Research Report

2012-10

Machine Design Optimization Based on Finite Element Analysis in a High-Throughput Computing Environment

W. Jiang, T.M. Jahns, T.A. Lipo, W. Taylor, Y. Suzuki*

Dept. Elect. & Comp. Engr.
University of Wisconsin-Madison
1415 Engineering Drive
Madison WI 53706

* JSOL Corp.
Middleton, WI, USA

Wisconsin
Electric
Machines &
Power
Electronics
Consortium

University of Wisconsin-Madison
College of Engineering
Wisconsin Power Electronics Research Center
2559D Engineering Hall
1415 Engineering Drive
Madison WI 53706-1691

Machine Design Optimization Based on Finite Element Analysis in a High-Throughput Computing Environment

Wenying Jiang T.M. Jahns T.A. Lipo William Taylor Yusaku Suzuki
wjiang4@wisc.edu jahns@engr.wisc.edu lipo@ece.wisc.edu bt@cs.wisc.edu y.suzuki@powersys.solutions.com
Dept. of Electrical & Computer Engineering Dept. of Computer Science JSOL Corporation
University of Wisconsin-Madison Middleton, WI USA
Madison, WI USA

Abstract—**Although finite element (FE) analysis is a powerful analytical tool for electric machines, it is rarely used in iterative machine design optimization programs since it is computationally intensive, requiring excessive calculation times. This paper describes an approach for overcoming this obstacle using a high-throughput computing (HTC) environment that harnesses the parallel processing capabilities of large numbers of computers to evaluate many candidate designs simultaneously. Differential evolution has been selected as the optimization algorithm that applies FE analysis to maximize the electromagnetic performance according to an objective function in a computationally-efficient manner. This software has been applied using available HTC resources to optimize the design of a 30 kW (continuous) fractional-slot concentrated winding (FSCW) surface permanent magnet (SPM) machine for high torque density. Tests comparing the computational speeds achieved using the same optimization software with the HTC resources and a single computer have demonstrated a major reduction (approx. 30:1) of the computation time using the HTC approach.**

I. Introduction

In the development of machine design optimization algorithms, high accuracy is desired throughout the whole solution space along with fast computational speed. Finite element (FE) analysis is often the preferred analysis tool for electric machines because of the highly nonlinear nature of the machine electromagnetic characteristics. Unfortunately, the adoption of FE analysis in multi-dimensional search programs to find optimal designs has generally been impractical because of the intensive computational load imposed by FE analysis of such large numbers of candidate designs. As a result, only a few researchers have reported efforts to pursue this approach [1-3], despite the fact that the idea of applying distributed computing techniques to finite element analysis and optimization of electric machines has been recognized for many years [4].

A high-throughput computing (HTC) environment is available at UW-Madison known as Project Condor that harnesses the parallel processing capabilities of more than 10,000 cores [5]. The availability of these distributed parallel computing resources offers opportunities for significantly reducing the total computational time for iterative optimization of electric machines, making FE analysis much more attractive for this task.

There are many different optimization algorithms used in the area of machine design optimization. Differential evolution (DE), particle swarm optimization (PSO), and genetic algorithm (GA) are the three optimization algorithms discussed most widely in publications [6-8]. DE, PSO, and GA all fall into the class of population-based, stochastic function minimizers. As such, they are all well-suited for machine design optimization, which is a nonlinear and constrained problem involving both continuous and discrete variables. Comparisons between DE and many other optimization algorithms have been reported in a recent study [9]. The results show that DE may not always be the fastest method, but it is usually the one that produces the best results, and there are frequent cases in which it is the fastest algorithm as well. Since DE is a population-based algorithm, it is suitable for adoption in a distributed parallel computing environment.

The paper presents the implementation of a machine design optimization algorithm based on electromagnetic FE analysis using the Project Condor HTC environment. Differential evolution is used as the optimization algorithm. The software is applied to optimize a 30 kW (continuous) surface PM machine for achieving maximum torque density using both the Project Condor parallel computing resources and, for comparison, a single computer in the Condor pool. Comparison of the test results delivered by the two computational environments demonstrates that a major reduction in the required computational time can be achieved using the Condor HTC resources.

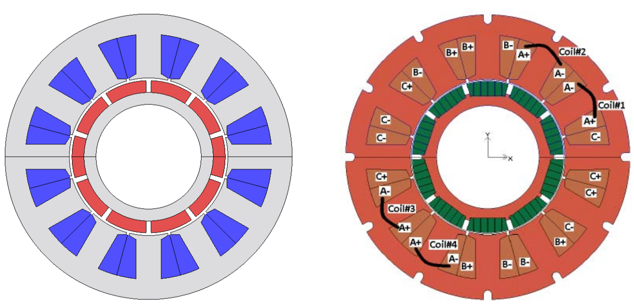
II. Implementation of FE Analysis based Machine Design Optimization in Condor

