

COBEM-2017-0831

A NEURAL GENETIC HYBRID MODEL FOR EIGENSTRUCTURE ALLOCATION IN THE LQR PROJECT IN DFIG

Rildenir Silva

Ivanildo Abreu

State University of Maranhão - (UEMA) University City Paulo VI s/n - St. Louis

e-mails:

ril.ifmatematico@gmail.com

ivanildoabreu@yahoo.com.br

Henrique Amaral

State University of Maranhão - (UEMA) University City Paulo VI s/n - St. Louis

e-mail:hmca13@gmail.com

Cristovam Filho

State University of Maranhão - (UEMA) University City Paulo VI s/n - St. Louis

e-mail: cristovamfilho17013@gmail.com

Paulo Sousa

State University of Maranhão - (UEMA) University City Paulo VI s/n - St. Louis

e-mail: renato.yotta@gmail.com

Abstract. A hybrid neuronal genetic model is proposed with the objective of solving the Riccati Algebraic Equation (RAE) that is associated to the restricted optimization structure of the Linear Quadratic Regulator (LQR) problem. The application of this hybrid model of artificial intelligence will be performed in a wind power generation system, in particular, the double fed induction generator (DFIG). For this, a recurrent neural network with multiple layers is used where its performance is realized by metrics of the norm of infinity associated with RAE and energy surfaces as a function of the positive definite symmetric matrix and the Cholesky factor.

Keywords: Artificial Neural Networks (RNA), genetic algorithm (GA), DFIG, LQR.

1. INTRODUCTION

In Brazil, the first experience with the generation and use of wind energy was through a wind turbine installed in the Fernando de Noronha archipelago in the 90's. Over the years, several wind farms have been installed throughout Brazil in which are installed wind turbines that are distinguished in two types:

- Generator driven directly by the Turbine shaft;
- Generator driven by the Turbine shaft, using multiplier.

Wind Energy can be converted into electrical energy through aero generators which are systems that harness wind energy through the propellers, converting it into mechanical energy which in turn is converted into electrical energy through an electricity generator.

The principle of operation of the DFIG is identical to AC electric generators, but has more relevant features that allow them to run at speeds slightly above or below their natural synchronous speed. These generators are widely used in large wind turbines of varying speed, due to the speed of the wind being variant with the time. Figure (1) illustrates a DFIG installation model:

According to Hansen *et al.* (2004), the operating principle of the DFIG is based on the slip that can be determined by $s = \frac{w_s - w_m}{w_s}$ which represents the power flow.

The demand for new renewable energy sources for the development of increasingly efficient energy generation systems has been made possible by new computational techniques to optimize these activities. In this sense, we will approach in this work computational techniques based on artificial intelligence such as: Artificial Neural Networks and genetic algorithms in order to synthesize optimal control systems related to the Linear Quadratic Regulator (LQR) problem.

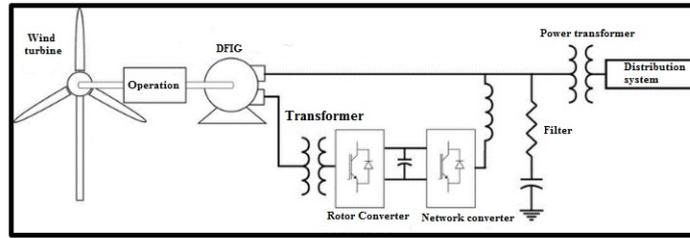


Figure 1. : DFIG model coupled to the turbines

In this context, (Muyeen *et al.*, 2008) lists the main areas of application where control systems models are concentrated, where we will find the LQR project fundamentals such as industrial sector, power generation, noise attenuation, machines of direct current, among others. According to (Pinto, 2007), the wide use of the LQR project has brought great results in wind power systems that use as power generation the dual-fed induction generators.

2. METHODOLOGY

These goals will be achieved through the use of Artificial Intelligence paradigms, in particular, genetic algorithms and neural networks. The first one is responsible for the allocation of eigenvalues and eigenvectors, while the second justifies the solution of the Riccati Algebraic Equation and the convergence of GA and RNA stability are guaranteed.

3. INTELLIGENT CONTROLLER

Here presents a hybrid neuronal genetic model to be applied to solve the LQR problem. According to (Wang and Wu, 1998) these evolutionary models come from studies using a genetic algorithm to select the Q and R weighting matrices and a recurrent artificial neural network that solves the Riccati Algebraic Equation. Figure (2) illustrates a scheme for these procedures:

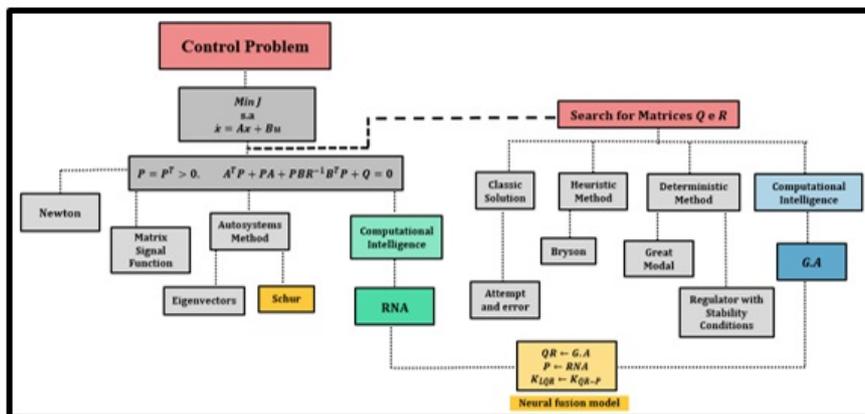


Figure 2. Intelligent control scheme.

