Interior Permanent Magnet Synchronous Motor (IPMSM) Adaptive Genetic Parameter Estimation

Javad Rezaie, Mehdi Gholami, Reza Firouzi, Tohid Alizadeh, Karim Salashoor

Abstract - **Interior permanent magnet synchronous motors (IPMSMs) are receiving increased attention for high performance drive applications because of their high power density, high efficiency and flux weakening capability. However, their high efficiency characteristic is influenced by applied control strategies. Thus much effort has been directed towards the efficiency optimization of the IPMSM by minimizing motor copper loss .This loss is influenced by machine parameters. Thus online estimation of these parameters is essential. On the other hand Genetic Algorithm (GA) is a tool for optimization and can be used for solving some problems that can be formulate in those forms that this algorithm can handle it. In this paper we formulate nonlinear state equation of this motor in such form that we can use GA for estimating the unknown parameters. Simulation results show that the estimated parameters converge to correct values after several iterations.**

Index Terms— **Genetic algorithm; parameter estimation; nonlinear system; IPMSM; copper loss; efficiency.**

I. INTRODUCTION

If all model parameters of a linear/nonlinear system are known we can use KF/EKF for state estimation. In some nonlinear cases EKF diverges. In order to solving this problem we must use the other tools for estimation.

One of these tools is Genetic Algorithm (GA) that uses the principles of evolution, natural selection, and genetics to offer a method for parallel search of complex spaces. This paper describes a GA that can perform on-line adaptive parameter estimation for nonlinear systems (IPMSM in this paper).

Manuscript received July 22, 2007. This work was supported in part by the International Association of Engineers (IAENG), engineering letters and CSA (Cambridge Scientific Abstracts).

J. Rezaie, Control and Intelligent Processing Center of Excellence, University of Tehran, Iran (e-mail: j.rezaie@ece.ut.ac.ir)

M. Gholami, Automation and Instrumentation Department, Petroleum University of Technology, Iran (phone: 912-644-6597; e-mail: meti.gholami@gmail.com).

N. Firouzi, Automation and Instrumentation Department, Petroleum University of Technology, Iran (e-mail: na.firouzi@yahoo.com).

T. Alizadeh, Automation and Instrumentation Department, Petroleum Uuniversity of Technology, Iran (e-mail: alizadeh_tohid @ yahoo.com).

K. Salashoor, Automation and Instrumentation Department, Petroleum University of Ttechnology, Iran (e-mail: selahshour @ yahoo.com).

It is a stochastic process which attempts to find an optimal solution for a problem by using techniques that are based on Mendel's genetic inheritance concept and Darwin's theory of evolution and survival of the fittest.

This paper introduces a modified approach to adaptive parameter and state estimation where a genetic algorithm is used to estimate the plant parameters and hence to obtain an estimate of the state.

Parameter and state estimation problem formulate in such a form which we can use GA for solving the problem. Finally simulate the problem and results will show. The model that we used in this paper is the state space model of Interior Permanent Magnet Synchronous motor (IPMSM) which is nonlinear.

The remainder of this paper is organized as follows. A base 10 genetic algorithm is presented in Section 2. IPMSM model is discussed in Section 3. In Section 4 formulate nonlinear parameter and state estimation problem formulate in such a form which we can use GA for solving the problem. Finally simulate the problem and results will show.

II. A BASE 10 GA

In order for the GA to find the optimal solution to a particular problem, the parameters that comprise a solution must be encoded into a form upon which the GA can operate. To borrow a term from biology and genetics, any set of parameters which may be a solution to the given problem is called a chromosome, and the individual parameters in that possible solution are called traits. Since the GA will most likely be implemented on a digital computer, each trait must be encoded with a finite number of digits (called genes). The more genes in a given trait (or in a chromosome), the longer the GA will take for encoding and decoding purposes and in other operations, so a reasonable length should be chosen. The entire set of chromosomes (that is, the entire set of candidate solutions to the given problem) upon which the GA will operate is called a population [1].

To evolve the best solution candidate (or chromosome), the GA employs the genetic operators of selection, crossover, and mutation for manipulating the chromosomes in a population. A brief description of these operators follows. The GA uses these operators to combine the chromosomes of the population in different arrangements, seeking a chromosome that maximizes some user-defined objective function (called the fitness function). This combination of the chromosomes results in a new population (that is, the next generation). The GA operates repetitively, with the idea that, on average, the members of the population defining the current generation should be as good (or better) at maximizing the fitness function than those of the previous generation. The most fit member of the current generation (that is, the member with the highest fitness function result, or "fitness value") at the time the GA terminates is often taken to be the solution of the GA optimization problem [1].

