Multiobjective Genetic Algorithm for Product Design

Jyotirmaya Nanda
Industrial and Manufacturing
Engineering Department
Penn State University
University Park, PA 16802

Abstract

In this paper multi-objective evolutionary algorithm is proposed to design a universal electric motor. Designing a product often incorporates multiple objectives. Designing a product family has an added tradeoff, between commonality and individual product performance. The presence of multiple objectives gives rise to a set of Pareto-optimal solutions for individual products as well as the product family. The multi-objective evolutionary algorithm is used to help design a universal electric motor with competing design requirements. This is a step towards designing a family of electric motors with an acceptable balance between commonality in the product family and desired performance of individual products.

1 INTRODUCTION

The consumer market today is more turbulent and varied. Robust economy but cautious consumers; increasing globalization; increasing competition, often using new technology; diversified society of better educated and informed individuals; and many other factors has created a demand for unique custom products and services. Consumers no longer prefer to get directed to purchase pre-built products; rather, they want to have products and services that meet their particular needs. According to Pine (1993), "Customers can no longer be lumped together in a huge homogeneous market, but are individuals whose individual wants and needs can be ascertained and fulfilled".

By introducing higher levels of commonality within a product family, a manufacture is able to reduce the part inventory, reduce the procurement cost, be more capable of handling volatility in consumer demand, and increase diversity in product offerings. The only snag being that increased commonality has an associated trade-off in individual product performance. The argument being, a single product designed specifically for a set of requirements is capable of performing better than a product belonging to a family of products that shares components. The components designed to be shared among different products will not perform as good as a custom component. When designing a product family the focus is more on the family as a whole rather than a particular member of the family. Consequently decision maker has to make a tradeoff between commonality and performance during product family design. The tradeoff between commonality and performance is generally captured using one or more of the many commonality indices for product family design (Jio and Tseng, 2000; Kota, et al., 2000; Martin and Ishii, 1997; Siddique, et al., 1998; Simpson, et. al., 2001b) based on the direct and indirect benefits of commonality. Numerous examples of successful product families can be found in the literature: Swiss army knives and Swatch Watches (Ulrich and Eppinger, 2000), Xerox (Paula, 1997) and Cannon (Yamanouchi, 1989) photocopiers, Dell Computers (Schonfeld, 1998), Hewlett Packards printers (Feitzinger and Lee, 1997), Kodak (Wheelwright and Clark, 1995), Volkswagen (Bremmer, 1999), and Sony Walkmans (Sanderson and Uzumeri, 1995).

Many design researchers have started to use multi-objective optimization to examine the trade-off between commonality and individual product performance. Gonzalez-Zugasti, et al. (1999), use real options concepts to help select the most appropriate product family design from a set of alternatives; they also investigate the use of multi-objective optimization to design modular product platforms (Gonzalez-Zugasti and Otto, 2000; Gonzalez-Zugasti et al., 2000). Simpson et al., (2001) proposed a formal method that facilitates the synthesis and exploration of a common Product Platform Concept that can be scaled into an appropriate family of products known as Product Platform Concept Exploration Method (PPCEM). The product platform is modeled as a compromise Decision Support Problem (DSP) to model the necessary constraints and goals for the product platform. The compromise DSP is a multi-objective mathematical construct which is a hybrid formulation based on mathematical programming and goal programming (Mistree et al., 1993).

Genetic Algorithms are well suited for solving combinatorial problems. Li and Azarm (2002) present a two stage approach that employs a multiobjective genetic algorithm for product line design selection under uncertainty and with competitive advantage. D'Souza and Simpson (2003) present a method of using non-dominated sorting genetic algorithm (NSGA) to design a family of General Aviation Aircraft while optimizing the performance of the individual products.

In this paper a Universal Electric motor is optimized using NSGA-II (Deb et al., 2000) as a step towards designing a family of Universal motors. The designing of a Universal Electric motor involves simultaneous optimization of multiple objectives that are competing in nature. The aim of using NSGA-II is to find out the Pareto-optimal or noninferior solutions of designing the motor. The problem is also highly constrained. Genetic Algorithm is used to explore the design space so that it can direct the designers towards feasible design variables that can later be used in designing the product family. The problem will later be extended to find commonality between different motors.

2 THE UNIVERSAL MOTOR PROBLEM

Universal electric motors are so named for their capability to function on both direct current (DC) and alternating current (AC). Universal motors deliver more torque for a given current than any other kind of AC capable motor (Chapman, 1991). The high performance characteristics and flexibility of universal motors have led to a wide range of applications, especially in household use where they are found in, e.g. electric drills and saws, blenders, vacuum cleaners, and sewing machines (Veinott and Martin 1986).

