

An approximate high gain observer for speed-sensorless estimation of induction motors

Wang, Y.; Zhou, L.; Bortoff, S.A.; Satake, A.; Furutani, S.

TR2018-155 December 07, 2018

Abstract

Rotor speed estimation for induction motors is a key problem in speed-sensorless motor drives. This paper performs nonlinear high gain observer design based on the full-order model of the induction motor. Such an effort appears nontrivial due to the fact that the full-model at best admits locally a non-triangular observable form (NTOF), and its analytical representation in the NTOF can not be obtained. This paper proposes an approximate high gain estimation algorithm, which enjoys a constructive design, ease of tuning, and improved speed estimation and tracking performance. Experiments demonstrate the effectiveness of the proposed algorithm

IEEE/CAA Journal of Automatica Sinica

This work may not be copied or reproduced in whole or in part for any commercial purpose. Permission to copy in whole or in part without payment of fee is granted for nonprofit educational and research purposes provided that all such whole or partial copies include the following: a notice that such copying is by permission of Mitsubishi Electric Research Laboratories, Inc.; an acknowledgment of the authors and individual contributions to the work; and all applicable portions of the copyright notice. Copying, reproduction, or republishing for any other purpose shall require a license with payment of fee to Mitsubishi Electric Research Laboratories, Inc. All rights reserved.

An approximate high gain observer for speed-sensorless estimation of induction motors

Abstract—Rotor speed estimation for induction motors is a key problem in speed-sensorless motor drives. This paper performs nonlinear high gain observer design based on the full-order model of the induction motor. Such an effort appears non-trivial due to the fact that the full-model at best admits locally a non-triangular observable form (NTOF), and its analytical representation in the NTOF can not be obtained. This paper proposes an approximate high gain estimation algorithm, which enjoys a constructive design, ease of tuning, and improved speed estimation and tracking performance. Experiments demonstrate the effectiveness of the proposed algorithm.

Index Terms—Induction motor, speed-sensorless motor drive, nonlinear state estimation, industrial applications.

I. INTRODUCTION

SPEED regulation of induction motors has evolved from open-loop variable frequency control to closed-loop vector control with/without an encoder and their adaptive variants [1], [2]. Speed-sensorless motor drives, in which the rotor position/speed is unmeasured, are particularly interesting from both theoretical and practical perspectives. On one hand, they are favored due to the improved reliability and the reduced cost by removing the rotor shaft encoder. On the other hand, the resultant speed-sensorless control design problem is challenging due to the absence of the encoder, and thus attracts a lot of theoretical interests [3]–[5]. Prevailing speed-sensorless motor drives suffer significant performance degradation from the elimination of the encoder, and thus their applications remain limited within fields requiring low or medium performance. In fact, state estimation, identified as the main bottleneck to speed regulation performance, has been consistently the research focus.

Existing approaches, e.g. adaptive or Kalman filter-based, have limitations such as unnecessarily assuming rotor speed as a constant parameter or failure to ensure convergence of estimation error dynamics. Adaptation idea, where the rotor speed is treated as an unknown parameter to avoid dealing with nonlinear dynamics, was initially exploited and remains appealing [6]–[10]. Although the design is simple, adaptation-based estimation fails to offer satisfactory balance between estimation bandwidth and robustness to measurement noises. This motivates designs treating the rotor speed as a state variable, e.g. [5], [11]–[17] and reference therein. More or less, these methods are subject to limitations. For instance, work [11]–[14] rely on a triangular observable form where the system dynamics have triangular state dependence [18]. Certain nonlinear state-dependent terms in the state equation are treated as disturbances to ensure that the system dynamics admit the triangular observable form. Such a treatment necessarily results in a conservative design. Relying on singular

perturbation analysis, work [5] establishes local stability results.

Focusing on new estimation algorithms with certain guarantee of stability and practical effectiveness, this paper explores speed-sensorless estimation through nonlinear geometric observer design, which typically entails the system being in certain normal forms. This paper develops and verifies new estimation algorithms for speed-sensorless motor drives based on a non-triangular observable form (NTOF) [19]. Main contributions are: first, we show that the induction motor model admits the NTOF by a change of state coordinates, and thus high gain observer design based on the NTOF can be conducted; second, we address, by examining several observers without involving the closed-form inverse state coordinate, the implementation issues of the resultant high gain observer; finally, we verify the effectiveness of the proposed estimation algorithm by experiments.

The rest of this paper is organized as follows. Problem formulation is provided in Section II. Section III presents speed-sensorless estimation algorithms. Experimental results in Section IV verify that the proposed algorithm is meaningful and effective in practice. This paper is concluded by Section V.

Notations. Let ζ be a dummy variable. Denote $\hat{\zeta}$ as the estimate of the variable, ζ^* as the reference, $\tilde{\zeta} = \zeta - \hat{\zeta}$ as the estimation error, and $e_\zeta = \zeta^* - \hat{\zeta}$ as the tracking error. Given a C^∞ vector field $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$, and a C^∞ function $h : \mathbb{R}^n \rightarrow \mathbb{R}$, the function $L_f h(\zeta) = \frac{\partial h(\zeta)}{\partial \zeta} f$ is the *Lie derivative* of $h(\zeta)$ along f . Repeated Lie derivatives are defined as $L_f^k h(\zeta) = L_f(L_f^{k-1} h(\zeta))$, $k \geq 1$ with $L_f^0 h(\zeta) = h(\zeta)$.

II. PRELIMINARY

In a frame rotating at an angular velocity ω_1 , the induction motor model is given by

$$\begin{aligned} \dot{i}_{ds} &= -\gamma i_{ds} + \omega_1 i_{qs} + \beta(\alpha \Phi_{dr} + \omega \Phi_{qr}) + \frac{u_{ds}}{\sigma} \\ \dot{i}_{qs} &= -\omega_1 i_{ds} - \gamma i_{qs} + \beta(\alpha \Phi_{qr} - \omega \Phi_{dr}) + \frac{u_{qs}}{\sigma} \\ \dot{\Phi}_{dr} &= -\alpha \Phi_{dr} + (\omega_1 - \omega) \Phi_{qr} + \alpha L_m i_{ds} \\ \dot{\Phi}_{qr} &= -\alpha \Phi_{qr} - (\omega_1 - \omega) \Phi_{dr} + \alpha L_m i_{qs} \\ \dot{\omega} &= \mu(\Phi_{dr} i_{qs} - \Phi_{qr} i_{ds}) - \frac{T_l}{J} \\ y &= [i_{ds}, i_{qs}]^\top, \end{aligned} \quad (1)$$

where notation is defined in Table I. The frame with $\omega_1 = 0$ is called the stationary frame. Readers are referred to [1], [20] for details on the motor modeling.

This paper deals with the speed-sensorless state estimation problem, i.e., to reconstruct the full state of system (1) based on u_{ds}, u_{qs} and y . For linear systems, detectability ensures

TABLE I: Notations

Notation	Description
i_{ds}, i_{qs}	stator currents in d - and q -axis
Φ_{dr}, Φ_{qr}	rotor fluxes in d - and q -axis
ω	rotor angular speed
u_{ds}, u_{qs}	stator voltages in d - and q -axis
ω_1	angular speed of a rotating frame
Φ^*	rotor flux amplitude reference
ω^*	rotor angular speed reference
i_{ds}^*, i_{qs}^*	references of stator currents in d - and q -axis
T_l	load torque
J	inertia
L_s, L_m, L_r	stator, mutual, and rotor inductances
R_s, R_r	stator and rotor resistances
σ	$\frac{L_s L_r - L_m^2}{L_r}$
α	R_r / L_r
β	$L_m / (\sigma L_r)$
γ	$R_s / \sigma + \alpha \beta L_m$
μ	$3p L_m / (2J L_r)$
p	number of pole pairs

convergent state estimation. For nonlinear systems, uniform observability, although stronger than detectability, is typically needed [18]. Detailed observability analysis of system (1) can be found in [2], [21]. What follows is a brief summary. When the rotor speed is constant, the induction motor model (1) is locally observable provided that the rotor flux vector rotates or its modulus (amplitude) is not constant. For general scenarios, local observability might not hold at certain operation regions. In fact, work [21] shows the existence of operation regimes that the induction motor model (1) is neither observable nor detectable. Lack of local (uniform) observability poses fundamental limitation to the state estimation problem. For simplicity, this paper assumes that the operation conditions suffice the uniform observability.