The first genetic operator used by the GA for creating a new generation is selection. To create two new chromosomes (or children) two chromosomes must be selected from the current generation as parents. As is seen in nature, those members of the population most likely to have a chance at reproducing are the members that are the most fit. The technique used in [2] for selection uses a "roulette wheel" approach [1].

Consider a roulette wheel that is partitioned according to the fitness of each chromosome. The fitter chromosomes occupy a greater portion of the wheel and are more likely to be selected for reproduction. In [3], a selection method is chosen so that a given segment of the population corresponding to the most fit members (that is, the D most fit members) are automatically selected for reproducing. Therefore, the least fit members have no chance of contributing any genetic material to the next generation. Of the D most fit members of the current population, parents are randomly chosen, with equal probability. The latter method of selection is used in this study [1].

Once two parents have been selected, the crossover operation is used. Crossover mimics natural genetics (that is, "inheritance") in that it causes the exchange of genetic material between the parent chromosomes, resulting in two new chromosomes. Given the two parent chromosomes, crossover occurs with a user defined probability p_c. According to [1], if crossover occurs, a randomly chosen "cross site" is determined. All genes from the cross site to the end of the chromosome are switched between the parent chromosomes, and the children are created. Another approach to crossover, one that is used in [3] and in this study, is that crossover occurs exactly once (that is, $p_c = 1$) for every trait, with the cross site within that trait chosen randomly. That is, all genes between the cross site and the end of the trait are exchanged between the parent chromosomes. Crossover helps to seek for other solutions near to solutions that appear to be good [1].

After the children have been created, each child is subjected to the mutation operator. Mutation occurs on a gene-by-gene basis, each gene mutating with probability pm. If mutation does occur, the gene that is to mutate will be replaced by a randomly chosen allele (in this case, a randomly chosen value between 0 and 9). The mutation operator helps the GA avoid a local solution to the optimization problem. If all of the members of a population should happen to converge to some local optimum, the mutation operator allows the possibility that a chromosome could be pulled away from that local optimum, improving the chances of finding the global optimum. However, since a high mutation rate results in a random walk through the GA search space, pm should be chosen to be somewhat small [4]. We have found, however, that in some instances in real-time systems, we need a slightly higher mutation rate. This is the case since the fitness function depends on the dynamically changing state of a system, so the locality of an optimum is time-dependent and we must ensure that the GA is readily capable of exploring new opportunities for maximizing the time-varying fitness functions [1].

If a chromosome is generated by crossover and mutation, it is possible that one or more of its traits will lie outside the allowable range(s). If this occurs, each trait that is out of range should be replaced with a randomly selected trait that does fall within the allowable range [1].

In addition to selection, crossover, and mutation, a fourth operator can be used by the GA. This operator, known as elitism, causes the single most fit chromosome of a population to survive, undisturbed, in the next generation. The motivation behind elitism is that after some sufficiently small amount of time, a candidate solution may be found to be close to the optimal solution. To allow manipulation of this candidate solution would risk unsatisfactory performance by the GA. Therefore, with elitism, the fitness of a population (seen as the fitness of the best member of the population) should be a non decreasing function from one generation to the next. If elitism is selected, the most fit member of the current generation is automatically chosen to be a member of the next generation. The remaining members are generated by selection, crossover, and mutation [5]. Notice, also, that this allows us to raise the mutation probability since we know that we have a good solution available. In [3] as well as in this study, elitism can involve more than just one member. That is, a certain number rd (possibly more than one) of the most fit members will survive in the next generation without manipulation by crossover or mutation. If the most fit member would point to a local optimum in the GA search space, but a slightly less fit member points to the global optimum, they might both survive in the next generation with this new form of elitism [1].

To initialize the GA, a chromosome length must be chosen, along with the length and position of each trait on the chromosome. The allowable range for each trait must also be specified. The population size (denoted N) must be specified, along with the method of generating the first population. The individual members may be randomly generated, or they may be initialized to some set of ``best guesses". In this study, a randomly generated initial population is always used. In addition, pc and pm must be specified. After this initialization, the GA can operate freely to solve its optimization problem [1].