3.1 Classical approach LQR

This approach covers the formulation and solution of the Linear Quadratic Regulator problem, as well as the Schur method, which solves RAE. In this classic analysis of LQR, the input vector u is sought which aims to minimize a cost function J with some restrictions for the dynamic model of the system. According to some authors, one of the advantages of the linear quadratic regulator is the guaranteed stability margin where the phase margin is 60° and the gain margin is infinite, thus having a stability quality. The Linear Quadratic Regulator is modeled by a quadratic performance index and has some constraints on its linear and time invariant state cost function as follows:

$$\min J = \int_0^\infty [x(t)^T Q x(t) + u(t)^T R u(t)] dt \quad (1)$$

s.a,

$$\dot{x} = Ax(t) + Bu(t), \quad \text{com } x(0) = x_0. \quad (2)$$

at where $x \in R^n$, $u \in R^m$, the pair of matrices A and B is controllable, the pair A and C is observable, $Q \geq 0$, $R > 0$ and all are symmetrical. Respectively, The matrices $Q \in R^{n \times n}$ and $R \in R^{m \times m}$ are definite and semi-definite positive matrices.

According to (Athans and Falb, 1966), some variational calculation methods can be used to solve the optimization problem that is inherent to the optimization structure of equations (1) and (2). Such methods will provide the control law that will minimize the cost function J and, at the same time, obey the constraints of the vector \dot{x} related to the optimization structure that follows,

$$u = Kx \quad (3)$$

At where $K = -R^{-1}B^TP$ is the gain of the controller. The gain K depends on the matrix P symmetric and positive definite, which is the solution of EAR given by:

$$A^T + PA - PBR^{-1}B^TP + Q = 0 \quad (4)$$

3.1.1 RAE and the LQR

According to (Laub, 1979), the Riccati Algebraic Equation (RAE) can be solved by reducing the Hamiltonian matrix. This matrix has the following structure:

$$H = \begin{bmatrix} A & -BR^{-1}B^T \\ -Q & -A^t \end{bmatrix} \quad (5)$$

Using the Schur method to obtain a quasi-triangular shape using orthogonal similarity transformations, we can find an orthogonal matrix U , which transforms the Hamiltonian matrix H into the real Schur form, that is:

$$T = W^T H W = \begin{bmatrix} T_{11} & T_{12} \\ 0 & T_{22} \end{bmatrix} \quad (6)$$

Where the blocks T_{11} e T_{22} are quasi-triangular upper matrices. W is a matrix partitioned into four dimension blocks $n \times n$. If the matrix W_{11} are non-singular matrices, the solution of the Riccati Algebraic Equation is definite positive, that is:

$$P = W_{21}W_{11}^{-1} \quad (7)$$

The matrix T_{11} has eigenvalues that represent the closed-loop spectrum of the matrix $(A - BR^{-1}B^TP)$.

Another important consideration is that the polar region of the optimal closed-loop system is due to the following relationship:

$$(A - BR^{-1}B^TP) = W_{11}T_{11}W_{11}^{-1} \quad (8)$$

4. GENETIC ALGORITHM ARCHITETURE

For the proposed GA model, it has been as weights matrices Q e R which establishes the free parameters of the LQR project, since they will be used to adjust the optimal gain of the controller. The main objective is the search for the weighting matrices that form the quadratic performance index J . Given this context, a nonlinear programming problem will arise to optimize the Q and R weighting matrices in terms of computational evolution algorithms and data structures, according to the following equation:

$$\min_{Q,R} \sum_{i=1}^n p_i s_i(Q, R) \quad (9)$$

s.a,

$$\begin{aligned} S_i(Q, R) &\leq 1 & i &= 1, \dots, n \\ \lambda_{ei} &\leq \lambda_{ci}(Q, R) \leq \lambda_{di} & i &= 1, \dots, n \end{aligned}$$

