Engineering Applications of Artificial Intelligence 26 (2013) 1781–1794

Contents lists available at SciVerse ScienceDirect

Engineering Applications of Artificial Intelligence

journal homepage: www.elsevier.com/locate/engappai

Hybridization of multi-objective evolutionary algorithms and artificial neural networks for optimizing the performance of electrical drives

Alexandru-Ciprian Zăvoianu [a,c], Gerd Bramerdorfer [b,c], Edwin Lughofer [a,*], Siegfried Silber [b,c], Wolfgang Amrhein [b,c], Erich Peter Klement [a,c]

[a] Department of Knowledge-based Mathematical Systems/Fuzzy Logic Laboratory Linz-Hagenberg, Johannes Kepler University of Linz, Austria
[b] Institute for Electrical Drives and Power Electronics, Johannes Kepler University of Linz, Austria
[c] ACCM, Austrian Center of Competence in Mechatronics, Linz, Austria

ARTICLE INFO

Article history:
Received 28 August 2012
Received in revised form
31 May 2013
Accepted 2 June 2013
Available online 29 June 2013

Keywords:
Electrical drives
Performance optimization
Multi-objective evolutionary algorithms
Feed-forward artificial neural networks
Surrogate fitness evaluation
Hybridization

ABSTRACT

Performance optimization of electrical drives implies a lot of degrees of freedom in the variation of design parameters, which in turn makes the process overly complex and sometimes impossible to handle for classical analytical optimization approaches. This, and the fact that multiple non-independent design parameter have to be optimized synchronously, makes a soft computing approach based on multi-objective evolutionary algorithms (MOEAs) a feasible alternative. In this paper, we describe the application of the well known Non-dominated Sorting Genetic Algorithm II (NSGA-II) in order to obtain high-quality Pareto-optimal solutions for three optimization scenarios. The nature of these scenarios requires the usage of fitness evaluation functions that rely on very time-intensive finite element (FE) simulations. The key and novel aspect of our optimization procedure is the *on-the-fly automated creation of highly accurate and stable surrogate fitness functions* based on artificial neural networks (ANNs). We employ these surrogate fitness functions in the middle and end parts of the NSGA-II run (→ hybridization) in order to significantly reduce the very high computational effort required by the optimization process. The results show that by using this hybrid optimization procedure, the computation time of a single optimization run can be reduced by 46–72% while achieving Pareto-optimal solution sets with similar, or even slightly better, quality as those obtained when conducting NSGA-II runs that use FE simulations over the whole run-time of the optimization process.

1. Introduction

1.1. Motivation

Today, electrical drives account for about 70% of the total electrical energy consumption in industry and for about 40% of used global electricity (EMSA). In De Keulenaer et al. (2004) it is stated that, each year, in the European Union, the amount of wasted energy that could be saved by increasing the efficiency of electrical drives is around 200 TWh and for this reason, in 2009, a European regulation was concluded forcing a gradual increase of the energy efficiency of electrical drives (European Union, 2009). However, manufacturers of electrical machines need to take more than just the efficiency into account to hold their own value in the global market. To be able to successfully compete, the electrical drives should be fault-tolerant and should offer easy to control operational characteristics and compact dimensions. Apart from these, the most important quality factor is the price. During the development of an electrical machine, a multi-objective optimization approach (Chiong, 2012; Chiong et al., 2012) is required in order to address all of the above aspects and to find an appropriate tradeoff between the final efficiency and the cost of the drive.

1.2. State-of-the-art in electrical drive design

In the past, electrical machines were designed by applying a parameter sweep and calculating a maximum of several hundred designs (Johansson et al., 1994). Calculating a design actually means predicting the operational behavior of the electrical drive for a concrete set of parameter settings. Because of the nonlinear behavior of the materials involved, such a prediction needs to be based on time intensive finite element simulations. This, combined with the need to have an acceptable duration of the overall analysis, imposed a severe limitation in the number of designs to be calculated. As such, only major design parameters could be taken into consideration and only a rather coarse parameter step size could be applied.

* Corresponding author. Tel.: +43 7236 3343 435.
E-mail address: edwin.lughofer@jku.at (E. Lughofer).

0952-1976/$ - see front matter
http://dx.doi.org/10.1016/j.engappai.2013.06.002

During the last decade, the use of response surface methodology (Hwang et al., 207), genetic algorithms (Bianchi and Bolognani, 1998; Jannot et al., 2011), particle swarm optimization (del Valle et al., 2008) and other techniques (Russenschuck, 1990) for the design of electrical machines and the associated electronics has become state-of-the-art. For a detailed comparisons of these modern approaches and additional reviews of the state-of-the-art in electrical drive design, the reader is kindly directed to consult (Duan et al., 2009; Duan and Ionel, 2011; Skaar and Nilssen, 2003).

Although the above mentioned search methods have proved to be far more suitable for the task of multi-objective optimization than basic parameter sweeps, they are still plagued by the huge execution times incurred by the need to rely on FE simulations throughout the optimization procedure. The usage of computer clusters where multiple FE simulations can be performed in parallel can partially address this problem, but the following drawbacks still remain severe:

- The FE evaluation of one particular design still takes a long time and conventional methods need to evaluate each individual design.
- There are high costs associated with the usage of computer clustering architectures and various software licenses.