According to Meyer and Lehnerd (1997), in the 1970s Black & Decker developed a family of universal motors for its power tools in response to a new safety regulation: double insulation. Prior to that, they used different motors in each of their 122 basic tools with hundreds of variations, from jig saws and grinders to edgers and hedge trimmers. Through redesign and standardization of the product line, they were able to produce all of their power tools using a line of motors that varied only in the stack length and the amount of copper wrapped within the motor. As a result, all of the motors could be produced on a single machine with stack lengths varying from 0.8 in to 1.75 in, and power output ranging from 60 to 650 W. By paying attention to standardization and exploiting platform scaling around the motor stack length, material costs dropped from \$0.77 to \$0.42 per motor while labor costs fell from \$0.248 to \$0.045 per motor, yielding an annual savings of \$1.82 million per year. Tool costs decreased by as much as 62%, boosting sales, increasing production volumes, and further improving savings. Furthermore, new designs were developed using standardized components such as the redesigned motor, which allowed products to be introduced, exploited and retired with minimal expense related to product development. Our goal is to demonstrate the use of the Genetic Algorithms to design a family of universal motors in a similar manner, starting with designing a single motor.

A schematic of a universal motor is shown in Figure 1. As shown in the figure, a universal motor is composed of an armature and a field which are also referred to as the rotor and stator, respectively. The armature consists of a metal shaft and slats (armature poles) around which wire is wrapped longitudinally as many as thousands times. The field consists of a hollow metal cylinder within which the armature rotates. The field also has wire wrapped longitudinally around interior metal slats (field poles) as many as hundreds of times.

In this example problem, the design variables of interest for the universal motor are: the wire cross-sectional areas and numbers of turns in both the field and the armature; the radius, thickness, and stack length of the motor; and the current drawn by the motor.

Several textbooks are available for analyzing the performance of universal motors (Shultz 1992; Nasar and Unnewehr 1983; Veinott and Martin 1986; Chapman 1991). Such texts make use of the performance equations in terms of variables and constants (such as the magnetic field strength, the magnetic flux, and the motor constant K) which vary with respect to the physical dimensions of the motor; however, they do not detail the specific relationships between the physical dimensions of the motor and the resulting performance parameters.

A more sophisticated approach is presented by Kawanda (1965) to analyze the design factors associated with universal motor. The approach is based upon measurements of an existing universal motor, and it is neither intended, nor applicable, to an original design problem. Similarly, Dickin-Zangger (1962) presents a method for estimating the distribution of energy in a universal motor based on purposeful testing, as a basis for analytical and comparison of existing universal motor designs. Again, this approach does not provide relationships between physical motor dimensions and performance before a motor is actually built. In summary, the design literature of universal motors, where available, does not include a model (of any complexity) which relates physical parameters to resulting performance as we seek to develop.

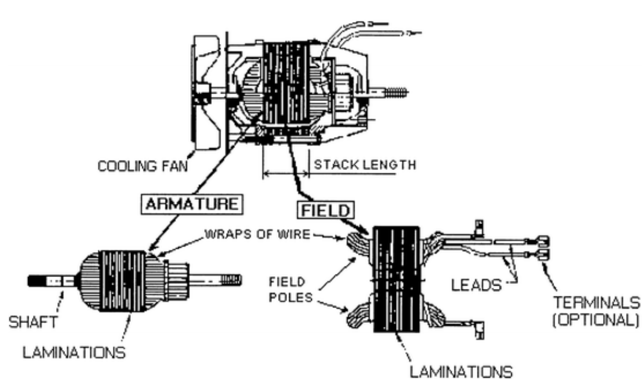
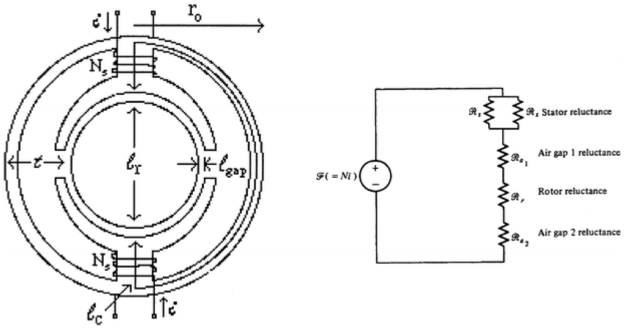


Figure 1: Universal motor schematic (GS Electric 1997)

For more detailed motor schematics, see Figure 2.



(a) Physical model (b) Magnetic circuit

Figure 2: Detailed Universal motor schematic (Simpson et al., 2001)

In design related work, Wijenayake et al. (1995) develop a model for design optimization of permanent magnet motors that is rather complex with 53 input variables and 36 output variables. It is not applicable to universal motor design because permanent magnet motors use permanent magnets instead of wire coils to create a magnetic field. Boules (1990) present a similar approach for the design optimization of permanent magnet motors. Therefore, in order to provide computer simulation of universal motors, a mathematical model needs to be developed with input of physical motor dimensions and output of motor performance measures. Such a model is derived next.