At where p_i are the weights related to the sensitivity s_i/ϵ_i normalized with respect to i th project specification where we should have $\epsilon_i > 0$. In this model, the genetic representation of the Q and R matrices is approached through the operations between the chromosomes arriving at the performance evaluations. According to Abreu *et al.* (2008), the selection operation is based on the values of the performance function and chromosomal model that represents the weights matrices Q and R is presented below:

$$QR_z = \bigcup_{i,j=1}^n q_{i,j} \wedge \bigcup_{j,i=1}^m r_{i,j} \quad (10)$$

with $i \leq j$, $z = 1, \dots, n_{indiv}$, $q_{ij} \in r_{ij}$ Represent the genes of the matrices QR_z and the term n_{indiv} Characterizes the number of individual chromosomes of a population. The random generation model of the search matrices is given by:

$$q_{ij} = \begin{cases} P_{Q,\alpha} + P_{Q,\beta}k_{Q_{i,j}}, & i = j \\ P_{Q,\lambda}k_{Q_{i,j}}, & i \neq j \\ i, j = 1, \dots, n \end{cases} \quad (11)$$

with $P_{Q,\alpha}$ and $P_{Q,\beta}$ are respectively the fixed and variable parameters belonging to the elements of the diagonal of $q_{i,j}$. The model of generation of the individual for the matrix R is identical to the model of generation of the individual for the matrix Q , as it shows the equation that follows:

$$r_{ij} = \begin{cases} P_{R,\alpha} + P_{R,\beta}k_{R_{i,j}}, & i = j \\ P_{R,\lambda}k_{R_{i,j}}, & i \neq j \\ i, j = 1, \dots, n \end{cases} \quad (12)$$

5. ARCHITECTURE OF RNAR

For the synthesis of the LQR project, the new formulation can be considered as an unrestricted optimization problem, so that, given the matrices A , B , Q and R , we intend to find the matrices P and L in order to minimize an energy function, given by the new formulation:

$$\min_{P,L} \xi(PSP - A^T P - PA - Q + LL^T - P) \quad (13)$$

where $S = BR^{-1}B^T$. Considering that the activation function ξ_{ij} , is a convex function bounded at the bottom, then it is shown that the sum of the Riccati Algebraic Equation, using Cholesky's factorization, can be rewritten as follows:

After reformulating the LQR simplification problem, it can now be considered an unrestricted optimization problem, so that, knowing the arrays A , B , Q and R , we must find the arrays P and L which, in fact, will minimize an energy function such that:

$$\min_{P,L} E(P, L) = \sum_{i=1}^n \sum_{j=1}^n \{e_{ij} [g_{ij}(P)] + e_{ij} [h_{ij}(P, L)]\} \quad (14)$$

where:

e_{ij} : is a convex function limited inferiorly; $G(P) = [g_{ij}(P)] = PSP - A^T P - PA - Q$; $H(P, L) = [h_{ij}(P, L)] = LL^T - P$; and $h_{ij}(P, L) = \sum_{k=1}^{\min(i,j)} (l_{ik}l_{jk} - p_{ij})$.

Figure 3 shows the architecture of the recurrent neural network to solve Riccati's algebraic equation which consists of four bidirectionally connected layers: an output layer represented by $\mathbf{V}(t)$ With the EAR solution, an input layer represented by $\mathbf{U}(t)$ and two hidden layers $\mathbf{Y}(t)$ e $\mathbf{Z}(t)$ the latter being the Cholesky factor of \mathbf{P} :

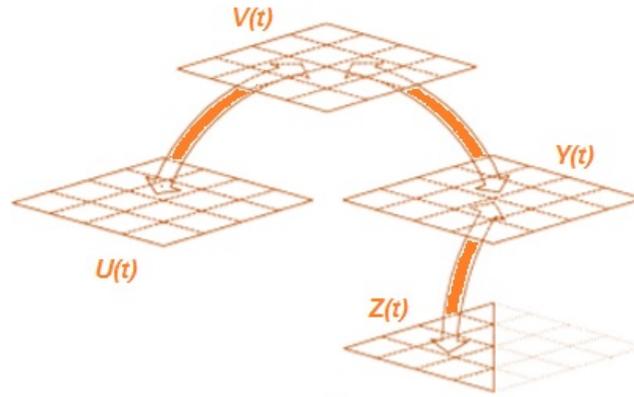


Figure 3. Recurrent Neural Network Architecture with multiple layers.

The matrix form of dynamic equations that symbolize the recurrent neural network, being, $i = 1, \dots, n$ and $j = 1, \dots, n$ are given by:

$$\frac{d\mathbf{V}(t)}{dt} = -\eta_v[\mathbf{V}(t)\mathbf{S}\mathbf{U}(t) + \mathbf{U}(t)\mathbf{S}\mathbf{V}(t) - \mathbf{A}\mathbf{U}(t) - \mathbf{U}(t)\mathbf{A}^T - \mathbf{Y}(t)] \quad (15)$$

$$\frac{d\mathbf{Z}(t)}{dt} = -\eta_z\mathbf{Y}(t)\mathbf{Z}(t) \quad (16)$$

$$\mathbf{U}(t) = \mathbf{F}[\mathbf{V}(t)\mathbf{S}\mathbf{V}(t) - \mathbf{A}^T\mathbf{V}(t) - \mathbf{V}(t)\mathbf{A} + \mathbf{Q}] \quad (17)$$

$$\mathbf{Y}(t) = F[\mathbf{Z}(t)\mathbf{Z}(t)^T - \mathbf{V}(t)] \quad (18)$$

which $\mathbf{U}(t) = [u_{ij}(t)]$, $\mathbf{V}(t) = [v_{ij}(t)]$ and $\mathbf{Y}(t) = [y_{ij}(t)]$ are square matrices of activation of the states and $\mathbf{Z}(t) = [z_{ij}(t)]$ is a lower triangular square matrix of states.

