Development of a Resolver-to-Digital Converter based on Second-Order Difference Generalized Predictive Control †

Thyago Estrabis [ID]**, Gabriel Gentil** [ID] **and Raymundo Cordero** [ID]

> Graduation Program in Electrical Engineering, Federal University of Mato Grosso do Sul, Campo Grande, MS, Brazil; thyago.estrabis@gmail.com (T.E.); gabrielgent@gmail.com (G.G.); raymundo.garcia@ufms.br (R.C.)

*   Correspondence: thyago.estrabis@gmail.com; Tel.: (+55 67 99333-2171)

†   This paper is an extended version of "Application of Model Predictive Control in a Resolver-to-Digital Converter" published in Proceeding of the 2019 IEEE 15th Brazilian Power Electronics Conference and 5th IEEE Southern Power Electronics Conference (COBEP/SPEC) on Santos, SP, Brazil, 1–4 December 2019.

Abstract: High-performance motor drives that operate in harsh conditions require an accurate and robust angular position measurement to correctly estimate the speed and reduce the torque ripple produced by angular estimation error. For that reason, a resolver is used in motor drives as a position sensor due to its robustness. A resolver-to-digital converter (RDC) is an observer used to get the angular position from the resolver signals. Most RDCs are based on angle tracking observers (ATOs). On the other hand, generalized predictive control (GPC) has become a powerful tool in developing controllers and observers for industrial applications. However, no GPC-based RDC with zero steady-state error during constant speed operation was proposed. This paper proposes an RDC based on a second-order difference GPC (SOD-GPC). In SOD-GPC, the second-order difference operator is applied to design a GPC model with two embedded integrators. Thus, the SOD-GPC is used to design a type-II ATO whose steady-state angle estimation error tends to zero during constant speed operation. Simulation and experimental results prove that the proposed RDC system has better performance than other literature approaches.

Keywords: angle tracking observer, generalized predictive control, resolver, resolver-to-digital converter, tracking.

---

1. Introduction

High-performance motor drives require a robust and accurate motor shaft angular position measurement to operate efficiently in harsh conditions [1–3]. The angular position is used to estimate the motor shaft speed and perform vector control techniques [4–6]. A high error in the angular position estimation produces a high torque ripple that may lead to motor drive malfunctions [6]. Many sensorless techniques were proposed to estimate the angular position without using position sensors [7–10]. However, most sensorless techniques depend on motor parameters whose values cannot be measured with accuracy. Hence, sensorless techniques have angular estimation errors [11]. As a result, many applications such as electric vehicles, antennas, and servomechanism still depend on sensors to guarantee precise angular position information [11–16].

Many types of angular position sensors were developed for industrial applications [12–16]: hall sensors, magnetoresistive sensors, capacitive sensors, inductive sensors, encoders, resolver, and others. Among these sensors, the resolver is widely applied in harsh environments as it can

resist high temperatures, vibrations, shocks, and contamination produced by oil, dust and mist [17–20]. Resolver receives a high-frequency excitation voltage and generates two amplitude-modulated voltages that depend on the angular position.

An observer called resolver-to-digital converters (RDC) estimates the angular position from resolver outputs. Many RDCs were proposed in literature [21–28]. These approaches can be divided into open-loop RDCs and closed-loop RDCs. Closed-loop RDCs are more robust to noise and are usually based on an angle tracking observer (ATO): a closed-loop estimator that reduces the difference between the actual and the estimated angular position [23–28].

On the other hand, model predictive control (MPC) becomes an interesting research topic due to its fast response and robustness [29–32]. In the '60s, the modern control theory began to diffuse by the need to control more complex plants [33]. Interest in MPC begins in the late 1970s with the emergence of several papers and consolidating MPC in the industry [34]. This interest existed for **its behavior** in the time domain, and its robustness since the industrial engineers tend to use robust control [32]. Clarke, in [35,36], presented a generalized predictive control (GPC) model, one of MPC's robust methods [31]. The objective of GPC is to calculate a sequence of future control signals that minimize a cost function that represents the system performance [31,32]. That cost function is defined over a prediction window (a set of predicted plant responses). An augmented model of the plant is used to predict the plant responses. In recent works, predictive control is applied in driving systems [37–39], microgrids [40–42], HVAC systems [43,44], and other applications.

In [45], it was proposed for the first time the application of the conventional GPC described in [32] in the development of an ATO. In that approach, the ATO was modeled as a tracking system where the reference is the actual angular position, and the system output is the estimated angle. Thus, the angle estimation is represented as a tracking problem. However, the ATO proposed in [45] has a steady-state angle estimation error during constant speed operation. To explain this error, let define $R(s)$, $G(s)$ and $C(s)$ be, respectively, the transfer function of the reference, the plant transfer function, and the controller. According to the internal model principle, if the product $G(s)C(s)$ contains $R(s)$, then the plant output will asymptotically track the reference [46]. The angular position during constant speed operation is modeled as a ramp signal ($\theta(t) = \omega t$). Besides, the Laplace transform of a ramp signal is $1/s^2$. However, the conventional GPC system used in [45] has only one embedded integrator ($1/s$). Furthermore, the GPC cost function used in [45] considers that the reference signal is constant for the predicted outputs, which is false when the reference is a ramp signal. As a result, the ATO in [45] has an estimation error during constant speed operation.

Adaptations of GPC for ramp reference tracking were proposed [47–49]. Most of them require information about the reference, and the solution of their cost functions to get their control law is not straightforward. **However, Cordero et al. proposed in [49] a simple adaptation of the conventional GPC described in [32] for ramp reference tracking**. That approach applies the second-order difference operator to create an augmented prediction model with two embedded integrators. Thus, according to the internal model principle, the GPC approach in [49] asymptotically tracks ramp references. Besides, the prediction window in [49] is composed of predicted tracking errors. The desired value of an error is always zero; then, the prediction window reference vector is a zero set (i.e., a constant reference). Thus, the optimization and the receding horizon techniques used in [32] can be used to get the control law for ramp reference tracking. In this paper, to simplify the notation, the GPC approach described in [49] is called second-order difference generalized predictive control (SOD-GPC).

This paper proposes an ATO based on the second-order difference generalized predictive control (SOD-GPC) described in [49], for developing a system with the advantages of the GPC and zero steady-state angle estimation error during constant speed operation. As in [45], the ATO is modeled as a tracking system whose reference is the actual angular position and whose output is the estimated angle. During constant speed operation, the angular position (the reference) is a ramp signal. Thus, the SOD-GPC can be used for the angle estimation error to be asymptotically zero during constant speed operation.

The remainder of the paper is organized as follows. Section II describes the structure of the resolver and GPC. The proposed ATO is described in Section III. Simulation and experimental results shown in Section IV **prove that the proposed ATO has a faster response than other ATOs in literature, without reducing its robustness against noise**. Finally, conclusions are outlined. In this paper, $A \in \Re^{n \times m}$ denotes that A is a $n \times m$ matrix.

2. Theoretical Foundations

2.1. Resolver

Figure 1 shows the schematics of the resolver. **This sensor consists of an excitation winding and two output windings**. The excitation winding is coupled to the motor shaft and receives a sinusoidal excitation voltage $v_e(t)$ of several kHz (1-10 kHz) [21–28]:

$$v_e(t) = a_r cos(\omega_r t), \tag{1}$$

where a_r is the excitation voltage amplitude (in V), $\omega_r = 2\pi f_r$ and f_r is the excitation voltage frequency (in Hz). Two amplitude-modulated voltages $v_s(t)$ and $v_c(t)$ **are induced** in the output windings. Those signals depend on the angular position [24]:

$$v_s(t) = k_r v_e(t) sin(\theta(t)), \tag{2}$$

$$v_c(t) = k_r v_e(t) cos(\theta(t)). \tag{3}$$

where $\theta(t)$ is the angular position to be measured, while k_r is the transformation ratio. The objective of a resolver-to-digital converter (RDC) is to get the angular position from $v_e(t)$, $v_s(t)$ and $v_c(t)$.

2.2. Difference Operation

Let $a(k)$ and $b(k)$ be discrete-time signals (vectors or scalars). The first-order difference operator (Δ) and the second-order difference operator (Δ^2) are defined as follows [50]:

$$\Delta a(k) = a(k) - a(k-1), \tag{4}$$

$$\Delta^2 a(k) = \Delta[\Delta a(k)] = \Delta a(k) - \Delta a(k-1). \tag{5}$$

Let M_1 and M_2 be matrices such as $M_1 a(k) + M_2 b(k)$ exists. The aforementioned operators are linear [49,50]:

$$\Delta[M_1 a(k) + M_2 b(k)] = M_1 \Delta a(k) + M_2 \Delta b(k), \tag{6}$$

$$\Delta^2[M_1 a(k) + M_2 b(k)] = M_1 \Delta^2 a(k) + M_2 \Delta^2 b(k). \tag{7}$$

The operator Δ corresponds to the discrete transfer function $C(z) = 1 - z^{-1}$ [32].
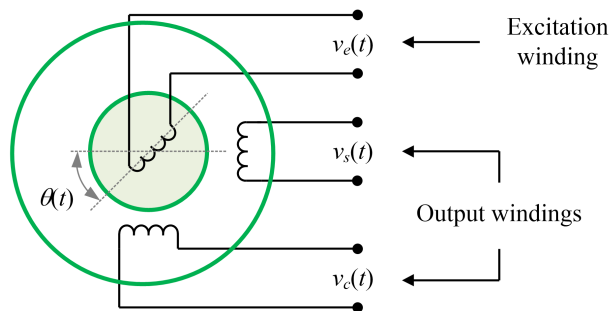
Figure 1. Schematic diagram of the resolver sensor.