A. Target Machine Configuration

The target machine selected for this study is a 30 kW (continuous) fractional-slot concentrated winding (FSCW) surface PM machine configured with 10 poles and 12 stator slots. This machine configuration has been developed as part of an investigation of FSCW-SPM machines for high-

TABLE I. PROTOTYPE FSCW-SPM TRACTION MACHINE KEY DIMENSIONS AND MATERIAL CHARACTERISTICS

Parameter/Metric	Value
Continuous Power Rating [kW]	30
Rated Speed [r/min]	2800
Stator Outer Diameter [mm]	274.2
Airgap Thickness [mm]	1
Banding Thickness [mm]	1.5
Current Density [A/mm^2]	4.6
Copper Fill Factor [%]	50%
Copper Resistivity [nOhm-m]	25 @ 150 °C
Steel Lamination Material, Packing Factor	M19, 95%
Remanent Flux Density [T]	1.1 @ 150 °C
Magnet Coercive Force [kA/m]	800
Hysteresis Loss Coeff. [W/m^3/Hz]	256
Eddy Current Loss Coeff. [W/m^3/Hz^2]	0.37
Copper Density [kg/m^3]	8940
Steel Density [kg/m^3]	7850
Magnet Density [kg/m^3]	7450



a) FE model cross-section b) Prototype machine cross-section [10]

Fig. 1: FE model crosssection compared with the prototype machine

performance traction machine applications [11]. A prototype FSCW machine was designed and built for this application with an active mass of 27.8 kg that includes the stator and rotor electromagnetic assemblies [10].

The prototype machine was designed for the FreedomCar (now renamed as U.S. DRIVE) advanced traction motor specifications, while the specifications used for this study is shown in Table I. The specifications for this study differ from the FreedomCar specs primarily by concentrating their focus on machine performance at the corner point operating conditions, i.e.,$n = 2800$ r/min, $P = 30$ kW (continuous).

An electromagnetic FE model has been built based on the parameters provided in [10], and its cross-section is compared with that of the prototype machine in Fig. 1. Based on the operating condition specified in Table I, the rated torque is $T = 102.3$ Nm. The FE model of the prototype PM machine predicts an average torque of 100.7 Nm for corner

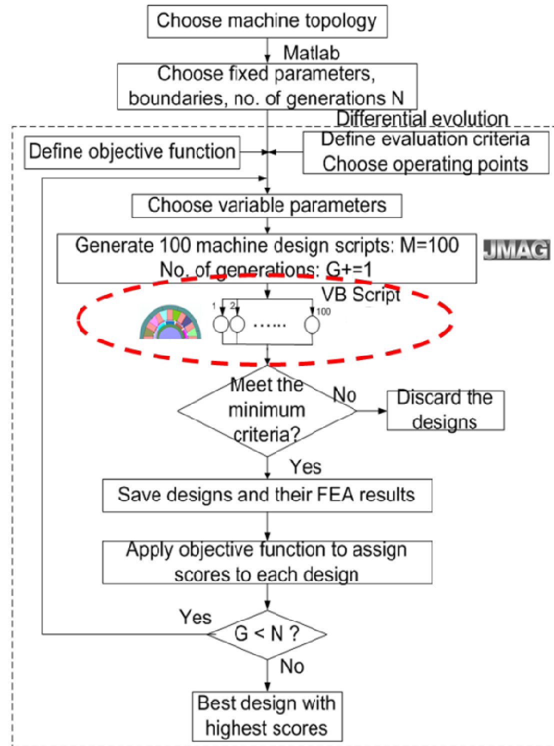


Fig. 2: Flow chart of machine design optimization algorithm in HTC environment.

point operation, while the experimental test results show that the prototype machine is able to deliver an output torque of 99.7 Nm under these conditions [11]. As a result of this good agreement, there is confidence that the FE model can be adopted for this study to accurately evaluate the torque production characteristics of candidate FSCW-SPM machine designs.

B. Optimization Algorithm: Differential Evolution (DE)

The implementation of the FE analysis is based on the machine design optimization flow diagram presented in Fig. 2. The DE optimization algorithm written in MATLAB provides the outer shell of the optimization program that governs the optimization process. It is a generation-based algorithm, so it provides the opportunity for parallel analysis of all designs within each generation. Through initialization, mutation, crossover and selection, DE is able to explore the entire design space and locate the global optimum with high confidence [9]. Each is described briefly as follows:

1. Initialization

$$x_{j,i,0} = rand_j(0,1) * \left(b_{j,U} - b_{j,L}\right) + b_{j,L} \quad (1)$$

where $\boldsymbol{x}$ represents the vector of adjustable machine design variables. The subscript *0* represents the initial generation (i.e. $g = 0$), j and i represent the j^{th} parameter of the i^{th} vector in the initial generation of variable vectors, *rand* is a scalar random number between 0 and 1, and $\boldsymbol{b}_L$, $\boldsymbol{b}_U$ are initialization lower- and upper-bound vectors, respectively.