1.3. Our approach

In our attempt to create an efficient optimization framework for electrical drive design, we are exploiting well known and widely applied genetic algorithms used for multi-objective optimization. These specialized algorithms are generally able to efficiently handle several optimization objectives. For us, these objectives are electrical drive target parameters like efficiency, cogging torque, total iron losses, etc. In our implementation, the goal is to minimize all the objectives. If a target needs to be maximized in the design (e.g. efficiency), during the optimization, its negative value is taken to be minimized. The FE simulations required by each fitness function evaluation are distributed over a high throughput computer cluster system. Although it is able to evolve electrical drive designs of remarkable high quality, the major drawback of this initial, and somewhat conventional, optimization approach (**ConvOpt**) is that it is quite slow as it exhibits overall optimization run-times that vary from ≈44 to ≈70 h. As a particular multi-objective genetic algorithm, we employ the well-known and widely used NSGA-II (Deb et al., 2002).

One main method aimed at improving the computational time of a multi-objective evolutionary algorithm that has a very time-intensive fitness function is to approximate the actual function through means of *metamodels/surrogate models* (Santana-Quintero et al., 2010). These surrogate models can provide a very accurate estimation of the original fitness function at a fraction of the computational effort required by the latter. Three very well documented overviews on surrogate based analysis and optimization can be found in Queipo et al. (2005), Forrester et al. (2008) and Tenne and Goh (2010).

In our case, the idea is to substitute the time-intensive fitness functions based on FE simulations with very-fast-to-evaluate surrogates based on highly accurate regression models. The surrogate models act as direct mappings between the design parameters (inputs) and the electric drive target values which should be estimated (outputs). For us, in order to be effective in their role to reduce overall optimization run-time, the surrogate models need to be constructed *on-the-fly, automatically, during the run of the evolutionary algorithm*. This is because they are quite specific for each optimization scenario and each target value (i.e., optimization goal or optimization constraint) that we consider.

In other words, we would like that only individuals (i.e., electrical drive designs) from *the first N generations* will be evaluated with the time-intensive FE-based fitness function. These initial, FE evaluated, electrical drive designs will form a training set for constructing the surrogate models. For the remaining generations, the surrogate models will substitute the FE simulations as the basis of the fitness function. As our tests show, this yields a significant reduction in computation time.

The novelty of our research lies in the analysis of how to efficiently integrate automatically created on-the-fly-surrogate-models in order to reduce the overall optimization run-time without impacting the high quality of the electrical drive designs produced by ConvOpt.

Artificial Neural Networks (ANNs) (Haykin, 1999) are among the popular methods used for constructing surrogate models because they possess the universal approximation capability (Hornik et al., 1989) and they offer parameterization options that allow for an adequate degree of control over the complexity of the resulting model. Another advantage of ANNs is the fact that they are known to perform well on non-linear and noisy data (Paliwal and Kumar, 2009) and that they have already been successfully applied in evolutionary computation for designing surrogate models on several instances (Jin et al., 2004; Hong et al., 2003). For the purpose of this research, the particular type of ANN we have chosen to use is the multilayered perceptron (MLP). MLP is a popular and widely used neural network paradigm that has been successfully employed to create robust and compact prediction models in many practical applications (Gupta et al., 2007; Wefky et al., 2011). However, our choice for the MLP is first and foremost motivated by the fact that, for our specific modeling requirements, MLP-bases surrogate models have proved to be both relatively fast and easy to create as well as extremely accurate.

There is a wide choice of methods available for constructing surrogate models. In this paper, we describe in details how we created surrogates based on MLPs, but our hybridization schema itself is general and suitable for a multitude of modeling methods. In Section 5.1 we present results obtained with other non-linear modeling methods that can be used as alternatives for constructing the surrogate models. These modeling methods are, support vector regression (SVR) (Collobert and Bengio, 2001), RBF networks (Buhmann, 2003) and a regression orientated adaptation of the instance based learning algorithm IBk (Aha et al., 1991). In the aforementioned section, we also further motivate our current preference for MLP surrogate models.

Regardless of the modeling method used, the automatic surrogate model construction phase involves testing different parameter settings (e.g. different number of neurons and learning rates in the case of MLPs, different values of C and γ in the case of SVR), yielding many models with different complexities and prediction behaviors. Given a certain target parameter we propose a new, automated model selection criterion, aimed at selecting the best surrogate to be integrated in the optimization process. The selected surrogate model should deliver the best tradeoff between smoothness, accuracy and sensitivity, i.e., the lowest possible complexity with an above-average predictive quality.

The rest of this paper is organized in the following way: Section 2 presents an overview of multi-objective optimization problems (MOOPs) in general with a special focus on the particular complexities associated with MOOPs encountered in the design and prototyping of electrical drives. Section 3 contains a description of our hybrid optimization procedure (**HybridOpt**) focusing on the creation and integration of the MLP surrogate models. Section 4 provides the description of the experimental setup. Section 5 contains an evaluation of the performance of the hybrid optimization process