Optimized Sizing of High Speed PM Generator for Renewable Energy Applications

Adel El Shahat and Ali Keyhani

Member, IEEE, Fellow, IEEE
Department of Electrical and Computer Engineering
The Ohio State University,
Columbus, Ohio 43210, USA
{ahmed.210 & keyhani.1}@osu.edu

Hamed M. El Shewy
Electrical Power and Machines Department
Faculty of Engineering, Zagazig University
Zagazig, Egypt
prof.hamed_elshewy@yahoo.com

Abstract--This paper proposes optimized sizing of High Speed Permanent Magnet Synchronous Generators (HSPMSGs) for renewable energy applications to be driven by micro-turbines. These designs have more significant improvement in weights and volumes than usual or classical. This paper presents at the beginning a step by step sizing procedure of (HSPMSGs) to be used in optimization functions formulations. Then, Unconstrained Optimization for Total Losses Minimization is presented. Simple constrained optimized Total Mass with bounded constraints is introduced. Finally, Minimum Mass Genetic Sizing, Constrained by Minimum Losses is proposed. The optimization variables are the same in every optimization process. The optimized and genetic results are well depicted by some variables 3D figures for initial and detailed sizing. The simulation of generators sizing is performed using MATLAB Optimization Toolbox, and Genetic Algorithm.

Index Terms--Genetic, High-speed, Optimization, generators, Size.

I. INTRODUCTION

High speed PM generators provide a substantial reduction in size and weight, also higher in power density; because as a machine’s speed increases, its size decreases for a given output power. The size, weight, and cost are from the major factors for successful design. The increase of the rotating speed of electrical machines is a way to improve their power/mass ratio and thus the dimensions and weight are reduced and the total efficiency is increased [1–5]. For high-speed applications, the rotor aspect ratio, defined as length-to-diameter, is a critical parameter. Stator core losses may be minimized by using laminated steel in stator construction and by not generating frequencies that are too high. The main applications of PMSG are for power generation as part of renewable energy resources and main generators for aircraft, etc. [2–9]. The sizing of HSPMSG design must address system topology for good power/volume, low cost, and superior efficiency [1–14]. Optimum design of high speed PM alternator is proposed to be used in distributed power generation system with experimental tests are conducted to verify the FEM predictions [8], [9]. The high speed permanent magnet (PM) machine has been widely used in distributed power generation. The high speed generator distributed generation system in comparison with PM Doubly Fed Reluctance Generator for the same application has better electromagnetic property and the PM doubly fed reluctance machine has better mechanical behaviour [31]. Seven configurations of both radial-flux and axial-flux machines give comparisons among PM wind generators of different topologies from 1 kW to 200 kW. Optimization of design data are done and verified by finite element analysis and commercial generator test results [32]. Aspects of PM motor technology and design of brushless PM machines are introduced in [12] and [13]; to be used in this paper.

II. BASIC SELECTIONS

NdFeB (Neodymium Iron-Boron) is preferred because it is cheaper and more readily available. So, NdFeB magnets are selected for use in PMG, with some conservatively assumed values [13-15]. TM19, 29 gauge electrical silicon steel is selected for the PMG stator and rotor because it is economical [13], [16], its thin laminations minimize power losses due to the circulating eddy current, and because it has a saturation flux density of about 1.8 T [13], [14].

III. MACHINE DESIGN PARAMETERS

A. Stator Mechanical Design

Slotted stators consist of openings around the stator for the armature windings, as shown in Fig. 1 (a) are selected.

Fig. 1 Slotted Stator Design (a) and Stator Slot Geometry (b).

