Computers and Mathematics with Applications 57 (2009) 1929–1937

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

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

Product platform two-stage quality optimization design based on multiobjective genetic algorithm☆

Wei Wei, Yixiong Feng *, Jianrong Tan, Zhongkai Li

States Key Lab of Fluid Power Transmission and Control, Zhejiang University, Hangzhou 310027, PR China

ARTICLE INFO

Keywords:
Product platform
Multiobjective optimization
NSGA-II
Product family
Traction machine

ABSTRACT

Product platform design (PFD) has been recognized as an effective means to satisfy diverse market niches while maintaining the economies of scale and scope. Numerous optimization-based approaches have been proposed to help resolve the tradeoff between platform commonality and the ability to achieve distinct performance targets for each variant. In this study, we propose a two-stage multiobjective optimization-based platform design methodology (TMOPDM) for solving the product family problem using a multiobjective genetic algorithm. In the first stage, the common product platform is identified using a nondominated sorting genetic algorithm II (NSGA-II); In the second stage, each individual product is designed around the common platform such that the functional requirements of the product are best satisfied. The design of a family of traction machine is used as an example to benchmark the effectiveness of the proposed approach against previous approachs.

1. Introduction

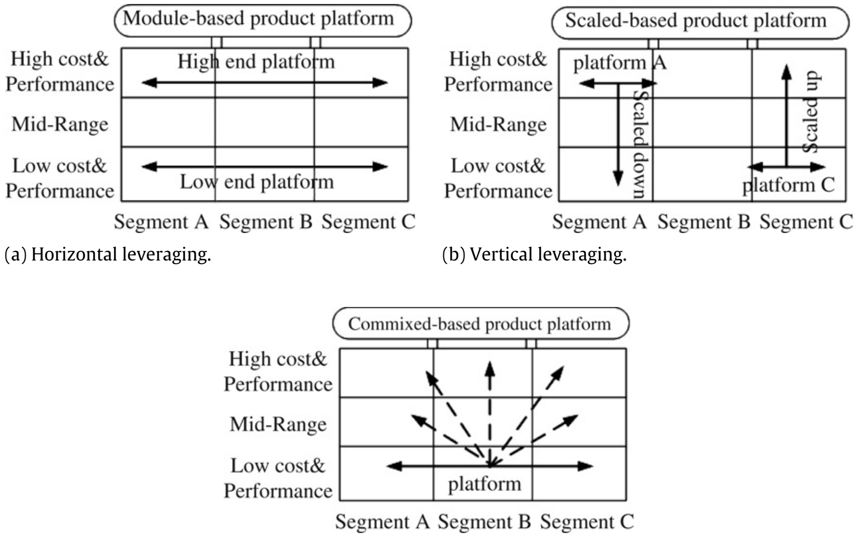
In an effort to improve customization for today's highly competitive global marketplace, many companies are utilizing product families to increase variety, shorten lead-times, and reduce costs. The product platform, which was provided by Meyer in 1997 [1] is a set of subsystems and interfaces intentionally planned and developed to form a common structure from which a stream of derivative products can be efficiently developed and produced. Platform-based product development offers a multitude of benefits including reduced development time and system complexity, reduced development and production costs, and improved ability to upgrade products. According to different platform leveraging strategies, there are two basic types of platforms; they are the module-based and the scale-based.

Simpson et al. [2] proposed the product platform concept exploration method (PPCEM) to develop the scale-based product platform using the compromise decision support problem (CDSP). Based on the similar principle as that of the PPCEM, the product variety tradeoff evaluation method (PVTEM) is presented by Conner [3] to assess appropriate product family tradeoffs using the commonality and performance indices. In each of the above approaches, the set of the scale factor and that of the common platform parameters are pre-selected by the designer. Ideally, the product family optimization design should explore varying levels of commonality during the optimization process to determine the best product platform rather than having the designer specify the common and unique variables a priori. Toward that end, Messac et al. [4] and Nayak et al. [5] have proposed a product family penalty function and a commonality goal, respectively, to help determine which variables have the largest impact on product performance and commonality. The variation-based

☆ The project was supported by the National Natural Science Foundation of China (Grant Nos. 50835008, 50875237), High-Tech Research and Develop Program of China (Grant Nos. 2008AA042301, 2007AA04Z190) and Science and Technology Support Program of China (Grant No. 2006BAF01A37).

* Corresponding author.
E-mail addresses: boyweiwei@zju.edu.cn (W. Wei), fyxtv@zju.edu.cn (Y. Feng).

0898-1221/$ – see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.camwa.2008.10.022



(a) Horizontal leveraging.

(b) Vertical leveraging.

(c) Beachhead approach.

