FOREX Publication
Open Access | Rapid and quality publishing

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 746-755 | e-ISSN: 2347-470X

Speed Control of Sensorless Induction motor based on Grey Wolf Optimizer Fractional Order Controller using MRAS based Speed Estimation

Saravanan T Y[1] and Dr. Ponnambalam P[2*]

[1]*Research Scholar SELECT, VIT, Vellore, India; saravanan651988@gmail.com*
[2] *Professor SELECT, VIT, Vellore.India; ponnambalam.p@vit.ac.in*

***Correspondence:** Dr. Ponnambalam P; ponnambalam.p@vit.ac.in

ABSTRACT- Traditional induction motor control methods typically require feedback from sensors like encoders or resolvers to determine the motor's rotor position and speed accurately. The speed control of a sensorless induction motor is critical, so this study provides a novel method that combines the Model Reference Adaptive System (MRAS) for speed estimate with the Fractional Order PID controller for speed control. This controller's parameters are optimized using the Grey Wolf Optimizer Algorithm. After being implemented in the MATLAB/Simulink environment, the suggested approach's performance is compared to that of a standard PI controller. From the findings, it is clear that the proposed method effectively maintaining the specified speed as compared with PI controller. The proposed controller performance is also validated through experimental results.

Keywords: Model Reference Adaptive System (MRAS), Fractional Order PID Controller (FOPID), Grey Wolf Optimizer (GWO).

ARTICLE INFORMATION

Author(s): Saravanan T Y and Dr. Ponnambalam P;
Received: 09/05/2024; **Accepted**: 26/06/2024; **Published**: 25/07/2024;
e-ISSN: 2347-470X;
Paper Id: IJEER 0905-07;
Citation: 10.37391/ijeer.120304
Webpage-link:
https://ijeer.forexjournal.co.in/archive/volume-12/ijeer-120304.html

1. INTRODUCTION

Due to advancements in renewable energy technology and fuel demand, electric vehicles (EVs) are now being used as an alternative to traditional transportation systems. However, EVs have some limitations, including thirster complaint times, larger weight, and a lower battery life. DC motors have been used for EVs due to their decoupling effect, but they require frequent maintenance and have a short lifespan. To address these issues, Upendra et al. (2017) used an induction motor (IM) in an electric vehicle application.

The inverter controls the frequency and amplitude of the power supply to the electric motor, allowing the motor's speed and torque to be adjusted. Power electronic converter and control schemes were selected based on the system requirements and complexity of the industrial process. Motor-load dynamics have a significant impact on the efficiency of electric drive motors. The power loss in an induction motor drive system must be reduced in order to boost system efficiency while also reducing environmental pollutants. The overall driving losses are determined by the motor, power electronic converter, semiconductor switching device, and power converter modulation methods used. Because of the advancement of contemporary power electronic components, machine theory, and rapid controllers, DC motor-like independent control may now be applied in an induction machine. As a result, selecting the right control strategy is critical in industrial drives in order to achieve the desired results.

There are two main types of speed/torque control for induction motors:

1. Scalar Control
2. Vector Control

The scalar control approach is straightforward to develop, however it does not provide good dynamic performance. Carlos et al. (2001). The main disadvantage of scalar control is the coupling effect. In IM, voltage and frequency can be used to control flux and torque. However, flux and torque are affected by both the voltage and frequency of the electric supply input to the motor. Torque cannot be regulated independently without impacting flux. In another manner, both torque and flux of an induction motor are dependent on stator current, and there is a mutual reliance between torque and flux. This dependency, known as the coupling effect, causes sluggish responsiveness and instability.

IM operates continually in VFD mode to provide the optimal transient response. At low speeds, the VSI supply to the induction motor generates current harmonics, resulting in additional losses. Optimal Direct Torque Control (ODTC) has been shown to reduce losses by Tazerart et al. (2015). The machine consumes less current during acceleration and deceleration, and the flux varies with load. It is suitable for electric vehicles and will extend the battery's life by preventing discharge.

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 746-755 | e-ISSN: 2347-470X

Vector control can also be used in IMs without using shaft sensors. Thus, sensorless vector control is the name given to this control scheme. The stator voltage and current values can be used to calculate the IM's speed. In any case, variations in machine parameters impact sensorless techniques. Temperature and humidity levels also play a role. Farasat et al. (2014) presented sensorless control for energy-efficient operation of an electric powertrain. Instead of the speed encoder, this technique makes use of an enlarged Kalman observer. Abdellatif et al. (2014) developed sensorless control for DFIM drives employing multiple DTCs. Zhang et al. (2012) developed a novel way for controlling sensor-less DTC strategies for instant messaging systems. In this situation, the stator current was raised due to the pre-excitation approach. Wang et al. (2014) offered a new approach for IM. In which, the predicted torque was determined at the end of switching period. The size of the stator flux was used to control machine losses. Sanjay et al. (2017) used fuzzy logic to eliminate ripples in induction motor. The significant torque ripples caused by DTC is its main drawback. This approach substitutes two fuzzy logic controllers for the two hysteresis controllers. Because of the buildup of angular errors in the velocity integration process, Poor control quality has been seen in speed estimation when no angular corrective block is included. At the same time, the transition processes are being tracked by undesirable nonlinearities.

Solodkiy et al. (2017) proposed an adaptive rotor speed observer-based sensorless induction motor control system as a solution to this problem. Phase-locked loop coordinate transformation is used in this continuous angle correction. The method has a straightforward model implementation for actual object adjustment. The speed observer was constructed utilizing error estimations based on stator current mismatch.
Vinay and Srinivasa (2013) presented slip speed control in DTC for induction motor drives, with a focus on reducing torque pulsations. To create the stator flux reference, this control approach adds the actual stator flux angle to the change in stator flux angle.

Amiri et al. (2018) proposed a unique Predictive Torque Control (PTC) algorithm for induction motors that uses the DSVM scheme. The PTC method is then used to value the increased number of voltage vectors produced by the DSVM system. A low sampling frequency is offered by this PTC-DSVM. However, the computational overhead has increased dramatically due to the enormous number of virtual vectors.

In order to control the approximate stator flux's size and speed, Sangsefidi et al. (2014) introduced the Approximate Stator Flux Manage (ASFC) method. Neither mechanical sensors nor current transducers are needed for this method.