Slots are trapezoidal, but assumed here to be rectangular, as shown in Fig. 1 (b). They contain form-wound windings so that the depression width is the same as the top slot width. Slotting is used because of its advantages [14]. The initial design of the generator assumes a three-phase machine with 36 slots [13].
B. Rotor Mechanical Design

For high-speed applications, the rotor aspect ratio, defined as length-to-diameter (L/D), is a critical parameter. PM machines offer flexibility in selecting pole sizes, which allows for smaller diameters. A normal L/D ratio for a wound rotor machine is 0.5 – 1.0, and 1 – 3 for a PM machine [17]. The rotor radius and rotational speed also determine the tip speed of the machine, which is the surface velocity of the rotor.

\[ v_{tip} = r \omega_m \]  

where \( \omega_m \) = angular speed (rad/sec); \( r \) = rotor radius (m)

The upper limit on tip speed is between 100-250 m/s, depending on the design of the machine. In this design, a range of tip speed is taken to be (50:250).

C. Number of Poles and Magnets Pole Design

An even number of poles is always used, (here pole pairs number = 3). Assuming a constant mechanical rotation speed, electrical frequency is given as.

\[ N = \frac{2 \times P \times f}{120} \]  

where \( N \) = speed (rpm); \( P \) = pole pairs; \( f \) = electrical frequency (Hz)

Magnet poles skew factor is selected to reduce cogging torque and smooth out variations in air gap reluctance, flux, and voltage waveforms.

\[ k_m = \sin(\frac{n \theta_s}{2}) \]  

where \( \theta_s \) = Skew angle, rad; \( n \) = Harmonic number

D. Magnetic Dimensions

The magnetic dimensions that affect a PM machine are air gap and magnet height. Air gap flux density (\( B_g \)) represents in Eq. 4. The radial air gap is made as small as possible to maximize the air gap flux density, minimize the flux leakage, and to produce a lower reluctance value.

\[ B_g = \frac{h_m}{h_m + g} B_r \]  

where \( h_m \) = Magnet height (mm); \( g \) = Air gap (mm); \( B_r \) = Magnet Remnant Flux Density (T)

Magnets losses are reduced, using smaller magnets. For uniform magnetic fields, the magnet height is usually larger than the air gap, by a factor 5 – 10.

E. Slots Per Pole, Per Phase

Three-phase machines are typically used in this study as the standard choice for most motors and generators. Another important design parameter is the number of slots per pole, per phase (m), as in Eq. 5.

\[ m = \frac{N}{2 \times P \times q} \]  

F. Stator Windings

The pitch of a winding (\( \alpha \)) refers to the angular displacement between the sides of a coil. The breadth of a stator winding results from the coils occupying a distribution of slots within a phase belt. In smaller machines, coils are composed of round insulated wires that are placed in the stator slot, along with insulation material. A slot fill factor (\( \lambda_s \)) is used to determine how much of the slot’s cross-sectional area is occupied by winding material, as in Eq. 6.

\[ \lambda_s = \frac{\text{Winding Area}}{\text{Total Slot Area}} \]  

Typically, machines contain two coils sides per slot, making the winding a double-layer design [13]. Overall, slot fill factors vary in value from 0.3 – 0.7, depending on the number and size of the conductors in the slots, as well as the amount of labour utilized. A slot fill factor of 0.5 is assumed.

G. Machine Calculated Parameters

Each phase of the machine is modelled, as shown in Fig. 2.

Fig. 2 A Per Phase Electrical Model.

\[ R_a: \text{Armature resistance; } L_s: \text{Synchronous inductance; } E_a: \text{Back e.m.f voltage and } V_a: \text{Terminal voltage.} \]

H. Winding Resistances

Copper phase windings resistance is calculated in Eq. 7.

\[ R_s = \frac{l}{\sigma \times A_w} \]  

where \( l \) = length of conductor, \( \sigma \) = winding conductivity, \( A_w \) = winding cross – sectional area

\[ A_w = \frac{A_s \times \lambda}{2 \times N_c} \]  

where \( A_s \) = slot Area, \( N_c \) = turns per coil

But the above stator resistance equation may be used as in low frequencies applications, so it has to be developed. Since the machine rotates at high speed, and high frequency and so the skin depth may be affected. In conductors that carry high frequency currents, skin effect can become an issue and affect the operation of the machine. Skin effect is caused by eddy currents in the windings themselves due to the changing magnetic field. These eddy currents force the current flowing in the conductor to crowd to the outer edges of the conductor. This in turn causes the current to flow through a smaller cross – sectional area and increase the resistance of the conductor. It is well known that, when conductive material is exposed to an ac magnetic field, eddy currents are induced in the material in accordance with Lenz’s law. The power loss resulting from eddy currents which can be induced in the slot conductors appears as an increased resistance in the winding. To understand this phenomenon, let us consider a rectangular conductor as shown in fig. 3. The average eddy current loss in the conductor due to a sinusoidal magnetic field in the y direction is given approximately by Hanselman [12].

