FIXED SPEED WIND GENERATOR MODEL PARAMETER ESTIMATION USING IMPROVED PARTICLE SWARM OPTIMIZATION AND SYSTEM FREQUENCY DISTURBANCES

F. González-Longatt†, P. Regulski*, P. Wall*, W. Terzija**

* *The University of Manchester, United Kingdom, fglongatt@ieee.org, pawel.regulski@postgrad.manchester.ac.uk, peter.wall@postgrad.manchester.ac.uk, vterzija@ieee.org*

† *Vice-President of Venezuelan wind Energy Association, Caracas, Venezuela, fglongatt@aveol.org.ve*

Keywords: generator modelling, parameter estimation, particle swarm optimization, wind power, wind turbine.

Abstract

When planning power system operation it is important to have reliable models of the elements of the power system. Fixed speed wind turbines are a widely installed generation technology that use a single squirrel cage induction generator. The local wind profile and the properties of the induction machine constitute the main considerations when modeling these wind turbines. Existing methods for estimating the parameter values of induction machine models use a wide variety of parameter estimation algorithms but primarily use active and reactive power measurements made during start-up or direct mechanical testing to fit the model to. Proposed here is a parameter estimation method that applies improved particle swarm optimization to active and reactive power measurements made during a deviation in system frequency to estimate the parameter values of a induction machine model. This method has shown good accuracy and the use of on-line data may prove beneficial in future applications.

1 Introduction

The single squirrel cage induction generator (SCIG) is a simple, but effective, generator design. This simple design makes the SCIG a practical and reliable solution in fixed speed wind turbine (FSWT) applications. The basic design of a FSW is shown in Figure 1.

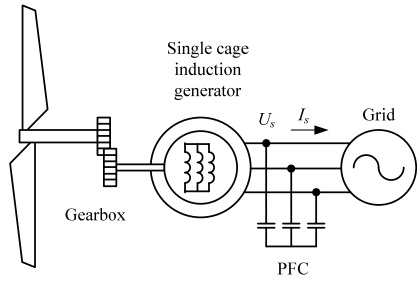

Figure 1: The basic configuration of a FSWT.

To ensure proper operation it is important to have a reliable model of the wind farm. This equivalent model is usually formed in one of two ways: reduce the wind farm to a single equivalent wind turbine [1], [2], [3], [4], [5]; or create several equivalent wind turbine models based on the variable wind profile of the site and aggregate these into a single equivalent wind turbine [5], [6]. These methods were compared in [7] and both found to be effective. Regardless of how the equivalent model is produced accurate parameter values will be necessary.

Existing solutions to the problem of estimating SCIG parameter values have two main characteristics: the parameter estimation technique employed and the source of the SCIG power output data the model is fitted to.

The parameter estimation techniques that have been employed include: non-linear least squares, Kalman Filters [10], genetic algorithms (GA) [11], [12], [13], [14], local search algorithms (LSA), simulated annealing (SA), differential evolution [15] and various forms of particle swarm optimization (PSO) [11], [16], [17], [18].

The data used is usually the active and reactive power output of the SCIG during start-up [12], [15], [16], [17], or direct mechanical testing.

The method proposed in this paper uses improved particle swarm optimization (IPSO) and the SCIG response to a change in the grid frequency.

PSO achieves good solution quality by allowing constructive interaction between population members based on the best solutions found in past iterations [11], [12]. [19]. IPSO differs from PSO as it defines the concept of inertia weight, introduced in [20], as a function of the iteration count. This can be done in a variety of ways [17], [21], [22] and affords an improvement in convergence and accuracy [22].

During a disturbance the power factor correction that is necessary when using SCIGs will reduce the voltage deviation at the SCIG terminals but leave the frequency deviation relatively unaffected. Therefore, fitting the SCIG model to the response of the SCIG to a change in grid frequency could be a useful source of data for parameter estimation in FSWT wind farm applications.

The intent of this paper is to demonstrate the validity of using IPSO to estimate the parameter values of a SCIG model based on the response of the SCIG to a change in grid frequency. The formulation of the SCIG model and the IPSO algorithm are describe in sections 2 and 3, respectively. Section 4 contains the results of some of the simulations performed to verify the method.