Let us briefly recall provable state estimation results to expose challenges in the pursuit of probable state estimation for the induction motor. Consider a locally uniformly observable multi-input and multi-output (MIMO) system represented by

$$\begin{aligned} \dot{\zeta} &= f(\zeta) + g(\zeta)u \\ y &= h(\zeta), \end{aligned} \quad (2)$$

where the state $\zeta \in \mathbb{R}^n$, the control input $u \in \mathbb{R}^m$, the output $y \in \mathbb{R}^p$, $f, g: \mathbb{R}^n \rightarrow \mathbb{R}^n$ are C^∞ vector fields, and $h: \mathbb{R}^n \rightarrow \mathbb{R}^p$ is a vector of C^∞ functions. System (2) is lack of structures. One can perform general observer designs, such as Thau's [22], a Linear Matrix Inequality-based extension [23], and a neural-adaptive variant [24]. These approaches, although barely imposing restrictions on the system structure, suffer non-constructive nature or conservative design to certain extent. Take [24] as an example, where nonlinear terms in the system dynamics are approximated by a linear combination of basis functions (implemented as a neural network). A neural-adaptive observer estimates the system state, and identifies the weight vector of basis functions. Key idea is to render the state estimation error dynamics being input-to-state stable (achieved by following linear observer design), and the parameter estimation error being uniformly bounded. The state estimation error is bounded, and the bound depends on the infinity norms of the approximation error, basis functions, and

the weight vector. By resorting to a high observer gain, the state estimation error bound can be made arbitrarily small, while inevitably sacrificing robustness to measurement noises. Without explicit exploitation of any knowledge of the system structure, one will expect that the neural-adaptive observer requires a larger gain than the traditional high gain observer. It is noteworthy that the proposed neural-adaptive result is essentially equivalent to a linear observer design where uncertainties including nonlinear terms being treated as bounded and non-vanishing disturbances. This paper leverages another class of convergence-guaranteed nonlinear observer designs, e.g. exact error linearization [25]–[28], block triangular-based decentralized observer [29]–[31], high gain observer [18], [32]–[38], sliding mode observer [12], [39], and switched observers for triangular systems [40]. Exploiting the system structure, these approaches are typically more restrictive, but render less conservative design, better estimation performance, as well as provably convergent estimation error dynamics. In order to design such observers, it is necessary to transform system (2) into certain special structures by a change of coordinates. For simplicity, this paper considers a change of state coordinates.

III. OBSERVER DESIGN FOR SPEED-SENSORLESS ESTIMATION

This section verifies that the induction motor model (1) admits the NTOF (13) in Appendix and the corresponding observer design in the new coordinates. Next focus is to overcome two challenges while implementing the observer: 1) it is not implementable in the new coordinates, because the inverse state transformation cannot be explicitly derived; 2) its implementation in the original coordinates involves the inverse of the observability matrix, which might be ill-conditioned and run into the numerical issue.

A. Verifying Transformability to the NTOF

A concise representation of system (1) is given as follows

$$\dot{\zeta} = f(\zeta) + g^1 u_{ds} + g^2 u_{qs},$$

where $\zeta = [i_{ds}, i_{qs}, \Phi_{dr}, \Phi_{qr}, \omega]^\top$, $g^1 = [1/\sigma, 0, 0, 0, 0]^\top$, $g^2 = [0, 1/\sigma, 0, 0, 0]^\top$, and f can be determined readily. The induction motor model (1) cannot be put into the triangular form defined in [33]. We show that system (1) is transformable to the NTOF by the following change of state coordinates, for the k th subsystem

$$x^k = \phi^k(\zeta) = \begin{bmatrix} h_k(\zeta) \\ \vdots \\ L_f^{\lambda_k - 1} h_k(\zeta) \end{bmatrix}, \quad (3)$$

where λ_k is observability indices [26], [41]. Local observability ensures that $x = \phi(\zeta) = [(\phi^1(\zeta))^\top, \dots, (\phi^p(\zeta))^\top]^\top$ locally defines new coordinates, i.e., $\phi(\zeta)$ is a local diffeomorphism. The observable form is not uniquely defined due to the non-uniqueness of observability definitions and observability indices. Readers are referred to [29], [42] for details about observability and observable forms.

For system (1) with outputs $h_1 = i_{ds}$ and $h_2 = i_{qs}$, observability indices can be taken as (3, 2) and (2, 3). For illustrative purpose, we take $(\lambda_1, \lambda_2) = (3, 2)$ as an example and verify that the change of state coordinates, given by

$$x = [h_1, L_f h_1, L_f^2 h_1, h_2, L_f h_2]^\top, \quad (4)$$

transforms (1) into the NTOF (13). Given (4), the system in x -coordinates is written as

$$\begin{aligned} \dot{x} &= Ax + \underline{\varphi}(x, u) \\ y &= Cx, \end{aligned}$$

where $A = \text{diag}\{A^1, A^2\}$ with $A^1 \in \mathbb{R}^{3 \times 3}$ and $A^2 \in \mathbb{R}^{2 \times 2}$, $C = \text{diag}\{C^1, C^2\}$ with $C^1 \in \mathbb{R}^3$ and $C^2 \in \mathbb{R}^2$, and

$$\begin{aligned} \underline{\varphi} &= [(\underline{\varphi}^1)^\top \quad (\underline{\varphi}^2)^\top]^\top \\ \underline{\varphi}^1 &= \{L_{g^1} h_1(\phi^{-1}(x))u_{ds} + L_{g^2} h_1(\phi^{-1}(x))u_{qs}\} \frac{\partial}{\partial x_1^1} \\ &\quad + \{L_{g^1} L_f h_1(\phi^{-1}(x))u_{ds} + L_{g^2} L_f h_1(\phi^{-1}(x))u_{qs}\} \frac{\partial}{\partial x_1^2} \\ &\quad + \{L_f^3 h_1(\phi^{-1}(x)) + L_{g^1} L_f^2 h_1(\phi^{-1}(x))u_{ds} \\ &\quad + L_{g^2} L_f^2 h_1(\phi^{-1}(x))u_{qs}\} \frac{\partial}{\partial x_1^3} \\ \underline{\varphi}^2 &= \{L_{g^1} h_2(\phi^{-1}(x))u_{ds} + L_{g^2} h_2(\phi^{-1}(x))u_{qs}\} \frac{\partial}{\partial x_2^1} \\ &\quad + \{L_f^2 h_2(\phi^{-1}(x)) + L_{g^1} L_f h_2(\phi^{-1}(x))u_{ds} \\ &\quad + L_{g^2} L_f h_2(\phi^{-1}(x))u_{qs}\} \frac{\partial}{\partial x_2^2}. \end{aligned} \quad \text{where}$$

The rest is to verify that $\underline{\varphi}$ satisfies the triangular condition (14) in Appendix. As mentioned in Section II, one considers system (1) in the stationary frame. With this in mind, it is not difficult to verify that

$$\begin{aligned} L_{g^1} h_1(\phi^{-1}(x)) &= 1/\sigma, & L_{g^2} h_1(\phi^{-1}(x)) &= 0 \\ L_{g^1} L_f h_1(\phi^{-1}(x)) &= 0, & L_{g^2} L_f h_1(\phi^{-1}(x)) &= 0 \\ L_{g^1} h_2(\phi^{-1}(x)) &= 0, & L_{g^2} h_2(\phi^{-1}(x)) &= 1/\sigma \\ L_{g^1} L_f h_2(\phi^{-1}(x)) &= 0, & L_{g^2} L_f h_2(\phi^{-1}(x)) &= 0. \end{aligned}$$

The terms $\underline{\varphi}^1$ and $\underline{\varphi}^2$ are therefore rewritten as follows

$$\begin{aligned} \underline{\varphi}^1 &= \frac{u_{ds}}{\sigma} \frac{\partial}{\partial x_1^1} + \{L_f^3 h_1(\phi^{-1}(x)) \\ &\quad + L_{g^1} L_f^2 h_1(\phi^{-1}(x))u_{ds} + L_{g^2} L_f^2 h_1(\phi^{-1}(x))u_{qs}\} \frac{\partial}{\partial x_1^3} \\ \underline{\varphi}^2 &= \frac{u_{qs}}{\sigma} \frac{\partial}{\partial x_1^1} + L_f^2 h_2(\phi^{-1}(x)) \frac{\partial}{\partial x_2^2}. \end{aligned}$$

Evidently, $\underline{\varphi}_1^1$, $\underline{\varphi}_2^1$ and $\underline{\varphi}_1^2$ satisfy the triangular condition (14). The rest two components $\underline{\varphi}_3^1$ and $\underline{\varphi}_2^2$, though complicated, are allowed to have general dependence on x . We therefore verify that system in coordinates (3) is in the NTOF.

Remark 3.1: One can verify that state coordinates induced by observability indices $\lambda_1 = 2$ and $\lambda_2 = 3$ transform system (1) into the NTOF as well. The ordering of subsystems should, however, be adjusted to ensure the triangular condition (14).