Fig. 3 Rectangular conductor geometry
Using Eq. (13), the total slot resistance can be written as

\[ R_s = \rho \frac{\pi n_s^2 L}{k_{cp} \omega_s d_s} \]  

Eq. (9) can be written as

\[ R_s = \frac{L \omega_s h_s^2}{6 \sigma d_s^4} \]  

Using this expression it is possible to compute the ac resistance of the slot conductors. If the slot conductors are distributed uniformly in the slot, and substituting the field intensity into Eq. (11) and summing over all \( n_s \) conductors gives a total slot eddy current loss of

\[ p_e = \frac{1}{12} \sigma L \omega_s h_s^2 \omega_s^2 \sigma_0 H^2 \]  

\( \sigma_0 \): the turn field intensity value; \( \sigma_0 \): perm. of free space. Since skin depth is defined as

\[ \delta = \sqrt{\frac{2}{\omega \sigma_0 \rho}} \]  

Eq. (9) can be written as

\[ R_s = \frac{L \omega_s h_s^2}{6 \sigma d_s^4} \]  

\( I \): rms conductor current; \( \omega_s \): Slot width (m); \( d_s \): Slot depth (m)

The slot resistance of a single slot containing \( n_s \) conductors connected in series is

\[ R_s = \frac{1}{2} \frac{n_s^2 L}{k_{cp} \omega_s d_s} \]  

This result shows that the resistance increases not only as a function of the ratio of the conductor height to the skin depth but also as a function of the slot depth to the skin depth. Thus, to minimize ac losses, it is desirable to minimize the slot depth as well as the conductor dimension. For a fixed slot cross-sectional area, this implies that a wide but shallow slot is best.

### I. Winding and Magnet Factors

Winding are short-pitched and have breadth associated with them. To account for these effects, a winding factor \( k_w \) is utilized, as in Eq. (16).

\[ k_w = k_m \ast k_{bw} \]  

Short-pitching is an important means for eliminating harmonics and improving the power quality of the machine. The pitch factor is shown in Eq. (17).

\[ k_m = \sin \left( \frac{n_w \frac{\pi}{2}}{2} \right) \ast \sin \left( \frac{n_w \frac{\pi}{2}}{2} \right) \]  

The breadth factor explains the effect of the windings occupying a distribution or range of slots within a phase belt. The breadth factor is derived in Eq. (18).

\[ k_{bw} = \frac{\sin \left( \frac{n_w \frac{\pi}{2}}{2} \right)}{m \ast \sin \left( \frac{n_w \frac{\pi}{2}}{2} \right)} \]  

where \( m \): slots per pole per phase; \( \gamma \): coil electrical angle

The magnetic flux factor equation [12], for the slotted stator and surface magnet configuration is shown in Eq. 19.

\[ k_B = \frac{R_s \omega^2}{R_s^2 + (R_s + \omega^2 \delta_s)^2} \ast (R_s + \omega^2 \delta_s)^2 \ast (R_s + \omega^2 \delta_s)^2 \]  

\( R_s \): outer magnetic boundary, \( R_s \): outer boundary of magnet; \( R_i \): inner magnetic boundary, \( R_i \): inner boundary of magnet

\[ \lambda = 2 \ast R_s \ast L_{sl} \ast N_s \ast k_w \ast k_{bw} \ast B_s \]  

\[ B_s = \frac{k_B C_s}{1 + k_s \ast u_{rec} / PC} \]  

where \( u_{rec} \): recoil permeability; \( B_s \): remnant flux density

\[ PC = \frac{k_B}{C_s} \]  

where \( k_C \): permeance coeff.; \( C_s \): flux concentration factor (Am/Ag)

\[ N_s = 2 \ast P \ast N_a \]  

where \( N_a \): Turns per coil; \( N_s \): Number of armature turns (each slot has 2 half coils)

\[ r_s = w_s + w_s \]  

\[ \delta \]: effective air gap; \( w_s \): average slot width; \( w_s \): tooth width