91   *2.3. Conventional GPC*

Consider that a n-order SISO (single-input single-output) plant has the following discrete-time space-state model:

$$x_m(k+1) = A_m x_m(k) + B_m u(k), \tag{8}$$

$$y(k) = C_m x_m(k), \tag{9}$$

92   where $u(k)$ is the plant input, $y(k)$ is the system output, $x_m(k) \in \Re^{n \times 1}$ is the plant state vector, while
93   $A_m \in \Re^{n \times n}$, $B_m \in \Re^{n \times 1}$ and $C_m \in \Re^{1 \times n}$. Let consider that the plant does not have an embedded
94   integrator. The control system must have an integrator so that the steady-state error is zero for a step
95   reference. Conventional GPC requires the development of an augmented model with an embedded
96   integrator. This model will be used to predict plant responses.
97   Applying the first-order difference operator (Δ) into (8) and (9), and based on (6), yields:

$$\begin{aligned} \Delta x_m(k+1) &= \Delta[A_m x_m(k) + B_m u(k)] \\ &= A_m \Delta x_m(k) + B_m \Delta u(k), \end{aligned} \tag{10}$$

$$\begin{aligned} \Delta y(k) &= \Delta[C_m x_m(k)] \\ &= C_m \Delta x_m(k), \end{aligned} \tag{11}$$

98   where $\Delta x_m(k) = x_m(k) - x_m(k-1)$, $\Delta u(k) = u(k) - u(k-1)$ and $\Delta y(k) = y(k) - y(k-1)$. Note that
99   $y(k+1) = y(k) + \Delta y(k+1)$ and $y(k+1) = C_m \Delta x_m(k+1)$. Based on these results and (10), $y(k+1)$
100   is defined in terms of $\Delta x_m(k)$, $\Delta u(k)$ and $y(k)$ [32]:

$$\begin{aligned} y(k+1) &= y(k) + \Delta y(k+1) \\ &= y(k) + C_m \Delta x_m(k+1) \\ &= y(k) + C_m A_m \Delta x_m(k) + C_m B_m \Delta u(k). \end{aligned} \tag{12}$$

The augmented model defined in (13) is obtained by placing (10) and (12) together in matrix form [32]:

$$\begin{bmatrix} \Delta x_m(k+1) \\ y(k+1) \end{bmatrix} = \overbrace{\begin{bmatrix} A_m & o_m^T \\ C_m A_m & 1 \end{bmatrix}}^{A} \overbrace{\begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix}}^{x(k)} + \overbrace{\begin{bmatrix} B_m \\ C_m B_m \end{bmatrix}}^{B} \Delta u(k)$$

$$y(k) = \overbrace{\begin{bmatrix} o_m & 1 \end{bmatrix}}^{C} \overbrace{\begin{bmatrix} \Delta x_m(k) \\ y(k) \end{bmatrix}}^{x(k)}, \tag{13}$$

where A, B and C, in (13), are the matrices of the augmented model, while $o_m \in \Re^{1 \times n}$ vector composed of zeros. This augmented model allows predicting the plant responses. Consider that the state vector at the instant k_i, $x(k_i)$, is known. The future control trajectory is defined as:

$$\Delta u(k_i), \ \Delta u(k_i+1), \cdots, \ \Delta u(k_i + N_c - 1), \tag{14}$$

where N_c is a variable called the control horizon. On the other hand, the future state variables are:

$$x(k_i+1|k_i), \ x(k_i+2|k_i), \cdots, \ x(k_i + N_p|k_i), \tag{15}$$

where $x(k_i + j|k_i)$ is the predicted state vector at instant $k_i + j$ based on $x(k_i)$, while N_p is the prediction horizon, being $N_c \leq N_p$. The value of N_p defines the length of the prediction window [32]. In GPC, the future state variables are predicted using the control future trajectory and $x(k_i)$ [32]:

$$
\begin{aligned}
x(k_i + 1|k_i) &= Ax(k_i) + B\Delta u(k_i), \\
x(k_i + 2|k_i) &= A^2 x(k_i) + AB\Delta u(k_i) + B\Delta u(k_i + 1), \\
&\vdots \\
x(k_i + N_p|k_i) &= A^{N_p} x(k_i) + A^{N_p - 1} B\Delta u(k_i) \\
&+ A^{N_p k - 2} B\Delta u(k_i + 1) + \cdots + A^{N_p - N_c} B\Delta u(k_i + N_c - 1).
\end{aligned}
\tag{16}
$$

As $y(k) = C_m x(k)$, the plant outputs are predicted based on (16):

$$
\begin{aligned}
y(k_i + 1|k_i) &= CAx(k_i) + CB\Delta u(k_i), \\
y(k_i + 2|k_i) &= CA^2 x(k_i) + CAB\Delta u(k_i) + CB\Delta u(k_i + 1), \\
&\vdots \\
y(k_i + N_p|k_i) &= CA^{N_p} x(k_i) + CA^{N_p - 1} B\Delta u(k_i) \\
&+ CA^{N_p - 2} B\Delta u(k_i + 1) + \cdots + CA^{N_p - N_c} B\Delta u(k_i + N_c - 1).
\end{aligned}
\tag{17}
$$

Putting (17) into a matrix form, yields:

$$
Y = Fx(k_i) + \phi U,
\tag{18}
$$

where $Y \in \Re^{N_p \times 1}$ is vector with the predicted responses, $U \in \Re^{N_c \times 1}$ is the vector with the future control actions, while F and ϕ are defined in (21) :

$$
Y = \begin{bmatrix} y(k_i + 1|k_i) & y(k_i + 2|k_i) & \cdots & y(k_i + N_p|k_i) \end{bmatrix}^T,
\tag{19}
$$

$$
U = \begin{bmatrix} \Delta u(k_i) & \Delta u(k_i + 1) & \cdots & \Delta u(k_i + N_c - 1) \end{bmatrix}^T,
\tag{20}
$$

$$
F = \begin{bmatrix} CA \\ CA^2 \\ \vdots \\ CA^{N_p} \end{bmatrix}, \quad
\phi = \begin{bmatrix}
CB & 0 & \cdots & 0 \\
CAB & CA & \cdots & 0 \\
\vdots & \vdots & \cdots & 0 \\
CA^{N_p - 1} & CA^{N_p - 1} & \cdots & CA^{N_p - N_c}
\end{bmatrix}.
\tag{21}
$$

Let $r(k)$ be the reference signal of the plant. The objective of GPC is to calculate an optimal vector of U that minimizes the error between the reference and the predicted plant outputs. Let $R_s \in \Re^{N_p \times 1}$ be a vector that contains the references of the predicted responses that composes Y. It is considered that R_S is constant inside the prediction window (the set of predictions) [32]. Equation (22) defines the reference vector R_s according to [32] :

$$
R_S^T = \underbrace{\begin{bmatrix} 1 & 1 & \cdots, 1 \end{bmatrix}}_{N_p} r(k_i).
\tag{22}
$$

The cost function J, which reflects the control system performance, is defined in [32] as:

$$
J = (R_S - Y)^T (R_S - Y) + U^T \overline{R}_w U, \quad \overline{R}_w = r_w I_c,
\tag{23}
$$

where $I_c \in \Re^{N_c \times N_c}$ is an identity matrix, while r_w is a tuning parameter used to control the magnitude of U [32]. Equation (24) shows the value of $\frac{\partial J}{\partial U}$. The optimal solution of (23) makes

$\frac{\partial J}{\partial U} = 0$ [32]. **By replacing (18) into (24), and after mathematical manipulations, the optimal solution** U **is defined as indicated in (25) [32]:**

$$\frac{\partial J}{\partial U} = -2\phi^T(R_S - Fx(k_i)) + 2(\phi^T\phi + \overline{R}_w)U = 0. \tag{24}$$

$$U = (\phi^T\phi + \overline{R}_w)^{-1}\phi^T(R_S - Fx(k_i)). \tag{25}$$

According to receding horizon approach, only the first element of the optimal control trajectory, i.e., $\Delta u(k_i)$, **is used to calculate the control law [32]. Thus:**

$$
\begin{aligned}
\Delta u(k_i) &= \overbrace{\begin{bmatrix} 1 & 0 & \cdots 0 \end{bmatrix}}^{N_c} (\phi^T\phi + \overline{R}_w)^{-1}(\phi^T R_S - \phi^T Fx(k_i)) \\
&= K_y r(k_i) - K_{mpc}x(k_i),
\end{aligned}
\tag{26}
$$

where K_y is the first element of $(\phi^T\phi + \overline{R}_w)^{-1}(\phi^T R_S)$ and K_{mpc} is the first line of $(\phi^T\phi + \overline{R}_w)^{-1}(\phi^T Fx(k_i))$. The gain vector K_{mpc} can be expressed as $K_{mpc} = [K_x \ K_y]$, where $K_y \in \Re^{1\times1}$ is the last element of K_{mpc}, while $K_x \in \Re^{n\times1}$. Based on (26) and the definition of $x(k)$ in (13), it is possible to prove [32]:

$$\Delta u(k_i) = K_y(r(k_i) - y(k_i)) - K_x\Delta x_m(k_i). \tag{27}$$

The closed-loop GPC system described in [32] can be described as shown in Figure 2, where z^{-1} block indicates the discrete delay operator, $1 - z^{-1}$ is the discrete transfer function equivalent to the Δ operator, while $\frac{1}{1-z^{-1}}$ corresponds to the discrete-time integrator. The plant input is obtained by integrating $\Delta u(k_i)$ in (27).

3. Proposed RDC System based on SOD-GPC

3.1. Second Order Difference GPC (SOD-GPC)

According to the internal model principle, if the plant has no integrator, then the controller must have two embedded integrators to track ramp references [46]. However, the conventional GPC approach in [32] only has one embedded integrator. Besides, the cost function of the GPC described in (23) considers that the prediction window reference vector (R_S) is constant within the prediction window [32]. However, the reference vector defined in (22) is not constant when the plant reference $r(k)$ is a ramp signal.

Cordero et al. proposed in [49] an adaptation of the GPC algorithm for track reference tracking. This approach is based on applying the second-order difference operator in (4) to create an augmented model with two embedded integrators and whose output is the tracking error. For simplicity sake, the GPC approach in [49] will be called second-order difference GPC (SOD-GPC) in this paper.
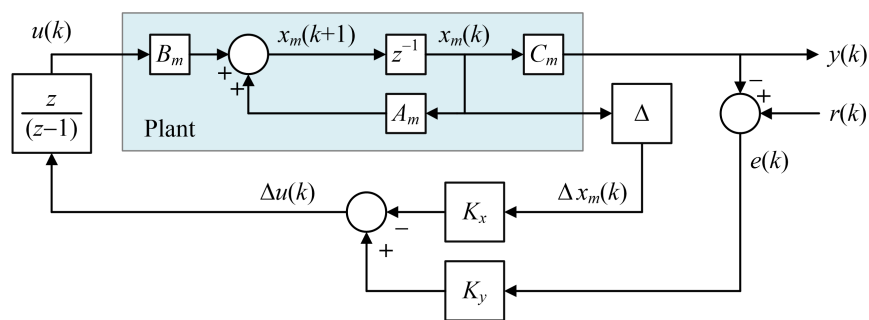
Figure 2. Conventional Generalized Predictive Control described in [32].