B. Exact Observer Design

Given the observer (15) (in Appendix) for the k th subsystem, an observer can be readily designed for each subsystem in the x -coordinates. For system (1), we take $\lambda_1 = 3, \lambda_2 = 2, p = 2$, and have design parameters

$$\begin{aligned} \theta &> 0, \quad \delta_1 = 1, \quad \delta_2 = 1 \\ \Delta_1(\theta) &= \text{diag}\{1, \frac{1}{\theta}, \frac{1}{\theta^2}\} \quad \Delta_1(\theta) = \text{diag}\{1, \frac{1}{\theta}\} \\ S_1^{-1} C_1^\top &= [3, 3, 1]^\top, \quad S_2^{-1} C_2^\top = [2, 1]^\top. \end{aligned}$$

Substituting the aforementioned design parameters into the k th subsystem, one can obtain the observer in x -coordinates as follows

$$\begin{aligned} \dot{\hat{x}}^1 &= A_1 \hat{x}^1 + \hat{\varphi}^1 + G^1 C_1 \tilde{x}^1 \\ \dot{\hat{x}}^2 &= A_2 \hat{x}^2 + \hat{\varphi}^2 + G^2 C_2 \tilde{x}^2 \\ \hat{y}_1 &= C_1 \hat{x}^1 \\ \hat{y}_2 &= C_2 \hat{x}^2, \end{aligned} \quad (5)$$

$$\begin{aligned} G^1 &= \theta^{\delta_1} \Delta_1^{-1}(\theta) S_1^{-1} C_1^\top \\ &= \text{diag}\{\theta, \theta^2, \theta^3\} \begin{bmatrix} 3 \\ 3 \\ 1 \end{bmatrix} = \begin{bmatrix} 3\theta \\ 3\theta^2 \\ \theta^3 \end{bmatrix} \\ G^2 &= \theta^{\delta_2} \Delta_2^{-1}(\theta) S_2^{-1} C_2^\top \\ &= \text{diag}\{\theta, \theta^2\} \begin{bmatrix} 2 \\ 1 \end{bmatrix} = \begin{bmatrix} 2\theta \\ \theta^2 \end{bmatrix}. \end{aligned}$$

Without the closed-form inverse state transformation $\zeta = \phi^{-1}(x)$, the expression of $\varphi(x, u)$ cannot be obtained. The observer in x -coordinates (5) needs additional rearrangement for implementation. Rewrite $\varphi(x, u)$ as a function of $(\phi(\zeta), u)$. The x -observer is rearranged as follows

$$\begin{aligned} \dot{\hat{x}}^k &= A_k \hat{x}^k + \varphi^k(\phi(\hat{\zeta}), u) + \theta^{\delta_k} \Delta_k^{-1}(\theta) S_k^{-1} C_k^\top C_k \tilde{x}^k \\ \hat{y}_k &= C_k \hat{x}^k, \end{aligned} \quad (6)$$

where $\varphi^k(\phi(\hat{\zeta}), u)$ are the rows corresponding to the k th subsystem in $\varphi(\phi(\hat{\zeta}), u)$ given by

$$\varphi(\phi(\hat{\zeta}), u) = \frac{\partial \phi(\hat{\zeta})}{\partial \hat{\zeta}} (f(\hat{\zeta}) + g(\hat{\zeta})u) - A\phi(\hat{\zeta}).$$

As a result, the observer (6) can be implemented as

$$\begin{aligned} \dot{\hat{x}} &= \frac{\partial \phi(\hat{\zeta}, u)}{\partial \hat{\zeta}} (f(\hat{\zeta}) + g(\hat{\zeta})u) + \Theta \Delta^{-1}(\theta) S^{-1} C^\top C \tilde{x} \\ \hat{y} &= C \hat{x}, \end{aligned} \quad (7)$$

where $\hat{\zeta}$ is numerically constructed by solving the following n nonlinear algebraic equations

$$\hat{x} = \phi(\hat{\zeta}). \quad (8)$$

Remark 3.2: If the state transformation $\phi(\zeta)$ is a global diffeomorphism, $\hat{\zeta}$ can be uniquely solved from (8). Otherwise, two issues arise. First, the x -coordinates may not be well-defined where $\frac{\partial \phi(\hat{\zeta}, u)}{\partial \hat{\zeta}}$ is singular, and thus the observer (7) is invalid. Second, even if the x -coordinates are well-defined, $\phi(\zeta)$ is non-convex, indicating that (8) may have multiple

solutions. Solving (8) for $\hat{\zeta}$ could be easily trapped into to a local minimum (the state estimate $\hat{\zeta}$ is erroneous).

If the state transformation $\phi(\zeta)$ is a local diffeomorphism in a neighborhood of ζ_0 , its Jacobian $\frac{\partial x}{\partial \zeta}$ is non-singular in the neighborhood. One can therefore implement the observer (5) as follows

$$\begin{aligned} \dot{\hat{\zeta}} &= f(\hat{\zeta}) + g(\hat{\zeta})u + \left(\frac{\partial \hat{x}}{\partial \hat{\zeta}}\right)^{-1} \Theta \Delta^{-1}(\theta) S^{-1} C^T (y - \hat{y}) \\ \hat{y} &= h(\hat{\zeta}). \end{aligned} \quad (9)$$

Compared with the observer (7)-(8), observer (9) costs less computation by avoiding the calculation of numerical solutions of nonlinear equations (8). However, the observer (9) also suffers incorrect state estimation.

C. Approximate Observer Design

To shorten the notation, we introduce $Q_o = \frac{\partial \hat{x}}{\partial \hat{\zeta}}$. Implementation of the observer (9) requires solving the inverse of Q_o in realtime. More accurately, to solve the observer gain G_ζ in ζ -coordinates from the following linear system

$$G_x = Q_o G_\zeta, \quad (10)$$

where $G_x = \Theta \Delta^{-1}(\theta) S^{-1} C^T$. Numerical issue arises when Q_o is ill-conditioned, because one cannot easily rely on the solutions coming out of an ill-conditioned Q_o . For instance the least square solution

$$G_\zeta = (Q_o^T Q_o)^{-1} Q_o^T G_x$$

becomes useless. Let $\text{cond}(Q_o) = |Q_o|_\infty |Q_o^{-1}|_\infty$ be the condition number of Q_o . One can expect a loss of roughly $\log_{10}(\text{cond}(Q_o))$ decimal places in the accuracy of the solutions.

Next, we discuss how to remedy the aforementioned numerical instability, by using techniques such as pre-conditioning, scaling, etc. An intuitive pre-conditioning, also called the Tikhonov regularization, introduces a positive constant h into (10) and tries to solve

$$G_\zeta = (Q_o^T Q_o + hI)^{-1} Q_o^T G_x.$$

By introducing h , the Tikhonov regularization reduces the condition number of $Q_o^T Q_o$ and thus results in improvement of the solution accuracy.

Alternatively, one can regularize the problem by adding extra information, for instance the smoothness of G_ζ . The basic idea is that the correct one among all (near) solutions is characterized by requiring the ‘‘smoothness’’ of some function, curve or surface constructed from G_ζ at a fixed time instant. Interested readers are referred to [43] for an example of imposing a smoothness condition on G_ζ . As a smooth function of $\zeta(t)$ which is continuous over time, G_ζ is continuous over time as well. This priori however contradicts the fact that an ill-conditioned Q_o typically results in discontinuity of G_ζ over time. During algorithm implementation, one would force that G_ζ be continuous over time. This is viable by, for instance, introducing a low pass filter on G_ζ , or the following scheme

$$G_\zeta(k) = \begin{cases} \text{solved from (10),} & \text{if } \text{cond}(Q_o) \leq c \\ G_\zeta(k-1), & \text{otherwise,} \end{cases}$$

where k is the time step, $\text{cond}(\cdot)$ is the condition number, and c is a positive constant.

The aforementioned preconditioning methods concentrate on (10) and try to resolve the ill-conditionedness of Q_o at a fixed time instant. One can also take an approach from a systematic perspective, where Q_o is treated as a function of ζ . The idea is to replace the Q_o , viewed as a matrix of functions, with \hat{Q}_o , a matrix of approximate functions. \hat{Q}_o is often accomplished intuitively, for instance, by truncating the Taylor series expansion of Q_o , or by eliminating certain components from Q_o on the basis of structural analysis. Note that it is difficult to analyze how the approximation error, due to either pre-conditioning or functional approximation, affects the stability and accuracy of estimation error dynamics.

D. Stability and Robustness

We recite main results in [19] as follows.

Theorem 3.3: [19, Thm. 3.1] Given Assumption A.1, $\forall N > 0; \exists \theta_0 > 0; \forall \theta \geq \theta_0; \exists \lambda_\theta > 0, \mu_\theta > 0$ such that for $1 \leq k \leq p$

$$|x(t) - \hat{x}(t)| \leq \lambda_\theta e^{-\mu_\theta t} |x(0) - \hat{x}(0)|$$

for every admissible control u with $|u|_\infty \leq N$. Moreover, λ_θ is a polynomial in θ and $\lim_{\theta \rightarrow \infty} \mu_\theta = +\infty$.

Remark 3.4: Although [19, Thm. 3.1] establishes globally exponential stability for the estimation error \tilde{x} , it does not necessarily hold for the estimation error $\tilde{\zeta}$, because x is not globally defined. In fact, for the motor model (1), only local stability result can be obtained in $\tilde{\zeta}$: both ζ and $\hat{\zeta}$ should start from and remain inside an open domain \mathcal{B}_x belonging to a compact domain \mathcal{D} where $\phi(\zeta)$ and its inverse are diffeomorphic. The convergence of the zero solution of \tilde{x} also implies that of $\tilde{\zeta}$ because

$$\begin{aligned} |\zeta - \hat{\zeta}| &= |\phi^{-1}(x) - \phi^{-1}(\hat{x})| \\ &\leq \left| \frac{\partial \phi^{-1}(\epsilon)}{\partial \epsilon} \right|_\infty |x - \hat{x}| \\ &\leq \left| \frac{\partial \phi^{-1}(\epsilon)}{\partial \epsilon} \right|_\infty \lambda_\theta e^{-\mu_\theta t} |x(0) - \hat{x}(0)|, \end{aligned}$$

where $\left| \frac{\partial \phi^{-1}(\epsilon)}{\partial \epsilon} \right|_\infty$ is computed over all $\epsilon \in \mathcal{B}_x$.