Here, a leakage factor \( K_l \sim 0.95 \) and a reluctance factor \( K_r \sim 1.05 \) are both used for surface magnets. The presence of the slots in the stator also affects the air gap flux density because of the difference in permeance caused by the slots. Carter’s coefficient \( (k_s) \) is used to account for this effect [12].

\[ k_s = [1 - \frac{1}{w_s} \ast (5 \ast \frac{R_s}{w_s} + 1)]^{-1} \]  

The terminal voltage \( (V_a) \) is calculated from the internal voltage \( (E_a) \), and the synchronous reactance voltage drop. The armature resistance is usually ignored because it is much smaller than synchronous reactance. The voltage is found as a result in output power \( (P_{w}) \), e.m.f., and reactance from the resulting quadratic equation.

\[ V_a = \sqrt{\frac{BB + \sqrt{BB^2 - 4CC}}{2}} \]  

\[ BB = 2 \ast X_P \ast P_{w} - E_a \]  

\[ CC = \frac{2}{9} \ast X_s \ast P_{w} \]  

\[ K \]: Machine Inductances

In a slotted PM machine, there are three distinct components of inductance: the largest, air gap inductance slot leakage inductance, and the smallest, end-turn inductance. The total inductance for the phase is the sum of the three inductances, ignoring other small factors.

\[ L_s = L_{sl} + L_{slot} + L_s; \quad X_s = \omega_a \ast L \]
The air gap inductance is given by Eq. 30.

\[ L_{ag} = \frac{\lambda}{I} = \frac{q}{2} \frac{4}{n \pi} \frac{R_{st} * N}{a_s * p^2 * (g + h_m)} \] (30)

The slot leakage inductance is presented in Eq. 31. Assume the slot is rectangular with slot depressions, as in Fig. 1, and assume (m) slots per pole per phase, with a standard double layer winding.

\[ L_{slot} = L_{ax} - L_{om} \quad (3 \text{ phase}) \] (31)

\[ L_{om} = 2 * P * L_{nt} * \text{Perm} * N_{st} * N_r^2 \] (32)

\[ L_{ax} = 2 * P * L_{nt} * \text{Perm} * [4 * N_r^2 (m - N_r) + 2 * N_{st} * N_r^2] \] (33)

A slot permeance per unit length is shown in Eq. 34.

\[ \text{Perm} = \frac{1}{3} * \frac{h_{d}}{w_{d}} + \frac{k_{c} * h_{d}}{w_{d}} \] (34)

The end turn inductance is introduced in Eq. 35, assuming the end turns are semi-circular, with a radius equal to one-half the mean coil pitch.

\[ L_{e} = \frac{u_{om} * N_r * N_r^2}{2} * \tau \text{ sinh} \left( \frac{\tau}{\sqrt{2} * A} \right) \] (35)

L. Basic Losses

Losses in a machine consist of core losses, conductor losses, friction and windage losses, and rotor losses. Rotor losses will be discussed later. Stator core losses, per weight, can be greater than normal in machines because of higher frequencies. These losses are minimized by using laminated steel in stator construction and by not generating frequencies that are too high. Core losses consist of hysteresis and eddy current losses. The best way to approximate core losses is to use empirical loss data. An exponential curve fitting is applied to the empirical data for M-19, 29 gauge material, in order to obtain an equation for estimating core losses, as in Eq. 36, with constant values in [19].

\[ P_{c} = P_{0} * \left( \frac{B}{B_{0}} \right)^{a_1} * \left( \frac{f}{f_{0}} \right)^{a_2} \] (36)

where \( P_{0} \): Base power; \( B_{0} \): Base flux density; \( a_1 \): Flux density exponent; \( f_{0} \): Base frequency; \( a_2 \): Frequency exponent.