Applying the second-order difference operator into (8), and as this operator is linear, yields:

$$\Delta^2 x_m(k+1) = C_m A_m \Delta^2 x_m(k) + C_m B_m \Delta^2 u(k). \tag{28}$$

Let $e(k)$ be the tracking error:

$$e(k) = r(k) - y(k). \tag{29}$$

Hence

$$e(k+1) = r(k+1) - y(k+1). \tag{30}$$

Applying the first-order difference operator into (12), (29) and (30) yields [49]:

$$\Delta e(k) = \Delta r(k) - \Delta y(k), \tag{31}$$

$$\Delta e(k+1) = \Delta r(k+1) - \Delta y(k+1), \tag{32}$$

124

$$\Delta y(k+1) = \Delta \left[y(k) + C_m A_m \Delta x_m(k) + C_m B_m \Delta u(k) \right] \tag{33}$$
$$= \Delta y(k) + C_m A_m \Delta^2 x_m(k) + C_m B_m \Delta^2 u(k).$$

On the other hand, let $r(k) = \alpha k + \beta$ be the ramp reference signal. Observe that $\Delta r(k) = r(k) - r(k-1) = \alpha$. Besides, $\Delta r(k+1) = r(k+1) - r(k) = \alpha$. Therefore:

$$\Delta r(k+1) = \Delta r(k). \tag{34}$$

125   Replacing (31), (33) and (34) into (32), yields

$$\Delta e(k+1) = \Delta r(k) - \left[\Delta y(k) + C_m A_m \Delta^2 x_m(k) + C_m B_m \Delta^2 u(k) \right] \tag{35}$$
$$= \Delta e(k) - C_m A_m \Delta^2 x_m(k) - C_m B_m \Delta^2 u(k).$$

Observe that $\Delta e(k+1) = e(k+1) - e(k)$. Hence:

$$e(k+1) = e(k) + \Delta e(k+1). \tag{36}$$

126   Replacing (35) into (36), yields:

$$e(k+1) = e(k) + \Delta e(k+1) \tag{37}$$
$$= e(k) + \Delta e(k) - C_m A_m \Delta^2 x_m(k) - C_m B_m \Delta^2 u(k).$$

The augmented model proposed defined in (38) is obtained from (28), (35) and (37) [49]:

$$x(k+1) = A_2 x(k) + B_2 \Delta^2 u(k),$$
$$\text{Output: } e(k) = C_2 x(k), \tag{38}$$

where

$$x(k) = \begin{bmatrix} \Delta^2 x_m(k) \\ \Delta e(k) \\ e(k) \end{bmatrix}, \quad A_2 = \begin{bmatrix} A_m & o_o^T & o_o^T \\ -C_m A_m & 1 & o_o^T \\ -C_m A_m & 1 & 1 \end{bmatrix}, \quad B_2 = \begin{bmatrix} B_m \\ -C_m B_m \\ -C_m B_m \end{bmatrix},$$
$$C_2 = \begin{bmatrix} o_o & 0 & 1 \end{bmatrix}. \tag{39}$$

127   Note that the input and the output of (38) are $\Delta^2 u(k)$ and $e(k)$, respectively. In [49], the prediction
128   window is composed by the predicted tracking errors $e(k_i+1|k_i), e(k_i+2|k_i), \cdots, e(k_i+N_p|k_i)$, while
129   the future control trajectory is $\Delta^2 u(k_i), \Delta^2 u(k_i+1), \cdots, \Delta^2 u(k_i+N_c-1)$. Thus, the vector with the

130　predicted response (Y_a) and the vector with the future control signals (U_a) for the SOD-GPC system
131　proposed in [49] are defined as follows:

$$Y_a = \left[e(k_i + 1|k_i) \quad e(k_i + 2|k_i) \quad \cdots \quad e(k_i + N_p|k_i) \right]^T,　\quad (40)$$

$$U_a = \left[\Delta^2 u(k_i) \quad \Delta^2 u(k_i + 1) \quad \cdots \quad \Delta^2 u(k_i + N_c - 1) \right]^T.　\quad (41)$$

The proposed augmented model in (38) has a similar structure of the GPC augmented model in (13). Thus, it is possible to apply the method defined in (18) **to predict the tracking error** (the outputs of the augmented mode) in (38):

$$Y_a = Fx(k_i) + \phi U_a,　\quad (42)$$

132　**where F and ϕ are defined in (21) but replacing the matrices A, B and C by A_2, B_2 and C_2 in (39),**
133　**respectively.**
134　　Observe that the desired values of the predicted errors that composes Y_a is zero. Therefore, the
135　reference vector is a set of zeros, i.e., $R_S = 0$, which is constant. Replacing replacing Y, U and R_S by Y_a,
136　U_a and 0 into (23) and (24), yields [49]:

$$J = Y_a^T Y_a + U_a^T \overline{R}_w U_a,　\quad (43)$$

$$\frac{\partial J}{\partial U_a} = 2\phi^T Fx(k_i) + 2(\phi^T \phi + \overline{R}_w) U_a = 0.　\quad (44)$$

The optimal solution U_a for the SOD-GPC makes $\frac{\partial J}{\partial U_a} = 0$ in (44). This solution is:

$$U_a = -(\phi^T \phi + \overline{R}_w)^{-1} \phi^T Fx(k_i).　\quad (45)$$

Equation (45) defines the optimal control trajectory of the SOD-GPC. According to the receding horizon control principle, only the first element of U_a in (45), i.e., $\Delta^2 u(k_i)$, is used to calculate the control law. Thus:

$$\Delta^2 u(k_i) = - \overbrace{\left[1 \quad 0 \quad \cdots 0 \right]}^{N_c} (\phi^T \phi + \overline{R})^{-1} \phi^T Fx(k_i).　\quad (46)$$

137　　The plant input is obtained by integrating $\Delta^2 u(k_i)$ in (46) twice. Figure 3 shows the structure of
138　the SOD-GPC system described in [49]. The dynamics of the control system depends on the tuning of
139　N_p, N_c and $\overline{R}_w$.

$$\Delta u(k_i) = \Delta^2 u(k_i) + \Delta u(k_i - 1), \quad u(k_i) = \Delta u(k_i) + u(k_i - 1).　\quad (47)$$
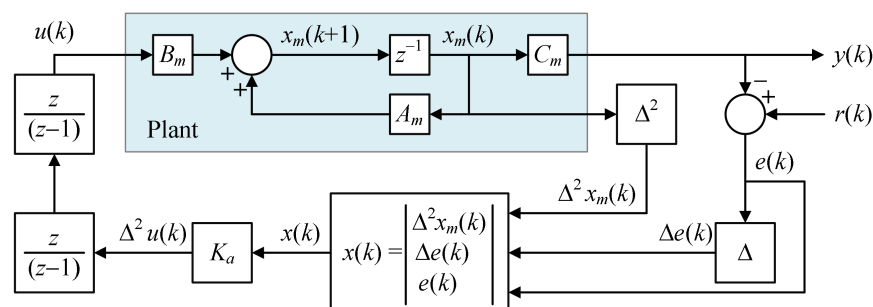
Figure 3. Schematic of the Second-Order Difference GPC system (SOD-GPC).

Let $G(z)$ and $G_a(z)$ be the plant's discrete transfer functions and the SOD-GPC augmented model. It is proved in [49]:

$$G_a = - \left[\frac{z}{z-1}\right]^2 G(z). \tag{48}$$

Equation (48) proves that the augmented model in (38) has two embedded integrators, when the plant has no zeros at $z = 1$. Hence, the SOD-GPC can asymptotically track ramp references [49].

3.2. Proposed GPC-Based ATO

Figure 4 shows the structure of the proposed ATO based on SOD-GPC. **Let consider that the resolver excitation signal with frequency $\omega_r = 2\pi f_r$ is generated by the RDC. Thus, the value of this signal is known. Let $t = t_s k$, where t is the discretized time, t_s is the sampling time, $f_s = \frac{1}{t_s}$ is the sampling frequency, while k denotes the k-th sample. Let define ω_{rs} as the normalized resolver excitation frequency:**

$$\omega_{rs} = \omega_r t_s = \frac{2\pi f_r}{f_s}. \tag{49}$$

Let $\theta(k) = \theta(kt_s)$ and $\theta_e(k) = \theta_e(kt_s)$ be the actual and the estimated angle. Thus, **based on (49),** the resolver signals in (1), (2) and (3) can be discretized as follows:

$$
\begin{aligned}
v_e(k) &= a_r cos(\omega_r t_s k) = a_r cos(\omega_{rs} k), & (50) \\
v_s(k) &= k_r v_e(k) sin(\theta(k)), & (51) \\
v_c(k) &= k_r v_e(k) cos(\theta(k)), & (52)
\end{aligned}
$$

According to Figure 4, the term $g(k)$ is:

$$g(k) = \frac{2}{k_r a_r^2} \left[v_s(k)cos(\theta_e(k)) - v_c(k)sin(\theta_e(k))\right] v_e(k). \tag{53}$$

Replacing (50), (51) and (52) into (53), and applying trigonometry, yields:
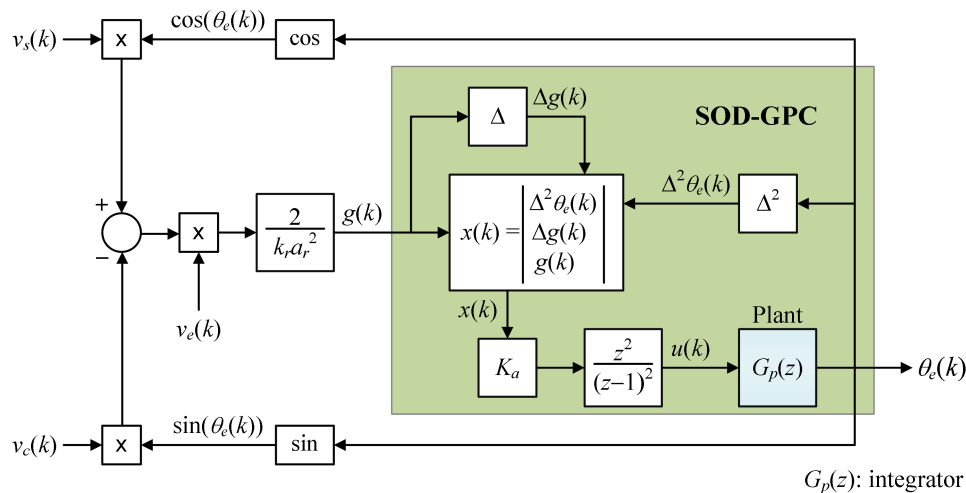
Figure 4. Proposed ATO based on SOD-GPC.

$$
\begin{aligned}
g(k) &= \frac{2}{k_r a_r^2}\left[k_r v_e(k)sin(\theta(k))cos(\theta_e(k)) - k_r v_e(k)cos(\theta(k))sin(\theta_e(k))\right]v_e(k) \quad (54)\\
&= \frac{2}{a_r^2}\left[sin(\theta(k))cos(\theta_e(k)) - cos(\theta(k))sin(\theta_e(k))\right]v_e^2(k)\\
&= \frac{2}{a_r^2}sin(\theta(k)-\theta_e(k))a_r^2 cos^2(\omega_{rs}k)\\
&= sin(e_\theta(k))\left[1 + cos(2\omega_{rs}k)\right]\\
&= sin(e_\theta(k)) + sin(e_\theta(k))cos(2\omega_r k),
\end{aligned}
$$

where $e_\theta(k) = \theta(k) - \theta_e(k)$ is the angle estimation error and assuming that $e_\theta(k) \approx 0$, then $sin(e_\theta(k)) \approx e_\theta(k)$. Thus, (54) is rewritten as follows:

$$
g(k) \approx e_\theta(k) + e_\theta(k)cos(2\omega_{rs}k). \quad (55)
$$

Let $Z[s(k)]$ denotes the Z-transform of the discrete signal $s(k)$. Let $H(z)$ be the transfer function of the SOD-GPC system (including the integrator $G_p(z)$) shown in Figure 4. $H(z)$ represents the relationship between $Z[\theta_e(k)]$ and $Z[g(k)]$, i.e., $H(z) = \frac{Z[\theta_e(k)]}{Z[g(k)]}$. Let $h(k)$ be the impulse response of the SOD-GPC system, i.e., $Z[h(k)] = H(z)$. Hence [51]:

$$
Z[\theta_e(k)] = H(z)Z[g(k)] \quad \rightarrow \quad \theta_e(k) = h(k)*g(k), \quad (56)
$$

where * denotes convolution. As the SOD-GPC is a linear system, it is possible to apply the superposition principle to analyze the ATO response: the estimated angular position (the ATO output) is the summation of the responses produced by $e_\theta(k)$ and $e_\theta(k)cos(2\omega_{rs}k)$:

$$
\begin{aligned}
\theta_e(k) &= h(k)*g(k) \quad (57)\\
&= h(k)*[e_\theta(k) + e_\theta(k)cos(2\omega_{rs}k)]\\
&= h(k)*e_\theta(k) + h(k)*[e_\theta(k)cos(2\omega_{rs}k)].
\end{aligned}
$$