In order to minimize power losses within the core of the motor when operating on AC power, a universal motor is constructed with slightly thinner laminations in both the field and the armature and less field windings. However, the governing electromagnetic equations for the operation of a series DC motor and a universal motor running on DC current are identical (Chapman 1991). The performance at full-load torque of a universal motor running on AC current is only slightly less than the performance of the same motor running on DC current. This discrepancy in performance is due to losses caused by the inherent oscillation in alternating current; for an overview of the losses associated with AC operation (see Chapman 1991).

These extra losses incurred in AC operation of a universal motor are difficult, if not impossible, to model analytically; thus, complicated finite element analyses are becoming more popular for modeling motor behavior under AC current. Since such a detailed analysis is beyond the scope of this work, the derived model for the performance of the universal motor is for DC operation for which simple analytical expressions are known or can be derived. Moreover, several texts indicate that the performance of universal motors under AC and DC conditions is quite comparable up until full load torque (Shultz 1992; Nasar and Unnewehr 1983; Veinott and Martin 1986; Chapman 1991); Shultz states that "Universal motors ... will operate either on DC or AC up to 60 Hz. Their performance will be essentially the same when operated on DC or AC at 60 Hz." For this work, all motors are designed for operation at full-load torque. Thus, it is assumed that designing a universal motor for DC conditions yields satisfactory performance for AC conditions as well.

The formulae used for calculation of motor outputs, such as, power, torque, mass, and efficiency are illustrated in the following sections.

2.1 POWER

The basic equation for power output of a motor is the input power minus losses, where the input power is the product of the voltage (V) and current (I).

$$P = P_{in} - P_{losses} = VI - P_{losses} \quad (1)$$

For a universal motor, power is lost in (i) the copper wires as they heat-up (copper losses), (ii) at the interface between the brushes and the armature (brush losses), (iii) in the core due to hysteresis and eddy currents (core losses), (iv) in mechanical friction in the bearings supporting the rotor (mechanical losses), (v) in heating up the core and copper which adversely effects the magnetic properties of the core (vi) and the current carrying ability of the wires (thermal losses). For this analysis thermal losses, core losses, mechanical losses, and stray losses are neglected. The combined effects of all the aforementioned neglected losses will, conversely, decrease the output power and efficiency from the predicted value from the model. However, the following equations serve as a sufficiently accurate model for the DC operation of a universal motor.

$$P_{losses} = P_{copper} + P_{brush} \quad (2)$$

$$P_{copper} = I^2(R_a + R_s) \quad (3)$$

R_a and R_s are the resistances of the armature and field windings.

$$P_{brush} = 2I \quad (4)$$

2.2 TORQUE

Te torque of a DC motor is given by the product of a motor constant, K, the magnetic flux, f, and the current, I.

$$T = K\ f\ I \quad (5)$$

2.3 MASS

The mass of the motor includes mass of the stator, armature, and windings. The motor is modeled as a solid steel cylinder with length L for the armature and a hollow steel cylinder with length L, outer radius r_o, and inner radius (r_o-t) for the stator.

$$M_{stator} = \Pi\ .\ (r_0^2 - (r_0 - t)^2.L.\ r_{steel} \quad (6)$$

$$M_{armature} = \Pi.\ (r_0 - t - l_{gap})^2.L.\ r_{steel} \quad (7)$$

$$M_{windings} = \Pi.\ (r_0 - t - l_{gap})^2.L.\ r_{steel} \quad (8)$$

2.4 EFFICIENCY

The basic equation for efficiency, expressed as a decimal and not a percentage, is given by,

$$h = \frac{P}{P_{in}} \tag{9}$$

For more detailed motor schematics, operation and performance measure see Simpson et al. (1999, 2001).

3 EVOLUTIONARY ALGORITHMS AND NSGA II

Evolutionary Algorithms (EAs) are search and optimization algorithms inspired by the process of natural evolution. Current evolutionary approaches include Evolutionary Programming, Evolution Strategies, Genetic Algorithms, and Genetic Programming (Fonseca and Fleming, 1998).

The Genetic Algorithm (GA) technique was first conceived by Professor John Holland of University of Michigan, Ann Arbor in 1975. GAs are adaptive stochastic optimization algorithms involving search and optimization. Instead of working with a single solution at each iteration, a GA works with a number of solutions (collectively known as a population). GAs are based on the notion of the "survival of the fittest", and they operate by searching for and choosing optimal solutions in much the same way that natural selection occurs. GAs only use the objective function while searching for optimized result and not the derivatives, therefore it is a direct search method. GAs work with a coding of the parameter set (set of strings/individual chromosomes), and not the parameters themselves and use probabilistic transition rules (Goldberg, 1989).