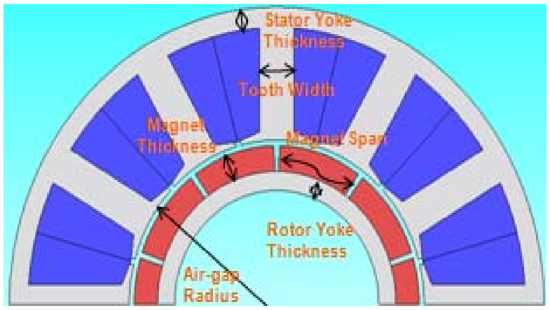


Fig. 3: FSCW-SPM machine design variables

TABLE II. FSCW-SPM VARIABLE DEFINITIONS AND RANGES

Variable Parameters	Range
Tooth Width to Slot Pitch Ratio	[0.1, 0.8]
Stator Yoke Thickness to Tooth Width Ratio	[0.1, 0.8]
Magnet Span to Rotor Pole Pitch Ratio	[0.5, 0.95]
Rotor Yoke Thickness to Rotor Pole Pitch Ratio	[0.1, 0.6]
Magnet Thickness to Airgap Thickness Ratio	[1, 7]
Airgap Radius to Stator Outer Radius Ratio	[0.3, 0.75]

2. *Mutation*

$$\mathbf{v}_{i,g} = \mathbf{x}_{r0,g} + F * (\mathbf{x}_{r1,g} - \mathbf{x}_{r2,g}) \quad (2)$$

In the g^{th} generation, $\mathbf{x}_{r0,g}$, $\mathbf{x}_{r1,g}$ and $\mathbf{x}_{r2,g}$ are three randomly selected vectors of machine design variables. The mutant vector $\mathbf{v}_{i,g}$ is created by adding a scaled vector difference to the third vector. The scale factor, $F \in (0, 1+)$, is an adjustable user-defined scalar variable that controls the rate at which the population evolves.

3. *Crossover*

$$\mathbf{u}_{i,g} = u_{j,i,g} = \begin{cases} v_{j,i,g} & if\ (rand_j(0,1) \leq Cr) \\ x_{j,i,g} & otherwise. \end{cases} \quad (3)$$

The trial vector $\mathbf{u}_{i,g}$ is assembled from parameter values that have been copied from two different source vectors: the mutant vector $\mathbf{v}_{i,g}$ and the current vector $\mathbf{x}_{i,g}$. The crossover probability, $Cr \in [0, 1]$, is a user-defined value that controls the fraction of parameter values that are selected from the mutant vector.

4. *Selection*

$$\mathbf{x}_{i,g+1} = \begin{cases} \mathbf{u}_{i,g} & if\ f(\mathbf{u}_{i,g}) \leq f(\mathbf{x}_{i,g}) \\ \mathbf{x}_{i,g} & otherwise. \end{cases} \quad (4)$$

A user-defined objective function f is used to evaluate performance metrics of all candidate designs. DE uses objective function results to formulate choices for the next generation of designs. The design performance improves with each generation. Once the pre-specified termination criterion is satisfied, the iterative optimization process ends.

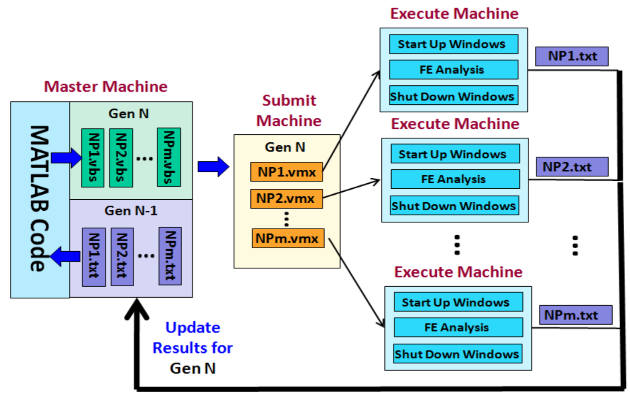


Fig. 4: Data flow in the Project Condor HTC environment

C. *High Throughput Computing (HTC) Environment: Project Condor*

The highlighted circle in Fig. 2 is the key to this optimization, identifying the site of the major of this work from conventional methods. More specifically, the Condor HTC environment provides the opportunity for parallel analysis that reduces the required computational time by simultaneously analyzing the *N* designs in each generation.

Condor is a specialized workload management system for computation-intensive jobs. It can be used to manage a cluster of dedicated compute nodes or to harness CPU power from otherwise idle desktop workstations [6]. Condor utilizes open-source software developed by the UW Condor Team that can be run on Linux, Macintosh, and Windows machines. It matches jobs to available computers, and transfers required data to and from remote resources. Moreover, Condor enables workflow execution that allows its users to utilize many resources while maintaining the complex job inter-dependencies and dataflow.