The above commonly used equation considering hysteresis and eddy-current loss is not completely satisfactory, because the measured iron loss is much higher than theoretically calculated. This is so because it assumes a homogenous magnetization of the laminations, which is not a valid representation of what happens during the magnetization process. The loss caused by the movements of the magnetic domain walls is higher than the loss calculated with the commonly used equation. The difference between measured and calculated loss is called the excess loss or the anomalous loss. Sometimes, this anomalous or excess loss is considered as a third contribution to the iron loss. Great efforts have been made to calculate this excess loss, because of the complexity of the domain patterns. For reasons mentioned before, it is useful to represent the core loss by core loss resistance, which is placed in equivalent circuit. The core loss resistance is connected across the voltage \( V_{ag} \). Therefore, the power dissipated in this resistance is [22-30].
A. Trust-Region Methods

Many of the methods used in Optimization Toolbox solvers are based on trust regions, a simple yet powerful concept in optimization. To understand the trust-region approach to optimization, consider the unconstrained minimization problem, minimize f(x). Suppose we are at a point x in n-space and we want to improve. The basic idea is to approximate f with a simpler function q, which reasonably reflects the behavior of function f in a neighborhood N around the point x. This neighborhood is the trust region. A trial step s is computed by minimizing over N. This is the trust - region sub - problem, min s \{ q(s), s ∈ N \} [33]. The current point is updated to be x + s if f(x + s) < f(x); otherwise, the current point remains unchanged and N, the region of trust, is shrunk and the trial step computation is repeated. The key questions in defining a specific trust – region approach to minimizing f(x) are how to choose and compute the approximation q (defined at the current point x), how to choose and modify the trust region N, and how accurately to solve the trust – region sub – problem. In the standard trust-region method, the quadratic approximation q is defined by the first two terms of the Taylor approximation to F at x; the neighborhood N is usually spherical or ellipsoidal in shape. Mathematically the trust – region sub – problem is typically stated

\[
\min \{ \frac{1}{2} s^T H s + s^T g \text{ such that } \| Hs \| \leq \Delta \}
\]  

(43)

where g is the gradient of f at the current point x, H is the Hessian matrix (the symmetric matrix of second derivatives), D is a diagonal scaling matrix, \(\Delta\) is a positive scalar, and \(\| \) is the 2-norm.

Good algorithms exist for solving previous equation; such algorithms typically involve the computation of a full Eigen system and a Newton process applied to the secular Equation [33]

\[
\Delta = \frac{1}{\lambda} \| s \| = 0
\]

(44)

Such algorithms provide an accurate solution to the equation. However, they require time proportional to several factorizations of H.

B. Fminsearch Algorithm

Fminsearch uses the Nelder-Mead simplex algorithm as described in [34]. This algorithm uses a simplex of n + 1 points for n-dimensional vectors x. The algorithm first makes a simplex around the initial guess x0 by adding 5% of each component x0(i) to x0, and using these n vectors as elements of the simplex in addition to x0. (It uses 0.00025 as component i if x0(i) = 0.) Then the algorithm modifies the simplex repeatedly according to the following procedure.

1. Let x(i) denote the list of points in the current simplex, i = 1,...,n+1.
2. Order the points in the simplex from lowest function value f(x(1)) to highest f(x(n+1)). At each step in the iteration, the current worst point x(n+1) is discarded, and another point is accepted into the simplex (or, in the case of step 7 below, all n points with values above f(x(1)) are changed).
3. Generate the reflected point \(r = 2m - x(n+1)\), where \(m = \Sigma x(i)/n, i = 1...n\), and calculate f(r).
4. If f(x(1)) ≤ f(r) < f(x(n)), accept r and terminate this iteration. Reflect
5. If f(r) < f(x(1)) (i.e., r is better than x(n+1)), calculate c = m + (r – m)/2 and calculate f(c).
6. If f(c) < f(r), accept c and terminate the iteration.
7. Calculate the n points v(i) = x(1) + (x(i) – x(1))/2 and calculate f(v(i)), i = 2,...,n+1. The simplex at the next iteration is x(1), v(2),...,v(n+1). Shrink

C. Total Losses Minimization Sizing

Using the previous technique the total losses could be minimized. The first step is to choose the optimizing variables x1, x2, x3, these variables here are L/D ratio, Rotor radius, and Stack length respectively. These variables will be stayed also with constrained ones, also with Genetic Algorithm.