Usually there are only two main components of most genetic algorithms that are problem dependent: the problem encoding and the evaluation function. When the GA is implemented it is usually done in a manner that involves the following cycle: Evaluate the fitness of all of the individuals in the population. Create a new population by reproduction. The reproduction process for a pair of chromosomes involves duplicating the two individual chromosomes (the "parents") and then choosing a place (site) on the chromosomes to crossover (or switch) information between them. This results in two new "children" chromosomes in the population, which could have higher fitness values than their "parents". Mutation can also occur when decision variable values in a chromosome are randomly changed. Then the old population is discarded and iteration is started using the new population. Every iteration of the algorithm is referred to as a generation.

The exchange of information between chromosomes during crossover allows the algorithm to converge to a global, rather than a local, optimum (Goldberg, 1989). Even though the operators are simple, GAs are highly nonlinear, massively multifaceted, stochastic, and complex.

The notion of Non-dominated Sorting Genetic Algorithms was first suggested by Goldberg (1989) and then presented by Srinivas and Deb (1995). Primarily, there are two tasks that a multi-objective GA should do well in solving multi-objective optimization problems (Deb, 1999):

1. Guide the search towards the global Pareto-optimal region, and
2. Maintain population diversity in the current non-dominated front.

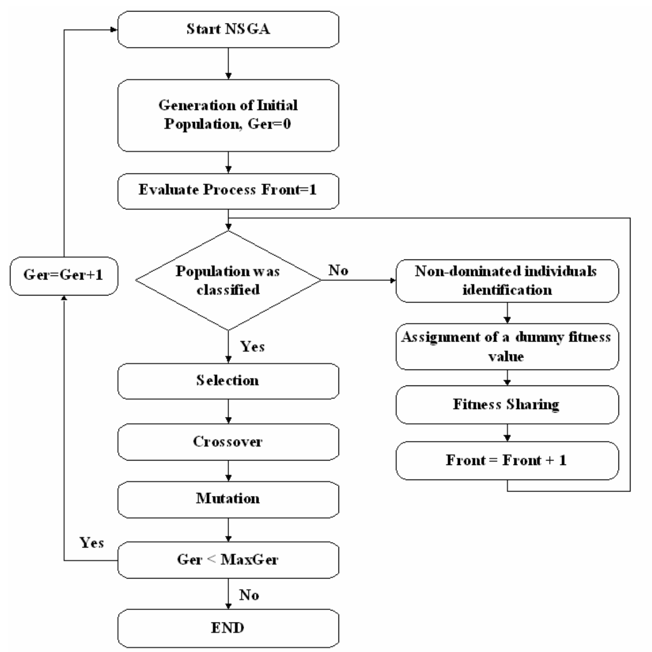


Figure 3: Working Principle of a NSGA

(Dias and Vasconcelos, 2002)

Weile et al. (1996), indicate that the NSGA finds more of the Pareto frontier and maintains diversity of the population in subsequent generations when compared with Niched Pareto GA and the Crowded Tournament Pareto GA. Figure 1 shows the working principle of NSGA .

Mathematically Multiobjective optimization problems are represented as,

Maximize

$$y = f(x) = \{ f_1(x),\ f_2(x), \ldots\ldots, f_M(x)\}$$

Subject to,

$$\begin{cases} g(x) = \{g_1(x), g_2(x), \ldots\ldots\ldots, g_J(x)\} \le 0 \\ h(x) = \{h_1(x), h_2(x), \ldots\ldots\ldots, h_K(x)\} = 0 \end{cases} \tag{10}$$

Where, $\begin{cases} x = \{x_1, x_2, \ldots . x_N\} \in X \\ y = \{y_1, y_2, \ldots . y_M\} \in Y \end{cases}$

and x is the vector of decision variables, y is the objective vector, X is the decision space, and Y is called the objective space. The solution of (1) is usually no unique, but a set of equally efficient, noninferior or nondominated solutions (Dias and Vasconcelos, 2002).

