu, Wanying Qu, Long Jiang wrote the paper.

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

References

1. Xia, C.; Wang, S.; Gu, X.; Yan, Y.; Shi, T. Direct Torque Control for VSI-PMSM Using Vector Evaluation Factor Table. *IEEE Trans. Ind. Electron.* **2016**, *63*, 4571–4583. [CrossRef]
2. Lin, F.J.; Hung, Y.C.; Tsai, M.T. Fault-Tolerant Control for Six-Phase PMSM Drive System via Intelligent Complementary Sliding-Mode Control Using TSKFNN-AMF. *IEEE Trans. Ind. Electron.* **2013**, *60*, 5747–5762. [CrossRef]
3. Kommuri, S.K.; Defoort, M.; Karimi, H.R.; Veluvolu, K.C. A Robust Observer-Based Sensor Fault-Tolerant Control for PMSM in Electric Vehicles. *IEEE Trans. Ind. Electron.* **2016**, *63*, 7671–7681. [CrossRef]
4. Ahmed, A.; Sozer, Y.; Hamdan, M. Maximum Torque per Ampere Control for Buried Magnet PMSM Based on DC-Link Power Measurement. *IEEE Trans. Ind. Electron.* **2017**, *32*, 1299–1311. [CrossRef]
5. Lee, K.; Ahmed, S.; Lukic, S.M. Universal Restart Strategy for High-Inertia Scalar-Controlled PMSM Drives. *IEEE Trans. Ind. Appl.* **2016**, *52*, 4001–4009. [CrossRef]
6. Liu, K.; Zhu, Z. Fast Determination of Moment of Inertia of Permanent Magnet Synchronous Machine Drives for Design of Speed Loop Regulator. *IEEE Trans. Control Syst. Technol.* **2017**, *25*, 1816–1824. [CrossRef]
7. Zhang, G. Speed control of two-inertia system by PI/PID control. *IEEE Trans. Ind. Electron.* **2000**, *47*, 603–609. [CrossRef]
8. She, J.H.; Fang, M.; Ohyama, Y.; Hashimoto, H.; Wu, M. Improving Disturbance-Rejection Performance Based on an Equivalent-Input-Disturbance Approach. *IEEE Trans. Ind. Electron.* **2008**, *55*, 380–389. [CrossRef]
9. O'Sullivan, T.M.; Bingham, C.M.; Schofield, N. Observer-Based Tuning of Two-Inertia Servo-Drive Systems with Integrated SAW Torque Transducers. *IEEE Trans. Ind. Electron.* **2007**, *54*, 1080–1091. [CrossRef]
10. Andoh, F. Moment of Inertia Identification Using the Time Average of the Product of Torque Reference Input and Motor Position. *IEEE Trans. Ind. Electron.* **2007**, *22*, 2534–2542. [CrossRef]
11. Babau, R.; Boldea, I.; Miller, T.J.E.; Muntean, N. Complete Parameter Identification of Large Induction Machines from No-Load Acceleration–Deceleration Tests. *IEEE Trans. Ind. Electron.* **2007**, *54*, 1962–1972. [CrossRef]
12. Kim, N.-J.; Moon, H.-S.; Hyun, D.-S. Inertia identification for the speed observer of the low speed control of induction machines. *IEEE Trans. Ind. Appl.* **1996**, *32*, 1371–1379.
13. Shi, T.; Wang, Z.; Xia, C. Speed Measurement Error Suppression for PMSM Control System Using Self-Adaption Kalman Observer. *IEEE Trans. Ind. Electron.* **2015**, *62*, 2753–2763. [CrossRef]
14. Xiaoquan, L.; Heyun, L.; Junlin, H. Load disturbance observer-based control method for sensorless PMSM drive. *IET Electr. Power Appl.* **2016**, *10*, 735–743. [CrossRef]

15. Ouhrouche, M.; Errouissi, R.; Trzynadlowski, A.M.; Tehrani, K.A.; Benzaioua, A. A Novel Predictive Direct Torque Controller for Induction Motor Drives. *IEEE Trans. Ind. Electron.* **2016**, *63*, 5221–5230. [CrossRef]
16. Huang, W.S.; Liu, C.W.; Hsu, P.L.; Yeh, S.S. Precision Control and Compensation of Servomotors and Machine Tools via the Disturbance Observer. *IEEE Trans. Ind. Electron.* **2010**, *57*, 420–429. [CrossRef]
17. Choi, J.W.; Lee, S.C.; Kim, H.G. Inertia identification algorithm for high-performance speed control of electric motors. *IEE Proc. Electr. Power Appl.* **2006**, *153*, 379–386. [CrossRef]
18. Li, S.; Liu, Z. Adaptive Speed Control for Permanent-Magnet Synchronous Motor System with Variations of Load Inertia. *IEEE Trans. Ind. Electron.* **2009**, *56*, 3050–3059.
19. Niu, L.; Xu, D.; Yang, M.; Gui, X.; Liu, Z. On-line Inertia Identification Algorithm for PI Parameters Optimization in Speed Loop. *IEEE Trans. Power Electron.* **2015**, *30*, 849–859. [CrossRef]
20. Yin, Z.G.; Zhao, C.; Zhong, Y.R.; Liu, J. Research on Robust Performance of Speed-Sensorless Vector Control for the Induction Motor Using an Interfacing Multiple-Model Extended Kalman Filter. *IEEE Trans. Power Electron.* **2014**, *29*, 3011–3019. [CrossRef]
21. Alonge, F.; Cirrincione, M.; D'Ippolito, F.; Pucci, M.; Sferlazza, A.; Vitale, G. Descriptor-Type Kalman Filter and TLS EXIN Speed Estimate for Sensorless Control of a Linear Induction Motor. *IEEE Trans. Ind. Appl.* **2014**, *50*, 3754–3766. [CrossRef]
22. Antonello, R.; Ito, K.; Oboe, R. Acceleration Measurement Drift Rejection in Motion Control Systems by Augmented-State Kinematic Kalman Filter. *IEEE Trans. Ind. Electron.* **2016**, *63*, 1953–1961. [CrossRef]
23. Kweon, T.-J.; Hyun, D.-S. High-performance speed control of electric machine using low-precision shaft encoder. *IEEE Trans. Power Electron.* **1999**, *14*, 838–849. [CrossRef]
24. Lin, F.J. Robust speed-controlled induction-motor drive using EKF and RLS estimators. *IEE Proc. Electr. Power Appl.* **1996**, *143*, 186–192. [CrossRef]
25. Sayed, A.H. *Fundamentals of Adaptive Filtering*; Wiley: New York, NY, USA, 2003.
26. Leung, S.-H.; So, C.F. Gradient-based variable forgetting factor RLS algorithm in time-varying environments. *IEEE Trans. Signal Process.* **2005**, *53*, 3141–3150. [CrossRef]
27. Zarei, J.; Shokri, E. Convergence analysis of non-linear filtering based on cubature Kalman filter. *IET Sci. Meas. Technol.* **2015**, *9*, 294–305. [CrossRef]
28. Cortesao, R.; Koeppe, R.; Nunes, U.; Hirzinger, G. Compliant motion control with stochastic active observers. In Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems, Maui, HI, USA, 29 October–3 November 2001; Volume 4, pp. 1876–1881.