Fig. 1. Platform leveraging strategies.

platform development methodology (VBPDM) proposed by Nayak identified the common parameters firstly, and calculated the scaling variables of instances in the second stage, which is later adapted by Hernandez et al. [6] to design a family of absorption chillers while considering the associated manufacturing costs. Meanwhile, Dai et al. [7] introduces a single-stage approach using preference aggregation to simultaneously optimize a product platform. The dimensionality of the single-stage optimization problems, however, is considerably higher than in two-stage approaches, which often leads to many computational challenges. Thus, the approach was only suited for the design of small scale product platform. Wang et al. [8] incorporated the robust design methodology with the CDSP to identify the right components and design variables to be shared among products in order to maintain economies of scale with minimum sacrifice in the performance of each product in the family.

Recently, genetic multiobjective optimization, which applies genetic computation to multiobjective optimization has attracted a great deal of attention [9,10]. This paper focuses on multiobjective genetic algorithms (MOGA) and presents a two-stage multiobjective optimization-based platform design methodology (TMOPDM) to design and optimize a scale-based product platform. Many varieties of multiobjective genetic algorithms have been proposed. The recent researches include SPEA [11], NPGA [12], NSGA [13] etc and some improved versions SPEA2 [14], SPEA2 + [15] and NSGA-II [16]. Among these approaches, the improved nondominated sorting genetic algorithm (NSGA-II) proposed by Deb can solve multiobjective optimization efficiently with population ranking, crowding distance, constraints Pareto optimality and provide excellent results, while no user-defined parameters are needed in the process. In addition, the NSGA-II can handle both real and binary representations. So it is recommended in this paper to solve the problems in the proposed multiobjective optimization-based platform development methodology.

The modeling of scale-based product platform is presented in Section 2. In Section 3, the implementation framework of TMOPDM is presented. The issues about multiobjective optimization and implementation of NSGA-II are presented in Section 4, followed by an example involving the design of a family of traction machines and the efficiency and effectiveness comparison of the TMOPDM against previous approaches in Section 5. Conclusions and future research are discussed in Section 6.

2. Product family design and modeling of scale-based product platform

2.1. Platform configuration and product family design

In response to the varieties of customer needs with as little differences as possible between products, platform-based product family design methodologies [17] are advocated for their effectiveness in achieving product variety and scale benefit. Depending on the market segmentation grid which articulates the platform leveraging strategies, there are three different types of leveraging strategies for platform-based product family design. As shown in Fig. 1, three platform leveraging strategies can be identified within the grid: (a) vertical leveraging, (b) horizontal leveraging, and (c) the beachhead approach, which combines both vertical and horizontal leveraging. While most horizontal leveraging strategies take advantage of modular platforms, which changes products through adding, substituting, and/or removing of modules vertical leveraging for a scale-based product platform accommodates features through different values of scaling variables to form series

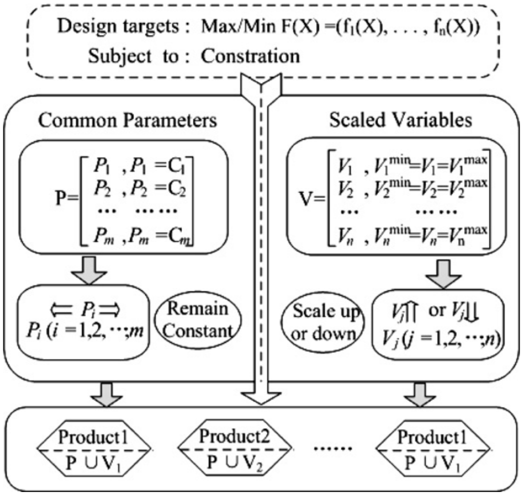


Fig. 2. Optimization model for scale-based product platform.

products with different performances. While the beachhead approach combines both of them which has the largest potential benefit, it is also the most difficult platform leveraging strategy to implement. Regardless of whether the platform is modular or scalable, the basic development strategy within any product family is to leverage the product platform across multiple market segments or niches.

2.2. Optimization model for scale-based product platform

Based on the platform vertical leveraging strategy, which variables should be made common in the platform and how to leverage the scaling variables need to be identified. The process model for scale-based product platform is illustrated in Fig. 2. The platform is composed of the common parameters set P and the scaling variables set V.

According to the design targets and constraints of the product family, while keeping the common parameters constant, the scaling variables are scaled up or down to form a series of derivative products. During the instantiation process, each scaling variable is restricted within its lower or upper end. The combination of platform common parameters and scaling variables must satisfy the design constraints of topological, geometric, performance etc. For a single derivative product i, design constants, platform common parameters and scaling variables constitute all its design variables. Furthermore, the product family is composed of the scale-based product platform and its derivative instances.