The effectiveness of the NSGA comes with an added computational expense due to its ranking and sharing functions. The NSGA arranges all solutions into fronts that are Pareto optimal with respect to the multiple objectives, and the NSGA can generate a Pareto tradeoff curve faster than it would take a simple GA using the weighted sum or constraint methods, which are more traditional multiobjective optimization techniques. In order to sort a population size N according to the level of non-domination, each solution must be compared with every other solution in the population to find if it is dominated. This requires O(mN) comparisons for each solution where m is the number of objectives. When this process is continued to find the members of the first non-dominated class for all population members, the complexity is $O(mN^2)$. At this stage, all individuals in the first non-dominated front are found. In order to find the individuals in the next front, the solutions of the first front are temporarily discounted, and the procedure is repeated. in worst case, the task of finding the second front also requires $O(mN^2)$ computations. The procedure is repeated to find the subsequent fronts. As can be seen, the worst case (when there exists only one solution in each front), the complexity of this algorithm is $O(mN^3)$. A fast non-dominated sorting approach that requires at most O(mN2) computations is described next; it is part of the NSGA-II algorithm that is employed in this work. The source code for the NSGA -II algorithm is written in C and is available from Kanpur Genetic Algorithm Lab at: <http://www.iitk.ac.in/kangal/soft.htm>.

The implementation of Non-dominated Sorting Genetic Algorithm is briefly explained in this section. First, for each solution two entities are calculated: (i) n_i, the number of solutions which dominate the solution i, and S_i, a set of solutions which the solution i dominates. All points that have $n_i = 0$ are identified and placed in a list f_1. This f_1 is called the current front. Now, for eachsolution in the current front each member j is visited in its set Si and its nj count is reduced by one. In doing so, if for any member j the count becomes zero, it is placed in a separate list j. When all members of the current front have been checked, the members in the list f_1 are declared as members of the firt front. This process is continued using the newly identified front H as the current front. This procedure is called the fast non-dominated sort method.

In the NSGA -II, a random parent population P_0 is first created. The population is then sorted based on the non-domination. Each solution is assigned a fitness equal to its non-dominated level (1 is the best level) where, minimization of fitness is assumed. Binary tournament selection, recombination, and mutation operators are used to create a child population Q_0 of size N. From the first generation onwards the procedure is different.

For generations $t = 1$, a combined population $R_t = P_t \cup Q_t$ is formed first, which is of size 2N. The population R_t is then sorted according to non-domination. The new parent population P_{t+1} is formed by adding solutions from the first front until the size exceeds N. Thereafter, the solutions of the last accepted front are sorted according to $=_n$, and the first N points are picked. This is how the population P_{t+1} of size N is constructed. This population of size N is now used for selection, crossover, and mutation to create a new population Q_{t+1} of size N. It is important to note that a binary tournament selection operator is used, but the selection criterion is now based on the niched comparison operator $=_n$.

4 PROBLEM REPRESENTATION AND PARAMETER SELECTION

There are eight design variables that are used to evaluate the motors. The following sections describe the design variables, as well as the constraints used in the design problem.

4.1 DESIGN VARIABLES

The design variables and ranges of interest for each motor are as follows.

1. Number of wire turns on the motor armature, N_c ($100 = N_c = 1500$ turns)
2. Number of wire turns on each field pole, N_s ($1 = N_s = 500$ turns)
3. Cross-sectional area of the armature wire, Awa ($0.01 = Awa = 1.0\ mm^2$)
4. Cross-sectional area of the field wire, Awf ($0.01 = Awf = 1.0\ mm^2$)
5. Radius of the motor, r_o ($0.01 = r_o = 0.10$ m)
6. Thickness of the stator, t (0.0005 = t = 0.10 m)
7. Current drawn by the motor, I (0.1 = I =6.0 Amp)
8. Stack length, L (1 = L =0.2 m)

The terminal voltage, V_t, is fixed at 115 volts to correspond to standard household voltage, and the length of the air gap, l_{gap}, is set to 0.7 mm which is considered the minimum possible air gap length. A minimum air gap length is always desired because it maximizes torque while minimizing mass.

4.2 CONSTRAINTS

The constraints for each motor are listed in Table 1. The constraint on magnetizing intensity ensures that the magnetic flux within each motor does not exceed the physical flux carrying capacity of the steel (Chapman 1991). The constraint on feasible geometry ensures that the thickness of the stator does not exceed the radius of the stator since the thickness is measured from the outside of the motor inward. The required output power is taken as 300 W. This equality constraint is handled by putting it in the objective function of NSGA II, which gave better results than formulating it as constraints. The objective for power was multiplied by a large factor to direct the search towards the targeted power.

Table 1: Design Constraints

Sl. No.	Goal	Constraint Type	Value
1	r_0/t	>	1
2	*P*	=	300 watt
3	?	=	0.7
4	*T*	=	0.05 Nm
5	*M*	=	2.0 Kg
6	*SAT*	=	5000 A turns/m

4.3 GOALS

There are three goals for the motor:

1. Maximize Efficiency (?)
2. Minimize Mass (M)
3. Maximize Torque(T)

As shown in Table 1, a lower bound for efficiency, and torque, and a upper bound for mass have been imposed for the motor.