Second step is to formulate the total losses function to be minimized as fitness or objective function in the form of MATLAB m – file. Then, third step, using optimization tool box GUI with a proper choice for initial variables values also the accuracy with suitable number of iteration. Some sizing relations examples for this optimization problem are presented in the following figures. These examples at 500 kW, and 400 kW with tip speeds of 250, 200 and 150 m/s. Using x1, x2, and x3; all detailed parameters could be obtained. The following figures show the examples.

\[
P_{\text{Total Losses}} = P_{\text{Core}} + P_{\text{Conductor}} + P_{\text{Wind}}
\]

(45)
When trying to use unconstrained optimization with this case, from the beginning the maximum value for L/D ratio exceeds its upper limit. So, some simple constraints are used in this case as bounds from 0.0 to 3.0. The function used here is presented as follow.

A. fmincon Active Set Algorithm

In constrained optimization, the general aim is to transform the problem into an easier sub – problem that can then be solved and used as the basis of an iterative process. A characteristic of a large class of early methods is the translation of the constrained problem to a basic unconstrained problem by using a penalty function for constraints that are near or beyond the constraint boundary. In this way the constrained problem is solved using a sequence of parameterized unconstrained optimizations, which in the limit (of the sequence) converge to the constrained problem. These methods are now considered relatively inefficient and have been replaced by methods that have focused on the solution of the Karush-Kuhn-Tucker (KKT) equations. The KKT equations are necessary conditions for optimality for a constrained optimization problem. If the problem is a so-called convex programming problem, that is, f(x) and Gi(x), i = 1,...,m, are convex functions, then the KKT equations are both necessary and sufficient for a global solution point. The Kuhn-Tucker equations can be stated as

\[
\nabla f(x^*) + \sum_{i=1}^{m} \lambda_i \nabla G_i(x^*) = 0
\]

\[
\lambda_i g_i(x^*) = 0, i = 1,\ldots, m
\]

\[
\lambda_i \geq 0, i = m_v + 1,\ldots, m
\]

(46)

The first equation describes a canceling of the gradients between the objective function and the active constraints at the solution point. For the gradients to be canceled, Lagrange multipliers (\(\lambda_i, i = 1,...,m\)) are necessary to balance the deviations in magnitude of the objective function and constraint gradients. Because only active constraints are included in this canceling operation, constraints that are not active must not be included in this operation and so are given Lagrange multipliers equal to 0. This is stated implicitly in the last two Kuhn-Tucker equations. The solution of the KKT equations forms the basis to many nonlinear programming algorithms. These algorithms attempt to compute the Lagrange multipliers directly. Constrained quasi-Newton methods guarantee super linear convergence by accumulating second-order information regarding the KKT equations using a quasi-Newton updating procedure. These methods are commonly referred to as Sequential Quadratic Programming (SQP) methods, since a QP sub problem is solved at each major iteration (also known as Iterative Quadratic Programming, Recursive Quadratic Programming, and Constrained Variable Metric methods). The 'active-set' algorithm is not a large-scale algorithm.