In the Condor pool, all the computers are classified into two major types: Submit Machines and Execute Machines. Users send their jobs to the Submit Machine and specify which kind of Execute Machines they want their jobs to be run on. After job initiation, the Submit Machine begins searching the pool for Execute Machines. Condor plays the critical role of "Matchmaker" that efficiently matches submitted job requests with appropriate Execute Machines. Once it detects any suitable Execute Machine that becomes available, Condor automatically builds the links between the Submit Machine and the Execute Machines, sends the required input data to the Execute Machines, runs the analyses, and collects the results.

There are over 10,000 cores available for use by HTC users through the Center for High Throughput Computing (CHTC) at UW-Madison. Approximately one-third of the cores were purchased and are maintained by the center. The other two-thirds are owned by its users, who contribute their cores to make up the system and become part of the shared resource.

Condor used to be a very specialized research facility that was utilized by a small group of scientists. Today, it is open to a much wider community of users. The opportunity to install commercial software in this environment makes Condor an attractive environment for running FE analysis.

D. FE Analysis based Machine Design Optimization in Condor

The data flow in the Project Condor HTC environment is illustrated in Fig. 4. Each electric machine FE model can be defined by a set of parameters, including both fixed and variable types. The fixed parameters are defined by the machine specifications listed in Table I. The six variable parameters for the FSCW-SPM machine are identified in Fig. 3, consisting of the stator tooth width, stator yoke thickness, magnet span ratio, rotor yoke thickness, magnet thickness, and airgap radius. The numerical ranges for these variables, shown in Table II for this example, are defined as ratios rather than absolute values in order to help insure that each candidate design meets all of the geometric constraints. The range boundary values are chosen to insure that the optimum values fall within the variable ranges.

Next, all of the parameters in the parameter set are fed into the template Visual Basic(VB) script for FE analysis that contains information on the machine configuration, material, and excitation to create a specific script for each candidate design. All of these FE analysis conditions are set up in the same way as described for the FE model in Section II.A.

When all of the VB scripts in one generation (iteration) of the DE optimization are written, each VB script is mounted in Virtual Machine (VM) format, which corresponds to a CD-ROM image. The Virtual Machine approach is necessary because JMAG, the commercial FE analysis package used for the study, can only run in a Windows computer, while Condor is a Linux-based environment. The VM method helps to solve this problem since it allows Windows jobs to be run on a Linux machine.

Next, all of the VM images for each generation are sent to the Submit Machine (see Fig. 4). Every VM image will be mapped from the Submit Machine to an Execute Machine where each candidate machine design is analyzed by JMAG immediately after starting up Windows. When each FE analysis is completed, Windows is automatically shut down. Then the FE results for the candidate machine, which is written in a text file, is extracted from the VM image and transferred back to the Submit Machine for the main DE code to evaluate the objective (cost) function for the candidate machine. If the evaluated cost function value fulfills the convergence criteria, the optimization run will terminate. Otherwise, the whole process described in Figs. 2 and 4 will be repeated again until the optimum design is found.