The equations to calculate the goals are derived from Chapman (1991) and Cogdell (1996) for DC electric motors unless otherwise noted.

This genetic algorithm will later be extended to the family of motors and commonality between the products will be explored.

4.4 NSGA II PARAMETERS

Reed et al. (2000) present a 3-step methodology for the design of simple genetic algorithms that accounts for population sizing, selection pressure, and the influence of crossover and mutation on real-world computationally intensive applications.

To determine the range of potential population sizes, in the first run a small population size was used and the population size was doubled with each successive run. The percentage change in number of nondominated individuals for two successive run is calculated for each successive run. The population size increase is no longer done when the percentage change in number of non dominated individual fell below a pre specified value (see equation 11).

$$\Delta_{\text{nd}} < 100 \frac{|I_n - I_{n-1}|}{I_{n-1}} \tag{11}$$

Where, Δ nd is the pre specified percentage change. I_n and I_{n-1} are the number of non- dominated individuals in successive runs. The result of the analysis is shown in Figure 4. Majority of the applications use probability of crossover P_c between 0.6 and 0.9. To keep the disruptive effect of crossover for diversification, the crossover probability is kept at 0.7. Probability of mutation is taken as inverse of population. The number of real coded variables in the problem is eight with no binary representation. NSGA II code was modified to keep the first two variables, N_c and N_s, as integers by rounding off to the next integer value.

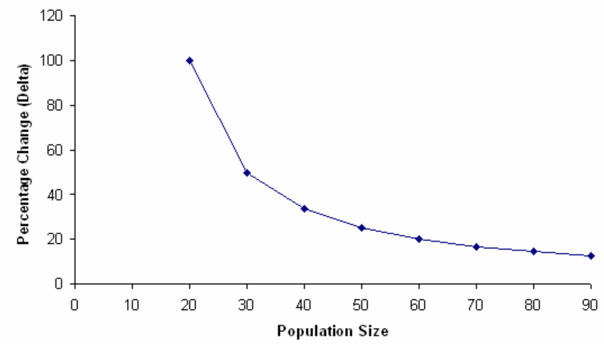


Figure 4: Change in number of Non-dominated Solutions

5 RESULTS

The results of the NSGA II run gave good insight to the interdependencies of design variables. It was observed that there is a significant increase in the value of torque with increase in the mass of the motor. The Pareto front for only torque and mass is shown in figure 5.

Figure 6 gives the 2D representation of the solution space for mass and torque.

The efficiency of the motor doesn't vary much over the solution space and it remains in the region of 0.7 to 0.96. As already mentioned in section 2, the efficiency of the motors will appear higher than the actual values due to the assumptions made in calculation of power loss.

Figure 7 shows the Pareto front for only efficiency and mass, and Figure 8 shows the whole solution space.

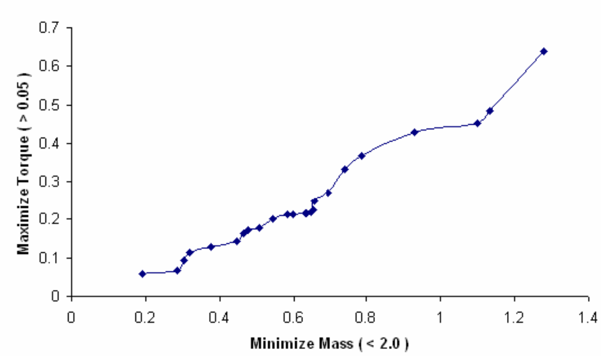


Figure 5: Pareto front for Mass vs. Torque of the Motor

It is clear from figure 5 that with increasing torque of the motor its mass increases.