In order to lessen flux and torque ripple, Ammar et al. (2017) suggested a predictive torque control model that makes use of an FLC. The ideal switching state for the inverter is determined without the need of a modulation approach. The cost function optimization is used to anticipate the control variable and minimize the difference between the reference and predicted values. The FLC, which is utilized to provide torque reference in the outer speed control loop, provides accurate speed tracking and powerful response to uncertainties.

The combination of predictive torque control and FLC results in effective steady-state performance as well as reversing operation with reduced current harmonics and torque ripple. Sivakumar et al. (2018) implemented a PI-based Shunt active filter into an artificial neural network to decrease harmonics caused by nonlinear loads in the source current.

A comparison was made between the performance of traditional DTCs and the ANN and ANFIS speed control solution proposed by Venkateswara Rao and Anand Kumar (2018) for switching table-based DTC to remove torque ripples in an induction motor drive. Bindal and Kaur (2019) developed an ANFIS-based DTC-IM drive with the purpose of delivering precise speed control over a wide range of speeds. Sekhar and Sekhar (2012) proposed five- and seven-level diode clamped multilevel inverters for DTC of induction motor drives. In this work, voltage distortions and torque ripples are significantly reduced. The major problems addressed by the researchers from the literature are torque ripple slow response, THD and drive losses. This paper presents a novel GWO-FOPID-MRAS controller for enhancing the speed response of induction motor.

2. MATHEMATICAL MODELLING OF INDUCTION MOTOR

This paper uses both synchronously rotating and stationary reference frames to create a dynamic machine model and design an appropriate controller for the IM.

$$V_{1A}(t) = R_1 i_{1A}(t) + \frac{d\Psi_{1A}(t)}{dt} \quad (1)$$

$$V_{1B}(t) = R_1 i_{1B}(t) + \frac{d\Psi_{1B}(t)}{dt} \quad (2)$$

$$V_{1C}(t) = R_1 i_{1C}(t) + \frac{d\Psi_{1C}(t)}{dt} \quad (3)$$

$$V_{2a}(t) = R_2 i_{2a}(t) + \frac{d\Psi_{2a}(t)}{dt} \quad (4)$$

$$V_{2b}(t) = R_2 i_{2b}(t) + \frac{d\Psi_{2b}(t)}{dt} \quad (5)$$

$$V_{2c}(t) = R_2 i_{2c}(t) + \frac{d\Psi_{2c}(t)}{dt} \quad (6)$$

After many simplifications and applying park's transformation:

$$V_{q1}^{s} = \frac{2}{3}V_{a1} - \frac{1}{3}V_{b1} - \frac{1}{3}V_{c1} = V_{a1} \quad (7)$$

$$V_{d1}^{s} = \frac{1}{\sqrt{3}}V_{b1} + \frac{1}{\sqrt{3}}V_{c1} \quad (8)$$

After further simplifications, the toque speed relation can be written as:

$$T_e = T_L + j\frac{d\omega_m}{dt} = T_L + \frac{2}{p}j\frac{d\omega_2}{dt} \quad (9)$$

3. MRAS SPEED OBSERVER DESIGN

One technique utilized in motor control systems, namely in

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 746-755 | e-ISSN: 2347-470X

sensorless vector control of induction motors, is MRAS speed estimation. It's a technique to estimate the speed of the motor without using a direct speed sensor, which can reduce cost and complexity in motor control systems. This involves the following steps:

Model Reference: The MRAS system consists of a mathematical model of the motor, typically a dynamic model representing the motor's electrical and mechanical behavior.

Adaptive Mechanism: The adaptive mechanism is in charge of changing the model's parameters to reflect the motor's real behaviour. This modification is essential because it enables the model to respond to variations in the motor's properties over time or as a result of operating circumstances.

Error Calculation: The difference between the model output and the actual measurements (usually stator current or voltage) is calculated. This error signal represents the mismatch between the model's prediction and the real-world behavior of the motor.

Speed Estimation: The speed of the motor is indirectly estimated based on this error signal. The adaptive mechanism adjusts the model parameters such that the error between the model output and the actual measurements is minimized. The parameters that best minimize this error are used to calculate the estimated speed of the motor.

Feedback Loop: In order to modify the control signals given to the motor, the estimated speed is transmitted back into the system. This eliminates the need for an additional speed sensor and enables accurate control of the motor's torque and speed.

Combining the voltage and current models is the technique [21-26]. Equations (10) and (13) provide descriptions of the voltage and current models, also referred to as the reference model and adaptive models.

$$\Psi_{ra}^{*} = \frac{L_T}{M_{ST}}(V_{s\alpha} - R_s i_{s\alpha} - \sigma L_s i_{s\alpha}) \quad (10)$$

$$\Psi_{r\beta}^{*} = \frac{L_T}{M_{ST}}\left(V_{s\beta} - R_s i_{s\beta} - \sigma L_s i_{s\beta}\right) \quad (11)$$

$$\dot{\Psi}_{r\alpha}^{*} = \frac{M_{ST}}{T_r} i_{s\alpha} - \frac{1}{T_r}\dot{\Psi}_{r\alpha} - \dot{\omega}\dot{\Psi}_{r\beta} \quad (12)$$

$$\dot{\Psi}_{r\beta}^{*} = \frac{M_{ST}}{T_r} i_{s\beta} - \frac{1}{T_r}\dot{\Psi}_{r\beta} - \dot{\omega}\dot{\Psi}_{r\alpha} \quad (13)$$

$$\dot{\omega} = \left(K_p + \frac{K_i}{s}\right)\theta_{\Psi r} \quad (14)$$

4. OVERVIEW OF FRACTIONAL ORDER PID CONTROLLER

An expansion of the conventional PID controller is a FOPID (Proportional-Integral-Derivative) controller., where the derivative and integral terms are replaced with fractional-order integrals and derivatives. In a FOPID, the order of integration and differentiation is not restricted to integer values, allowing for more flexibility in system modeling and control.

Proportional (P) Term: The proportional term in a FOPID behaves similarly to the traditional PID controller, when the difference between the intended set point and the actual process variable, or the error signal, is proportionate to the control signal.

Integral (I) Term: The integral term in a FOPID is a fractional-order integral of the error signal. Instead of integrating over time using integer-order calculus, a fractional-order integral integrates the error signal with a fractional order, which provides more flexibility in controlling systems with non-integer dynamics. The integral term helps in eliminating steady-state error and improving system stability.