2 Formulation of Parameter Estimation for an SCIG

2.1 Induction generator model

The induction generator model used in this paper is a standard induction machine model, detailed descriptions of which can be found in [23], [24], [25]. This is possible because an induction generator is fundamentally an induction machine with torque applied to the shaft.

Using an arbitrary reference at angular velocity ω the electrical dynamics can be modeled using the following second order state space model [24]:

$$\begin{aligned}
\dot{\lambda}_{qs} &= v_{qs} - R_s i_{qs} - \omega\lambda_{ds} \\
\dot{\lambda}_{ds} &= v_{ds} - R_s i_{ds} + \omega\lambda_{qs} \\
0 &= v'_{qr} - R'_r i'_{qr} - \left(\omega - \omega_r\right)\lambda'_{dr} \\
0 &= v'_{dr} - R'_r i'_{dr} + \left(\omega - \omega_r\right)\lambda'_{qr}
\end{aligned} \tag{1}$$

where the indices d, q, s and r refer to the d-q reference frame and the stator and rotor, respectively. The variable ω_r is the electrical angular velocity.

The electrical variables, namely voltage (v), current (i), resistance (R) and flux linkage (λ), are all referred to the stator, as indicated by the prime notation.

The flux linkages of a SCIG are expressed in terms of current and inductance (L), as follows [24]:

$$\begin{aligned}
\lambda_{qs} &= L_{ls} i_{qs} + L_m\left(i_{qs} + i'_{qr}\right) \\
\lambda_{ds} &= L_{ls} i_{ds} + L_m\left(i_{dr} + i'_{dr}\right)
\end{aligned} \tag{2}$$

The mechanical dynamics can be described using a fourth order model in terms of the rotor angular velocity (ω_{mec}) and angular position (θ_{mec}).

$$\begin{aligned}
\dot{\theta}_{mec} &= \omega_{mec} \\
\dot{\omega}_{mec} &= \frac{1}{2H}\left(P_{mec} - P_{ele}\right)
\end{aligned} \tag{3}$$

where H is the rotor inertia constant, P_{mec} is the shaft mechanical power, and P_{elec} is the electromechanical torque given by:

$$P_{ele} = v_{ds} i_{ds} + v_{qs} i_{qs} \tag{4}$$

The reactive power is calculated as follows:

$$Q_{ele} = v_{qs} i_{ds} - v_{ds} i_{qs} \tag{5}$$

Equations (1)-(5) are the model of a symmetrical induction machine that is used in this paper. The necessary transformation of voltage and current from the abc to the dq reference frame is achieved as follows:

$$\begin{bmatrix} v_{qs} \\ v_{ds} \end{bmatrix} = \frac{1}{3}\begin{bmatrix} 2\cos\theta & \cos\theta + \sqrt{3}\sin\theta \\ 2\sin\theta & \sin\theta + \sqrt{3}\cos\theta \end{bmatrix}\begin{bmatrix} v_{abs} \\ v_{bcs} \end{bmatrix} \tag{6}$$

$$\begin{bmatrix} i_{as} \\ i_{bs} \end{bmatrix} = \begin{bmatrix} \cos\theta & \sin\theta \\ \dfrac{-\cos\theta + \sqrt{3}\sin\theta}{2} & \dfrac{-\cos\theta + \sqrt{3}\sin\theta}{2} \end{bmatrix}\begin{bmatrix} i_{qs} \\ i_{ds} \end{bmatrix} \tag{7}$$

The three phases (a, b and c) are connected using an un-earthed Y configuration, so $i_{cs} = -i_{as} - i_{bs}$. This configuration makes it possible to use only two line-to-line input voltages in the model instead of three line-to-neutral voltages.

In the preceding equations, θ is the angular position of the arbitrary reference. A rotating reference frame with an angular velocity $\omega = \omega_s$, is suitable modelling work where the system frequency, f_s, is changing ($\omega_s = 2\pi f_s$). The increased complexity involved in using a rotating reference frame with variable speed is unavoidable as it is necessary to include the frequency behavior of the external system in the model.

2.2 Formulation of the parameter estimation problem

In essence, parameter estimation consists of comparing the response of the real system with the response of a system model and then updating the model parameter vector **x,** in some way, to reduce the difference, or error ε, between the response of the model and that of the real system.