A speed-sensorless control system should be robust w.r.t. parametric uncertainties, actuator offsets and noises, and measurement noises. The actuator offsets and noises appear as additive disturbances d_a in the motor model. The aforementioned observer design is relatively robust by showing that the resultant estimation error dynamics are input to state stable. The argument is as follows. From the expressions of φ^1, φ^2 , the actuator uncertainties d_a are also additive in x -coordinates, and thus are globally Lipschitz. According to [44, Lem. 4.6], the exponential convergence of $\tilde{\zeta}$ ensures that $\tilde{\zeta}$ -dynamics are input to state stable with respect to disturbances d_a as long as the dynamics of ζ is Lipschitz with respect to d_a . Hence, the estimation error \tilde{x} may converge to a bounded region containing the origin. A large tuning parameter θ can reduce the size of the region and improve the estimation accuracy.

With the presence of parametric uncertainties, the aforementioned observer design could not lead to concrete stability results. Assuming that true values of model parameters are

$\Theta = [\alpha, \beta, \gamma, \sigma, J]^\top$ and Θ_o is used in the observer, the motor model is rewritten as

$$\dot{\zeta} = f_o(\zeta, \Theta_o) + g^1 u_{ds} + g^2 u_{qs} + f(\zeta, \Theta) - f_o(\zeta, \Theta_o). \quad (11)$$

Applying the state transformation (3), with f replaced by f_o , to (11) gives

$$\dot{x} = Ax + \underbrace{\varphi(x, \Theta_o, u) + \frac{\partial x}{\partial \zeta} (f(\zeta, \Theta) - f_o(\zeta, \Theta_o))}_{\delta f} \Big|_{\zeta=\phi^{-1}(x)}.$$

Unless δf bears the same triangular structure as φ , the proposed observer could not guarantee the stability of the resultant estimation error dynamics. This observation implies the necessity of generalizing the proposed observer to incorporate parameter adaptation, a potential future work.

Regarding robustness to measurement noises, one can follow the similar argument as in the analysis of actuator uncertainties, measurement noises enter the estimation error dynamics additively, and the proposed observer is relative robust. However, the proposed observer, with the high gain nature, is susceptible to measurement noises and exhibits certain tradeoff. On one hand, the observer gain should be large to reject parametric and actuator uncertainties. On the other hand, a large observer gain amplifies the impact of measurement noises on estimation error dynamics.

IV. EXPERIMENTAL VALIDATION

We perform open-loop and closed-loop experiments when the motor operates below its rated speed. For the open-loop scenario, the encoder output is used as the feedback signal, and both the proposed and baseline algorithms run in open-loop. It demonstrates that the state estimation error of the proposed algorithm converges much faster than that corresponding to the baseline. In the closed-loop experiment, the estimated speed and rotor flux amplitude produced by the proposed algorithm are fed into the tracking controller, whereas the baseline remains in open-loop. It shows that the resultant speed-sensorless motor drive offers good speed tracking performance.

A. The Testbed

The testbed comprises Matlab/Simulink®, dSPACE® ACE Kit DS1104, a DC-AC inverter, and a Marathon® three-phase AC induction motor with an inertial load attached. The testbed is illustrated by Fig. 1 where the black and the red arrows represent signal and power flows, respectively. Control algorithm, including a tracking controller and state estimators, is implemented and compiled through Matlab/Simulink, and downloaded to dSPACE for real-time operation. The tracking controller determines the voltage reference $u_r(t)$ based on measured signals and references, and sends it to the PWM module in dSPACE; the PWM module controls the DC-AC inverter to generate three-phase voltages close to $u_r(t)$.

The dSPACE executes the data acquisition and real-time estimation and control tasks. It collects four signals: the motor position x , and three-phase stator currents i_a, i_b, i_c .

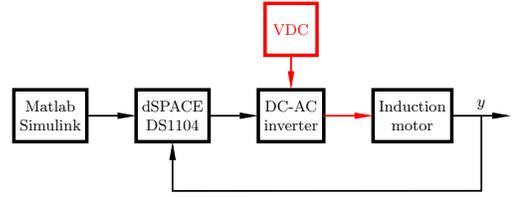


Fig. 1: The testbed architecture.

During experiments, the frequencies of both the control loop and PWM are 4kHz. The encoder has a resolution of 2048 pulses per revolution. The motor has model parameter values: $R_s = 11.05\Omega, R_r = 6.11\Omega, L_s = L_r = 0.3165H, L_m = 0.2939H, J = 5e - 3kgm^2$.

B. The Tracking Controller and State Estimator

The tracking controller implements an indirect field oriented control (IFOC) shown in Fig. 2. Four Proportional and Integral (PI) controllers $PI_i, 1 \leq i \leq 4$ regulate the speed, the rotor flux amplitude, the q -axis stator current, and the d -axis stator current, respectively. With signal and notation defined in Fig. 2 and Table I, the control law implemented in the tracking controller is represented by

$$\begin{aligned} i_{ds}^* &= K_\Phi^P e_\Phi + K_\Phi^I \int_0^t e_\Phi dt \\ i_{qs}^* &= K_\omega^P e_\omega + K_\omega^I \int_0^t e_\omega dt \\ u_{ds}^* &= K_{ids}^P e_{ids} + K_{ids}^I \int_0^t e_{ids} dt + u_{dsff} \\ u_{qs}^* &= K_{iqs}^P e_{iqs} + K_{iqs}^I \int_0^t e_{iqs} dt + u_{qsff}, \end{aligned} \quad (12)$$

where $u_{dsff} = -\sigma\omega_1 i_{qs}$ and $u_{qsff} = \sigma(\omega_1 i_{ds} + \beta\hat{\omega}\hat{\Phi}_{dr})$. ω_1 is the rotational speed of the estimated $d-q$ frame where the d axis is aligned with the estimated rotor flux. For open-loop experiments, $\omega_1 = \omega + \alpha i_{qs}/i_{ds}^*$, whereas for closed-loop experiments, $\omega_1 = \hat{\omega} + \alpha \hat{i}_{qs}/i_{ds}^*$. Park transformation and its inverse in Fig. 2 utilize the estimated rotor flux angle ρ , which is obtained according to the following equation

$$\dot{\rho} = \omega_1, \quad \rho(0) = 0.$$

All constants in (12) are determined by trial and error to achieve satisfactory speed tracking performance. Proportional gains K_{ids}^P and K_{iqs}^P are tuned to achieve 200Hz bandwidth for two stator current control loops, and K_Φ^P and K_ω^P are tuned to achieve 100Hz bandwidth for the speed control loop when the encoder feedback is used. Integral gains $K_{ids}^I, K_{iqs}^I, K_\Phi^I$ and K_ω^I are taken fairly small because the transient, instead of the steady state error, is our main concern.

The algorithm in [9] is chosen as the baseline, where the PI gains for the speed adaptation are tuned by trial and error to balance fast speed estimation (short transient) and the harmonics reduction in steady-state operation.

As aforementioned discussions, the matrix Q_o might be ill-conditioned, which necessarily leads to large fluctuations in the speed estimate. We follow the function approximation idea,

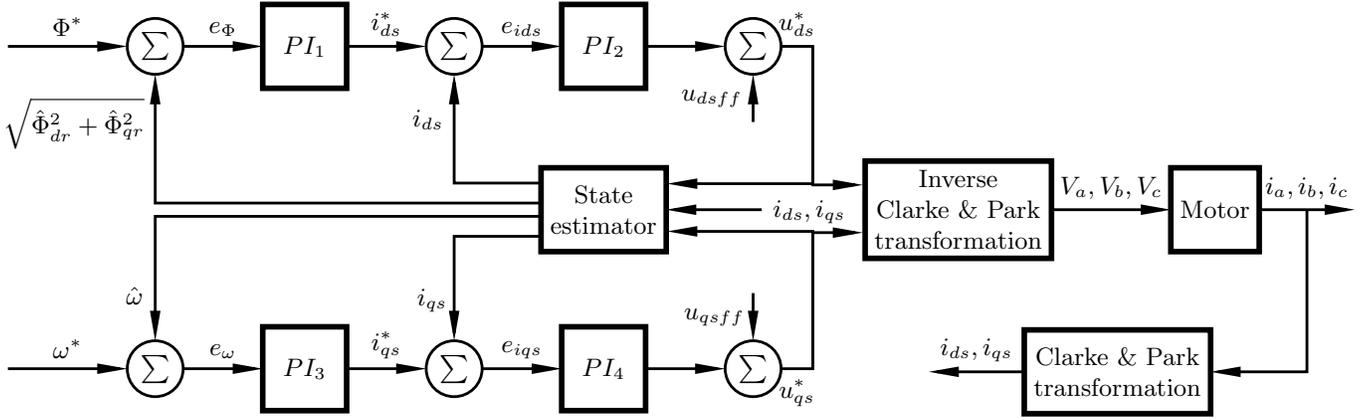


Fig. 2: The IFOC block diagram.