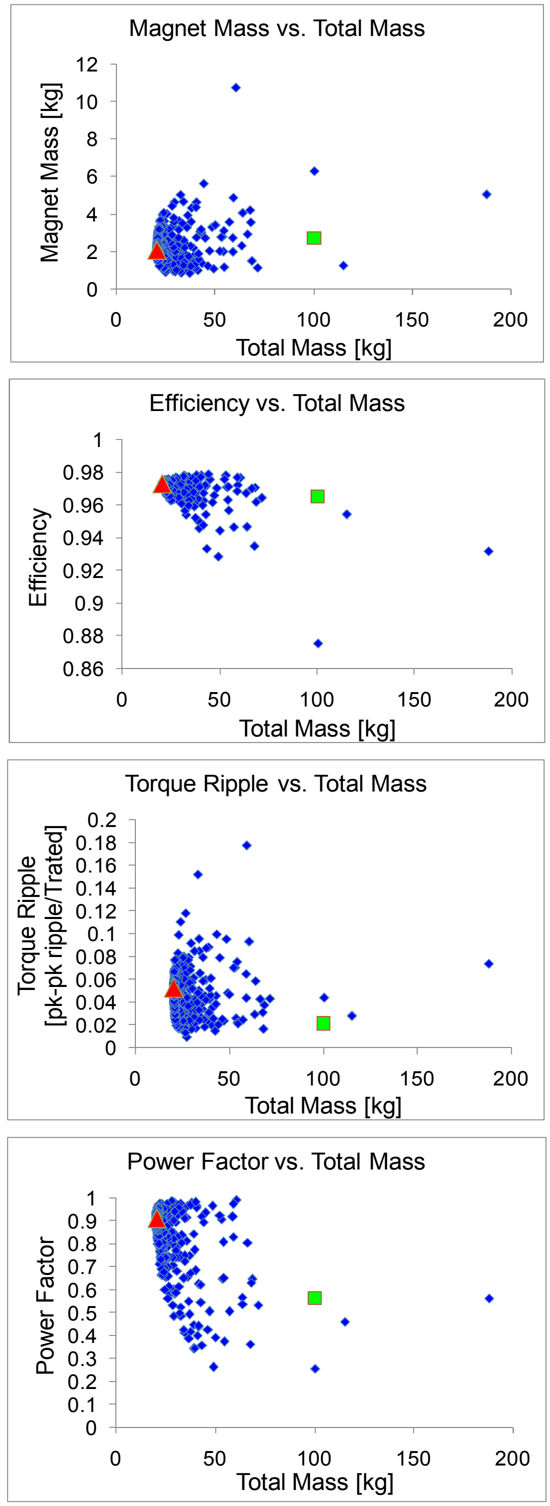


Fig. 5: Plots showing magnet mass, machine efficiency, torque ripple, and power factor, (all for rated power operation) vs. machine electromagnetic mass for all of the machine designs evaluated during the DE optimization. Points for initial design (green square) and final design (red triangle) are highlighted in each plot.

TABLE III. OPTIMAL FSCW-SPM MACHINE DETAILS

Parameter/Dimension	Value
Airgap Diameter [mm]	181.48
Active Stack Length [mm]	57.01
Magnet Height [mm]	10.46
Magnet Span [deg. elec.]	32.14
Volume [m^3]	0.0025
Copper Mass [kg]	10.11
Magnet Mass [kg]	2.07
Total Active Mass [kg]	20.65
Torque Ripple [pk-pk/T_{rated}]	0.05
Power Factor	0.91
Magnet Loss [W]	59.44
Core Loss [W]	282.39
Copper Loss [W]	609.64
Efficiency	0.97

III. COMPARISON BETWEEN HTC AND SINGLE COMPUTER OPTIMIZATION RESULTS

In order to quantify the reduction in computation time that can be achieved using the Condor HTC environment compared to using a single computer, a comparison test was launched. The same machine design optimization software has been applied to optimize the torque density for the FSPM-SPM machine using either the Condor HTC resources or a single computer that was chosen from the Condor pool in order to provide a fair comparison.

A. Comparison Test Conditions & Optimization Parameters

The candidate machine was designed for rated operating conditions, i.e. $n = 2800$ r/min, $P = 30$ kW, $T = 102.3$Nm, where n, P and T represent the rotor speed, output mechanical power, and torque, respectively. The objective function *OF* has been defined as

$$OF = \frac{Calculated\ Active\ Mass\ to\ Produce\ Required\ Torque}{Base\ Machine\ Active\ Mass} \quad (5)$$

where the base machine is the one studied previously in [10] with an active mass of 27.8 kg that includes the stator and rotor electromagnetic assemblies.

Based on previous experience, the control parameters for the DE optimizer were chosen to be: convergence tolerance $Tol = 10^{-6}$; number of designs in each generation $NP = 85$; crossover probability $Cr = 0.8$; and scale factor $F = 0.8$.

B. Optimized Design

The two optimizations were then run and both converge to the same optimal design at the 50th generation, with a total number of 4250 designs evaluated in each case. The best design is found to have an objective function value of 0.7426 for both cases, i.e. a total mass of 20.65 kg.

Key dimensions and metrics for the optimal machine are summarized in Table III. Plots of four machine metrics evaluated at the corner operating point – torque, efficiency, torque ripple, and power factor – are plotted in Fig 5 as a function of machine mass for all of the candidate machine designs that were evaluated during the optimization (i.e., the Pareto-optimal set). That is, each of the >4000 machine designs is represented by a point in each plot. Two data points are highlighted in each plot that identifies the metrics of the original machine and the final optimal machine.

It can be observed that optimizing the torque density helps to simultaneously reduce the magnet mass, improve the efficiency, and increase the power factor even though these other metrics are not part of the objective function. However, the optimization does not have much impact on the predicted torque ripple. Taken together, these results suggest the choice of torque density as the target for the objective function is a good one that results in improvements for several important performance metrics of the machine.

The optimized machine has a mass reduction of 25.7% compared to the base machine, representing a significant improvement in torque density. Although appealing, the significance of this improvement must be tempered by the fact that the specifications for this new design and the earlier prototype machine are not identical. Nevertheless, the fact that both optimization runs converged to the same optimal design in the same number of generations helps to build confidence in the fairness of the computation time comparison that is presented in the next section.

C. Comparison of Computational Time

Log files have been captured to record each step during the optimization runs for both the HTC and single computer cases. In Condor, there are three time stamps collected for each design, i.e. "submit", "execute" and "terminate". "Submit" refers to the time when the VM image is submitted to the Submit Machine. "Execute" refers to the time when the VM image is mapped from the Submit Machine to the Execute Machine. "Terminate" refers to the time when Windows has been shut down in the Execute Machine.