Through frequency shifting property, it is possible to prove that the term $e_\theta(k)cos(2\omega_{rs}k)$ is a high-frequency signal, while the ATO acts as a low-pass filter that rejects high-frequency signals [28]. As a result, the effect of the signal $e_\theta(k)cos(2\omega_{rs}k)$ in the angle estimation will be rejected by the ATO:

$$
h(k)*[e_\theta(k)cos(2\omega_{rs}k)] \approx 0. \quad (58)
$$

Replacing (58) into (57) yields:

$$
\theta_e(k) \approx h(k)*e_\theta(k). \quad (59)
$$

Equation (59) states that the estimated angle mainly depends on the term $e_\theta(k)$. In consequence, the ATO can be projected only considering the existence of $e_\theta(k)$. Thus, the ATO can be approximated as the closed-loop system in Figure 5, where the reference is $\theta(k)$, the output is $\theta_e(k)$ and the SOD-GPC is the controller that reduce the error $e_\theta(k)$. Besides, the plant whose output will be defined by the SOD-GPC algorithm is the integrator represented by $G_p(z)$ in Figure 5. **An ATO is an observer that estimates the angular position. However, many approaches in literature model the ATO as a closed-loop system whose objective is to reduce the angle estimation error [24,27,28,45]. In the proposed ATO, the angle estimation error is obtained from the signal $g(k)$. The discrete-time model of the integrator is described through (8) and (9), where $A_m = 1$, $B_m = t_s$, $C_m = 1$ and $y(k) = x_m(k) =$**

$\theta_e(k)$. Besides, the closed-loop error ($e(k)$) is the angle estimation error, i.e., $e(k) = e_\theta(k)$. Replacing these terms into (39) yields:

$$x(k) = \begin{bmatrix} \Delta^2\theta_e(k) \\ \Delta e(k) \\ e(k) \end{bmatrix}, \quad A_2 = \begin{bmatrix} 1 & 0 & 0 \\ -1 & 1 & 0 \\ -1 & 1 & 1 \end{bmatrix}, \quad B_2 = \begin{bmatrix} t_s \\ -t_s \\ -t_s \end{bmatrix}, \quad C_2 = \begin{bmatrix} 0 & 0 & 1 \end{bmatrix}. \tag{60}$$

The matrices A, B and C in (60) must be used to obtain the gain matrix K_a and the control law according to (21), (46) and (47). It is required the adequate tuning of the parameters N_p, N_c, and $\overline{R}_w$ to define the transient behavior of the ATO.

As $G_p(z)$ **in Figure 4 is an integrator, then the integral of $u(k)$ is the estimated position $\theta_e(k)$. Hence, the signal $u(k)$ produced by the SOD-GPC system is the estimated angular speed. However, if the noise power in the resolver signals is high, $u(k)$ should be filtered to get a good speed estimation (the design of that filter is beyond the objectives of this research). As a result, the proposed ATO allows estimating the angular speed.**

3.3. Effect of Resolver Parameter Variations in the Angle Estimation

Equations (2) and (3) are the ideal representation of the resolver outputs. However, imperfections in the mechanical resolver structure and the signal conditioning circuits introduce distortions to the resolver outputs [52–55]. The main distortions that affect the angle estimation are:

- **Amplitude imbalance: amplitude difference between the resolver outputs.**
- **Imperfect quadrature: spatial misalignment between the two resolver phases.**

Other distortions, such as DC offset in the envelopes of the resolver outputs and harmonics, have small effect [52–55]. The resolver outputs can be modeled, considering the effects of the amplitude imbalance and the imperfect quadrature, as follows:

$$v_s(t) = k_r v_e(t)sin(\theta(t) + \beta), \tag{61}$$

$$v_c(t) = (1 + \alpha)k_r v_e(t)cos(\theta(t)). \tag{62}$$

where α and β are, respectively, the amount of amplitude imbalance and imperfect quadrature error. These distortions produce low-frequency angle estimation errors [52–55]. The compensation of the resolver imperfection effects is beyond the objective of this research. However, many calibration techniques were proposed in the literature to measure and compensate these distortions, which can be applied with the proposed ATO [52–55].
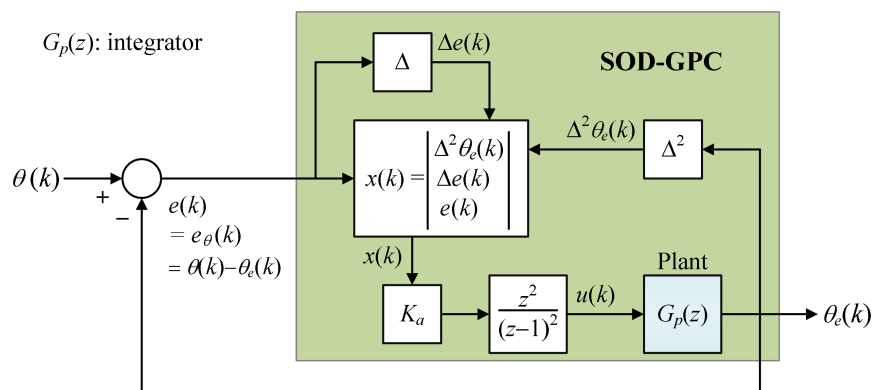
Figure 5. Simplified block diagram of the proposed ATO considering only the effect of $e_\theta(k)$ in the angle estimation.

4. Results

4.1. Simulation Results

A set of simulation tests was performed in **Simulink**. The sampling rate was set at 50 kHz. The parameters of the resolver sensor are listed in Table 1. The angular position **corresponds** to the speed curve shown in Figure 6. The proposed ATO was tested considering different values of N_p, N_c and $\overline{R}_w$:

- Configuration 1: $N_p = 102, N_c = 2, \overline{R}_w = 0.01$,
- Configuration 2: $N_p = 120, N_c = 2, \overline{R}_w = 0.01$,
- Configuration 3: $N_p = 102, N_c = 10, \overline{R}_w = 0.01$.

The proposed approach was compared with the ATO described in [56], and with an ATO where the SOD-GPC was replaced by a PI controller (PI-based ATO). Figures 7 and 8 illustrate the ATO in [56] and the PI-based ATO, respectively. The dynamic behavior of the ATO in Figure 7 depends on the solutions (λ_i) of the equation $(\lambda_i I - M) = 0$, where λ_i is the i-th eigenvalue of M, $i = 1, 2, 3$, I is the identity matrix, while $M = \begin{bmatrix} 0 & 0 & 0 \\ -1 & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} - \begin{bmatrix} 0.5\,(k_r a_r)^2 \\ 0 \\ 0 \end{bmatrix} \underbrace{[k_0 - k_1 - k_2]}_{K}$.

Ackermann formula was used to get the matrix K to set the eigenvalues of M. In this paper, those eigenvalues were set in $-100 + j100$, $-100 - j100$, and -500. These eigenvalues were selected after many simulations, trying to get a response similar to the proposed ATO. Observe that the complex-conjugated eigenvalues are dominant. The application of complex-conjugated eigenvalues to tune an ATO can be found in [28]. On the other hand, the PI regulator of the PI-based ATO in Figure 8 has the discrete transfer function $C(z) = \frac{500.52 \times (z-0.957)}{z-1}$. The PID

Table 1. Simulation Parameters of the Resolver Sensor.

Parameters	Values
Excitation amplitude (a_r)	8 V
Excitation frequency (f_r)	2.5 kHz
Transformation ratio (k_r)	0.5
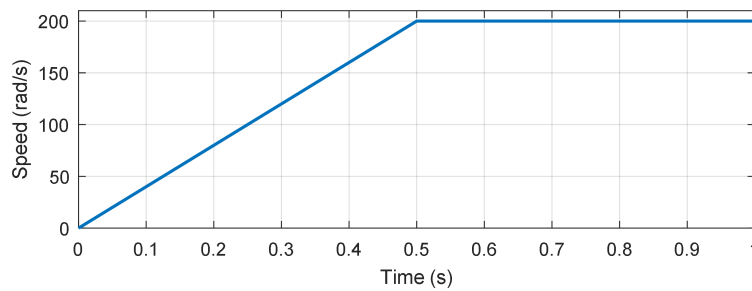
Figure 6. Speed curve used in the tests.
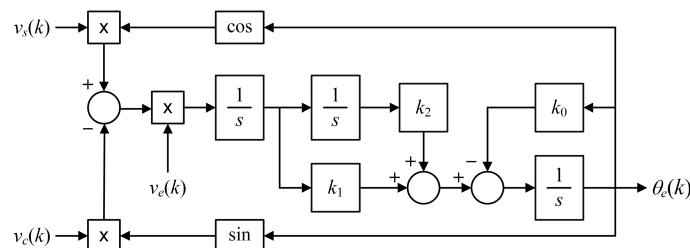
Figure 7. RDC system described in [56].
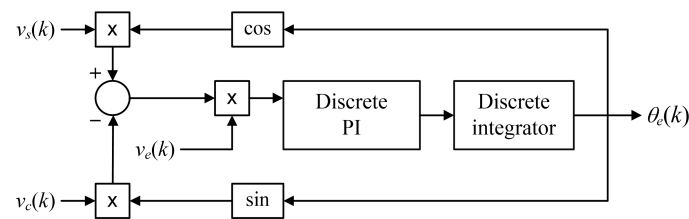
Figure 8. PI-based ATO.

tuning tool of MATLAB was used to set the parameters of the PI in order for the PI-based ATO and the proposed ATO to have a similar response. A sampling rate of 50 kHz was used for the discretization process. To avoid algebraic loops in the ATO models, each integrator whose output is the estimated angle was discretized through forward Euler approximation. In contrast, Backward Euler approximation was used to implement the other integrators.

Table 2 shows the number of multiplications and sums necessary to implement the SOD-GPC-based ATO, the ATO in [56], and the PI-based ATO, without considering the estimation of the trigonometric functions. The multiplication operation has a longer processing time than the sum operator. Table 2 shows that the SOD-GPC has eight multiplication operators in its structure, two more than the PI-based ATO, which is a low computational cost system due to the modeling's simplicity. On the other hand, the proposed ATO uses less multiplications than the ATO in [56]. The SOD-GPC uses one matrix multiplication, two accumulators $\left(\frac{z}{z-1}\right)$ (an accumulator uses only one sum) and the estimation of the difference operators (substractions). As a result, the SOD-GPC has low complexity and it can be easily implemented.