Derivative (D) Term: Comparably, a fractional-order derivative of the error signal represents the derivative term in a FOPID controller. It captures the rate of change of the error signal over time with a fractional order, allowing for better control of systems with complex dynamics. The derivative term helps in damping oscillations and improving system response time.

Fractional Order Parameters: In addition to the gains K_P, K_d and K_i, a fractional-order PID controller also includes fractional order parameters μ and λ which establish the derivative and integral terms' respective orders. By adjusting the controller's reaction according to these characteristics, the dynamics of the controlled system can be more precisely matched.

A Fractional Order Proportional Integral Derivative (FOPID) controller offers several advantages over traditional PID controllers in the context of speed control for induction motors. Some of the key benefits are:

Enhanced Flexibility and Tuning: FOPID controllers have two additional parameters, the fractional orders of integration (λ) and differentiation (μ), providing more tuning flexibility compared to traditional PID controllers. This allows for more precise control and adaptation to system dynamics.

Improved Robustness: FOPID controllers exhibit better robustness to parameter variations and external disturbances. The fractional orders can be fine-tuned to handle system uncertainties and non-linearities more effectively, leading to a more stable and reliable control system.

Better Performance: In terms of speed control of induction motors, FOPID controllers can offer superior performance characteristics such as reduced overshoot, lower settling time, and minimal steady-state error. This leads to smoother and more accurate speed control.

Enhanced Frequency Response: The fractional calculus used in FOPID controllers provides an inherent ability to manage the frequency response of the system more effectively. This is particularly beneficial for induction motors, which may exhibit complex dynamics across different operating frequencies.

Adaptive and Predictive Capabilities: FOPID controllers can be designed to adapt to changes in system dynamics and predict

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 746-755 | e-ISSN: 2347-470X

future behavior more accurately than traditional PID controllers. This is crucial for systems like induction motors, which may experience varying load conditions and operational environments.

Reduced Sensitivity to Noise: Due to the fractional nature of the differentiation and integration, FOPID controllers can be less sensitive to high-frequency noise, which is often a concern in motor control applications. This helps in maintaining smoother motor operation.

Better Handling of Non-linearities: Induction motors exhibit non-linear behavior, especially under varying load conditions. FOPID controllers are better suited to handle these non-linearities due to their fractional order properties, leading to improved control performance.

Energy Efficiency: Improved control accuracy and reduced oscillations in speed control can lead to more energy-efficient operation of induction motors. This can result in cost savings and reduced wear and tear on the motor.

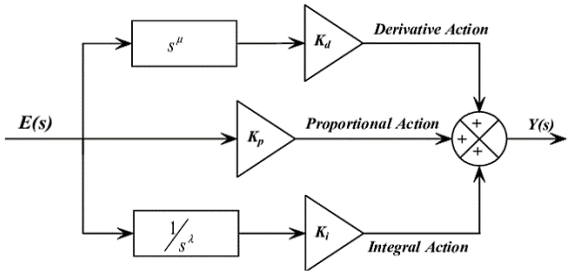

Figure 1. Structure of FOPID

Figure 1. shows the structure of Fractional Order PID controller.

5. GREY WOLF OPTIMIZER ALGORITHM FOR FRACTIONAL ORDER CONTROLLER PID CONTROLLER

Implementing a Grey Wolf Optimizer (GWO) algorithm for tuning a FOPID controller based on Model Reference Adaptive System (MRAS) speed estimation involves several steps. Below is a basic outline of how you could approach this implementation:

1. MRAS Speed Estimation:
To determine the system's speed, put the Model Reference Adaptive System (MRAS) algorithm into practice. To track the system's speed, this usually entails creating an adaptive model in addition to a reference model.

2. FOPID Controller Design:
Provide the structure for FOPID controller. The derivative, integral, and proportional terms' fractional orders and associated gains should be ascertained.

3. Grey Wolf Optimizer (GWO):
To optimize the FOPID controller's parameters, apply the Grey Wolf Optimizer algorithm. This entails specifying the GWO operators, including fitness calculation, initialization, and position updates.

4. Objective Function:
Define an objective function that represents the performance criteria of the FOPID controller. This could be based on speed tracking error, settling time, overshoot, or any other relevant metrics.

$$J = \frac{1}{ITAE}$$
$$ITAE = \int_0^t |\Delta N|\, dt$$

Selection of the Objective Function
The objective function is the function that the algorithm aims to optimize (minimize or maximize). This function represents the problem you want to solve.

Define the Objective Function: Clearly define the mathematical form of the objective function f(x). For example, if you are optimizing a speed control problem, the objective function might be the error between the desired and actual speed of an induction motor.

Type of Optimization: Determine whether it is a minimization or maximization problem. GWO can handle both types, but it typically involves minimizing a given objective function.

Parameter Initialization
Parameter initialization involves setting up initial values for various parameters used in the GWO algorithm.

Population Initialization: Initialize a population of grey wolves (solution candidates). Each wolf is represented by a position vector X_i in the search space.

Population Size: Choose the number of wolves N. A common choice might be between 20 and 50, but this depends on the problem complexity.

Position Vector Initialization: Initialize the position vectors X_i randomly within the search space bounds. If the search space for each dimension is defined by $[X_{min}, X_{max}]$, then each element of $X_i$can be initialized as:

$$X_{i,j} = X_{min} + (X_{max} - X_{min}) \times rand(0,1)$$

where $rand(0,1)$ generates a random number between 0 and 1.

Coefficient Initialization: Initialize the coefficients a, A and C. a decreases linearly from 2 to 0 over the course of iterations.

A and C are calculated as: $A = 2a.rand(0,1) - a;$ $C = 2.rand(0,1)$

Convergence Criteria
Convergence criteria determine when the algorithm should stop iterating.

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 746-755 | e-ISSN: 2347-470X

Maximum Number of Iterations: Set a maximum number of iterations T_{max}. The algorithm will stop after completing these iterations if no other stopping condition is met.

Tolerance Level: Set a tolerance level for the change in the objective function value. If the change in the best objective function value over a certain number of iterations is less than this tolerance, the algorithm can be stopped early.

Fitness Threshold: Define a fitness threshold. If the objective function value of the best solution reaches this threshold, the algorithm terminates.