and derive \hat{Q}_o by exploiting the structure of Q_o . Notice that in the stationary frame, the matrix Q_o is periodically time-varying, and thus makes the numerical stability issue more challenging. Hence, we rewrite the system (2) in a frame rotating at ω_1 so that at steady state, $i_{ds}, i_{qs}, \Phi_{dr}, \Phi_{qr}$ are constants. Then we trim Q_o to get \hat{Q}_o , which is suitable for real-time implementation as well as offers decent state estimation performance. For the illustration purpose, we express

$$Q_o = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \omega_1 & \alpha\beta & \beta\hat{\omega} & \beta\hat{\Phi}_{qr} \\ 0 & 1 & 0 & 0 & 0 \\ -\omega_1 & 0 & -\beta\hat{\omega} & \alpha\beta & -\beta\hat{\Phi}_{dr} \\ * & * & * & * & * \end{bmatrix}.$$

We have two options to approximate Q_o . Option 1: we choose the approximate matrix \hat{Q}_o by letting the first four elements of the last row in Q_o be zero, i.e.,

$$\hat{Q}_o = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \omega_1 & \alpha\beta & \beta\hat{\omega} & \beta\hat{\Phi}_{qr} \\ 0 & 1 & 0 & 0 & 0 \\ -\omega_1 & 0 & -\beta\hat{\omega} & \alpha\beta & -\beta\hat{\Phi}_{dr} \\ 0 & 0 & 0 & 0 & * \end{bmatrix}.$$

Option 2: the approximate matrix \hat{Q}_o is chosen as follows

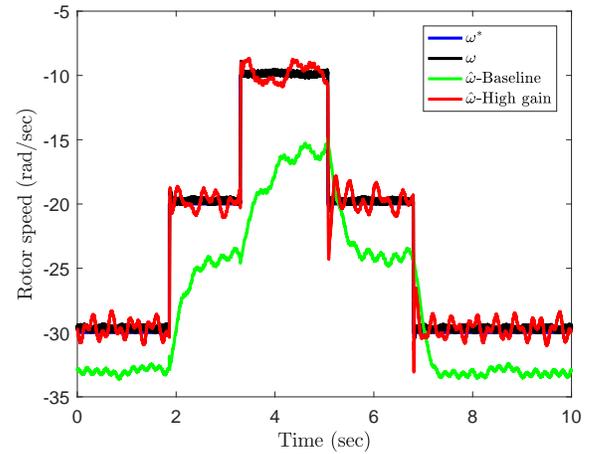
$$\hat{Q}_o = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & \omega_1 & \alpha\beta & \beta\hat{\omega} & 0 \\ 0 & 1 & 0 & 0 & 0 \\ -\omega_1 & 0 & -\beta\hat{\omega} & \alpha\beta & 0 \\ * & * & * & * & * \end{bmatrix}.$$

Both \hat{Q}_o is always non-singular, and their inverses can be solved reliably and efficiently. Option 2 is implemented in the experiment, and the observer parameter is $\theta = 5$.

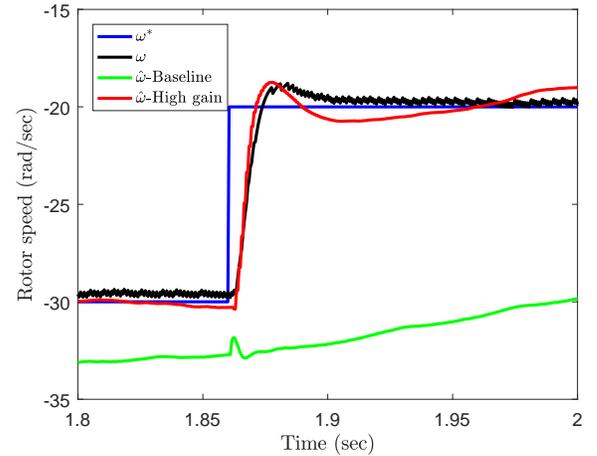
C. Experimental Results: Open-loop

The tracking controller uses the measured speed as a feedback signal for speed control, i.e., both the baseline and proposed estimation algorithms run in open-loop. With the knowledge of the rotor speed, the following open-loop estimator is employed to estimate the rotor flux for flux control

$$\dot{\hat{\Phi}}_{dr} = -\alpha\hat{\Phi}_{dr} - \omega\hat{\Phi}_{qr} + \alpha L_m i_{ds}$$



(a) Overall

(b) Zoom-in when ω^* step upFig. 3: Speed trajectories when ω^* jumps among -30rad/sec and -10rad/sec.

$$\dot{\hat{\Phi}}_{qr} = -\alpha\hat{\Phi}_{qr} + \omega\hat{\Phi}_{dr} + \alpha L_m i_{qs}.$$

We examine how quickly two estimation algorithms respond to step changes of the reference speed. Extensive tests are conducted and end up with the same conclusion: the proposed algorithm results in higher speed tracking performance

than the baseline, given the knowledge of the rotor inertia and a constant load torque. Figs. 3-5 present results for cases when the rotor speed is low. Low speed experiment is particularly interesting because the corresponding speed-sensorless operation is notoriously hard, thanks to weak observability, the low signal-to-ratio in sensed signals, and the distortion/uncertainties in the inverter voltages [45]. This is also manifested by Figs. 3-5, where the higher the rotor speed is, the better the baseline performs.

Fig. 3 shows the trajectories of reference speed ω^* , measured speed ω , and estimated speeds $\hat{\omega}$ by the baseline and the proposed algorithms. The reference speed contains four jumps: from -30rad/sec to -20rad/sec at $t = 1.85\text{sec}$, from -20rad/sec to -10rad/sec at $t = 3\text{sec}$, from -10rad/sec to -20rad/sec at around $t = 5\text{sec}$, and from -20rad/sec to -30rad/sec at $t = 7\text{sec}$. Upper plot offers a macroscopic view, while the bottom plot illustrates transient behavior incurred by reference steps. One can see that the estimated speed of the proposed algorithm converges to the neighborhood of the measured speed much faster than the baseline does. From the zoomed in plot, one can see that the proposed algorithm can closely follow the measured speed. In many cases, the transient lasts about 0.01sec .

Fig. 4 corresponds to the case where the speed reference jumps between -10rad/sec and 10rad/sec ; Fig. 5 corresponds to the case where the speed reference jumps between 10rad/sec and 30rad/sec . Even during the low speed, the proposed algorithm can estimate the speed quickly, although the baseline performs unsatisfactorily. Apparently, the same conclusion can be drawn from these cases.

D. Experimental Results: Closed-loop

We close the control loop with the estimated speed $\hat{\omega}$ from the proposed algorithm, and validate the speed tracking performance. The controller and estimator gains are the same as the open-loop. The baseline algorithm runs in open-loop.

Fig. 6 elaborates results when the reference speed jumps between -30rad/sec and -10rad/sec . As shown in the upper plots of Fig. 6, the measured speed can respond to positive steps of references quickly. The lower plot verifies that the speed transient is around 0.01sec . It is noteworthy that the estimated speed trajectories in the open-loop and the measured speed in the closed-loop contain harmonics. The harmonics are largely induced by model mismatches between the estimators and the true motor, and uncertainties in the output voltages of the inverter. The model mismatches are elegantly rejected by the tracking controller, and which render smooth measured speed in the open-loop and estimated speed in the closed-loop cases.

Although performing much better at the mid/high speed range, performance of the baseline is severely degraded during low-speed operation. This is consistent with observation disclosed in abundant literatures, due to low signal-to-noise ratio in sensed signals, and uncertainties in the output voltages of the inverter. Baseline, without considering the mechanical dynamics, benefits from high gains in the speed adaptation law to achieve fast estimation. This treatment does not cause any