Figure 6 shows the time duration between "Execute" and "Terminate" for each of the 85 designs evaluated in one of the 50 DE design generations, Generation #25. This time duration represents the time that is required for each design in an Execute Machine, including Windows start-up, FE analysis time, and Windows shutdown. The shortest time is 9 min 19 sec, while the longest one is 18 min 51 sec, indicating a range that exceeds 2:1. Because Condor is a shared computational environment, every computer in the Condor pool can be assigned to multiple users at the same time. If a machine design is submitted to a heavily loaded Execute Machine, it takes longer to analyze. Since this delay slows the overall computation time, efforts are under way to improve the assignment algorithm so that the designs are

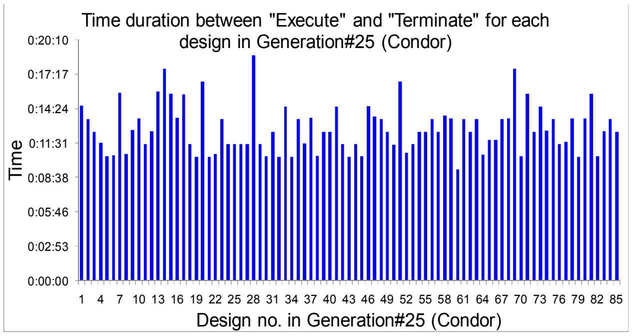


Fig. 6: Time duration between "Execute" and "Terminate" for each design in Generation #25 (Condor)

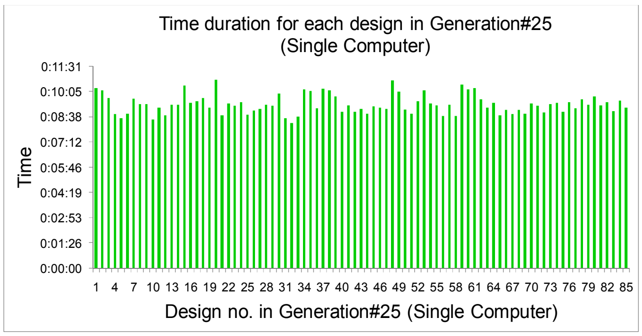


Fig. 7: Time duration for each design in Generation#25 (Single Computer)

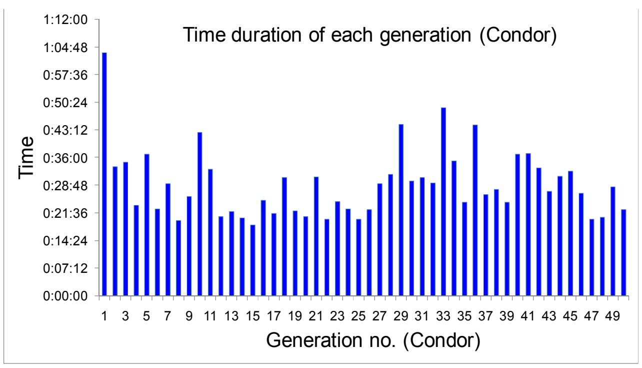


Fig. 8: Time duration of each generation over the 50 generations (Condor)

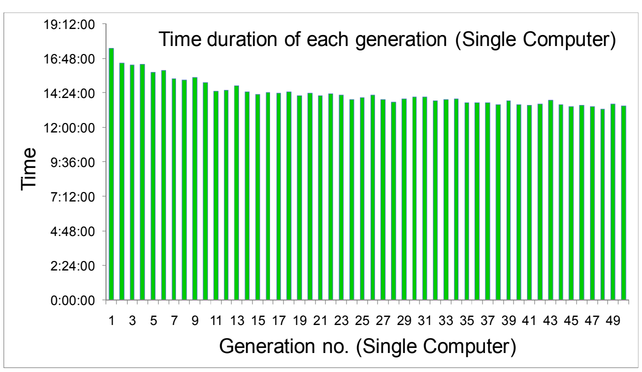


Fig. 9: Time duration of each generation over the total 50 generations (Single Computer)

preferentially matched with lightly-loaded Execute Machines to reduce this overhead time.

Unfortunately, the total time required to complete each generation of design evaluations in the Condor HTC environment is determined by the slowest of the computational periods for the 85 designs in the generation. For Generation #25, this total consist of the maximum “Execute” to “Terminate” value shown in Fig. 6 (18 min 51 sec) plus 1 min 9 sec to distribute the design parameters to all of the Execute Machines, plus 1 min 30 sec for processing the VM format conversions. Thus, the total time required for completing the computations for Generation #25 in the Condor HTC environment is 21 min 30sec.

In the single computer optimization run, there are no overhead or delay time components such as those encountered in the Condor HTC environment with shared parallel computers. One of the impacts of this difference can be observed in Fig. 7 showing the total time required for evaluating each of the 85 designs in Generation #25 with the single computer. This bar chart shows that the shortest time is 8 min 19 sec, while the longest one is 10 min 46 sec. The differences in computation times among the designs using a single computer are significantly smaller than those encountered in the Condor HTC environment, and the longest computation time for any of the 85 designs using a single computer is approx. 50% of the longest time for the Condor case.