III. IPMSM MODEL

Under assumption of linearity of the magnetic circuit a set of widely used model for IPMSM based on a synchronous d-q reference frame, including copper is presented in figure 1 and figure 2 [6,7]:

The model equations are given as follows:

$$\frac{di_d}{dt} = -\frac{R_s}{L_d} i_d + \frac{L_q}{L_d} \omega_e i_q + \frac{1}{L_d} v_d \qquad (1)$$

$$\frac{di_q}{dt} = -\frac{R_s}{L_q} i_q - \frac{L_d}{L_q} \omega_e i_d - \frac{1}{L_q} \lambda_m \omega_e + \frac{1}{L_q} v_q \qquad (2)$$

$$T_e = \frac{3P}{2}\left[\lambda_m i_q + \left(L_d - L_q\right) i_d i_q\right] \qquad (3)$$

$$J \frac{d\omega_m}{dt} = T_e - T_L - B\omega_m \qquad (4)$$

$$\omega_e = P\omega_m \qquad (5)$$

$$P_c = \frac{3}{2} R_s \left(i_d^{\,2} + \frac{4T_e^{\,2}}{9P^2\left[\lambda_m + \left(L_d - L_q\right) i_d\right]^2} \right) \qquad (6)$$

Where:

$i_q - i_d$ d- and q-axis components of armature current,

$v_q - v_d$ d- and q-axis components of terminal voltage,

ω_e Electrical angular speed,

R_s Armature winding resistance,

λ_m Magnetic flux linkage

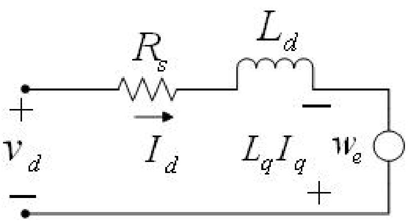

Figure 1: d-axis equivalent circuit

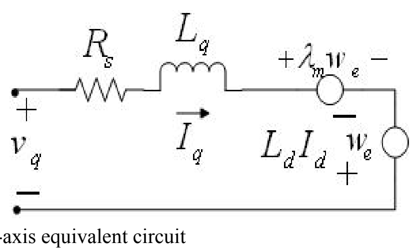

Figure 2: q-axis equivalent circuit

$\lambda_q - \lambda_d$ d-axis and q-axis components of armature self-inductance,

T_L Electro magnetic torque,

P Number of pole pairs,

P_c Copper loss.

All states of this machine are measurable and can be used for parameter estimation.

IV. PROBLEM FORMULATION

The above equation can be written in the following form:

$$\frac{d(X(t))}{dt} = F(X(t), u(t), t),$$
$$Y(t) = G(X(t), u(t), t). \qquad (7)$$

Discrete version of this equation with sampling time T_s is:

$$X(k+1) = H(X(k), u(k), k),$$
$$Y(k) = Z(X(k), u(k), k). \qquad (8)$$

In IPM synchronous machine all state (I_d, I_q and ω_m) are accessible and we can write the estimator model in the form of system model but with unknown parameters.

The cost function that must be minimized in the GA is the sum of square estimated values of state subtracted from real values of state. If the parameter chose such that this cost function minimized then the estimated value of system parameters is good other than we must run GA such that this cost function minimized or near minimization region.

From the above we define the following cost function:

$$J(R_s, L_d, L_q, \lambda_m) = \frac{1}{2}\sum_{k=t-n}^{t} \{(I_d(k) - \hat{I}_d(k))^2 +$$
$$(I_q(k) - \hat{I}_q(k))^2 + (\omega_m(k) - \hat{\omega}_m(k))^2\}. \qquad (9)$$

where:

$\hat{I}_d, \hat{I}_q$ and $\hat{\omega}_m$ are estimated values of I_d, I_q and ω_m, k is current time and n is length of data used in cost function.

V. MODIFIED GA FOR ADAPTIVE PARAMETER ESTIMATION

First we convert discrete, continues system equations then using above mentioned cost function in GA. In this example, GA has (i)Population size is 50, (ii)Generations per each iteration is 10, (iii)Mutation function is uniform with rate equal to 0.2, (iv)Crossover function is intermediate with rate equal to 0.5, (v)Selection function is roulette while .