VI. SIMPLE CONSTRAINED OPTIMIZATION
B. Large-Scale vs. Medium-Scale Algorithms

An optimization algorithm is large scale when it uses linear algebra that does not need to store, nor operates on, full matrices. This may be done internally by storing sparse matrices, and by using sparse linear algebra for computations whenever possible. Furthermore, the internal algorithms either preserve sparsity, such as a sparse Cholesky decomposition, or do not generate matrices, such as a conjugate gradient method. Large-scale algorithms are accessed by setting the Large Scale option to on, or setting the Algorithm option appropriately (this is solver-dependent). In contrast, medium-scale methods internally create full matrices and use dense linear algebra. If a problem is sufficiently large, full matrices take up a significant amount of memory, and the dense linear algebra may require a long time to execute. Medium-scale algorithms are accessed by setting the Large Scale option to off, or setting the Algorithm option appropriately (this is solver-dependent). Don’t let the name “large-scale” mislead you; you can use a large-scale algorithm on a small problem. Furthermore, you do not need to specify any sparse matrices to use a large-scale algorithm. Choose a medium-scale algorithm to access extra functionality, such as additional constraint types, or possibly for better performance.

C. Total Mass Minimization Sizing

The previous technique with the described function above could be used for minimizing the total mass. The optimizing variables x1, x2, x3, these variables are the same. Then formulate the total mass function to be minimized as fitness function in the form of MATLAB m – file. Later, using optimization tool box GUI with a proper choice for initial variables values also the accuracy with suitable number of iteration. Also, it should be notified that, Large scale (trust region) method does not currently solve this type of problem, but using medium scale (line search) instead. Some illustrating examples are proposed to show the validity of this minimization.

\[ M_{Total} = M_{Core} + M_{Magnet} + M_{Shaft} + M_{Conductor} + M_{Service} \]  \hspace{1cm} (47)

VII. GENETIC ALGORITHM HSPMSG SIZING
A. Description of the Nonlinear Constraint Solver

The genetic algorithm uses the Augmented Lagrangian Genetic Algorithm (ALGA) to solve nonlinear constraint problems. The optimization problem solved by the ALGA algorithm is \( \min f(x) \), such that
\[
\begin{align*}
\text{ci} (x) & \leq 0, \quad i = 1 \ldots m \\
\text{ceq} (x) & = 0, \quad i = m+1 \ldots mt \\
A \cdot x & \leq b \\
Aeq \cdot x & = beq \\
lb \leq x & \leq ub,
\end{align*}
\]
where \( c(x) \) represents the nonlinear inequality constraints, \( \text{ceq}(x) \) represents the equality constraints, \( m \) is the number of nonlinear inequality constraints, and \( mt \) is the total number of nonlinear constraints.

The Augmented Lagrangian Genetic Algorithm (ALGA) attempts to solve a nonlinear optimization problem with nonlinear, linear, and bounds. In this approach, bounds and linear constraints are handled separately from nonlinear constraints. A sub problem is formulated by combining the fitness function and nonlinear constraint function using the Lagrangian and the penalty parameters. A sequence of such optimization problems are approximately minimized using the genetic algorithm such that the linear constraints and bounds are satisfied. A sub - problem formulation is defined as
\[
\Theta(x, \lambda, s, \rho) = f(x) - \sum_{i=1}^{m} \lambda_i s_i \log(c_i(x)) + \sum_{j=1}^{mt} \sum_{i=1}^{n} \lambda_i c_i(x) + \frac{\rho}{2} \sum_{i=1}^{n} c_i(x)^2
\]
where the components \( \lambda_i \) of the vector (\( \lambda \)) are nonnegative and are known as Lagrange multiplier estimates. The elements \( s_i \) of the vector (\( s \)) are non - negative shifts, and \( \rho \) is the positive penalty parameter. The algorithm begins by using an initial value for the penalty parameter (Initial Penalty).

The genetic algorithm minimizes a sequence of the sub – problem, which is an approximation of the original problem. When the sub - problem is minimized to a required accuracy and satisfies feasibility conditions, the Lagrangian estimates are updated. Otherwise, the penalty parameter is increased by a penalty factor (Penalty Factor). This results in a new sub – problem formulation and minimization problem. These steps are repeated until the stopping criteria are met [35], [36].