Despite this advantage for the single computer case, the total time for each generation is calculated by summing up all of the computation times for the 85 designs when using a single computer since the designs must be evaluated serially. As a result, the total computation time for Generation #25 in the single computer run is 13 hr 16 min 52 sec, which is approx. 37 times longer than the corresponding time period for the Condor run.

Figure 8 shows the computation time required by each of the 50 design generations in the Condor HTC environment. There are variations in the computation time required for the generations ranging from 19 min 32 sec for Generation #15 to 1 hr 4 min 34 sec for Generation #1. By adding these individual generation times together, the total time required for the optimization to converge in the Condor HTC environment is 25 hr 0 min 26 sec., or 1.04 days.

The corresponding computation time for each generation in the single computer case is shown in Fig. 9. Although the variation in the computation times among the 50 generations in the single computer case are small, the total computation time required for the optimization to converge in a single computer environment is 29 d 22hr 17 min 8 sec, or 29.93 days.Thus, the time acceleration factor for completing the optimization using the Condor HTC environment instead of a single computer is 28.7, representing a very significant improvement in computational speed.

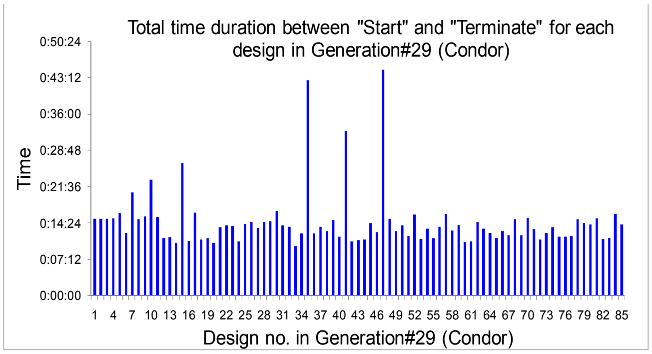


Fig. 10: Total time duration between "Start" and "Terminate" for each design in Generation #29 (Condor)

IV. Computation Time Breakdown and Acceleration Factor Improvements

The computation time acceleration factor of 28.7 that was achieved in the demonstration case using the Condor HTC environment is only approx. one-third of the ideal upper limit value of 85 that this project seeks to approach when using 85 computers performing computations in parallel. After breaking down of all the time components during the optimization, one finds that the required computation time for the DE main code performing most of the operations in Fig. 2 is negligible compared to the FE analysis times. A second important observation is that the variations in the FE analysis times among all of the 4250 designs that were evaluated during the optimization run using the single computer are small.

Using these numbers from the demonstration case, it is estimated that the acceleration factor could be significantly increased if a dedicated HTC network would be completely allocated to solving this same optimization problem. Much of this improvement comes from eliminating the overhead and delay times that are present in the shared Condor HTC environment.

There is clearly a large gap between the demonstrated acceleration factor and the estimated achievable value for a dedicated HTC environment. Some promising opportunities have been identified to improve the acceleration factor in the current Condor HTC environment. For example, Fig. 10 shows the total time duration between "Start" and "Terminate" for each of the 85 designs in Generation #29 for the HTC case. This figure indicates that 46 min 2 sec was required for completing this generation in the Condor HTC environment. This completion time is considerably longer than many of the other generations (see Fig. 6) because of 3 outlier designs that required much longer times to complete than the rest of the designs in the same generation. Alternative approaches to significantly reduce the number of these outliers cases are under investigation, including modifications of the DE algorithm and improvements of the machine assignment software in the Condor HTC environment.

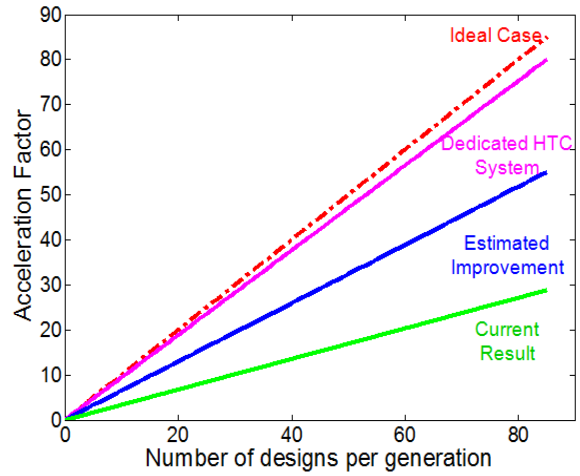


Fig. 11: Acceleration factor vs. number of machine designs in each generation

Figure 11 provides simplified estimates of the acceleration factor that can be achieved as a function of the number of designs in each DE generation for different HTC computing environments. The steepest line is for the unattainable ideal case for which the acceleration factor simply equals the number of designs that are being analyzed in parallel. A dedicated HTC environment is estimated to be capable of achieving approx. 90 to 95% of the ideal value, raising the acceleration factor to approx. 80 for the considered case with 85 designs in each generation.