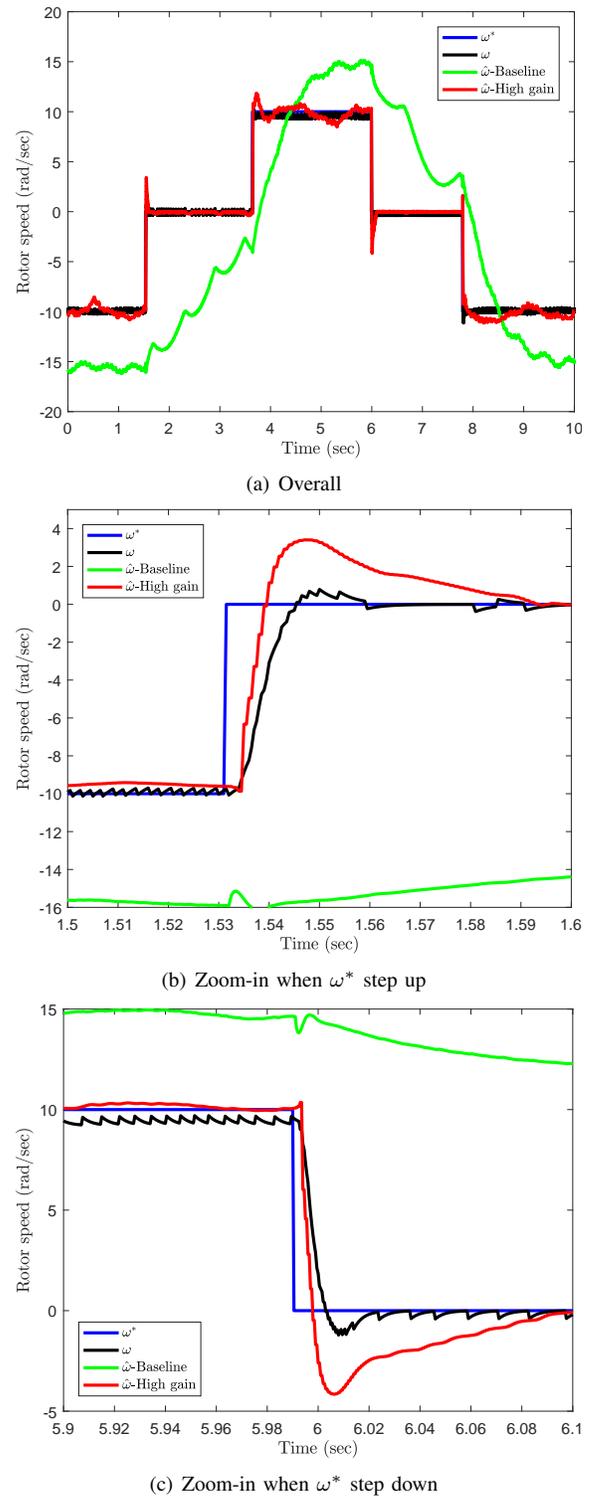
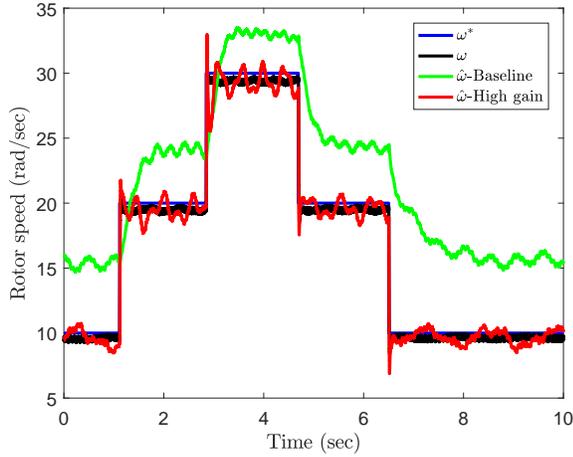
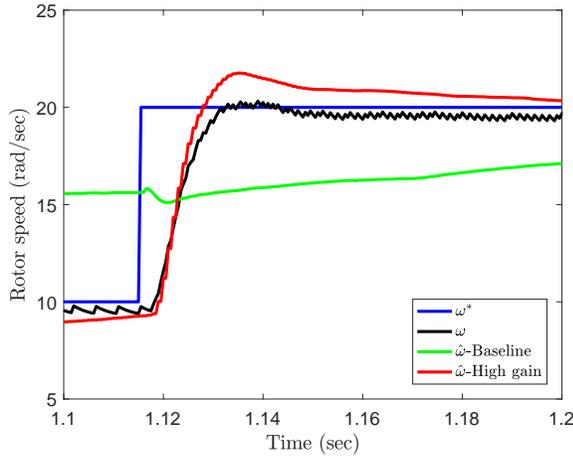


Fig. 4: Speed trajectories when ω^* jumps among -10rad/sec and 10rad/sec

problem during mid/high speed operation, where the signal-to-noise ratio is large. On the other hand, near zero speed, low gains are preferable to mitigate the large uncertainties in sensed signals, and necessarily leads to sloppy transient. Turning to the proposed algorithm, although it is coined as ‘high gain observer’, its gain in fact is not high at all, compared



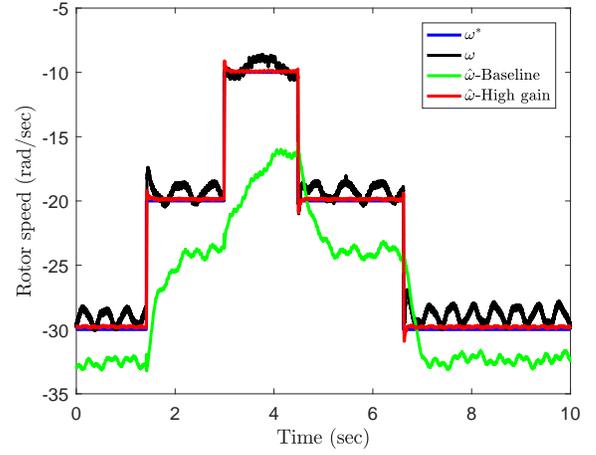
(a) Overall

(b) Zoom-in when ω^* step upFig. 5: Speed trajectories when ω^* jumps among 10rad/sec and 30rad/sec

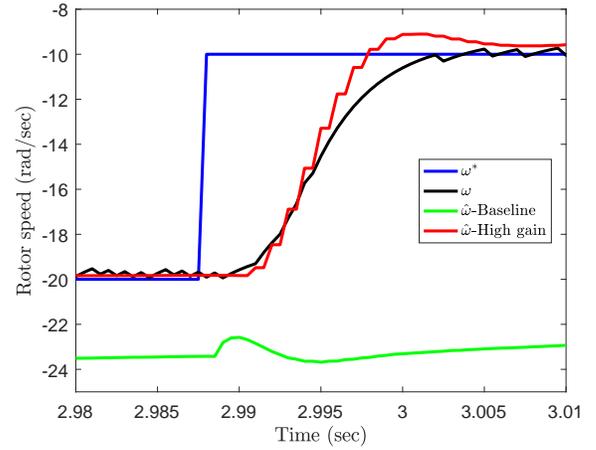
with the baseline. By utilizing the mechanical dynamics, the proposed algorithm can still achieve good performance with relatively low observer gain. As an apparent downside, the effectiveness of the proposed algorithm is contingent on the knowledge of the rotor inertia and a constant (maybe unknown) load torque.

V. CONCLUSION AND FUTURE WORK

This paper proposed and verified an approximate high gain estimation algorithm for speed-sensorless motor drives. The proposed algorithm is based on first transforming the motor model into a non-triangular observable form by a change of state coordinates, and then performing a high gain observer design in the new coordinates. Due to the local nature of the state transformation, only local stability can be obtained for the estimation error dynamics in the original coordinates. We provided approximate observers without solving the inverse state transformation. Experiments demonstrate the potential effectiveness and advantages of the proposed algorithm: fast speed estimation transient and ease of tuning. This paper also reveals a number of issues related to the proposed approach, for instance, relying on accurate values of model



(a) Overall

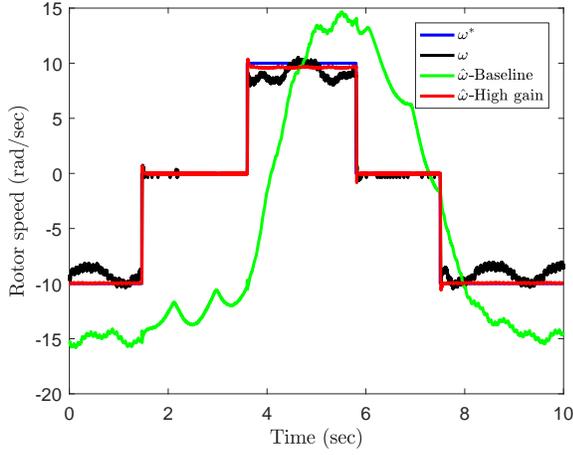
(b) Zoom-in when ω^* step upFig. 6: Speed trajectories when ω^* jumps among -30rad/sec and -10rad/sec

parameters, non-trivial numerical stability during real-time implementation, etc. Future work includes further development of more systematic solutions to address numerical stability, and generalization of the proposed algorithm to cope with parametric uncertainties.