$\mathbf{V}(0) = \mathbf{V}(0)^T$, with $\mathbf{Y}(0) \neq 0$, η_v and η_z are the weights, positive parameters of network and $F = [f_{ij}]$ is a symmetric matrix of non-decreasing activations functions.

In the proposed neural network architecture, there are four bidirectionally connected layers, where $\mathbf{V}(t)$ is the output layer, $\mathbf{U}(t)$ is the input layer, and $\mathbf{Y}(t)$ and $\mathbf{Z}(t)$ are two hidden layers that perform the interconnection between P and U . The matrices $\mathbf{V}(t)$, $\mathbf{U}(t)$ and $\mathbf{Y}(t)$ are square characterizing layers of order $n \times n$. The matrix $\mathbf{V}(t)$ represents the computational result of P and the state matrix $\mathbf{Z}(t)$ represents L which is the Cholesky factor of P . The matrix P , presented in equation (13) represents the solution of the RAE.

5.1 Stability Analysis

The stability of the recurrent neural network proposed here and analyzed in more detail in (Wang and Wu, 1998) is necessary since it is a continuous nonlinear dynamic system in time. This stability can be analyzed by using the direct Lyapunov method given by the following theorem:

Theorem 1. If every activation function is continuous, at least part-differentiable, and still non-decreasing, that is $\frac{df_{ij}(\xi)}{d\xi} \geq 0$ to $-\infty < \xi < +\infty$ and $f_{ij}(\xi) = 0$, if $\xi = 0$, with $(i, j = 1, 2, \dots, n)$. Then the recurrent neural network is asymptotically stable, that is, $\forall \mathbf{V}(0) \in \mathbf{Z}(0)$, $\exists \tilde{\mathbf{Z}}, \tilde{\mathbf{V}}$ such that $\lim_{t \rightarrow \infty} \mathbf{Z}(t) = \tilde{\mathbf{Z}}$ and $\lim_{t \rightarrow \infty} \mathbf{V}(t) = \tilde{\mathbf{V}}$.

5.2 Solvability analysis

Theorem 2: This theorem ensures a necessary and sufficient condition for the flow of the gradient function to converge to a symmetric and definite positive solution of Riccati Algebraic Equation, P , of any initial state (Wang and Wu, 1998), $\frac{df_{ij}(\xi)}{d\xi} \geq 0$ to $-\infty < \xi < +\infty$ and $f_{ij}(\xi) = 0$, if $\xi = 0$. The RNAR solution, V , is always symmetric and positive definite for the Riccati equation of any symmetric initial state $\mathbf{V}(0)$ and any non-zero initial state $\mathbf{Z}(0)$, this is, $\forall \mathbf{V}(0) = \mathbf{V}(0)^T, \forall \mathbf{Z}(0) \neq 0, \tilde{\mathbf{V}} = \mathbf{V}$, if and only if, $rank(\tilde{\mathbf{V}}\mathbf{S} - \mathbf{A}) = n$ and all diagonal elements in $\tilde{\mathbf{Z}}$ are non-zero, this is, for all $i, \tilde{z}_{ii} \neq 0$ and $i = 1, \dots, n$.

6. TURBINE GENERATOR MODEL

The wind turbine converts kinetic energy from the wind into rotational mechanical energy. The ideal turbine power, considering that there is possibility of converting all the kinetic energy of the wind is given by:

$$P_{wi} = \frac{1}{2}\rho\Lambda v^3 \quad (19)$$

Being, P_{wi} the theoretical power; ρ the air density; Λ the area covering the turbine; And v the wind speed. A diagram of the turbine connections, with generator with coiled rotor and the controller connected to the converters, is shown in figure (4).

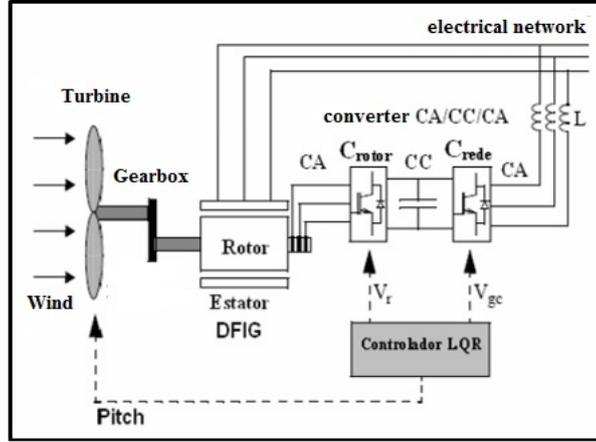


Figure 4. DFIG model connected to LQR controller

Equations (20) to (23) represent the linearized model of the wind plant:

$$\Delta \dot{x} = A\Delta x + B\Delta u \quad (20)$$

$$\Delta y = C\Delta x \quad (21)$$

At where Δx e Δu are vectors of state and input, respectively, being:

$$\Delta x = [\Delta i_{dr} \ \Delta i_{qr} \ \Delta \omega_r \ \Delta i_d \ \Delta i_q \ \Delta V_{dc}]^T \quad (22)$$

$$\Delta u = [\Delta v_{dr} \ \Delta v_{qr} \ \Delta T_e \ \Delta v_d \ \Delta v_q]^T \quad (23)$$