For each ATO configuration, two tests were performed: without adding noise to the resolver outputs, and adding a noise (zero mean and 0.0002 variance) to the resolver outputs. Figures 9, 10, 11, 12 and 13 show the angle estimation error for each simulation test.

Table 3 shows the root mean square error (RMSE) and the error settling time (the time required for the error signal to achieve its steady-state value) for each simulation test. **The errors without noise are negligible in all cases**. Thus, the main source of angular estimation error is the noise in the resolver outputs. As a result, the value of RMSE is a measurement of the robustness against noise of the ATO: the less the value of RMSE, the better the ATO robustness against noise. According to Table 3, it is possible to see that ATO in [56] has higher RMSE and settling time than the proposed approach. As a result, the proposed RDC based on SOD-GPC has a faster response and more

Table 2. Computational Cost of ATOs

ATO	Sum (+)	Multiplication (×)
SOD-GPC-based ATO	9	8
ATO in [56]	6	9
PI-based ATO	3	6

Table 3. RMSE and Settling Time in the Simulation Tests.

Configuration	RMSE (without noise)	RMSE (with noise)	Settling Time (s)
$N_p = 102, N_c = 2, \overline{R}_w = 0.01$	0.16×10^{-3}	0.48×10^{-3}	4.90×10^{-3}
$N_p = 120, N_c = 2, \overline{R}_w = 0.01$	0.20×10^{-3}	0.47×10^{-3}	5.10×10^{-3}
$N_p = 102, N_c = 10, \overline{R}_w = 0.01$	0.04×10^{-3}	0.68×10^{-3}	2.10×10^{-3}
ATO in [56]	0.61×10^{-3}	0.88×10^{-3}	55.0×10^{-3}
PI-based ATO	0.14×10^{-3}	0.45×10^{-3}	22.3×10^{-3}
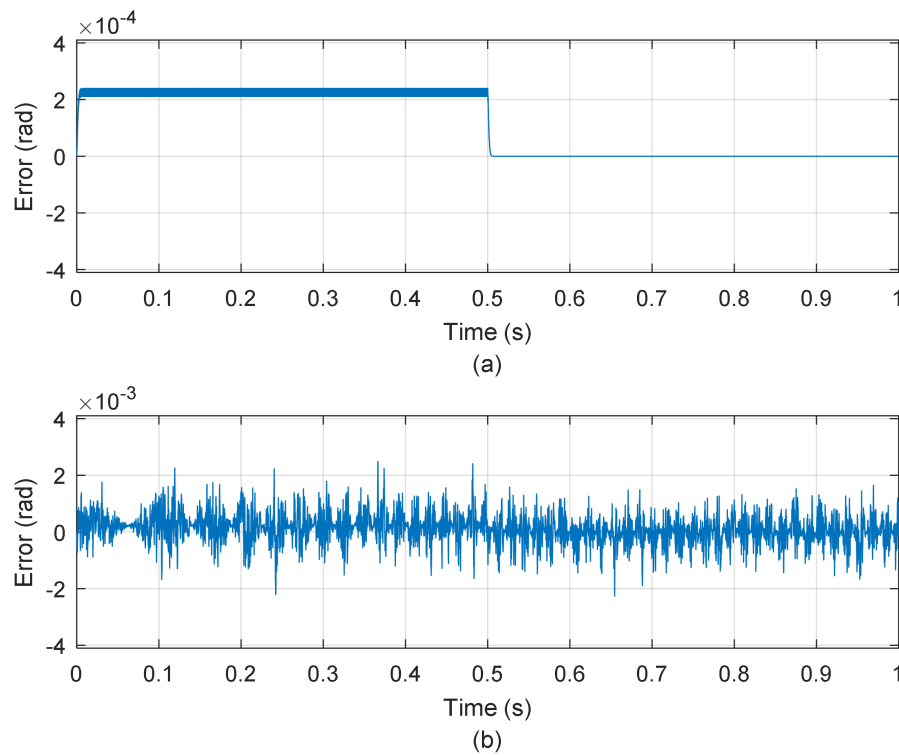
Figure 9. Simulation results for $N_p = 102, N_c = 2, \overline{R}_w = 0.01$. a) Without noise, b) With noise.
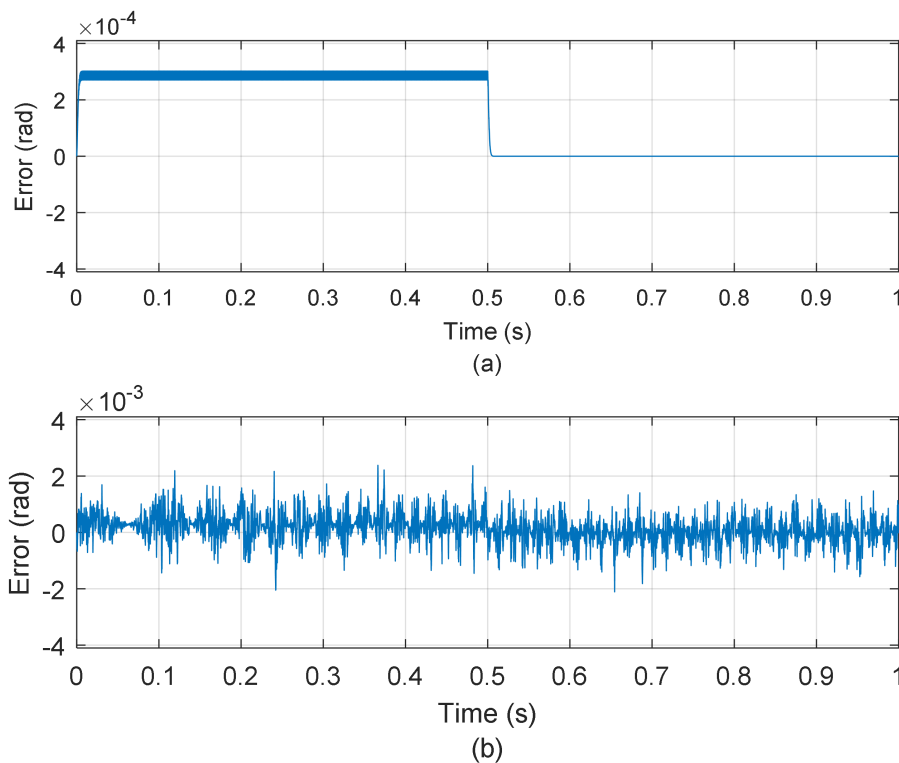
Figure 10. Simulation results for $N_p = 120, N_c = 2, \overline{R}_w = 0.01$. a) Without noise, b) With noise.

robustness against noise than the approach in [56]. Besides, the settling time for the proposed ATO is less than the value obtained for the PI-based ATO (both ATOs have similar RMSEs).
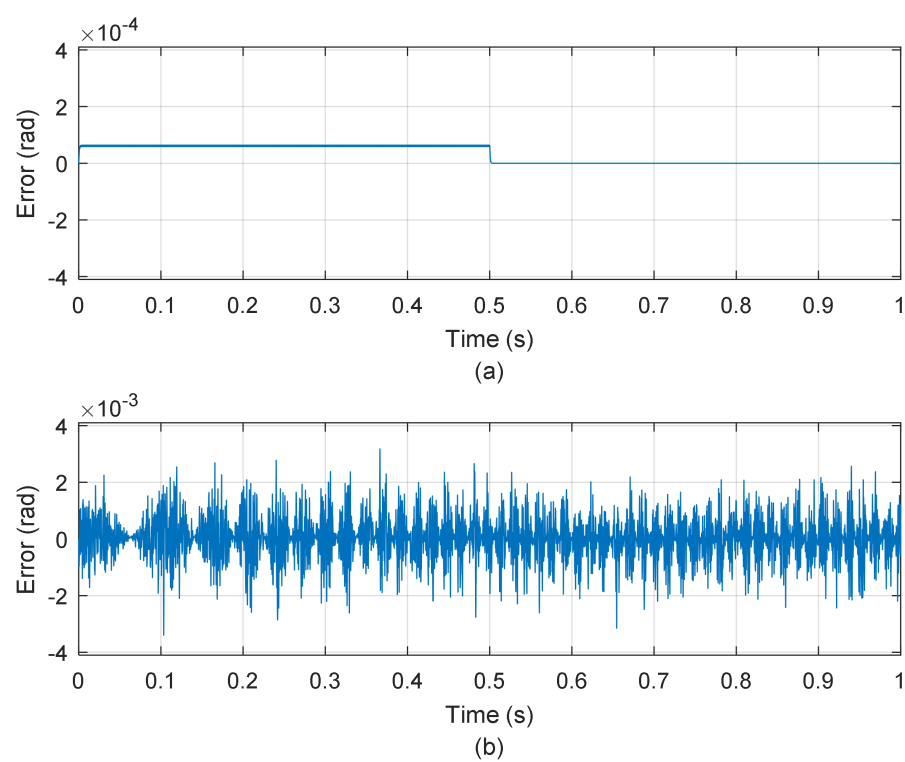
Figure 11. Simulation results for $N_p = 102, N_c = 10, \overline{R}_w = 0.01$. a) Without noise, b) With noise.
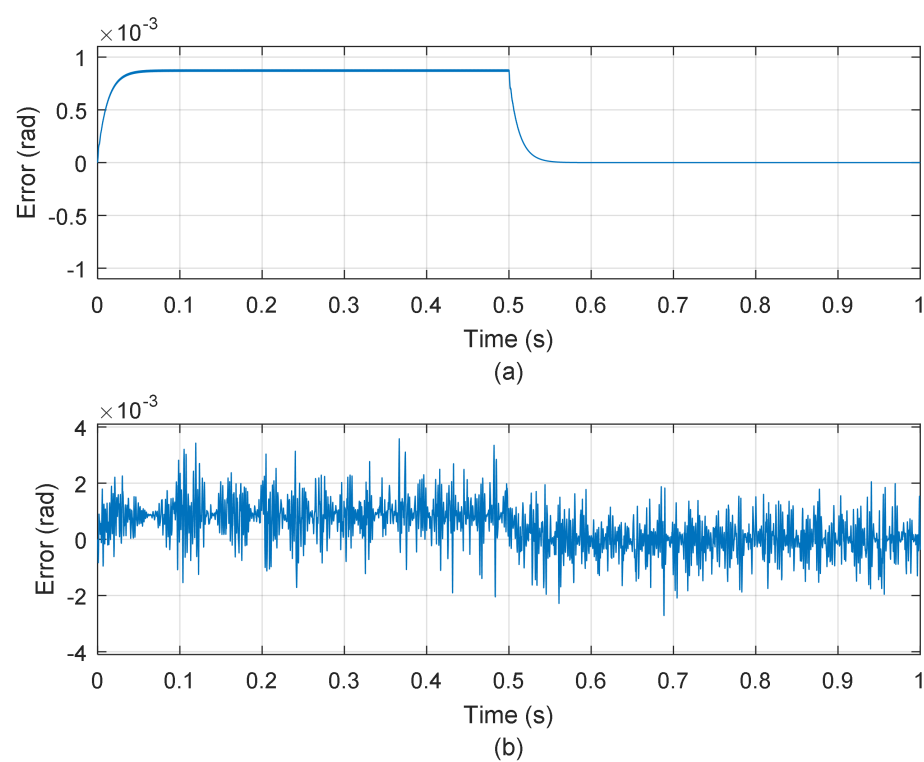
Figure 12. Simulation results for the ATO in [56]. a) Without noise, b) With noise.
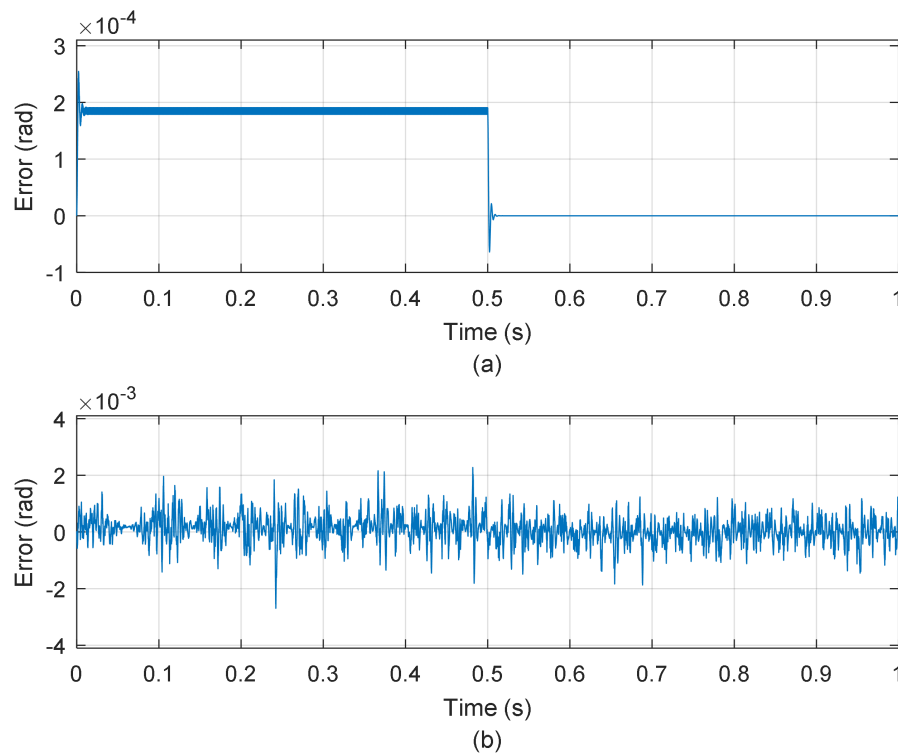
Figure 13. Simulation results for the PI-based ATO. a) Without noise, b) With noise.