6. OPTIMIZATION PROCESS:

Utilize the GWO technique to get the FOPID controller's ideal parameters. This involves initializing a population of wolves (solutions), evaluating their fitness using the objective function, and iteratively updating their positions until convergence.
Validation and Tuning:

Validate the performance of the optimized FOPID controller using simulation or experimental data. Fine-tune the parameters if necessary to achieve the desired performance.

The process code:

```
% Define the objective function (fitness function) for FOPID controller tuning
function fitness = objective_function(parameters)
   % Implement your objective function here, which evaluates the performance
   % of the FOPID controller based on MRAS speed estimation. It should return
   % a scalar value representing the fitness of the solution.
   % Example: fitness = my_fitness_function(parameters);
end
% Grey Wolf Optimizer (GWO) algorithm
function [best_solution, best_fitness] = grey_wolf_optimizer(population_size, num_variables, max_iterations)
   % Initialization
   wolves = rand(population_size, num_variables) * 2 - 1;
% Initialize wolf positions randomly within [-1, 1]
   alpha_position = zeros(1, num_variables);
   beta_position = zeros(1, num_variables);
   delta_position = zeros(1, num_variables);
   best_solution = [];
   best_fitness = inf;
      % Main loop
   for iteration = 1:max_iterations
      for i = 1:population_size
         % Evaluate fitness of the current wolf
         fitness = objective_function(wolves(i, :));

         % Update best solution
         if fitness < best_fitness
            best_fitness = fitness;
            best_solution = wolves(i, :);
         end
      end
         % Update alpha, beta, and delta positions
      [alpha_wolf, beta_wolf, delta_wolf] = select_alpha_beta_delta_wolves(wolves, best_solution);
            % Update positions of all wolves
     a = 2 - iteration * (2 / max_iterations); % Eq. (2.3) in [1]
      c = 2 * rand(); % Eq. (2.3) in [1]
      for i = 1:population_size
         D_alpha = abs(c * alpha_position - wolves(i, :));
         D_beta = abs(c * beta_position - wolves(i, :));
         D_delta = abs(c * delta_position - wolves(i, :));
         X1 = alpha_position - a * D_alpha;
         X2 = beta_position - a * D_beta;
         X3 = delta_position - a * D_delta;
         wolves(i, :) = (X1 + X2 + X3) / 3; % Update position using Eq. (4) in [1]
      end
   end
end
% Function to select alpha, beta, and delta wolves
function [alpha_wolf, beta_wolf, delta_wolf] = select_alpha_beta_delta_wolves(wolves, best_solution)
   % Sort wolves based on fitness (ascending order)
   [~, sorted_indices] = sort(objective_function(wolves));
      % Select alpha, beta, and delta wolves
   alpha_wolf = wolves(sorted_indices(1), :);
   beta_wolf = wolves(sorted_indices(2), :);
   delta_wolf = wolves(sorted_indices(3), :);
end
% Example usage
population_size = 20;
num_variables = 5; % Adjust based on the number of FOPID controller parameters
max_iterations = 100;
[best_solution, best_fitness] = grey_wolf_optimizer(population_size, num_variables, max_iterations);
disp('Best solution:');
disp(best_solution);
disp('Best fitness:');
disp(best_fitness);
```

7. . SIMULATION RESULTS

Case:A Simulation based performance comparison

The sensorless induction motor block design used in the simulation is shown in Figure 2.A PC running 64-bit software, has an Intel(R) Core (TM) i5-9400 CPU @ 2.90GHz processor, 8GB RAM, and is used for simulations. MATLAB 2021 is installed on this PC.

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 746-755 | e-ISSN: 2347-470X

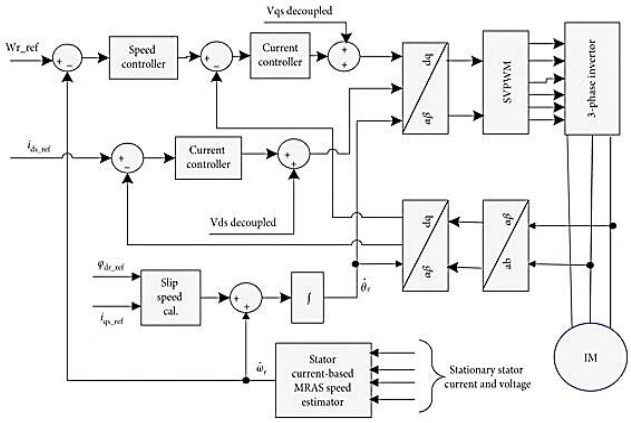

Figure 2. GWO-FOPID-MRAS method representation

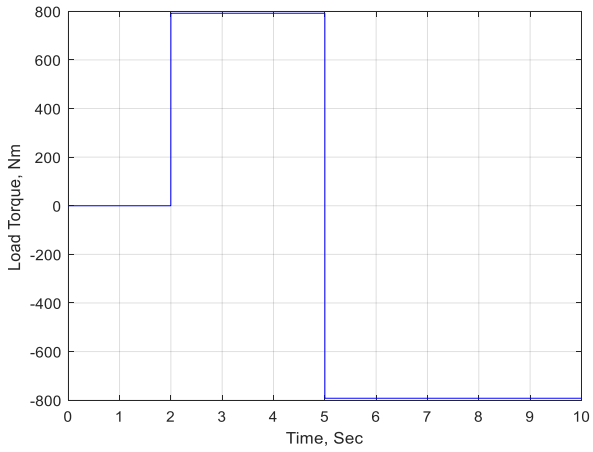

Figure 3. The load torque of an induction motor in relation to time

Initially, the torque on the motor is set to zero; at 2 seconds, it is increased to 800 Nm and kept at that level for 5 seconds; at 5 seconds, it is reduced to -800 Nm and maintained for 10 seconds, as shown in *figure 3*.