The state matrix A is formed by block matrices such as: $A_r^{3 \times 3}$ and $A_s^{3 \times 3}$ which represent, respectively, the rotor and stator part. The matrix of control B , is constituted of matrices of block $B_r^{3 \times 2}$ and $B_s^{3 \times 2}$. The state matrices take the forms:

$$A_r = \begin{bmatrix} -\frac{R_r}{L_{rr}\sigma} & S_0\omega_0 & 0 \\ -S_0\omega_0 & -\frac{R_r}{L_{rr}\sigma} & 0 \\ 0 & 0 & \mu \end{bmatrix} \quad (24)$$

$$B_r = \begin{bmatrix} -\frac{1}{L_{rr}\sigma} & 0 & 0 \\ 0 & -\frac{1}{L_{rr}\sigma} & 0 \\ 0 & 0 & -\frac{1}{2H} \end{bmatrix} \quad (25)$$

$$A_s = \begin{bmatrix} -\frac{R}{L} & \omega_0 & 0 \\ -W_0 & -\frac{R}{L} & 0 \\ -\frac{v_{d0}}{CV_{cc0}} & -\frac{v_{q0}}{CV_{cc0}} & -\frac{v_{d0}i_{d0} + v_{q0}i_{q0}}{CV_{cc0}^2} \end{bmatrix} \quad (26)$$

$$B_s = \begin{bmatrix} \frac{1}{L} & \omega_0 \\ 0 & \frac{1}{L} \\ -\frac{i_{d0}}{CV_{cc0}} & -\frac{i_{q0}}{CV_{cc0}} \end{bmatrix} \quad (27)$$

Where the coefficient μ belonging to the matrix A_r is represented by the expression:

$$\mu = -\frac{\frac{1}{2}\rho A v^3 C_p(\lambda, \beta)}{2H\omega_r^2} + \frac{\frac{1}{2}\rho A v^3}{2H\omega_r} \cdot \frac{dC_p(\lambda, \beta)}{d\lambda} \cdot \frac{R_p}{v} \quad (28)$$

According (Muyeen *et al.*, 2008), the term $C_p(\lambda, \beta)$ represents the power coefficient of fixed speed turbines that depends on the pitch angle β of the shovel where λ is the specific speed.

7. COMPUTATIONAL RESULTS

Based on the work of (Pinto, 2007), they follow the complete model and parameters of the wind turbine for application to DFIG and to the inverter that will be substituted in equations (23) and (24). After this application, the matrices of the state equation for the aforementioned system are obtained, investigated in relation to an operating point. The matrices are:

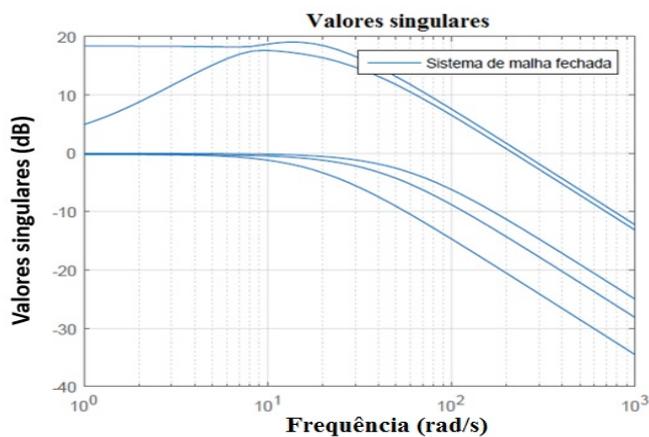
$$A_r = \begin{bmatrix} -39.4 & 0.39 & 0 & 0 & 0 & 0 \\ -0.27 & -55.7 & 0 & 0 & 0 & 0 \\ 0 & 0 & -21.9 & 0 & 0 & 0 \\ 0 & 0 & 0 & -24.5 & 0.63 & 19.8 \\ 0 & 0 & 0 & -1.27 & -18.0 & 0.72 \\ 0 & 0 & 0 & -69.1 & -0.08 & -1.78 \end{bmatrix} \quad (29)$$

The control matrix is given by:

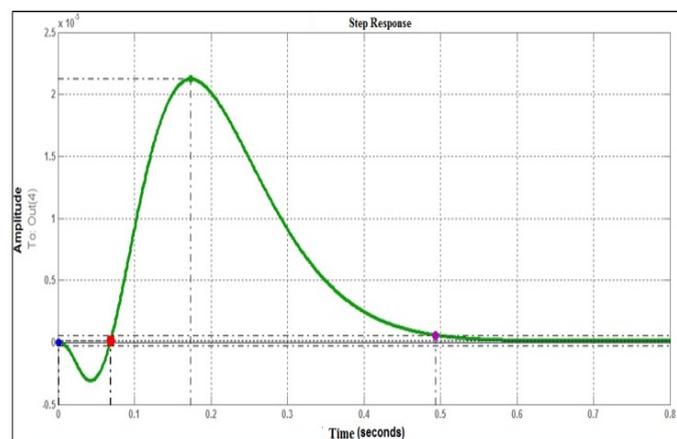
$$B = \begin{bmatrix} -12.46 & 0 & 0 & 0 & 0 \\ 0 & -12.46 & 0 & 0 & 0 \\ 0 & 0 & -21.9 & 0 & 0 \\ 0 & 0 & 0 & 5.71 & 0 \\ 0 & 0 & 0 & 0 & 5.71 \\ 0 & 0 & 0 & -0.71 & 0 \end{bmatrix} \quad (30)$$

$$C = \begin{bmatrix} 3.14 & 0 & 0 & 0 & 0 & 0 \\ 0 & 4.51 & 0 & 0 & 0 & 0 \\ 0 & 0 & -10.02 & 0 & 0 & 0 \\ 0 & 0 & 0 & -42.66 & 0.63 & 19.8 \\ 0 & 0 & 0 & 0 & 3.29 & 0 \\ 0 & 0 & 0 & 0 & 0 & -3.06 \end{bmatrix} \quad (31)$$

Figure (5a) shows the frequency response of the singular values to wind plant. It is observed in this figure the largest and smallest singular values in dB as a function of the rad/sec frequency of the base system and without controller and with the LQR. Before performing the tests, it is determined the eigenvalues that verify that the system is stable, $[-39.42; -55.74; -13.37 \pm j2.60; -17.68; -21.95]$. The step response analysis, as shown in Figure (5b), is performed with the purpose of evaluating more accurately the behavior of the DFIG plant dynamic system,



(a) Singular Values of the 6th Order Wind Plant.



(b) Step response of the 6th Order System

Figure 5. Singular Values and step response of the 6th Order Wind Plant.

7.1 Selection of Q and R matrices

Figure (6) shows the result of normalized final sensitivity for the selection of a group of twenty generations for each population of individuals:

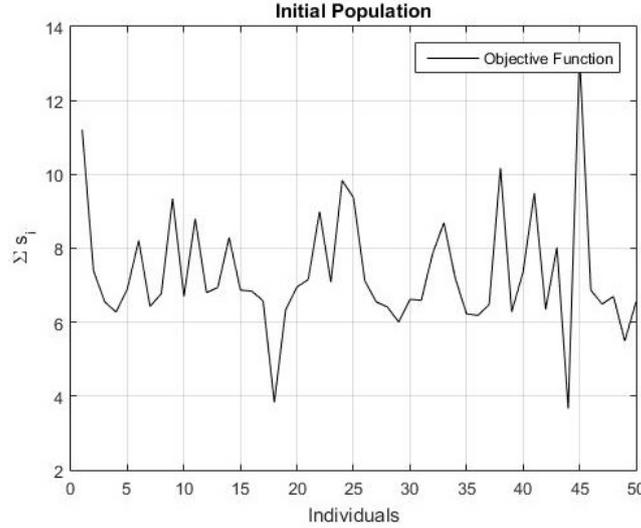


Figure 6. Total sensitivity for the final population.

Below, we show the Q and R weighting matrices:

$$Q_{lqr} = \begin{bmatrix} 2.59 & 0.34 & 1.13 & 0.55 & 0.35 & 1.83 \\ 0.34 & 4.39 & 1.42 & 0.29 & 1.59 & 4.57 \\ 1.13 & 1.42 & 3.49 & 5.00 & 0.39 & 0.36 \\ 0.55 & 0.29 & 5.00 & 11.15 & 0.33 & 0.36 \\ 0.35 & 1.59 & 0.39 & 0.33 & 12.34 & 0.29 \\ 1.83 & 4.57 & 0.36 & 0.36 & 0.29 & 14.93 \end{bmatrix} \quad (32)$$

$$R_{lqr} = \begin{bmatrix} 6.53 & 0.13 & 0.11 & 0.16 & 0.12 \\ 0.13 & 6.93 & 0.15 & 0.0832 & 0.13 \\ 0.11 & 0.15 & 7.45 & 0.12 & 0.15 \\ 0.16 & 0.08 & 0.12 & 9.68 & 0.13 \\ 0.12 & 0.13 & 0.15 & 0.13 & 10.56 \end{bmatrix} \quad (33)$$

According to (Goldberg, 1989), the GA uses strategies that are expressed by fitness functions, crossover and mutation whose goal is to reach a certain maximum number of generations or a certain fitness value. To obtain these results with GA, used the fitness function that scores each individual from the satisfaction of the project specifications. According to (Coley, 1999), the use of the fitness function using the elitist method is a great alternative of implementation, because, it keeps the best individuals in the course of generations.