There are six parameters in the non-linear model that cannot be measured directly. Therefore, the parameter vector contains six unknown variables that must be estimated:

$$\mathbf{x} = [H, R_s, L_{ls}, R'_r, L'_{lr}, L_m]^T \tag{8}$$

To obtain initial values for the parameter vector the model can be solved for the system state prior to the frequency deviation. This reduces the dimension of the problem affording an increase in the accuracy of the final solution and a reduction in the time taken to converge to this solution.

Parameter estimation can be treated as an optimization problem in which an *objective function* that describes the difference between the response of the system and the system model is minimized:

$$\min \varepsilon(\mathbf{x}) = \min \frac{1}{n}\sum_{j=1}^{n}\left[\left(P_{mea,j} - P_{sim,j}\right)^2 + \left(Q_{mea,j} - Q_{sim,j}\right)^2\right] \tag{9}$$

where P_{mea} and Q_{mea} are the measured active and reactive power from the response of the real system and P_{sim} and Q_{sim} correspond to the active and reactive power response of the system model. n is the number of samples of active and reactive power that are used during the estimation process.

3 Improved Particle Swarm Optimization

Particle swarm optimisation is an iterative method that was first proposed by Kennedy and Eberhart in 1995 [19]. They were inspired by the behaviour of animals that travel in large groups, in which each member communicates useful information to the other members.

To capture the benefit of this form of social behaviour a number of *particles* are generated to form a *swarm*. With each iteration these particles are moved around the searching space, based on information received from the other particles regarding the best solutions from past iterations, to find the best solution. The procedure for executing the method is depicted in a block diagram in Figure 2.

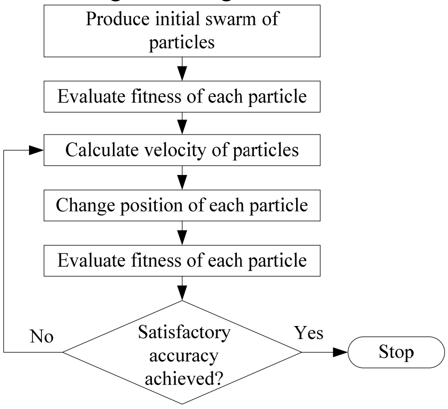

Figure 2: Flowchart of PSO algorithm.

In the first iteration each particle is a parameter vector **x** with randomly selected values (limited by a certain range particular to each parameter). The parameter values represented by the particle is the *position* of that particle within the swarm. Each particle represents a set of system parameter values. Using this position the response of the system model can be calculated.

The error in the particle position can be calculated by comparing the model response and the measured response using an expression like (9). The reciprocal of this error is the *fitness* of a particle and indicates the accuracy of the model response produced by the position the particle has within the searching space.

The position of each particle is updated each iteration using the *velocity* of the particle. Particle velocity describes the rate and direction in which the position, and therefore each of the model parameter values, changes with each iteration. The velocity of a particle and its next position can be calculated as follows:

$$\boldsymbol{\upsilon}_i^{k+1} = W\boldsymbol{\upsilon}_i^k + c_1 r_1(\mathbf{X}_i^k - \mathbf{x}_i^k) + c_2 r_2(\mathbf{X}_g^k - \mathbf{x}_i^k) \tag{10}$$

$$\mathbf{x}_i^{k+1} = \mathbf{x}_i^k + \boldsymbol{\upsilon}_i^{k+1} \tag{11}$$

where υ_i^k and υ_i^{k+1} are the current and next step velocity of the ith particle, respectively, W is the inertia weight, $\mathbf{X}_i^k$ is the *best* previous position of the ith particle, $\mathbf{X}_g^k$ is the *best* position for any iteration or particle, $\mathbf{x}_i^k$ is the actual ith particle position, c_1 and c_2 are the acceleration coefficients usually equal to 2.0 and r_1 and r_2 are random numbers ranging from 0.0 to 1.0. Here, the term best position refers to the position with the highest fitness.

The position and velocity of each particle is updated each iteration until the fitness of $\mathbf{X}_g^k$ reaches a pre-defined threshold or the achieved maximum number of iterations is reached.