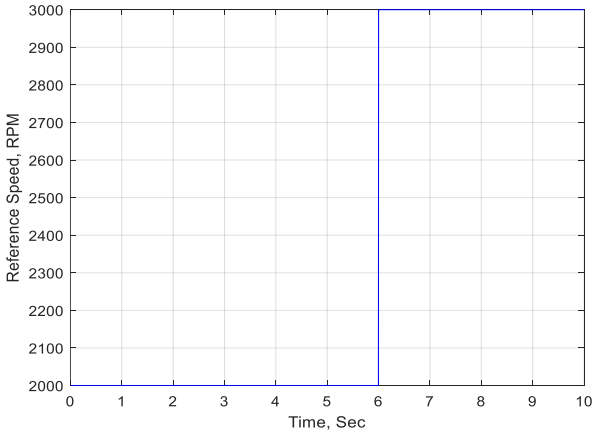

Figure 4. Induction motor Reference with respect to time

Initially, the motor's reference speed is established to 2000 rpm and kept at that level for 6 seconds; at 6 seconds, it is increased to 3000 rpm and maintained for 10 seconds, as shown in *figure 4.*

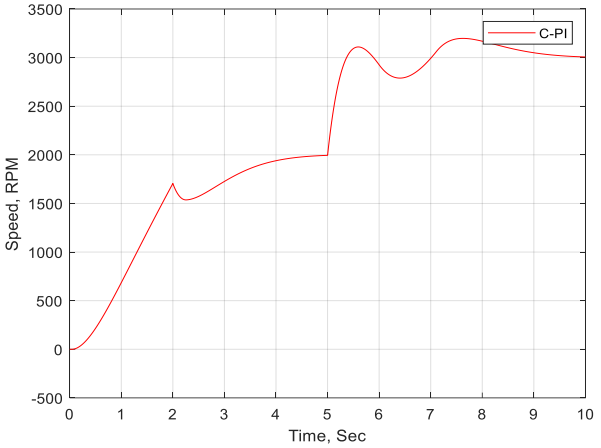

Figure 5. Actual Speed of IM with Conventional PI controller

The speed variations of IM using a typical PI controller are shown in *figure 5*. The speed is attempting to achieve 2000 rpm (the reference speed), however due to load torque divergence, the speed has not settled and has increased further, almost reaching 2000 rpm at 4.5 seconds. After 5 seconds, the speed changes again due to a quick change in torque, followed by a sudden shift in reference speed at 6 seconds. With these disturbances, the speed reached the reference speed of 3000 rpm in roughly 9.8 seconds using a typical PI controller.

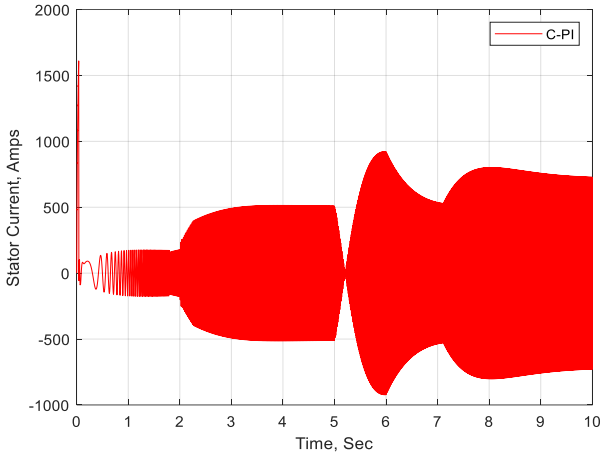

Figure 6. Stator current deviations of IM with Conventional PI controller

Figure 6 depicts the stator current variations of IM with respect to time owing to changes in load torque and reference speed. The first reference speed is 2000 rpm and the load torque is zero; at this point, the stator current increases slightly and then settle at 0.5 seconds. Following that, due to load variations at 2 seconds, the stator current fluctuated to support the load torque before stabilizing at approximately 3 seconds. Again, due to load torque deviation, the current begins to decrease before increasing further due to an increase in reference speed.

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 746-755 | e-ISSN: 2347-470X

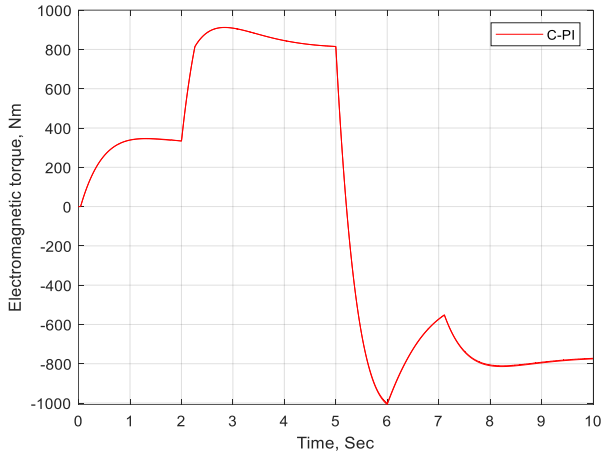

Figure 7. Electromagnetic Torque deviations of IM with Conventional PI controller

Figure 7 displays the electromagnetic torque (ET) variations of IM over time due to changes in load torque and reference speed. The first reference speed is 2000 rpm, and the load torque is zero; at this point, the ET climbs to 3000 Nm before settling at 1.9 seconds. Following that, due to load fluctuations around 2 seconds, the ET climbed to 950 Nm to support the load torque before stabilising at around 4.9 seconds. Again, due to load torque variation, the ET fell to -1000 Nm before rising to -800 Nm approximately 9 seconds later as the reference speed climbed.

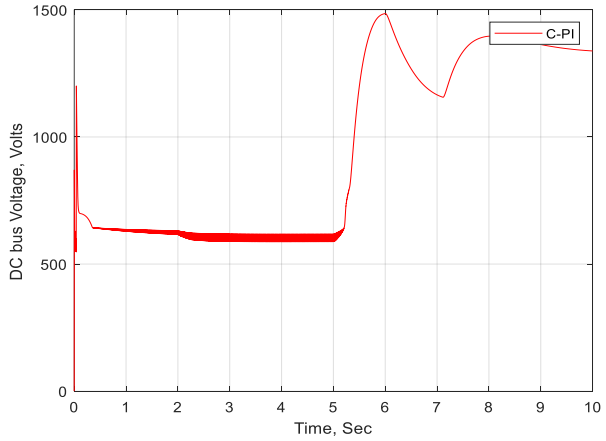

Figure 8. DC bus voltage deviations of IM with Conventional PI controller

Figure 8 depicts IM's DC bus voltage variations over time as a function of load torque and reference speed. The initial reference speed is 2000 rpm, and the load torque is zero; at this point, the DC bus voltage rises to 1200 volts before stabilising at 1.9 seconds. Following then, due to load oscillations lasting approximately 2 seconds, the DC bus voltage settled at 600 volts before settling at around 4.9 seconds. Again, due to load torque fluctuation, the DC bus voltage rose to 1500 volts before settling to 1300 volts about 9.9 seconds later as the reference speed increased.