As discussed in preceding sections, the current shared Condor HTC environment achieves only approx. 34% of the ideal acceleration factor value. It is currently estimated that efforts to reduce the accumulated overhead time in the Condor HTC environment can increase the acceleration factor to >50 for the case with 85 designs in each generation, corresponding to approx. 60% of the ideal value. Efforts are currently under way to demonstrate this increased acceleration factor within the constraints imposed by the Condor HTC environment.

One of the implicit assumptions for Fig. 11 is that the number of generations required by the DE algorithm to find an optimum design is fixed (50) independent of the number of designs per generation. While this topic extends beyond the scope of this paper, it is acknowledged that this assumption does not hold for all numbers of designs per generation. Nevertheless, it is interesting to note that the acceleration factor is largely independent of the number of generations because it is a ratio of computation times for two cases that each requires the same number of generations, even if that number varies. As a result, the key conclusions reached in this section about the acceleration factor are insensitive to the number of required DE generations.

V. Conclusions

This paper presents the implementation of an iterative FE-based machine design optimization algorithm based on differential evolution in a high-throughput computing (HTC) environment. The software has been applied to optimize a 30 kW (continuous) FSCW-SPM machine for high torque density using both an available HTC environment (Project Condor) and a single computer of the same type used in the HTC system. Commercially-available FE analysis software is used for these calculations.

The two optimization runs converge at the same optimal design, which exhibits a mass reduction of 25.7% compared to the baseline machine design. The comparison shows that the parallel HTC environment achieves a reduction in the computational time by a factor of 28.7 compared to the single computer. This result highlights the significant reductions in computational time of FE-based machine design optimization problems that can be achieved using currently-available HTC resources. Furthermore, closer examination has indicated that the acceleration factor can be improved considerably, and efforts are under way to reduce the overhead time associated with Condor environment.

To date, the focus of this study has been on the electromagnetic design. However, an electrical machine is also heavily constrained by thermal limits. Moreover, if the machine is operated at elevated speeds, the structural limits need to be taken into account as well. As a result, comprehensive optimization of an electrical machine design requires that its electromagnetic, thermal, and structural performance must be analyzed and optimized simultaneously. With the increasing accessibility of powerful HTC environments to many researchers, opportunities are opening for extending the optimization to include these critical multi-physics dimensions. Efforts are currently under way to expand the investigation, focusing initially on the integration of the electromagnetic and thermal machine design optimizations.

Acknowledgments

The authors are grateful to the Center for High Throughput Computing at UW-Madison for access to their facilities and technical support, and also to JSOL Corp. for making their JMAG software available for this investigation.

References

[1] Y. Duan, R.G. Harley, T.G. Habetler, “A useful multi-objective optimization design method for PM motors considering nonlinear material properties,” in Proc. of *2009 IEEE Energy Conversion Congress and Exposition (ECCE 2009)*, San Jose, pp. 187-19.

[2] P.D. Pfister, Y. Perriard, “Very-high-speed slotless permanent-magnet motors: analytical modeling, optimization, design, and torque measurement methods,” *Industrial Electronics, IEEE Transactions*, Vol.57, Issue. 1, pp. 296-303, Jan., 2010.

[3] G. Pellegrino, F. Cupertino, “FEA-based multi-objective optimization of IPM motor design including rotor losses,” in Proc. of *2010 IEEE Energy Conversion Congress and Exposition (ECCE 2010)*, Atlanta, pp. 3659-3666.

[4] C.C. Chan, K.T. Chau, “Design of electrical machines by the finite element method usig distributed computing”, *Computers in Industry*, vol. 17, pp. 367-374, 1991.

[5] http://research.cs.wisc.edu/condor/

[6] G. Sizov, P.Zhang, D. Ionel, N. Demerdash, M.Rosu, “Automated bi-objective design optimization of multi-MW direct-drive PM machines using CE-FEA and differential evolution,” in *Proc. of 2011 IEEE Energy Conversion Congress and Exposition (ECCE 2011),* Phoenix, pp. 3672-3678.

[7] R. Bhuvaneswar, V.P. Sakthivel, S. Subramanian, G.T. Bellarmine, “Hybrid approach using GA and PSO for alternator design,” in Proc. of *2009 IEEE Southeastcon Conference,* pp. 169-174.

[8] K.S. Chai, C. Pollock, “Using genetic algorithms in design optimization of the flux switching motor,” in *Proc. of 2002 IEE Power Electronics, Machines and Drives Conference (PEMD 2002),* pp. 540-545.

[9] K. Price, R. Storn, J. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*. Springer, 2005.

[10] P.B. Reddy, T.M. Jahns, P.J. McCleer, T.P. Bohn, “Design, analysis and fabrication of a high-performance fractional-slot concentrated winding surface PM machine”,in *Proc. of 2010 IEEE Energy Conversion Congress and Exposition (ECCE 2010),* Atlanta, pp. 1074-1081.

[11] P.B. Reddy, "Loss in high speed fractional slot concentrated winding surface permanent magnet machines". PhD Thesis, Univeristy of Wisconsin - Madison, 2010.