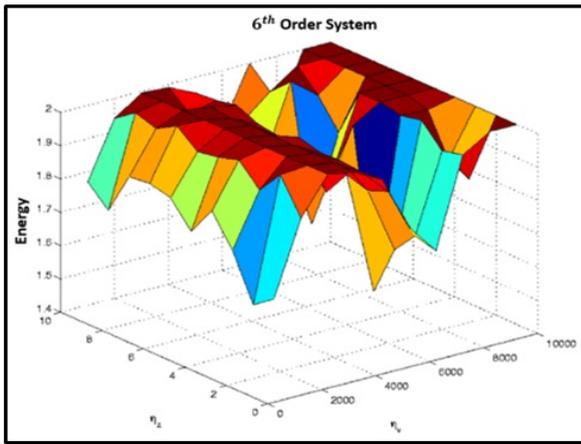
7.1.1 Neural Solution – (RAE)

Figure (7a) illustrates the energy of the RAE solution and the Cholesky factor. It has as goal to present the intervals for the weights of the network where the energy is minimum. Figure (7b) shows the infinity norm of the network layers. The elements of the U and Y layers are null when $t \rightarrow \infty$. Consequently, a solution of RAE reached convergence in 200 iterations.

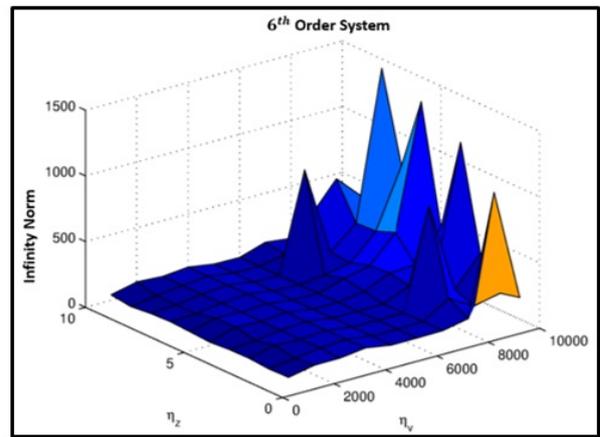
As an example of a stable solution, however, which does not present RAE behavior, it is shown in figure (8a) for the parameters $\eta_v = 1000$ and $\eta_z = 8$. As can be seen in figure (8b), the infinite norms of the U and Y layers are accompanied by values that have an unstable norm of infinity. Figure (9a) shows the results of the infinity norm of the layers $L, U, Y(t)$ and $P(t)$, and figure (9b) shows the results of the norm of infinity associated with the respective solutions.

The K_{RNA} gain matrix obtained from the LQR using the RNAR were:

$$K_{RNA} = \begin{bmatrix} -0.062 & -0.005 & -0.034 & -0.011 & -0.013 & -0.076 \\ -0.05 & -0.070 & -0.032 & 0.004 & -0.040 & -0.134 \\ -0.004 & -0.004 & -0.022 & -0.029 & -0.005 & -0.020 \\ -0.000 & -0.007 & 0.047 & 0.180 & -0.002 & -0.244 \\ 0.004 & 0.012 & 0.006 & -0.001 & 0.180 & 0.012 \end{bmatrix} \quad (34)$$

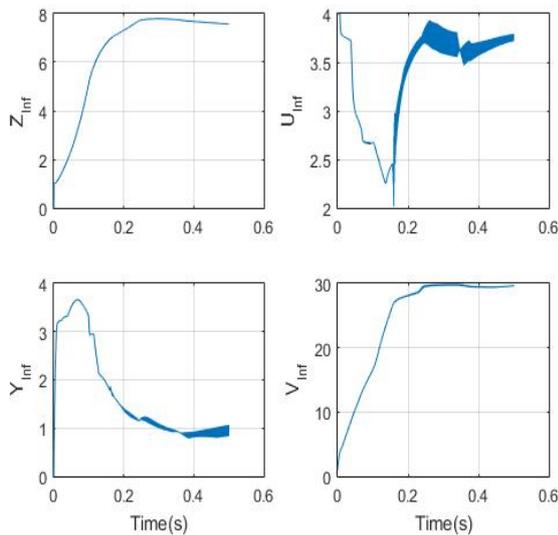


(a) Energy of the RAE solution and the Cholesky factor.

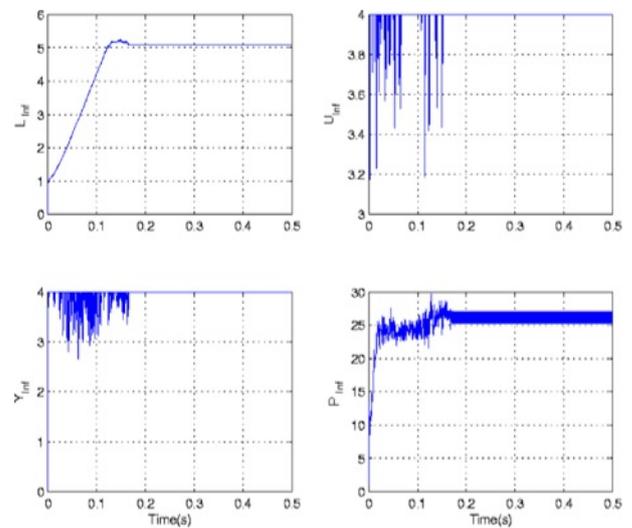


(b) Infinite norm of EAR.