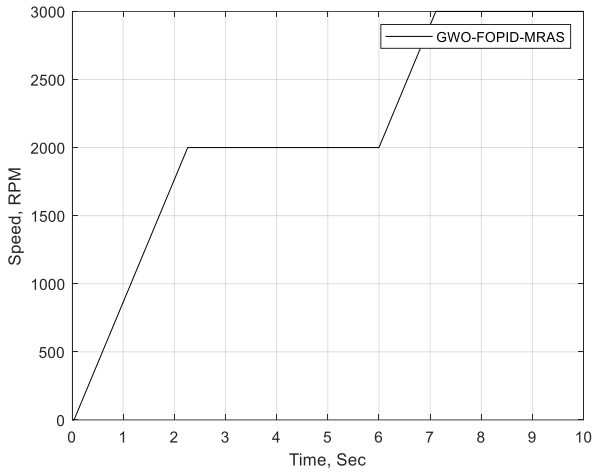

Figure 9. Actual Speed of IM with GWO-FOPID-MRAS controller

Figure 9 depicts the speed variations of IM with a GWO-FOPID-MRAS controller. The speed is attempting to achieve 2000 rpm (the reference speed), however due to load torque divergence, the speed has increased and settled at 2000 rpm at 2.2 seconds. After 5 seconds also the speed not changed due to change in torque also, followed by a sudden shift in reference speed at 6 seconds. With these disturbances, the speed reached the reference speed of 3000 rpm in roughly 7.1 seconds using GWO-FOPID-MRAS controller.

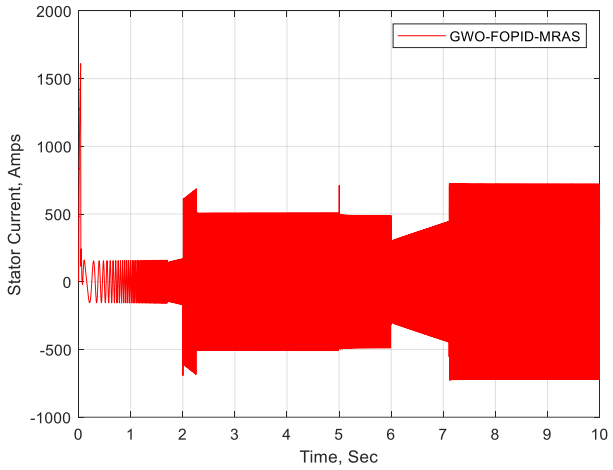

Figure 10. Stator current deviations of IM with GWO-FOPID-MRAS controller

Figure 10 depicts the stator current variations of IM with respect to time owing to changes in load torque and reference speed. The first reference speed is 2000 rpm and the load torque is zero; at this point, the stator current increases slightly and then settle at 0.1 seconds. Following that, due to load variations at 2 seconds, the stator current fluctuated to support the load torque before stabilizing at approximately 2.2 seconds. Again, due to load torque deviation, the current begins to decrease before increasing further due to an increase in reference speed.

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 746-755 | e-ISSN: 2347-470X

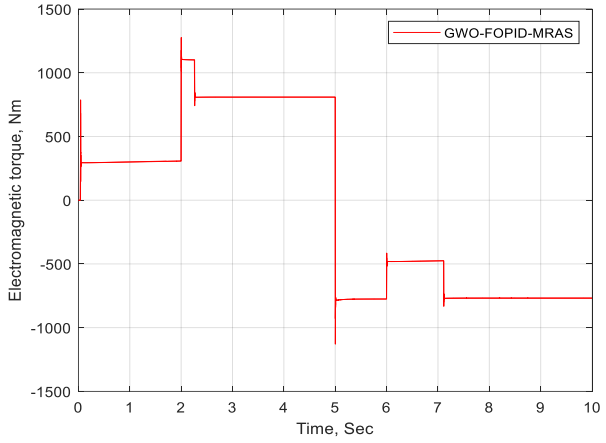

Figure 11. Electromagnetic Torque deviations of IM with GWO-FOPID-MRAS controller

Figure 11 displays the electromagnetic torque (ET) variations of IM over time due to changes in load torque and reference speed. The first reference speed is 2000 rpm, and the load torque is zero; at this point, the ET climbs to 800 Nm before settling at 0.1 seconds. Following that, due to load fluctuations around 2 seconds, the ET climbed to 800 Nm to support the load torque before stabilising at around 2.2 seconds. Again, due to load torque variation, the ET fell to -1100 Nm before rising to -800 Nm approximately 7.1 seconds later as the reference speed climbed.

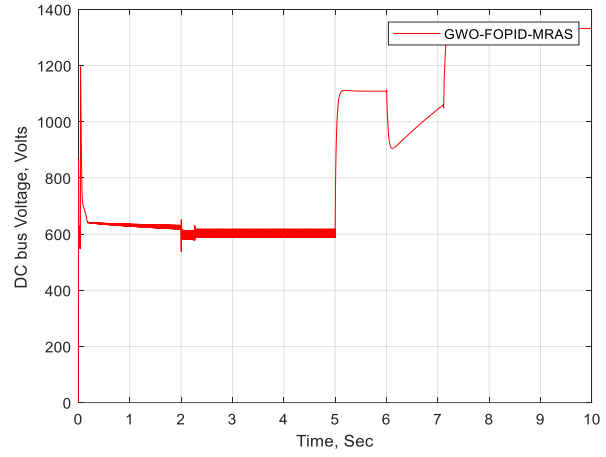

Figure 12. DC bus voltage deviations of IM with GWO-FOPID-MRAS controller

Figure 12 depicts IM's DC bus voltage variations over time as a function of load torque and reference speed. The initial reference speed is 2000 rpm, and the load torque is zero; at this point, the DC bus voltage rises to 1200 volts before stabilising at 0.1 seconds. Following then, due to load oscillations lasting approximately 2 seconds, the DC bus voltage settled at 600 volts before settling at around 2.1 seconds. Again, due to load torque fluctuation, the DC bus voltage rose to 1300 volts before settling to 1300 volts about 7.1 seconds later as the reference speed increased.

Case: B Experimental Results

Figure 12 depicts the experimental setup of the proposed controller. The proposed method is practically tested on 2kW, 415 volts, 50 Hz, 4.2 A, 1491 rpm induction motor. The performance results are shown in Figs 13 to 15. From these results, it is clear that the proposed method is effectively maintaining the speed as compared with conventional PI controller.