4.2. Experimental Results

An experimental platform based on hardware-in-the-loop (HIL) was set up to test the proposed ATO. The experimental setup shown in Figure 14 consists of a PC and the FPGA Development and Education Board DE2-115 of ALTERA (based on the FPGA EP4CE115F29C7). An Ethernet cable allows communication between the FPGA board and the PC.

In this paper's HIL testing method, the resolver signals are generated through a Simulink block diagram in the PC. These signals are sent to the FPGA board through the Ethernet cable. The FPGA board receives these signals, estimates the angular position through the proposed ATO algorithm, and sends this estimation to the PC block diagram to calculate the angle estimation error. **One main advantage of using the HIL method is that the angular position is known in each instant. Thus, the**
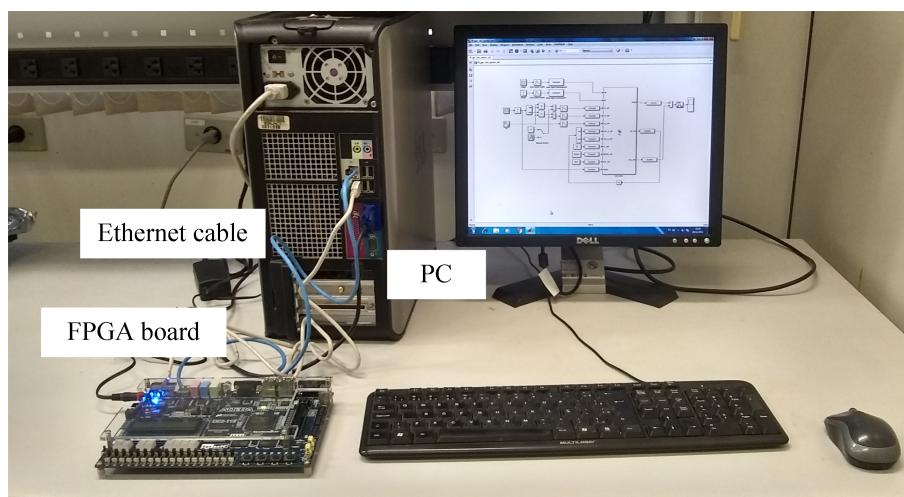
Figure 14. Experimental setup.

232 **angle estimation error can be calculated with accuracy without requiring high-precision sensors**
233 **(e.g., 16-bit absolute encoders, which are expensive). Furthermore, differently from simulations,**
234 **HIL tests allow evaluating the actual effects of implementation problems (e.g., processing delays**
235 **and the number of bits used in arithmetic operations) in the execution of an algorithm (the ATO**
236 **in this case) in a digital processor.** On the other hand, when a practical resolver is used, an additional
237 angular position sensor with better accuracy (which is difficult to obtain) is required to get the actual
238 angular position.

239  The proposed ATO was implemented in the FPGA board through a VHDL code, considering the
240 same signal sampling frequency (50 kHz) and the SOD-GPC parameters used in the simulation tests.
241 It was used as a fixed-point data format to perform the arithmetic operations with 32-bits. However,
242 the estimated angular position was represented using 40 bits (30 fractional bits) to better compare
243 the actual and the estimated angle. The emulated resolver signals are the same as those used in the
244 simulation tests.

245  It was not possible to properly implement the ATO in Figure 7 in the FPGA using 32 bits fixed-point
246 number representation. The main problem was the discretization of the integrators considering a
247 sampling rate of 50 kHz. On the other hand, the SOD-GPC is a discrete-time control system, being easier
248 to implement in a digital processor (no discretization is needed). As a result, the proposed RDC based
249 on a SOD-GPC based ATO has a fast response, robustness against noise, and suitable to be implemented
250 in digital processors such as FPGAs. **However, the PI-based ATO shown in Figure 8, whose structure**
251 **is simple, was implemented in the FPGA. The same PI regulator used in simulations,** $C(z) =$
252 $\frac{500.52(z-0.957)}{z-1}$**, was applied in the experimental tests. As in simulations, each integrator whose**
253 **output is the estimated angle was implemented in the FPGA through forward Euler approximation,**
254 **while Backward Euler approximation was used to implement the other integrators.**

255  Figures 15, 16, 17 and 18 show the results for the HIL tests. In the tests without noise, it is possible
256 to see a ripple in the angular position error during constant speed operation (from 0 s to 1 s). This
257 ripple is produced by the limited number of bits used for the arithmetic operations. However, that
258 ripple is negligible (that ripple has an amplitude less than 5×10^{-5} rad).

259  **Table 4 presents the RMSE and the settling time obtained in each HIL test. These values are**
260 **similar to those obtained in simulations. If the value of** N_c **rises, the ATO system will respond**
261 **faster, but RMSE will consequently be greater as well. Although the estimation error using PI has**
262 **a low RMSE, in the same order as the results obtained, the settling time for the PI-based ATO is**
263 **almost twice the value obtained for the proposed approach. Besides, the PI-based ATO generates**
264 **an overshoot (**2.8621×10^{-4} **rad) and an undershoot (**9.4647×10^{-5} **rad), which does not occur in**
265 **the proposed ATO.**

Table 4. RMSE and Settling Time in the Experimental HIL Tests.