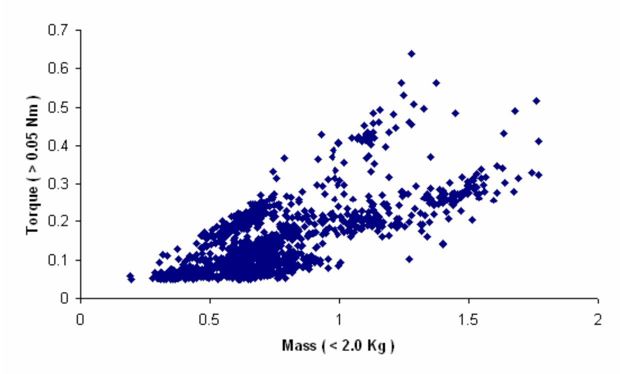


Figure 6: Solution space for Mass vs. Torque of the Motor

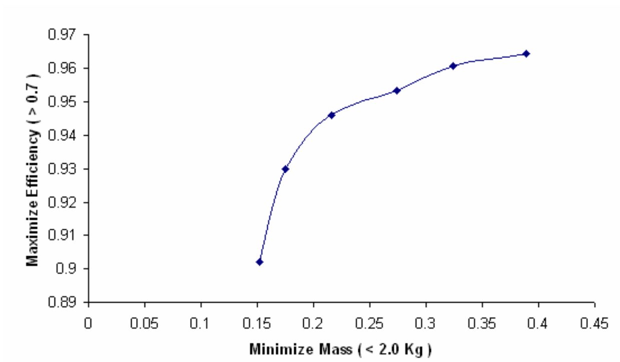


Figure 7: Pareto front for Mass vs. ? of the Motor

Also, increase in efficiency results in increase in mass. But it can be concluded from the results shown in figure 5-8 that it is easier to achieve the desired efficiency in the family of motors for a given mass than achieving the torque.

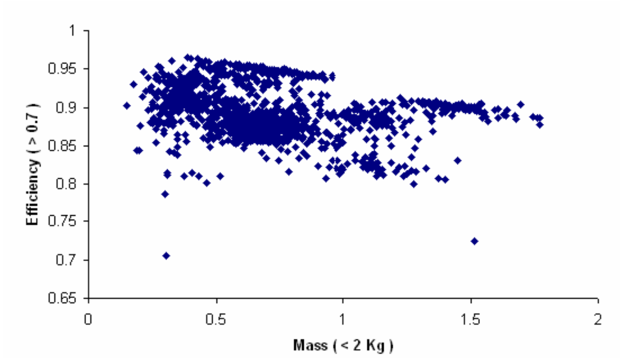


Figure 8: Solution space for Mass vs. Torque of the Motor

To find the common design variables in the search space a product family penalty function has to be introduced in the GA formulation which will try to find common design variables while maintaining the feasibility and desired output performance.

Also, offline analysis of the results will yield better analysis of the solution space and the trade off between various design variables.

6 CONCLUSIONS

As seen from the results the outputs form a Pareto frontier while designing the universal motor. The complexity of the problem will rise with the introduction of Product Family Penalty Function (Messac et al., 2002) to design the product family of universal motors.

The formulation of the problem will also change. Some of the possible methods of going forward are:

1. Individually optimize the products and do an offline analysis of the results to figure out the product family
2. Attach an extra binary header to each generation which will decide on which variables to keep common and which ones not, and then let the Genetic Algorithm find the product family
3. Introduce a penalty function for each variable that will give preference to commonality in product variables in the phenotypic space.

The decision regarding the parameters for the NSGA II for such a large problem will have to be carefully given to get any feasible results.

References

Boules, N., 1990, "Design optimization of permanent magnet DC motors," IEEE Trans Industry Application, Vol. 26(4), pp. 786-792

Bremmer, R., 1999, "Cutting-Edge Platforms," *Financial Times Automotive World*, pp. 30-38.

Chapman, S. J., 1991, Electric Machinery Fundamentals, McGraw-Hill, New York.

Cogdell, J. R., 1996, Foundations of Elcetrical Engineering, Prentice-Hall, Upper Saddle River, N 5.

Deb, K., Agrawal, S., Pratap, A., Meyarivan, T., 2000, "A Fast Elitist Non-Dominated Sorting Genetic Algorithm for Multi-Objective Optimization: NSGA-II", Proceedings of the Parallel Problem Solving from Nature VI Conference, 16-20 September, Paris, France, pp. 849--858.

Deb, K, 1999, "Multi-Objective Genetic Algorithms: Problem Difficulties and Construction of Test Problems," Evolutionary Computation Journal, Vol. 7(3), pp. 205-230.

Dias, A. H. F. and Vasconcelos, J. A. de., 2002, "Multiobjective Genetic Algorithms Applied to Solve Optimization Problems," IEEE Transactions on Magnetics, Vol. 38(2), pp. 1133-1136.

Dickin-Zangger C.W., 1962, "Performance analysis of universal motors," *Electro-Technology*, Vol. 70(6), pp. 127-130

D'Souza, B. and Simpson, T. W., 2003, "A Genetic Algorithm Based Method for Product Family Design Optimization," *Engineering Optimization*, Vol. 35(1), pp. 1-18

Feitzinger, E., Lee, H. L., 1997, "Mass Customization at Hewlett-Packard: the power of postponement," *Harvard Business Review*, Vol. 75(1), pp. 116-121.

Fonseca, C. M. , Fleming, P. J. , 1998, Multiobjective Optimization and Multiple Constraint Handling with Evolutionary Algorithms -Part I: A Unified Formulation, IEEE Transactions on Systems, Man, and Cybernetics, Part A: Systems and Humans, pp. 1-37.

Goldberg, D. E., 1989, "Genetic Algorithms in Search, Optimization and Machine Learning," Reading, MA, Addison-Wesley Publishing Company Inc.