Because sometimes the algorithm converge to some values (local optimum) that is not correct we modified this algorithm with checked this that if the current time is grater than some value(such as 30) and cost value ($J(.)$) is grater than some value (such as 0.5 ,this means that the converged parameters is not equal or near equal of the real values) and current values of estimated parameter subtracted from past estimated parameters is smaller than some value (such as 0.01, this means the parameters was converged to some value) then we increase the mutation rate to some value grater than current value to exit this point else algorithm runs with past mutation rate.

From the estimation theory we know that if the input signal was not Persistent Exciting (PE) of sufficient order the estimation was not converge to true values so we chose input signals (V_d and V_q) such that this condition satisfy.

VI. SIMULATION RESULTS

The results show that this algorithm estimates the parameters of nonlinear model with appropriate precision and time.

Simulation results show that if the range of parameters where used in GA is small then convergence is faster and estimation is more accurate than using wider range for parameters.

VII. CONCLUSION

In this paper first we present the state equation and model of IPMSM .We see that this equation is nonlinear in nature.

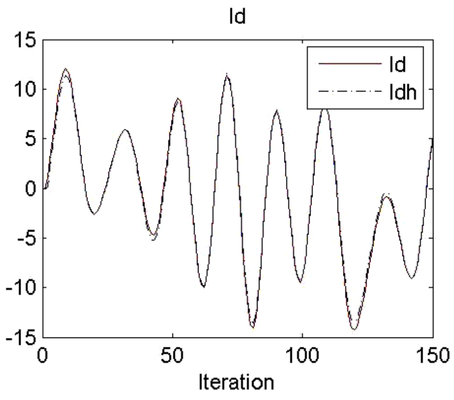

Figure 3: d-axis actual and estimated current.

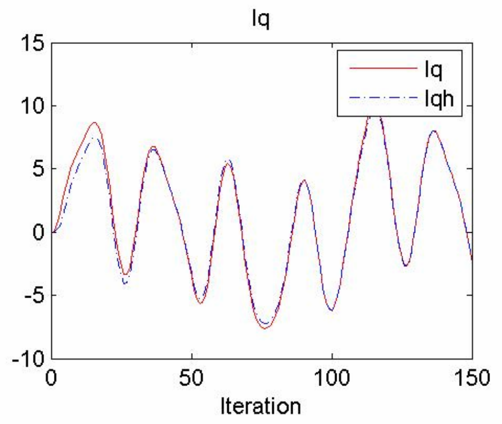

Figure 4: q-axis actual and estimated current.

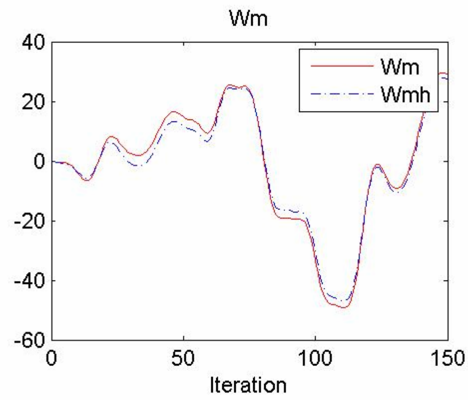

Figure 5: actual and estimated speed.

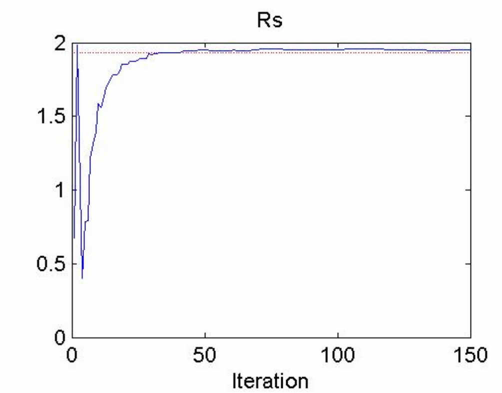

Figure 6: actual and estimated R_s .

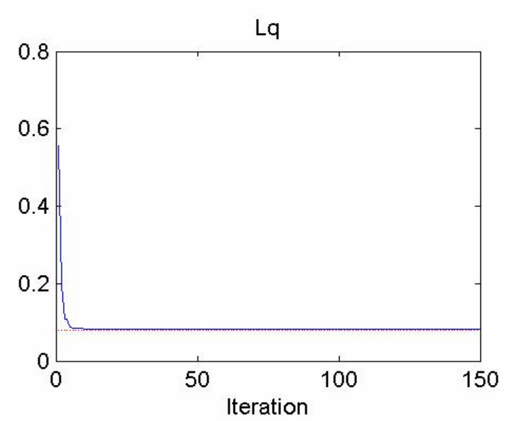

Figure 7: actual and estimated L_q .