B. Min. Mass Genetic Sizing, Constrained by Min. Loss

One case from the previous examples is proposed, using the total mass function (m.file) as a fitness function in GA with nonlinear constraint as minimum losses. The same optimizing variables with a simple constraints that are \([1 \ 0 \ 0]\) as lower limit, and \([3 \ 1 \ 1]\) as upper limit, also using non – linear constraint function with the aid of minimum losses values obtained from the part of minimum losses sizing one. The example here, selected randomly from the previous parts at tip speed = 200 m/s, and output power = 400 kW. The non – linear constraint function, which governs the optimization process, is implemented using the minimum losses value at the same values of tip speed, and output power. This design proposes machine sizing for minimum mass, under the condition of minimum total losses equal or under the minimum total losses value. This means make a benefit from limiting the machines losses at its absolute minimum value or lower, as a nonlinear constraint with total mass function as objective function to be minimized. This also will reduce the machine cost.

The following introduce the detailed result of this prescribed example in TABLE I:

### TABLE I

<table>
<thead>
<tr>
<th>Output Power</th>
<th>Tip Speed</th>
<th>X1 = 1.5317</th>
<th>X2 = 0.0388</th>
<th>X3 = 0.1189</th>
<th>Frequency</th>
</tr>
</thead>
<tbody>
<tr>
<td>400000</td>
<td>200</td>
<td></td>
<td></td>
<td></td>
<td>2.4602e+003</td>
</tr>
</tbody>
</table>

Min. Mass 16.9503 Total Loss 4.5192e+003 Rotor Diameter 0.0776 Speed(rpm) 4.9205e+004

Tooth Width 0.0053 Slot Top Width 0.0053 Slot Bottom Width 0.0045 Slator Core Back Iron Depth 0.0091

End Turn Travel (one end) 0.0289 End Length (half Coil) 0.0908 End Length (Axial direction) 0.0289 Magnetic Gap Factor 1.1404

Average Slot Width 0.0049 Width of Slot and Tooth 0.0102 Eff. Air Gap 0.0024 Air Magnetic Flux Density 0.8566

Magnetic Flux 0.0641 Internal Voltage 700.6569 Air Gap Inductance 1.0987e-005 Slot Leakage Inductance 3.3814e-006

Slot Area 4.9057e-006 Total Inductance 1.4327e-005 Total Reactance 0.2215 Armature Conductor Length 7.2104

Armature Conductor Area 1.2264e-005 Mass of Armature Conductor 2.3611 Overall Mach. Length 0.1767 Core Inside Radius 0.0712

Core Outside Radius 0.0803 Overall Mach. Diameter 0.1605 Back Iron Mass 3.9461 Teeth Mass 1.8500

Core Mass 5.7961 Magnet Mass 2.2489 Shaft Mass 4.3333 Service Mass 2.2109

Arm. Resistance 0.0098 Tooth Flux Density 1.7132 Back Iron Flux Density 1.2237 Core Back Iron Loss 1.0017e+003

Teeth Loss 966.9290 Total Core Loss 1.9776e+003 Term. Voltage 684.0544 Wind Loss 1.3199e+003

Current 203.8567 Conductor Loss 1.2216e+003 Efficiency 0.9888

VIII. Conclusions

Unconstrained optimization for the minimization of total loss is performed; this function of minimizing total losses is implemented and results are presented. The optimizing variables are rotor length to diameter ratio, rotor radius, and stack length, for each of the functions, in both constrained and unconstrained optimization and in the genetic algorithm. The constrained optimized total mass with some constraints is set up to keep the total mass to be minimized. It should be noted that the large-scale trust-region method does not currently solve this type of problem, but using medium scale line search has provided an acceptable performance. The results of the genetic algorithm are presented with the same optimization variables, as before, but the fitness functions, in which the constraints are varied. We have also presented minimum mass genetic sizing, constrained by minimum losses. The same optimizing variables and the same bounds are
used in the genetic example, with non-linear constraint function of minimum losses. This is done under the condition of minimum total losses, equal to, or less than, the minimum total losses value. We have observed that this will have the benefit of limiting machines losses. In our study, we have found that a noticeable improvement appears in the performance parameters.

REFERENCES