Configuration	RMSE (without noise)	RMSE (with noise)	Settling Time (s)
$N_p = 102, N_c = 2, \overline{R}_w = 0.01$	0.18×10^{-3}	0.48×10^{-3}	5.50×10^{-3}
$N_p = 120, N_c = 2, \overline{R}_w = 0.01$	0.23×10^{-3}	0.47×10^{-3}	5.90×10^{-3}
$N_p = 102, N_c = 10, \overline{R}_w = 0.01$	0.06×10^{-3}	0.69×10^{-3}	3.90×10^{-3}
PI-based ATO	0.14×10^{-3}	0.47×10^{-3}	11.8×10^{-3}
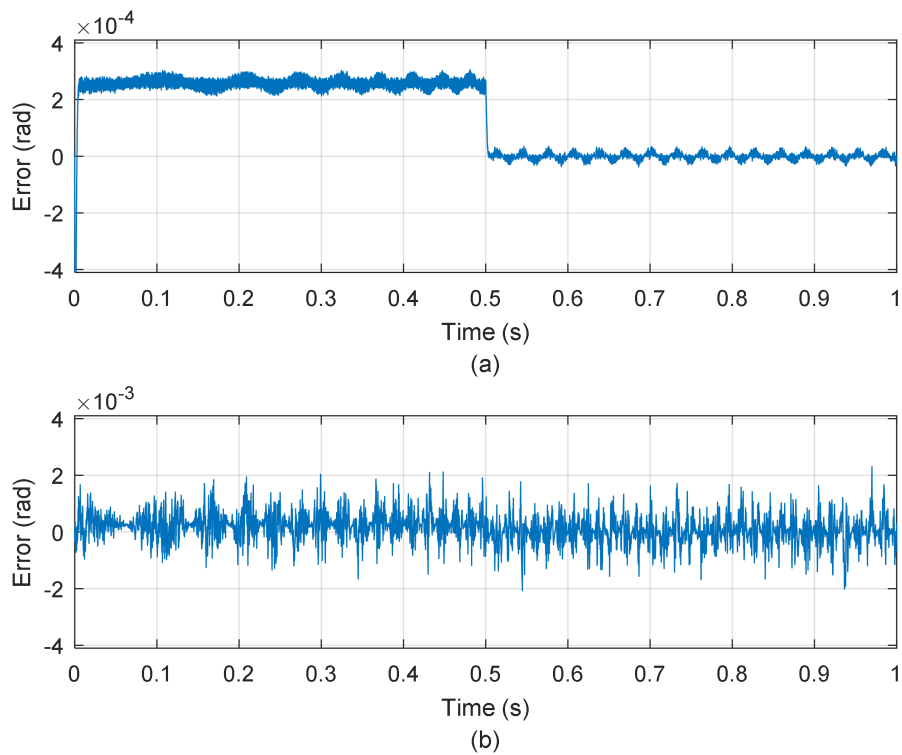
Figure 15. Experimental results for $N_p = 102$, $N_c = 2$, $\overline{R}_w = 0.01$. a) Without noise, b) With noise.
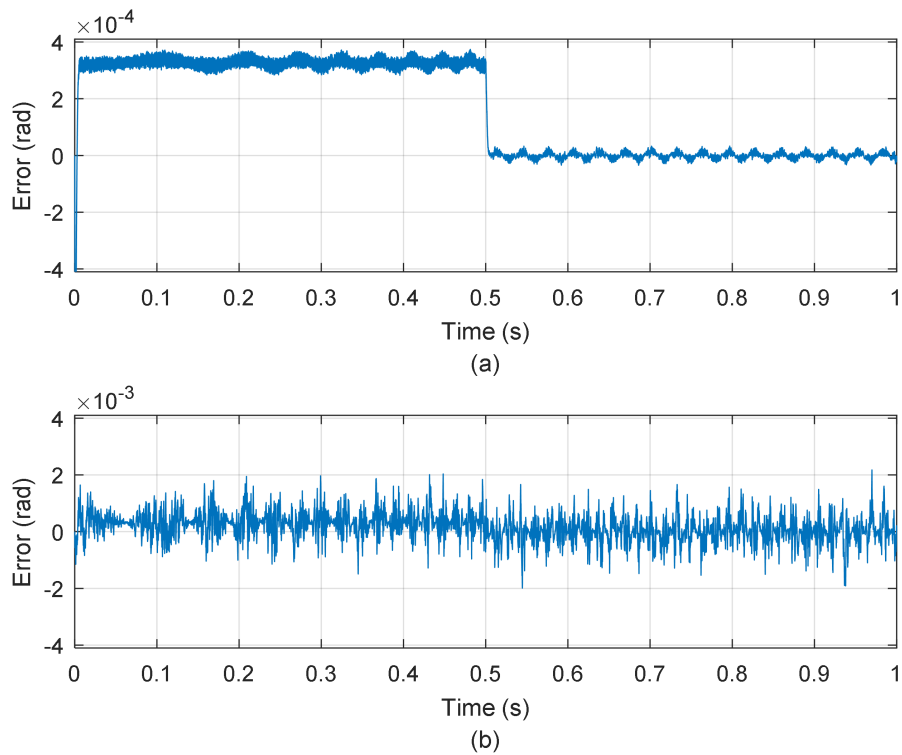
Figure 16. Experimental results for $N_p = 120$, $N_c = 2$, $\overline{R}_w = 0.01$. a) Without noise, b) With noise.
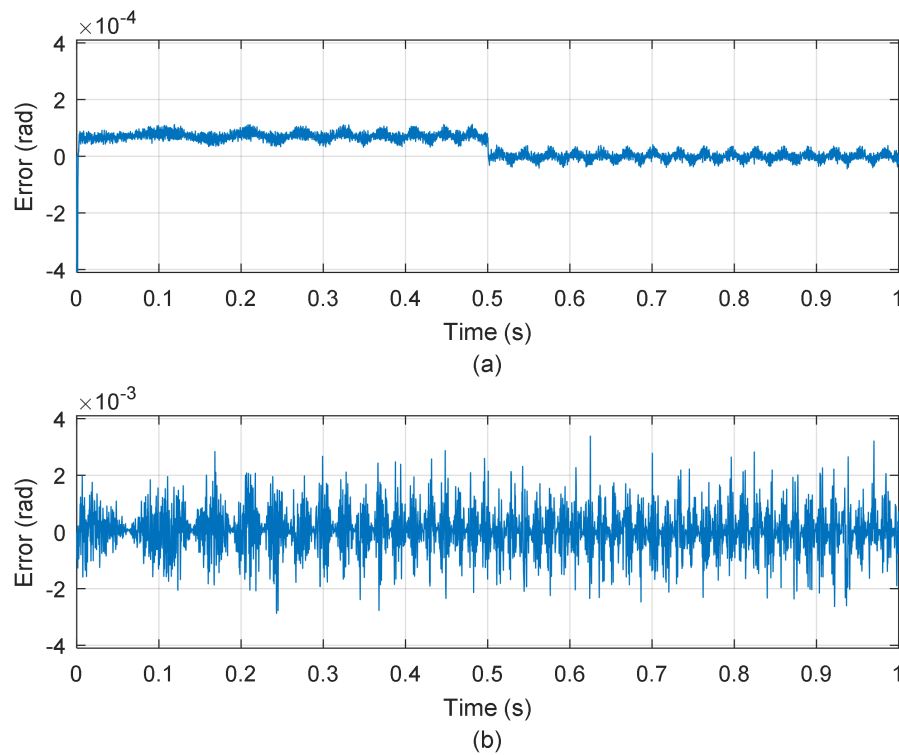
Figure 17. Experimental results for $N_p = 102$, $N_c = 10$, $\overline{R}_w = 0.01$. a) Without noise, b) With noise.
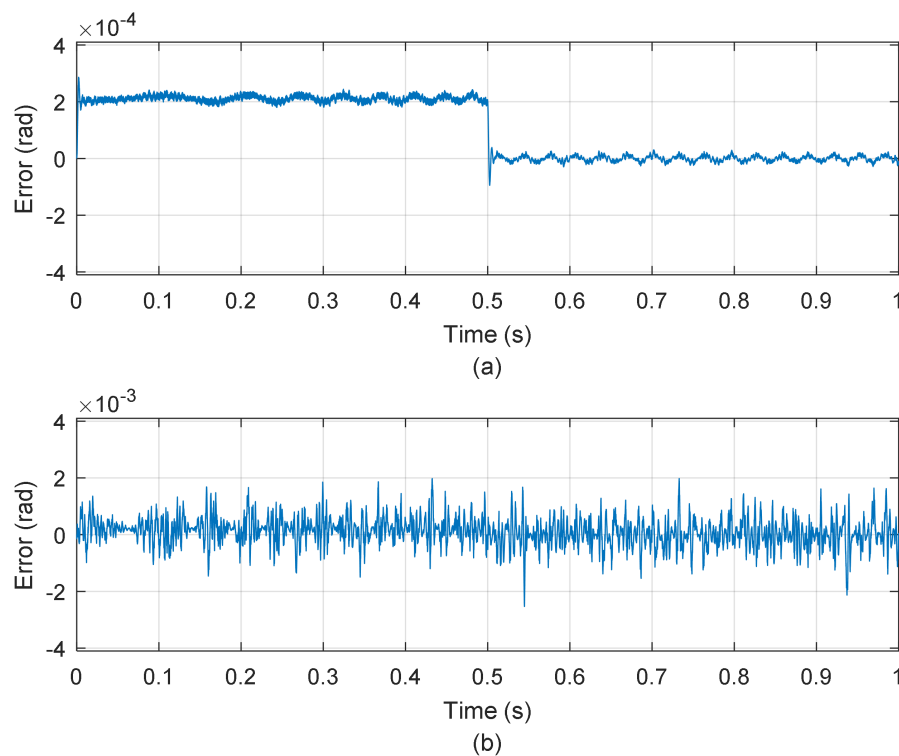
Figure 18. Experimental results for the PI-based ATO.

5. Conclusions

This paper presented a new angle tracking observer (ATO) based on a second-order difference GPC system (SOD-GPC). The proposed ATO has a steady-state error that tends to zero during constant

speed operation, faster and more robust than other ATOs. **Besides, the estimated angular speed can be obtained by filtering the signal u(k) generated by the SOD-GPC algorithm (the design of that filter is beyond the objectives of this research)**. The GPC approach applied in this work is a discrete-time control system. Hence, the GPC structure suitable for its implementation in a digital processor. Other ATOs, as in [56], are based on the discretization of continuous-time tracking systems. The discretization may reduce the performance of a control system. The proposed ATO's dynamic behavior depends on the values of N_p, N_c, and $\overline{R}_w$. Besides, the proposed ATO has faster response than the PI-Based ATO. **In future works, heuristic techniques such as genetic algorithms will be used for ATO tuning. Furthermore, experimental tests using a physical resolver and a reference angular position sensor (e.g., a 16-bit absolute encoder) will be performed. However, the tests done in this research indicate that the application of SOD-GPC algorithm to reduce the angle estimation error allows having a better estimation than using other types of closed-loop regulators (e.g., PI regulators).**

Author Contributions: G.G. and R.C. performed the simulation and experimental tests; T.E., G.G. and R.C. collaborated and supervised article writing; and T.E. did the theoretical analysis of the proposed system. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Acknowledgments: Authors want to thank Federal Univesity of Mato Grosso do Sul for the support to this research.

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

Nomenclature

a_r	resolver excitation amplitude (V)
f_r	resolver excitation frequency (Hz)
k_r	transformation ratio
N_c	control horizon
N_p	prediction horizon
$\overline{R}_w$	tuning coefficient
I	identity matrix
Δ	first-order difference operator
Δ^2	second-order difference operator
ω_r	resolver excitation frequency (rad/s)
λ	eigenvalue
$\theta(t)$	angular position (rad)
$\theta_e(t)$	estimated angular position (rad)
ATO	Angle tracking observer
DSP	Digital Signal Processor
FPGA	Field Programmable Gate Array
RDC	Resolver-to-digital converter

References

1. Datlinger, C.; Hirz, M. An Extended Approach for Validation and Optimization of Position Sensor Signal Processing in Electric Drive Trains. *Electronics* **2019**, *8*, 77.

2. Luo, P.; Tang, Q.; Jing, H.; Chen, X. Design and development of a self-calibration-based inductive absolute angular position sensor. *IEEE Sensors J.* **2019**, *19*, 135–142.

3. Jeon, N.; Lee, H. Integrated fault diagnosis algorithm for motor sensors of in-wheel independent drive electric vehicles. *Sensors* **2016**, *16*, 2106.

4. Mok, H.S.; Kim, S.H.; Cho; Y.H. Reduction of PMSM torque ripple caused by resolver position error. *Electron. Lett.* **2007**, *43*, 646 – 647.

5. Kovacs, I.; Iosub, A.; Țopa, M.; Buzo, A.; Pelz, G. On the influence of angle sensor nonidealities on the torque ripple in PMSM systems — An analytical approach. In Proceedings of the 2016 13th International Conference

on Synthesis, Modeling, Analysis and Simulation Methods and Applications to Circuit Design (SMACD), Lisbon, Portugal, 27–30 June 2016;

6.   Hwang, M.-H.; Lee, H.-S.; Cha, H.-R. Analysis of Torque Ripple and Cogging Torque Reduction in Electric Vehicle Traction Platform Applying Rotor Notched Design. *Energies* **2018**, *11*, 3053.

7.   Liu, Y.; Fang, J.; Tan, K.; Huang, B.; He, W. Sliding Mode Observer with Adaptive Parameter Estimation for Sensorless Control of IPMSM. *Energies* **2020**, *13*, 5991.

8.   Wachowiak, D. Genetic Algorithm Approach for Gains Selection of Induction Machine Extended Speed Observer. *Energies* **2020**, *13*, 4632.