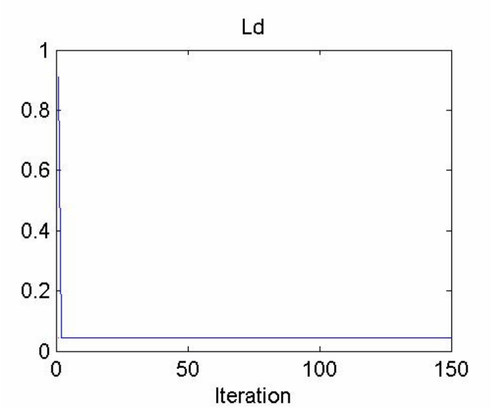

Figure 8: actual and estimated L_d .

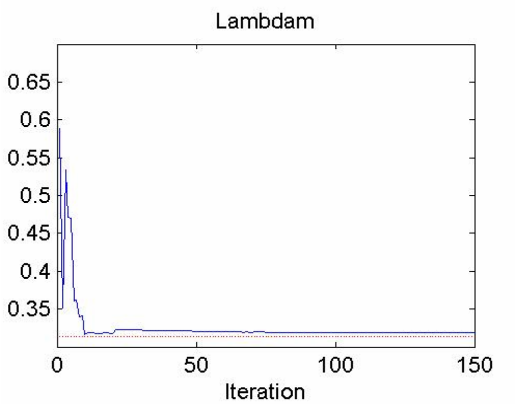

Figure 9: actual and estimated λ_m .

Then describe a base 10 GA and formulate the parameter estimation in such away that can be solve with GA. Because all state of this machine is measurable we use this measurement and estimated values of this states then difference of real values from estimated values and use this value in cost function we want to minimize, then at each iteration GA minimize this cost function and estimate parameters at each iteration. After this step we introduce a modification in online parameter estimation using GA in order to increase precision of the estimation and avoidance wrong convergence. Finally we simulate that algorithm using MATLAB

For a typical IPMSM those parameters of this motor will present at follow. Simulation results shows that by implementing this algorithm we can estimate parameters of this motor adaptively and states converge to true values after several iterations. This estimator is an alternative for parameter (or state) estimation for linear and nonlinear systems similar to KF and EKF. In some nonlinear case that EKF diverge we can used this algorithm for parameter estimation.

APPENDIX

Machine Specifications

Rated speed , rpm	1800
Rated torque, Nm	3.96
P, No. of pole pairs	2
R_s ,Ω	1.93
L_d , L_q , mH	42.44,79.57
λ_m ,Wb	0.314
J, rotor inertia constant, Nm/rad/sec	0.003

REFERENCES

[1] J. R Gremling and K. M. Passino,. “Genetic adaptive state estimation”, Engineering Applications of Artificial Intelligence, 13, 611-623, 2000.
[2] D. E. Goldberg, “Genetic Algorithms in Search, Optimization, and Machine Learning”, Addison-Wesley, 1989.
[3] L. Yao and W. A Sethares, “Nonlinear parameter estimation via the genetic algorithm”, IEEE Transactions on Signal Processing, 42, 927-935, 1994.
[4] S. H. Thilagar and R. G. Sridhara, “Parameter estimation of three-winding transformers using genetic algorithm, Engineering Applications of Artificial Intelligence”, 15, 429–437, 2002.
[5] C. S. Ong, J. J. Huang and G. H. Tzeng,”Model identification of ARIMA family using genetic algorithms”, Applied Mathematics and Computation, 164, 885–912, 2005.
[6] S. Morimoto, Y. Takeda and T. Hirasa,” Loss minimization control of permanent magnet synchronous motors”, IEEE Trans. Ind Electron., 41, 511-517,1994.
[7] S. Vaez and V. I. John, “ Minimum Loss Operation of PM Motor Drives”, In Proc. Of Canadian Conf. on Elec. & computer Eng., Montreal, pp. 284-287, 1995.

.

ISBN:978-988-98671-6-4