IPSO uses a variable inertia weight term, in this application this term is modulated as a function of the iteration count using the expression proposed in [17]:

$$W = \left\{ \frac{(iter_{\max} - iter)^{\gamma}}{(iter_{\max})^{\gamma}} \right\}(W_{initial} - W_{final}) + W_{final} \tag{12}$$

where $W_{initial}$ is the initial inertia weight, W_{final} is the final inertia weight, $iter_{\max}$ is the maximum number of iterations, $iter$ is the number of the current iteration and γ is the nonlinear modulation index. This definition of inertia weight causes it to decrease with each iteration reducing the contribution of the past velocity when calculating the future velocity. This means that as the iteration count increases the velocities will tend to be smaller, allowing more precise corrections of particle positions and hence improved accuracy and convergence.

3.1 Initialization and tuning of IPSO

The IPSO algorithm used here is initialized by randomly assigning a set of parameter values, bound within a given feasible range for each parameter, for each of the thirty particles that make up the swarm. Each element of each particle is selected according to the following formula:

$$x_i = x_{i\min} + rand(x_{i\max} - x_{i\min}) \tag{13}$$

where x_i is the ith element of the parameter vector, *rand* is a uniformly distributed random number between 0.0 and 1.0, $x_{i\min}$ and $x_{i\max}$ are the minimum and maximum possible value of the ith element of the parameter vector, respectively. The minimum and maximum values should be carefully selected for each parameter to ensure that any likely true value lies within the specified range.

To ensure the selection of suitable values for use in (12) a number of empirical studies were carried out. For each candidate configuration 100 trials have been performed and the mean fitness of the final solution calculated. The best settings were found by comparing the mean fitness for each configuration. This approach led to the selection of the following values: $W_{initial} = 0.90$, $W_{\text{final}} = 0.01$ and $\gamma = 1.20$.

4 Simulation and Results

Simulations were performed to demonstrate the validity of using the model and IPSO algorithm described in sections 2 and 3 for estimating the parameter values of a SCIG based on its response to a deviation in system frequency

The IPSO algorithm used a swarm size of thirty particles, the maximum number of iterations was fifty and it was implemented in MATLAB®.

The simulated test system that was used to generate the response of the 'real' system consisted of an AC programmable voltage source connected to a SCIG through a single busbar. This system layout was used as it is a rough representation of the connection of a FSWT wind farm to a large power system.

Two case studies were examined. In the first case, a step change in frequency was simulated. In the second case, a time series representing the frequency response of a synchronous generator during a load step change was simulated. The time series of frequency was obtained from a simulation performed with DIgSILENT® PowerFactory™ [26]. In both of these test cases the voltage output of the source was unchanged.

4.1 Case I: -2Hz Frequency Step Change

In Case I a step change of -2Hz was applied to the frequency of the programmable voltage source output to represent a disturbance. This very large change in frequency was simulated to verify that the parameter estimation method is viable when large disturbance occur.

The parameter values found using the IPSO algorithm are shown in Table I. Figure 3 contains a comparison of the simulated 'real' system and that model output. These results show that the parameter estimation method is capable of accuracy that should satisfy the needs of stability analysis.

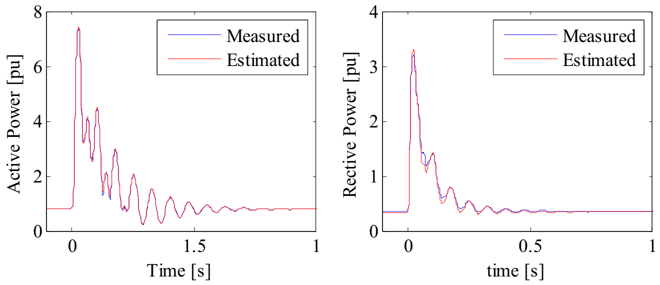

Figure 3: Results with IPSO: Case I.

	Real	*Estimated*
P_e Error [%]	2.0950	
Q_e Error [%]	8.4823	
H [s]	1.1877	1.1999
R_r [p.u]	0.0010	0.0013
L_{lr} [p.u]	0.0100	0.0116
R'_r [p.u]	0.0010	0.0010
L'_{ls} [p.u]	0.0100	0.0081
L_m [p.u]	3.0000	3.1193

Table 1: Summary of Results for Case I.