Gonzalez-Zugasti, J. P. and Otto, K. N., 2000, "Modular Platform-Based Product Family Design," *2000 Design Engineering Technical Conferences – Design Automation Conference (Renaud, J. E., Ed.)*, Baltimore, MD, ASME, Paper No. DETC-2000/DAC-14238.

Gonzalez-Zugasti, J. P., Otto, K. N. and Baker, J. D., 1999, Spetmber 12-15, "Assessing Value for Product Family Design and Selection," *Advances in Design Automation*, Las Vegas, NV, ASME, Paper No. DETC99/DAC-8613.

GS Electric, 1997, "Why universal motors turn on the appliance industry," http://www.gselectric.com/electric/univers4.htm.

Jiao, J. and Tseng, M. M., 2000, "Understanding Product Family for Mass Customization by Developing Commonality Indices," *Journal of Engineering Design*, Vol. 11(3), pp. 225-243.

Kawanda, T., 1965, Analysis of design factors of universal motors. EE Japan, Vol. 85(6), pp. 1-8

Kota, S., Sethuraman, K. and Miller, R., 2000, "A Metric for Evaluating Design Commonality in Product Families," *ASME Journal of Mechanical Design*, Vol. 122(4), pp. 403-410.

Messac, A., Martinez, M. P. and Simpson, T. W., 2002, "A Penalty Function for Product Family Design Using Physical Programming," *ASME Journal of Mechanical Design*, Vol. 124(2), pp. 164-172.

Mistree, F., Hughes, O. F., and Bras, B. A., 1993, "The Compromise Decision Support Problem and the Adaptive Linear Programming Algorithm in Kamat MP (ed) Structural optimization: Status and Promise", American Institute of Aeronautics and Astronautics (AIAA), Washington DC, pp 247-289.

Nasar S. A., Unnewehr L. E., 1983, Electromechanics and electric machines. Wiley, New York.

Paula, G., 1997, "Reinventing a Core Product Line," *Mechanical Engineering*, Vol. 119(10), pp. 102-103

Pine, B. Joseph II, 1993, "Mass Customization - The New Frontier in Business Competition," *Harvard Business School Press*, Boston, Massachusetts.

Reed, P., Minsker B., & Goldberg, D. E., 2000, "Designing a competent simple genetic algorithm for search and optimization. Water Resources Research," Vol. 36(12), pp. 3757-3761.

Sanderson, S., Uzumeri, M., 1995, "Managing Product Families: the case of the Sony Walkman," *Res Policy*, Vol. 24, pp. 761-782.

Schonfeld, E., 1998, "The customized, digitized, have-it-your way economy," *Fortune*, pp. 115-124.

Shultz, G. P., 1992, Transformers and Motors, Prentice Hall, Carmel.

Simpson, T. W., Maier, J. R. A. and Mistree, F., 1999, "A product platform concept exploration method for product family design," *Design Theory and Methodology – DTM'99*, Las Vegas, ASME, paper no. DETC99/DTM-8761.

Simpson, T. W., Maier, J. R. A. and Mistree, F., 2001, "Product Platform Design: Method and Application," *Research in Engineering Design*, Vol. 13(1), pp. 2-22.

Srinivas, N. and Deb, K., 1995, Multiobjective function optimization using nondominated sorting genetic algorithms, Evolutionary Computation Journal, Vol. 2(3), pp. 221-248.

Robertson, D. and Ulrich, K., 1998, "Planning Product Platforms," *Sloan Management Review*, Vol. 39(4), pp. 19-31.

Ulrich, K.T., Eppinger, S.D., 2000, "Product Design and Development," 2nd edn., *McGraw-Hill*, New York.

Veinott, C. G., Martin, J. E., 1986, Fractional and Subfractional horsepower Electric Motors, McGraw-Hill, New York.

Weile, D. S., Michielssen, E. and Goldberg, D. E., 1996, "Genetic Algorithm Design of Pareto-optimal Broadband Microwave Absorbers," IEEE Transactions on Electromagnetic Compatibility, Vol. 38(3), pp. 525-528.

Wheelwright, S. C. and Clark, K. B., 1995, "Leading Product Development," Free Press, New York.

Wijenayake A. H., Bailey, J.M., McCleer, P. J., 1995, "Design optimization of an axial gap permanent magnet brushless DC motor for electric vehicle applications," 1995 IEEE Industry Applications Conference, 13th IAS Annual Meeting (IAS'95), Orlando, October 8-12, Vol. 1. IEEE, pp. 685-692

Yamanouchi, T., 1989, "Breakthrough: the development of the Canon Personal Copier," *Long Range Planning*, Vol. 22(5), pp. 11-21.