APPENDIX A NON-TRIANGULAR OBSERVABLE FORM AND OBSERVER DESIGN

Work in [19] assumes that system (2) is transformable to the following non-triangular observable form by a change of state coordinates $x = \phi(\zeta)$

$$\begin{aligned} \dot{x} &= Ax + \varphi(x, u) \\ y &= Cx, \end{aligned} \quad (13)$$



(a) Overall

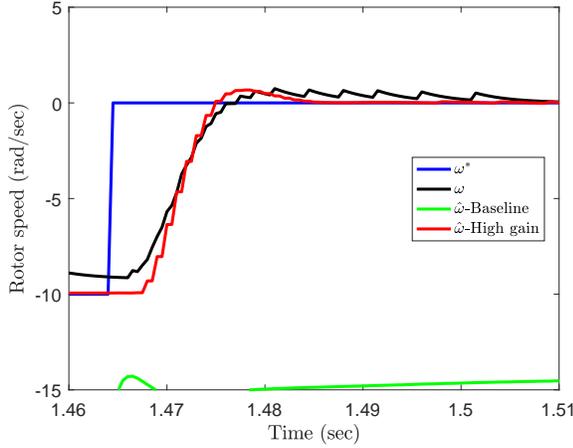
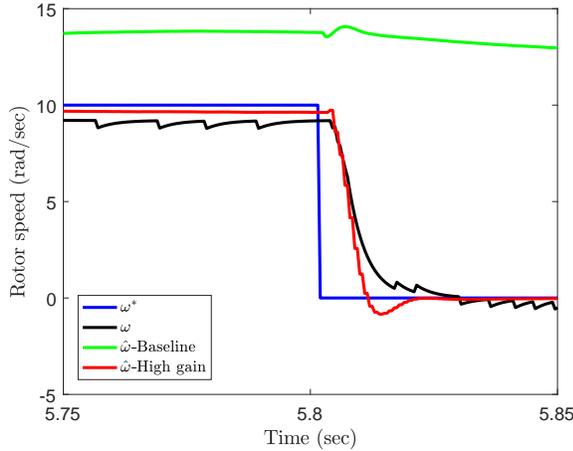

 (b) Zoom-in when ω^* step up

 (c) Zoom-in when ω^* step down

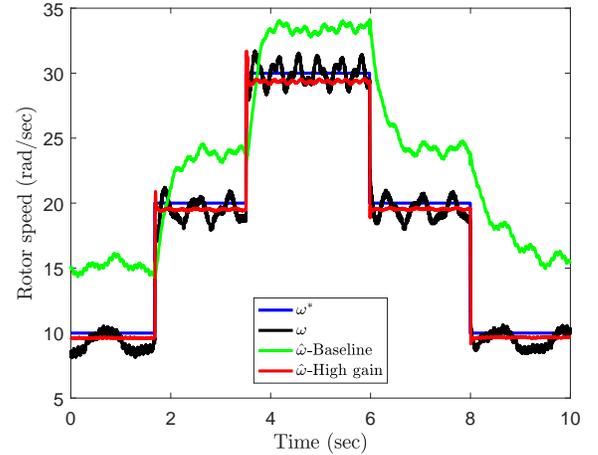
 Fig. 7: Speed trajectories when ω^* jumps among -10rad/sec and 10rad/sec

where the state $x \in \mathbb{R}^n$, and for $1 \leq k \leq p$,

$$x = \begin{bmatrix} x^1 \\ \vdots \\ x^p \end{bmatrix} \text{ with } x^k = \begin{bmatrix} x_1^k \\ \vdots \\ x_{\lambda_k}^k \end{bmatrix} \in \mathbb{R}^{\lambda_k}$$

$$A = \text{diag}\{A_1, \dots, A_p\}, A_k = \begin{bmatrix} 0 & I_{\lambda_k-1} \\ 0 & 0 \end{bmatrix} \in \mathbb{R}^{\lambda_k \times \lambda_k}$$

$$C = \text{diag}\{C_1, \dots, C_p\}, C_k = \begin{bmatrix} 1 & 0 \end{bmatrix} \in \mathbb{R}^{\lambda_k}$$



(a) Overall

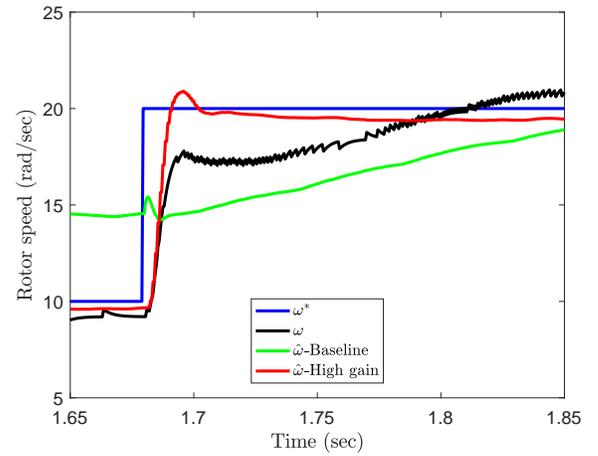

 (b) Zoom-in when ω^* step up

 Fig. 8: Speed trajectories when ω^* jumps among 10rad/sec and 30rad/sec

Literally, x^k denotes the state of the k th subsystem associated with the k th output y_k , and $\lambda_k, 1 \leq k \leq p$ are the dimensions of all subsystems. We call λ_k for $1 \leq k \leq p$ as subsystem indices, and have $\sum_{k=1}^p \lambda_k = n$. Note that the vector field $\varphi(x, u)$ is described as

$$\varphi = \begin{bmatrix} \varphi^1 \\ \vdots \\ \varphi^p \end{bmatrix} \text{ with } \varphi^k = \begin{bmatrix} \varphi_{\lambda_k}^k \\ \vdots \\ \varphi_{\lambda_k}^k \end{bmatrix} \in \mathbb{R}^{\lambda_k}, 1 \leq k \leq p.$$

Specifically, φ_i^k has the following structure: for $1 \leq i \leq \lambda_k - 1$,

$$\varphi_i^k = \varphi_i^k(x^1, \dots, x^{k-1}, x_1^k, \dots, x_i^k, x_1^{k+1}, \dots, x_1^p, u) \quad (14)$$

and for $i = \lambda_k$

$$\varphi_i^k = \varphi_{\lambda_k}^k(x^1, \dots, x^p, u).$$

The non-triangular observable form (13) does not have a triangular structure as in [33] because $\varphi_{\lambda_k}^k$ is dependent on all state x . It is actually a special case of the form defined in [19, Eqn. (1)] by taking $p_k = 1$. In fact, taking $p_k = 1$ for $1 \leq k \leq q$ in [19, Eqn. (1)] gives $q = p$ and $\lambda_k = n_k$.

It could be challenging to verify that system (2) is transformable to (13) and to construct the state transformation $x = \phi(\zeta)$. Verifying the transformability and solving the state transformation form another portfolio of research topics which are relatively separated from observer design.

Given the state transformation $x = \phi(\zeta)$ and its inverse $\zeta = \phi^{-1}(x)$, one can obtain the system representation in the non-triangular observable form. Observer design is performed in the non-triangular observable coordinates to estimate x . Given a system in the non-triangular observer form (13), a high gain observer can be constructed as follows. For the k th subsystem, the following observer is proposed

$$\begin{aligned}\dot{\hat{x}}^k &= A_k \hat{x}^k + \hat{\varphi}^k + \theta^{\delta_k} \Delta_k^{-1}(\theta) S_k^{-1} C_k^\top C_k \tilde{x}^k \\ \hat{y}_k &= C_k \hat{x}^k,\end{aligned}\quad (15)$$

where $\hat{x}^k = [\hat{x}_1^k, \dots, \hat{x}_{\lambda_k}^k]^\top$, $\tilde{x}^k = x^k - \hat{x}^k$,

$$\theta > 0$$

$$\delta_k = \begin{cases} 2^{p-k} \left(\prod_{i=k+1}^p (\lambda_i - \frac{3}{2}) \right), & \text{if } 1 \leq k \leq p-1 \\ 1, & \text{if } k = p \end{cases}$$

$$\Delta_k(\theta) = \text{diag}\left\{1, \frac{1}{\theta^{\delta_k}}, \dots, \frac{1}{\theta^{\delta_k(\lambda_k-1)}}\right\}$$

$$\hat{\varphi}^k = [\hat{\varphi}_1^k, \dots, \hat{\varphi}_{\lambda_k}^k]^\top$$

$$\hat{\varphi}_i^k = \varphi_i^k(\hat{x}_1^k, \dots, \hat{x}_{i-1}^k, \hat{x}_1^k, \dots, \hat{x}_i^k, \hat{x}_1^{k+1}, \dots, \hat{x}_1^p, u),$$

and S_k is solved from

$$S_k + A_k^\top S_k + S_k A_k = C_k^\top C_k. \quad (16)$$

It has been shown in [18] that the solution to (16) is symmetric positive definite and satisfies

$$S_k^{-1} C_k^\top = \left[C_{\lambda_k}^1, \dots, C_{\lambda_k}^{\lambda_k} \right]^\top$$

with $C_{\lambda_k}^i = \lambda_k! / (i!(\lambda_k - i)!)$ for $1 \leq i \leq \lambda_k$.