4.2 Case II: System frequency response

In Case II the frequency disturbance used is the response of a synchronous generator to a load step change, Figure 4. This frequency disturbance was selected as it is a better representation of the actual frequency deviation that occurs in a power system.

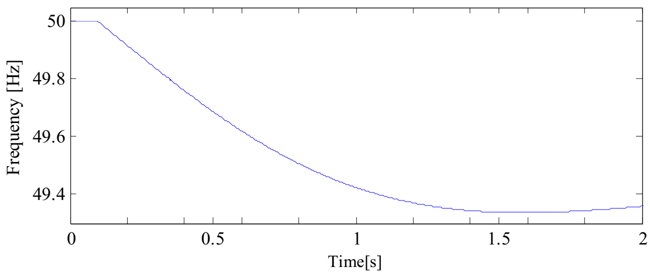

Figure 4: Frequency response of synchronous generator during a load step change.

Application of the parameter estimation method to the response of the SCIG to this frequency disturbance produced the results presented in Table 2.

	Real	*Estimated*
P_e Error [%]	2.5505	
Q_e Error [%]	16.4683	
H [s]	1.1877	1.2058
R_r [p.u]	0.0010	0.0009
L_{lr} [p.u]	0.0100	0.0078
R'_r [p.u]	0.0010	0.0010
L'_{ls} [p.u]	0.0100	0.0119
L_m [p.u]	3.0000	2.5819

Table 2: Summary of Results for Case II.

The IPSO produced estimates of the machine parameters with accuracy comparable to that of Case I for this more complex and realistic frequency disturbance.

4 Conclusions

When operating a power system, it is important to have accurate models of all of the significant components. This paper proposed a method for estimating the parameter values of a single squirrel cage induction generator model, typical of those used in fixed speed wind turbine applications, using the response of the machine to a deviation in frequency and the improved particle swarm algorithm.

This method for estimating the parameter values of the squirrel cage induction generator model produces good results for both a large step change in frequency and the frequency response of a synchronous generator.

The method uses the response of the machine to a frequency disturbance, instead of the start-up or mechanical test data usually used in other induction machine parameter estimation methods. This difference could well be beneficial as most wind turbines will be installed in relatively remote locations so direct testing once the generator is in operation could be awkward. Furthermore, the growing trend toward the installation of wide area monitoring equipment will mean that the data showing the response of a wind turbine to a disturbance will become increasingly available.

References

[1] M. Poller and S. Achilles, "Aggregated wind park models for analyzing power system dynamics," presented at the 4th International Workshop on Large-Scale Integration of Wind Power and Transmission Networks for Offshore Wind Farms, Billund, Denmark, 2003.

[2] J. G. Slootweg and W. L. Kling, "Modeling of large wind farms in power system simulations," in *Power Engineering Society Summer Meeting, 2002 IEEE*, 2002, pp. 503-508 vol.1.

[3] L. M. Fernández, J. R. Saenz, and F. Jurado, "Dynamic models of wind farms with fixed speed wind turbines," *Renewable Energy,* vol. 31, pp. 1203-1230, 2006.

[4] V. Akhmatov and H. Knudsen, "An aggregate model of a grid-connected, large-scale, offshore wind farm for power stability investigations--importance of windmill mechanical system," *International Journal of Electrical Power & Energy Systems,* vol. 24, pp. 709-717, 2002.

[5] Y. Q. Jin and P. Ju, "Dynamic equivalent modeling of FSIG based wind farm according to slip coherency," in *Sustainable Power Generation and Supply, 2009. SUPERGEN '09. International Conference on*, 2009, pp. 1-7.

[6] P. Ledesma, J. Usaola, and J. L. Rodríguez, "Transient stability of a fixed speed wind farm," *Renewable Energy,* vol. 28, pp. 1341-1355, 2003.

[7] Q. Wei, R. G. Harley, and G. K. Venayagamoorthy, "Dynamic Modeling of Wind Farms with Fixed-Speed Wind Turbine Generators," in *Power Engineering Society General Meeting, 2007. IEEE*, 2007, pp. 1-8.