Figure 12. Experimental Setup with proposed Controller

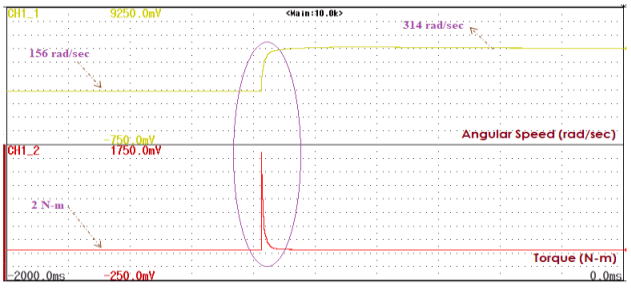

Figure 13. Angular Speed and Torque Variations of IM motor with Proposed Controller (Constant torque and variable speed)

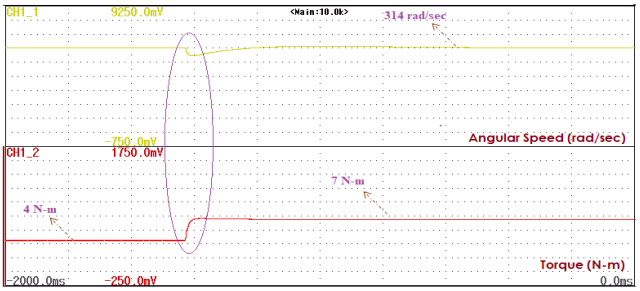

Figure 14. Angular Speed and Torque Variations of IM motor with Proposed Controller (Constant speed and variable torque)

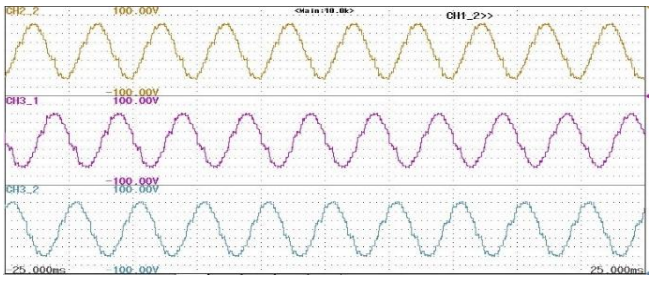

Figure 15. Stator Voltage deviations with respect to time

FOREX Publication
Open Access | Rapid and quality publishing
International Journal of Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 746-755 | e-ISSN: 2347-470X

8. CONCLUSIONS

The Fractional Order PID controller is used in conjunction with the Model Reference Adaptive System (MRAS) for speed estimate in this novel approach to address the critical issue of speed control in sensorless induction motors. The Grey Wolf Optimizer Algorithm is used to optimize the controller's parameters. The performance of the suggested method is evaluated against that of a conventional PI controller once it is implemented in the MATLAB/Simulink environment. The results demonstrate that, in comparison to the PI controller, the suggested method effectively maintains the specified speed. The suggested controller's functionality is confirmed via experiments.

8.1 Potential practical implementation challenges and future research

8.1.1 Practical Implementation Challenges

Harmonics and Power Quality: Issue: Variable frequency drives (VFDs), commonly used for speed control, introduce harmonics into the power system, affecting power quality.

Solution: Implementing harmonic filters and designing drives with lower harmonic distortion can help mitigate these issues.

Thermal Management: Issue: Speed control can affect the thermal performance of induction motors, especially at low speeds where cooling may be less effective.

Solution: Advanced thermal management techniques and efficient cooling systems need to be designed to ensure motor longevity and reliability.

Mechanical Stress: Issue: Frequent speed changes can introduce mechanical stress and wear on the motor and connected machinery.

Solution: Designing mechanical components to withstand variable speed operations and implementing smoother control strategies to reduce stress.

Cost: Issue: Advanced speed control solutions can be expensive due to sophisticated hardware and software requirements.

Solution: Research into cost-effective materials and components, and economies of scale can help reduce costs.

Maintenance and Reliability: Issue: Increased complexity in control systems can lead to higher maintenance requirements and potential reliability issues.

Solution: Developing robust and fault-tolerant control systems with easy diagnostic and maintenance capabilities.

8.1.2 Future Research Directions

Advanced Control Techniques: Focus: Investigating more advanced control strategies such as model predictive control (MPC), artificial intelligence (AI)-based control, and machine learning (ML) algorithms to enhance performance.

Goal: Improve efficiency, dynamic response, and adaptability of speed control systems.

Energy Efficiency: Focus: Researching new methods to further reduce energy consumption, especially under varying load conditions.

Goal: Enhance the overall energy efficiency of induction motor systems, contributing to sustainability goals.

Wireless and IoT Integration: Focus: Exploring wireless communication and Internet of Things (IoT) technologies for remote monitoring, control, and diagnostics.

Goal: Enhance the flexibility, convenience, and capabilities of motor control systems.

High-Power and High-Performance Applications: Focus: Tailoring control strategies for high-power and high-performance applications such as electric vehicles, aerospace, and industrial robotics.

Goal: Address the unique challenges and requirements of these demanding applications.

Safety and Standards Compliance: Focus: Ensuring that new control systems comply with safety standards and regulations.

Goal: Enhance the safety and regulatory compliance of induction motor control systems.

REFERENCES

[1] Upendra Reddy, C, Kashyap Kumar Prabhakar, Amit Kumar Singh & Praveen Kumar 2017, 'Performance evaluation of DTC IM Drive for an EV application', IEEE Power & Energy Society General Meeting, Electron., vol. 54, no. 5, pp. 2407-2416.

[2] Carlos A Martins & Adriano S Carvalho 2001, 'Technological trends in induction motor electrical drives', IEEE Porto Power Tech Conference, vol. 2.

[3] Farid Tazerart, Zahra Mokrani, Djamila Rekioua & Toufik Rekioua 2015, 'Direct torque control implementation with losses minimization of induction motor for electric vehicle applications with high operating life of the battery', International Journal of Hydrogen Energy, vol. 40, no. 39, pp. 13827-13838.

[4] Farasat, M, Trzynadlowski, AM & Fadali, MS 2014, 'Efficiency improved sensorless control scheme for electric vehicle induction motors', IET Electr. Syst. Transp., vol. 4, no. 4, pp. 122-131.