Assume that $\varphi(x, u)$ is globally Lipschitz with respect to x uniformly in u . That is

Assumption A.1: Given system (13), $\forall x_1, x_2 \in \mathbb{R}^n, \exists L_x > 0$

$$|\varphi(x_1, u) - \varphi(x_2, u)| \leq L_x |x_1 - x_2|. \quad (17)$$

REFERENCES

- [1] J. Holtz, "Sensorless control of induction motor drives," *Proceedings of the IEEE*, vol. 90, no. 8, pp. 1359–1394, 2002.
- [2] R. Marino, P. Tomei, and C. M. Verrelli, *Induction Motor Control Design*. London, UK: Springer, 2010.
- [3] J. Chiasson, "Dynamic feedback linearization of the induction motor," *IEEE Trans. Automat. Control*, vol. 38, no. 10, pp. 1588–1594, 1993.
- [4] L. Harnefors, "Globally stable speed-adaptive observers for sensorless induction motor drives," *IEEE Trans. Ind. Electron.*, vol. 54, no. 2, pp. 1243–1245, Apr. 2007.
- [5] M. Montanari, S. Peresada, and A. Tilli, "A speed-sensorless indirect field-oriented control for induction motors based on high gain speed estimation," *Automatica*, vol. 42, no. 10, pp. 1637–1650, Oct. 2006.
- [6] D. J. Atkinson, P. P. Acarnley, and J. W. Finch, "Observers for induction motor state and parameter estimation," *IEEE Trans. Ind. Appl.*, vol. 27, no. 6, pp. 1119–1127, Nov./Dec. 1991.
- [7] C. Schauder, "Adaptive speed identification for vector control of induction motors without rotational transducers," *IEEE Trans. Ind. Appl.*, vol. 28, no. 5, pp. 1054–1061, Sep./Oct. 1992.
- [8] H. Kubota, K. Matsuse, and T. Nakano, "DSP-based speed adaptive flux observer of induction motor," *IEEE Trans. Ind. Appl.*, vol. 29, no. 2, pp. 344–348, Mar./Apr. 1993.
- [9] H. Kubota and K. Matsuse, "Speed sensorless field-oriented control of induction motor with rotor resistance adaptation," *IEEE Trans. Ind. Appl.*, vol. 30, no. 5, pp. 1219–1224, Sep./Oct. 1994.
- [10] K. Ohyama, G. M. Asher, and M. Sumner, "Comparative analysis of experimental performance and stability of sensorless induction motor drives," *IEEE Trans. Ind. Electron.*, vol. 53, no. 1, pp. 178–186, Feb. 2006.
- [11] H. K. Khalil, E. G. Strangas, and S. Jurkovic, "Speed observer and reduced nonlinear model for sensorless control of induction motors," *IEEE Trans. Contr. Syst. Technol.*, vol. 17, no. 2, pp. 327–339, 2009.
- [12] C. Lascau, I. Boldea, and F. Blaabjerg, "A class of speed-sensorless sliding-mode observers for high-performance induction motor drives," *IEEE Trans. Ind. Electron.*, vol. 56, no. 9, pp. 3394–3403, Sep. 2009.
- [13] M. Ghanes and G. Zheng, "On sensorless induction motor drives: sliding-mode observer and output feedback controller," *IEEE Trans. Ind. Electron.*, vol. 56, no. 9, pp. 3404–3413, Sep. 2009.
- [14] S. Solvar, B. Le, M. Ghanes, J. P. Barbot, and G. Santomenna, "Sensorless second order sliding mode observer for induction motor," in *IEEE International Conference on Control Applications*, Yokohama, Japan, 2010, pp. 1933–1938.
- [15] Y. R. Kim, S.-K. Sul, and M.-K. Park, "Speed sensorless vector control of induction motor using extended kalman filter," *IEEE Trans. Ind. Appl.*, vol. 30, no. 5, pp. 1225–1233, Sep./Oct. 1994.
- [16] M. Barut, S. Bogosyan, and M. Gokasan, "Speed-sensorless estimation for induction motors using extended kalman filter," *IEEE Trans. Ind. Electron.*, vol. 54, no. 1, pp. 272–280, Jan. 2007.
- [17] M. Hilairt, F. Auger, and E. Berthelot, "Speed and rotor flux estimation of induction machines using a two-stage extended kalman filter," *Automatica*, vol. 45, no. 8, pp. 1819–1827, Aug. 2009.
- [18] J. P. Gauthier, H. Hammouri, and S. Othman, "A simple observer for nonlinear systems—applications to bioreactors," *IEEE Trans. Automat. Control*, vol. 37, no. 6, pp. 875–880, Jun. 1992.
- [19] M. Farza, M. M'Saad, M. Triki, and T. Maatoug, "High gain observer for a class of non-triangular systems," *Syst. Control Lett.*, vol. 60, no. 1, pp. 27–35, Jan. 2011.
- [20] W. Leonhard, *Control of Electrical Drives*. Springer, 2001.
- [21] S. Ibarra-Rojas, J. Moreno, and G. Espinosa-Pérez, "Global observability analysis of sensorless induction motors," *Automatica*, vol. 40, no. 6, pp. 1079–1085, Jun. 2004.
- [22] F. E. Thau, "Observing the state of nonlinear dynamic systems," *Int. J. Control*, vol. 17, no. 3, pp. 471–479, 1973.
- [23] R. Rajamani, "Observers for Lipschitz nonlinear systems," *IEEE Trans. Automat. Control*, vol. 43, no. 3, pp. 397–401, Mar. 1998.
- [24] X. Liu, C. Yang, Z. Chen, M. Wang, and C.-Y. Su, "Neuro-adaptive observer based control of flexible joint robot," *Neurocomputing*, vol. 275, pp. 73–82, 2018.
- [25] A. J. Krener and A. Isidori, "Linearization by output injection and nonlinear observers," *Syst. Control Lett.*, vol. 3, no. 1, pp. 47–52, Jun. 1983.
- [26] A. J. Krener and W. Respondek, "Nonlinear observers with linearizable error dynamics," *SIAM J. Control Optim.*, vol. 23, no. 2, pp. 197–216, Mar. 1985.
- [27] W. Respondek, A. Pogromsky, and H. Nijmeijer, "Time scaling for observer design with linearizable error dynamics," *Automatica*, vol. 40, no. 2, pp. 277–285, Feb. 2004.
- [28] Y. Wang and A. F. Lynch, "Multiple time scalings of a multi-output observer form," *IEEE Trans. Automat. Control*, vol. 55, no. 4, pp. 966–971, Apr. 2010.
- [29] J. Rudolph and M. Zeitz, "Block triangular nonlinear observer normal form," *Syst. Control Lett.*, vol. 23, no. 1, pp. 1–8, Jul. 1994.
- [30] Y. Wang and A. Lynch, "A block triangular form for nonlinear observer design," *IEEE Trans. Automat. Control*, vol. 51, no. 11, pp. 1803–1808, Nov. 2006.
- [31] —, "A block triangular observer forms for nonlinear observer design," *Int. J. Control*, vol. 81, no. 2, pp. 177–188, 2008.
- [32] G. Bornard and H. Hammouri, "A high gain observer for a class of uniformly observable systems," in *Proc. 30th CDC*, Brighton, England, 1991, pp. 1494–1496.
- [33] H. Shim, Y. I. Son, and J. H. Seo, "Semi-global observer for multi-output nonlinear systems," *Syst. Control Lett.*, vol. 42, no. 3, pp. 233–244, Mar. 2001.
- [34] H. Hammouri, B. Targui, and F. Armanet, "High gain observer based on a triangular structure," *Int. J. Robust Nonlin.*, vol. 12, pp. 497–518, 2002.
- [35] V. Andrien, L. Praly, and A. Astolfi, "High gain observers with updated gain and homogeneous correction terms," *Automatica*, vol. 45, no. 2, pp. 422–428, Feb. 2009.

- [36] M. Oueder, M. Farza, R. Ben Abdennour, and M. M'Saad, "A high gain observer with updated gain for a class of MIMO non-triangular systems," *Syst. Control Lett.*, vol. 61, no. 2, pp. 298–308, Feb. 2012.
- [37] H. Gao, W. He, C. Zhou, and C. Sun, "Neural network control of a two-link flexible robotic manipulator using assumed mode method," *IEEE Trans Ind. Informat.*, 2018, 10.1109/TII.2018.2818120.
- [38] W. He, Z. Yan, C. Sun, and Y. Chen, "Adaptive neural network control of a flapping wing micro aerial vehicle with disturbance observer," *IEEE Trans. Cybern.*, vol. 47, no. 10, pp. 3452–3465, 2017.
- [39] S. Drakunov and V. Utkin, "Sliding mode observers. Tutorial," in *Proc. 34th CDC*, New Orleans, LA, 1995, pp. 3376–3378.
- [40] D. Theodosis, D. Boskos, and J. Tsiniias, "Observer design for triangular systems under weak observability assumptions," *IEEE Trans. Automat. Control*, 2018, 10.1109/TAC.2018.2799904.
- [41] H. Nijmeijer, "Observability of a class of nonlinear systems: A geometric approach," *Ricerche Di Automatica*, vol. 12, no. 1, pp. 1107–1130, 1981.
- [42] R. Hermann and A. J. Krener, "Nonlinear controllability and observability," *IEEE Trans. Automat. Control*, vol. AC-22, no. 5, pp. 728–740, Oct. 1977.
- [43] A. Neumaier, "Solving ill-conditioned and singular linear systems: A tutorial on regularization," *SIAM Review*, vol. 40, no. 3, pp. 636–666, 1995.
- [44] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Englewood Cliffs, NJ: Prentice-Hall, 2002.
- [45] J. Holtz and J. Quan, "Sensorless vector control of induction motors at very low speed using a nonlinear inverter model and parameter identification," *IEEE Trans. Ind. Appl.*, vol. 38, no. 4, pp. 1087–1095, Jul./Aug. 2002.