Figure 7. Energy and norm of the RAE solution.



(a) Energy of the RAE solution and the Cholesky factor.



(b) Infinite norm of EAR.

Figure 8. Energy and norm of the RAE solution.

7.1.2 Solution via Schur method – (RAE)

In the structure of equation (4), it has been the matrices Q and R that are constant and are contained in the RAE, which in turn, is solved by the Schur method. The optimal gain matrix K_{Schur} , obtained by solving the RAE by the Schur method is given by:

$$K_{Schur} = \begin{bmatrix} -0.062 & -0.005 & -0.034 & -0.011 & -0.013 & -0.076 \\ -0.05 & -0.070 & -0.032 & 0.004 & -0.040 & -0.134 \\ -0.004 & -0.004 & -0.022 & -0.029 & -0.005 & -0.020 \\ -0.000 & -0.007 & 0.047 & 0.180 & -0.002 & -0.244 \\ 0.004 & 0.012 & 0.006 & -0.001 & 0.180 & 0.012 \end{bmatrix} \quad (35)$$

The eigenvalues obtained from the dynamic system with LQR: $[-56.62, -40.20, -22.00, -18.67, -13.98 \pm j1.86]$. Again, the SVD is used for analysis in the frequency domain of the system with LQR, where the elements of the matrix S of singular values obtained are: $[56.66, 40.22, 33.52, 21.99, 19.12, 5.79]$.

Analyzing the two results by the Schur method and by the RNA, see that both are well approximated, and the artificial neural network are more stable, since this difference becomes imperceptible, because it is in the decimal precision of the solutions after eighth decimal place. It is observed that the eigenvalues of the system and the diagonals of the Q and R matrices directly influence the performance and convergence of the RNA, so that the values of the Q and R matrices can be reduced or increased uniformly, thus improving the RAE solution.

According to the analysis of the parameters η_v and η_z , the convergence leads to two aspects of the RAE solution, the first regarding the existence of a single solution and the second focuses the speed of the neural network to reach the value of the solution at steady state. This analysis has as main objective to verify two types of results the first one is related to the best fit of the constants η_v and η_z and the second begins the choice of the best initial condition, and another point to consider is the number of iterations used to compute the solution Riccati Algebraic Equation.

8. CONCLUSION

After analyzing the parameters and obtaining the Q and R matrices presented by the proposed methodology of genetic-neural fusion, it can be observed that the hybrid model presented satisfactory results on the surfaces, metrics of the norm of infinity and decomposition in singular values, according to the analysis of the performance of the wind power generation system, with respect to the optimal control problem. As future work, about the proposed recurrent neural network, one can analyze the algorithmic complexity of this RNAR, as well as, to study the complexity of the neural network defining the problem size, aiming to study the computing time needed for the resolution of the Riccati Algebraic Equation based on the worst-case study. Another alternative as a proposal to be implemented and compared with the neural-genetic model in future works is the use of SVM, the Support Vector Machine, in order to find the best hyperplane between the analyzed data, seeking to maximize the distance between the points next, and with that, find the optimal gain of the plant.

9. REFERENCES

- Abreu, I.S. *et al.*, 2008. “Controle inteligente lqr neuro-genético para alocação de autoestrutura em sistemas dinâmicos multivariáveis”.
- Athans, M. and Falb, P.L., 1966. *Optimal control; an introduction to the theory and its application*. McGraw-Hill Book Company.
- Coley, D.A., 1999. *An introduction to genetic algorithms for scientists and engineers*. World Scientific Publishing Co Inc.
- Goldberg, D.E., 1989. “Genetic algorithms in search, optimization, and machine learning, 1989”. *Reading: Addison-Wesley*.
- Hansen, A.D., Jauch, C., Sørensen, P.E., Iov, F. and Blaabjerg, F., 2004. “Dynamic wind turbine models in power system simulation tool digsilent”. Technical report.
- Laub, A., 1979. “A schur method for solving algebraic riccati equations”. *IEEE Transactions on automatic control*, Vol. 24, No. 6, pp. 913–921.
- Muyeen, S., Tamura, J. and Murata, T., 2008. *Stability augmentation of a grid-connected wind farm*. Springer Science & Business Media.
- Pinto, V.P., 2007. “Modelagem e simulação de uma planta eólica controlada por um regulador linear quadrático conectada ao sistema de distribuição de energia elétrica”. *PhD tese, Universidade Federal do Ceará*.
- Wang, J. and Wu, G., 1998. “A multilayer recurrent neural network for solving continuous-time algebraic riccati equations”. *Neural Networks*, Vol. 11, No. 5, pp. 939–950.

10. RESPONSIBILITY NOTICE

Rildenir Ribeiro Silva, Ivanildo Silva Abreu, Henrique Mariano Amaral, Cristovam Dervalmar Filho e Paulo Renato Sousa.