[8] T. Littler, B. Fox, and D. Flynn, "Measurement-based estimation of wind farm inertia," in *Power Tech, 2005 IEEE Russia*, 2005, pp. 1-5.

[9] J. M. Kennedy, B. Fox, T. Littler, and D. Flynn, "Validation of Fixed Speed Induction Generator Models for Inertial Response Using Wind Farm Measurements," *Power Systems, IEEE Transactions on,* vol. PP, pp. 1-1, 2010.

[10] P. Marino, V. Mungiguerra, F. Russo, and F. Vasca, "Parameter and state estimation for induction motors via interlaced least squares algorithm and Kalman filter," in *Power Electronics Specialists Conference, 1996. PESC '96 Record., 27th Annual IEEE*, 1996, pp. 1235-1241 vol.2.

[11] D. C. Huynh and M. W. Dunnigan, "Parameter estimation of an induction machine using a dynamic particle swarm optimization algorithm," in *Industrial Electronics (ISIE), 2010 IEEE International Symposium on*, 2010, pp. 1414-1419.

[12] K. S. Huang, Q. H. Wu, and D. R. Turner, "Effective identification of induction motor parameters based on fewer measurements," *Energy Conversion, IEEE Transactions on,* vol. 17, pp. 55-60, 2002.

[13] R. R. Bishop and G. G. Richards, "Identifying induction machine parameters using a genetic optimization algorithm," in *Southeastcon '90. Proceedings., IEEE*, 1990, pp. 476-479 vol.2.

[14] T. Sag and M. Cunkas, "Multiobjective genetic estimation to induction motor parameters," in *Electrical Machines and Power Electronics, 2007. ACEMP '07. International Aegean Conference on*, 2007, pp. 628-631.

[15] R. K. Ursem and P. Vadstrup, "Parameter identification of induction motors using differential evolution," in *Evolutionary Computation, 2003. CEC '03. The 2003 Congress on*, 2003, pp. 790-796 Vol.2.

[16] A. Karimi, M. A. Choudhry, and A. Feliachi, "PSO-based Evolutionary Optimization for Parameter Identification of an Induction Motor," in *Power Symposium, 2007. NAPS '07. 39th North American*, 2007, pp. 659-664.

[17] G. Chen, W. Guo, and K. Huang, "On Line Parameter Identification of an Induction Motor Using Improved Particle Swarm Optimization," in *Control Conference, 2007. CCC 2007. Chinese*, 2007, pp. 745-749.

[18] C. Picardi and N. Rogano, "Parameter identification of induction motor based on particle swarm optimization," in *Power Electronics, Electrical Drives, Automation and Motion, 2006. SPEEDAM 2006. International Symposium on*, 2006, pp. 968-973.

[19] R. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Micro Machine and Human Science, 1995. MHS '95., Proceedings of the Sixth International Symposium on*, 1995, pp. 39-43.

[20] G. Yang, "A Modified Particle Swarm Optimizer Algorithm," in *Electronic Measurement and Instruments, 2007. ICEMI '07. 8th International Conference on*, 2007, pp. 2-675-2-679.

[21] M. Senthil Arumugam, A. Chandramohan, and M. V. C. Rao, "Competitive approaches to PSO algorithms via new acceleration co-efficient variant with mutation operators," in *Computational Intelligence and Multimedia Applications, 2005. Sixth International Conference on*, 2005, pp. 225-230.

[22] A. Ratnaweera, S. K. Halgamuge, and H. C. Watson, "Self-organizing hierarchical particle swarm optimizer with time-varying acceleration coefficients," *Evolutionary Computation, IEEE Transactions on,* vol. 8, pp. 240-255, 2004.

[23] P. Kundur, *Power System Stability and Control* Palo Alto , California: McGraw-Hill Inc., 1994.

[24] P. C. Krause, O. Wasiynczuk, and S. D. Sudhoff, *Analysis of Electric Machinery and Drive Systems*, 2nd ed.: Wiley-IEEE Press, 2002.

[25] M. Godoy-Simoes and F. A. Farret, *Renewable Energy Systems-Design and Analysis with Induction Generators*, 2nd ed. vol. I. New York: CRC Press, 2002.

[26] DIgSILENT, "DIgSILENT PowerFactory," 15.0.525.1 ed. Gomaringen / Germany, 1986-2011.