[5] Meriem Abdellatif, Mustapha Debbou, Ilhem Slama-Belkhodja & Maria Pietrzak-David 2014, 'Simple low-speed sensorless dual dtc for double fed induction machine drive', IEEE Transactions on Industrial Electronics, vol. 61, no. 8, pp. 3915-3922.

[6] Yongchang Zhang, Jianguo Zhu, Zhengming Zhao, Wei Xu & David G Dorrell 2012, 'An improved direct torque control for three-level inverter-fed induction motor sensorless drive', IEEE Transactions on Power Electronics, vol. 27, no. 3, pp. 1502-1513.

[7] Yukai Wang, Takumi Ito, Robert D Lorenz 2014, 'Loss manipulation capabilities of deadbeat-direct torque and flux control induction machine drives', Energy Conversion Congress and Exposition (ECCE) IEEE, pp. 5100-5107.

[8] Karpe, Suraj & Deokar, A, Sanjay, Dixit, M & Arati 2017, 'Switching losses minimization by using direct torque control of induction motor', Journal of Electrical Systems and Information Technology, vol. 4, no. 1, pp. 225-242.

[9] Solodkiy, E, Varzanosov, P & Belonogov, A 2017, 'Induction motor sensorless vector control with an adaptive speed observer and direct electrical angle correction in coordinate transformations', International

International Journal of
Electrical and Electronics Research (IJEER)
Research Article | Volume 12, Issue 3 | Pages 746-755 | e-ISSN: 2347-470X

Conference on Industrial Engineering, Applications and Manufacturing (ICIEAM) ISBN: 978-1-5090-5649-1.
[10] Chandra Sekhar & Marutheshwar, GV 2014, 'Modeling and field-oriented control of induction motor by using an adaptive neuro-fuzzy interference system control technique', International Journal of Industrial Electronics and Electrical Engineering, ISSN: 2347-6982, vol. 2, no. 10.
[11] Chandra Sekhar, D & Maruthes War, GV 2017, 'Direct torque control of three-phase induction motor with anfisand cuckoo search algorithms', International Journal of Pure and Applied Mathematics, ISSN: 1311-8080, vol. 114, no. 12, pp. 501-514.
[12] Vinay T Kumar & Srinivasa S Rao 2013, 'Modelling and simulation of direct torque-controlled induction motor drive using slip speed control', International Journal of Modelling and Simulation, ISSN: 0228-6203 (Print) 1925-7082 (Online), vol. 33, no. 2.
[13] Mohamad Amiri, Jafar Milimonfared & Davood Arab Khaburi 2018, 'Predictive torque control implementation for induction motors based on discrete space vector modulation', IEEE Transactions on Industrial Electronics, ISSN: 0278-0046, vol. 65, no. 9, pp. 6881-6889.
[14] Fengxiang Wang, Zhe Chen, Peter Stolze, Ralph Kennel, Mauricio Trincado & Jose Rodriguez 2015, 'A comprehensive study of direct torque control (DTC) and predictive torque control (PTC) for high performance electrical drives', EPE Journal, vol. 25, no. 1, pp. 12-21.
[15] Younes Sangsefidi, Saleh Ziaeinejad, Hamid Pairodin Nabi & Abbas Shoulaie 2014, 'Induction motor control based on approximate stator flux', Power Electronics IET, vol. 7, no. 11, pp. 2765-2777.
[16] Abdelkarim Ammar, Billel Talbi, Tarek Ameid, Younes Azzoug & Abdelaziz Kerrache 2017, 'Predictive direct torque control with reduced ripples and fuzzy logic speed controller for induction motor drive', 5th International Conference on Electrical Engineering – Boumerdes (ICEE-B), pp. 29-31.
[17] Sivakumar & Muthu Selvan, NB 2018, 'Reduction of source current harmonics in ANN controlled induction motor', Alexandria Engineering Journal, Elsevier B.V, vol. 57, pp. 1489-1499.
[18] Venkateswara Rao, M, Anand Kumar, A, & Obulesh, YP 2018, 'Artificial neural network and adaptive neuro fuzzy control of direct torque control of induction motor for speed and torque ripple control', WSEAS Transactions on Power Systems. E-ISSN: 2224-350X, vol. 13, pp. 1416-1422.
[19] Ranjit Kumar Bindal & Inderpreet Kaur 2019, 'Speed and torque control of induction motor using adaptive neuro-fuzzy interference system with DTC, 'Springer Nature Singapore Pte Ltd. ICAICR 2018, CCIS 955, pp. 815-825.
[20] Chandra Sekhar, O & Chandra Sekhar, K 2012, 'Multilevel inverter fed dtc control of induction motor drive', International Review on Modeling and Simulations (I. RE.MO. S.), N. 1ISSN 1974-9821, vol. 5.
[21] H. Karimi, S. Vaez-Zadeh and F. Rajaei Salmasi, "Combined Vector and Direct Thrust Control of Linear Induction Motors with End Effect Compensation", IEEE Transactions on Energy Conversion, vol. 31, no. 1, pp. 196-205, 2016.
[22] S. Gadoue, D. Giaouris and J. Finch, "MRAS Sensorless Vector Control of an Induction Motor Using New SlidingMode and Fuzzy-Logic Adaptation Mechanisms", IEEE Transactions on Energy Conversion, vol. 25, no. 2, pp. 394-402, 2010.
[23] I. Benlaloui, S. Drid, L. Chrifi-Alaoui and M. Ouriagli, "Implementation of a New MRAS Speed Sensorless Vector Control of Induction Machine", IEEE Transactions on Energy Conversion, vol. 30, no. 2, pp. 588-595, 2015.
[24] A. Smith, S. Gadoue and J. Finch, "Improved Rotor Flux Estimation at Low Speeds for Torque MRAS-Based Sensorless Induction Motor Drives", IEEE Transactions on Energy Conversion, vol. 31, no. 1, pp. 270-282, 2016.
[25] C. Schauder, "Adaptive speed identification for vector control of induction motors without rotational transducers", IEEE Transactions on Industry Applications, vol. 28, no. 5, pp. 1054-1061, 1992.
[26] H. Tajima and Y. Hori, "Speed sensorless fieldorientation control of the induction machine", IEEE Transactions on Industry Applications, vol. 29, no. 1, pp.175-180, 1993.