9.   Liu, T.-H.; Ahmad, S.; Mubarok, M.S.; Chen, J.-Y. Simulation and Implementation of Predictive Speed Controller and Position Observer for Sensorless Synchronous Reluctance Motors. *Energies* **2020**, *13*, 2712.

10. Woldegiorgis, A. T.; Ge, X.; Li, S.; Hassa, M. Extended sliding mode disturbance observer-based sensorless control of IPMSM for medium and high-speed range considering railway application. *IEEE Access* **2019**, *7*, 175302 – 175312

11. Chen, S.; Zhao, Y.; Qiu, H.; Ren, X. High-precision rotor position correction strategy for high-speed permanent magnet synchronous motor based on resolver. *IEEE Trans. Power Electron.* **2020**, *35*, 9716–9726.

12. Datlinger, C.; Hirz, M. Benchmark of Rotor Position Sensor Technologies for Application in Automotive Electric Drive Trains. *Electronics* **2020**, *9*, 1063.

13. Saneie, H.; Nasiri-Gheidari, Z.; Tootoonchian, F. Accuracy improvement in variable reluctance resolvers. *IEEE Energy Convers.* **2019**, *34*, 1563–1571.

14. Hou, B.; Zhou, B.; Song, M.; Lin, Z.; Zhang, R. A Novel Single-Excitation Capacitive Angular Position Sensor Design. *Sensors* **2020**, *16*, 1196.

15. Tang, T.; Chen, S.; Huang, X.; Yang, T.; Qi, B. Combining load and motor encoders to compensate nonlinear disturbances for high precision tracking control of gear-driven gimbal. *Sensors* **2018**, *18*, 754.

16. Ni, Q.; Yang, M.; S. A. Odhano,; M. Tang,; P. Zanchetta,; X. Liu,; D. Xu, A new position and speed estimation scheme for position control of PMSM drives using low-resolution position sensors. *IEEE Trans. Ind. Appl.* **2019**, *55*, 3747–3758.

17. Xiao, L.; Li, Z; Bi, C. An optimization approach to variable reluctance resolver. *IEEE Trans. Magn.* **2020**, *56*, 7509005.

18. Hajmohammadi,S.; Alipour-Sarabi, R.; Nasiri-Gheidari, Z.; Tootoonchian, F. Influence of different installation configurations on the position error of a multiturn wound-rotor resolver. *IEEE Sensors J.* **2020**, *20*, 5785–5792.

19. Sun,L.; Taylor, J.; Dorneles Callegaro, A.; Emadi, A. Stator-pm-based variable reluctance resolver with advantage of motional back-emf. *IEEE Trans. Ind. Electron.* **2020**, *67*, 9790–9801.

20. Bahari, M.; Davoodi, A.; Saneie, H.; Tootoonchian, F.; Nasiri-Gheidari, Z. A new variable reluctance pm-resolver. *IEEE Sensors J.* **2020**, *1*, 125–142.

21. Wang, K.; Wu, Z. Hardware-based synchronous envelope detection strategy for resolver supplied with external excitation generator. *IEEE Access* **2019**, *7*, 20801–20810.

22. Wang, S.; Kang, J.; Degano, M.; Buticchi, G. A resolver-to-digital conversion method based on third-order rational fraction polynomial approximation for PMSM control. *IEEE Trans. Ind. Electron.* **2019**, *66*, 6383–6392.

23. Idkhajine, L.; Monmasson, E.; Naouar, M.W.; Prata, A.; Boullaga, K.; Fully integrated FPGA-based controller for synchronous motor drive. *IEEE Trans. Ind. Electron.* **2009** *56*, 4006–4017.

24. Staebler, M.; Verma, A. *TMS320F240 DSP solution for obtaining resolver angular position and speed*; Texas Instrum. Inc., Dallas, TX, Appl. Rep. SPRA605A, Aug. 2017.

25. Abou Qamar, N.; Hatziadoniu, C.J.; Wang, H. Speed error mitigation for a DSP-based resolver-to-digital converter using autotuning filters. *IEEE Trans. Ind. Electron.* **2015**, *62*, 1134–1139.

26. Kaewjinda, W.; Konghirun, M. A DSP - based vector control of pmsm servo drive using resolver sensor. In Proceedings of the 2006 IEEE Region 10 Conference (TENCON), Hong Kong, China, 14–17 November 2006; 1–4.

27. Idkhajine, L.; Monmasson, E.; Naouar, M.W.; Prata, A.; Boullaga, K. Fully integrated FPGA-based controller for synchronous motor drive. *IEEE Trans. Ind. Electron.* **2009**, *56*, 4006–4017.

28. Garcia, R.C.; Pinto, J.O.P.; Suemitsu, W.I.; Soares, J.O. Improved demultiplexing algorithm for hardware simplification of sensored vector control through frequency-domain multiplexing. *IEEE Trans. Ind. Electron.* **2017**, *64*, 6538–6548.

29. Smidl, V.; Janous, S.; Peroutka Z.; Adam, L. Time-optimal current trajectory for predictive speed control of PMSM drive. In Proceedings of the 2017 IEEE International Symposium on Predictive Control of Electrical Drives and Power Electronics (PRECEDE), **2017**, 83–88.

30. Moreno, J. C.; Espi Huerta, J. M. ; Gil, R. G.; Gonzalez, S. A. A robust predictive current control for three-phase grid-connected inverters. *IEEE Trans. Ind. Electron.* **2009**, *56*, 1993–2004.

31. Camacho, E. F.; Bordons, C. *Model Predictive Control*, 2nd ed.; Springer-Verlag London: London, 2004; 13–124.

32. Wang, L. *Model Predictive Control System Design and Implementation Using Matlab®*; Springer-Verlag London: London, 2009; 1–20.

33. Ruchika, N. R. Model predictive control: history and development. *Int. J. Eng.. Trends and Technol.* **2013**, *4*, 2600–2602.

34. Qin, S. J.; Badgwell, T. A. A survey of industrial model predictive control technology. In Proceedings of the 2016 European Control Conference (ECC) **2016**, 733–764.

35. Clarke, D.W.; Mohtadi, C.; Tuffs, P.S. Generalized predictive control - part I. the basic algorithm. *Automatica* **1987**, *23*, 137–148.

36. Clarke, D.W.; Mohtadi, C.; Tuffs, P.S. Generalized Predictive Control - Part II. Extensions and Interpretations. *Automatica* **1987**, *23*, 149–160.

37. Prajwowski, K.; Golebiewski, W.; Lisowski, M.; Abramek, K.F.; Galdynski, D. Modeling of working machines synergy in the process of the hybrid electric vehicle acceleration. *Energies* **2020**, *13*, 5818.

38. Wang, F.; Zhang, Z.; Mei, X.; Rodríguez, J.; Kennel, R. Advanced control strategies of induction machine: field oriented control, direct torque control and model predictive control. *Energies* **2018**, *11*, 120.

39. Gonçalves, P.; Cruz, S.; Mendes, A. Finite control set model predictive control of six-phase asymmetrical machines—an overview. *Energies* **2019**, *12*, 4693.

40. Nguyen, T.-T.; Yoo, H.-J.; Kim, H.-M.; Nguyen-Duc, H. Direct phase angle and voltage amplitude model predictive control of a power converter for microgrid applications. *Energies* **2018**, *11*, 2254.

41. Jin, N.; Pan, C.; Li, Y.; Hu, S.; Fang, J. Model predictive control for virtual synchronous generator with improved vector selection and reconstructed current. *Energies* **2020**, *13*, 5435.

42. Abdelrahem, M.; Hackl, C.M.; Rodríguez, J.; Kennel, R. Model reference adaptive system with finite-set for encoderless control of PMSGs in micro-grid systems. *Energies* **2020**, *13*, 4844.

43. Turley, C.; Jacoby, M.; Pavlak, G.; Henze, G. Development and evaluation of occupancy-aware hvac control for residential building energy efficiency and occupant comfort. *Energies* **2020**, *13*, 5396.

44. Bahramnia, P.; Hosseini Rostami, S.M.; Wang, J.; Kim, G.-J. Modeling and controlling of temperature and humidity in building heating, ventilating, and air conditioning system using model predictive control. *Energies* **2019**, *12*, 4805.

45. Estrabis, T.; Cordero, R.; Batista, E.; Andrea, C.; Grassi, M.A.S. Application of model predictive control in a resolver-to-digital converter. In Proceedings of the 2019 IEEE 15th Brazilian Power Electronics Conference and 5th IEEE Southern Power Electronics Conference (COBEP/SPEC), Santos, Brazil, 1–4 December 2019, pp. 1–6.

46. Dorf, R.C.; Bishop, R.H. Modern Control Systems, 8th ed. Addison Wesley Longman, Inc.: Menlo Park, Ca., USA **1998**, 662–666.

47. Belda, K.; Vosmik, D. Explicit generalized predictive control of speed and position of PMSM drives. *IEEE Trans. Ind. Electron.* **2016**, *63*, 3889–3896.

48. Maeder, U.; Morari, M. Offset-free reference tracking with model predictive control. *Automatica* **2010**, *46*, 1469–1476.

49. Cordero, R.; Estrabis, T.; Batista, E. A., Andrea, C. Q.; Gentil, G. Ramp-tracking generalized predictive control system based on second-order difference. *IEEE Trans. Circuits Syst. II, Exp. Briefs* **2020**, to be published. DOI: 10.1109/TCSII.2020.3019028.

50. Jerry, A. J. Difference Equations with Discrete Transform Methods. *Springer Science+Business Media: Dordrecht* **1996**, 1–22.

51. Proakis, J.C.; Manolakis, D.G. Digital Signal Processing: Principles, Algorithms and Applications. 3rd ed. Prentice-Hall, Inc.:, Upper Saddle River. New Jersey, USA **1996**.

52. Wu, Z.; Li, Y. High-accuracy automatic calibration of resolver signals via two-step gradient estimators. *IEEE Sens. J.* **2018**, *18*, 2883–2891.

53. Sarma, S.; Venkateswaralu, A. Systematic error cancellations and fault detection of resolver angular sensors using a DSP based system. *Mechatronics* **2009**, *19*, 1303–1312.

54. Noori, N.; Khaburi, D. A. Diagnosis and compensation of amplitude imbalance, imperfect quadrant and offset in resolver signals. *2016 7th Power Electronics and Drive Systems Technologies Conference (PEDSTC)* **2016**, 76–81.

55. Gao, Z.; Zhou, B.; Hou, B.; Li, C.; Wei, Q.; Zhang, R. Self-calibration of nonlinear signal model for angular position sensors by model-based automatic search algorithm. *Sensors* **2019**, *19*, 2760.

56. Cordero, R.; Pinto, J.O.P.; Ono, I.E.; Fahed, H.D.S.; Brito, M. Simplification of the acquisition system for sensored vector control using resolver sensor based on fdm and current synchronous sampling. *In Proceedings of the 2018 IEEE 4th Southern Power Electronics Conference (SPEC), Singapure* **2018**, *1–4.*

Sample Availability: Samples of the compounds ...... are available from the authors.