3. Two-stage multiobjective optimization-based platform design methodology

With regard to the optimization procedure, the family design problem can be classified in two different ways: one-stage and two-stage. [17] One-stage approaches seek to optimize the platform settings and the corresponding members of the family simultaneously while two-stage approaches optimize the platform first and then instantiate the individual products during the second stage. When the size of the product family and the number of design variables increase, the dimensionality of the optimization problems can become so high that the one-stage method cannot deal with the complexity and computational expense. As a result, two-stage approaches can provide an effective means to divide the task into two stages: platform configuration to decide which variables are shared and their settings, and instantiation to generate the optimal values for non-platform variables for all product variants. As is shown in Fig. 3, the proposed multiobjective optimization platform development methodology is carried out based on two-stage.

CI denotes the similarity factor of design variables among instance products and PI expresses the general performances of the product family. They are two competing objectives during the design of scale-based product platform.

The particular optimization steps of the TMOPDM are given next.

Step 1: Identify the design variables through product analysis.

Step 2: Perform DOE to check for possible reduction of design variables.

Step 3: Identify the constraints and multiobjectives for optimization.

Step 4: Make sample runs to determine best MOEA parameters for the problem and optimize every instance independently based on MOEA.

Step 5: Calculate mean (m_i) and standard deviation (δ_i) of design variables, then identify platform common parameter and scaling variable sets by $\delta_i\,/m_i$.

Step 6: Set platform common parameters as m_i., identify the constraints and design variables.

Step 7: Optimization to derive products based on platform common parameters using MOEA.

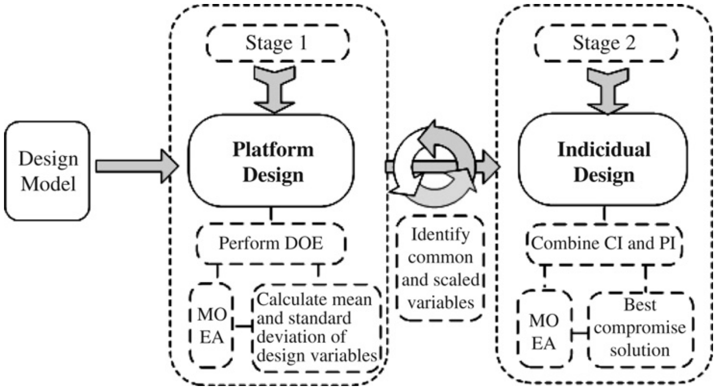


Fig. 3. Optimization framework of TMOPDM.

Steps 1 to 3 describe the construction procedure for the product optimization model used by TMOPDM. Screening designs such as fractional factorial design [18] and robust design [19] can help to reduce the design variables in Step 2. The product platform is established by TMOPDM during Steps 4 to Step 6. Afterwards, one reduced design model can be formulated for each derivative product to optimize the non-platform variables (i.e. scaling variables), while the values of common platform variables are set as constants in Step 7. Adopting the same optimization objectives and constraints, this process is similar with the multiobjective optimization in Step 4, but differs in a smaller scale. It is also referred to as the instantiation of the product platform to generate individual products within the family. The variable commonality MOEA is constructed such that it searches across varying levels of commonality for the platform while trying to resolve the tradeoff between commonality and individual product performance in the product family.

4. Multiobjective optimization based on genetic algorithm

4.1. Multiobjective optimization problem

The multiobjective optimization problem is a problem of minimization or maximization of multiple evaluation criteria that conflict with each other. In this paper, without the loss of generality, we consider the following multiobjective optimization problem (MOP) in the continuous search space. Mathematically, the product family design problem with p product, m design vectors and n objectives can be formulated as follows:

$$\begin{aligned}
&\text{Find } X = \{x_1^1, \ldots, x_1^p, x_2^1, \ldots, x_2^p, x_m^1, \ldots, x_m^p\}\\
&\text{minimize/maximize } F(X) = (f_1(X), \ldots, f_n(X))\\
&\text{subject to}\\
&g_j(X) \le 0, \quad j = 1; 2; \ldots; J;\\
&g_k(X) = 0, \quad k = 1; 2; \ldots; K;\\
&X_L \le X \le X_U, X = [X^d, X^c]^{\mathrm{T}}
\end{aligned} \tag{4.1}$$

where X is the vector of m controllable design variables, including possible platform settings and non-platform variables, $f_i(X)$ is the corresponding objective function, J and K are the number of inequality and equality constraints, respectively. The design space for some design variables involves both discrete and continuous variables (X^d and X^c). Contrary to single-objective problems, multiobjective problems have a set of alternative solutions. Because the multiple criteria have trade-off relationships with each other, it is impossible to determine objectively among such solutions which one is the best. Such solutions are called Pareto, nondominated, or efficient solutions. Once the set of Pareto-optimal solutions is identified, the designers can choose the best overall optimum design scheme based on the requirement and their experience. Praiseworthily, many genetic algorithms (GAs) have successfully been employed to tackle MOOPs over the past decade [20,21].

4.2. Application of NSGA-II to the multiobjective optimization of product platform

The nondominated sorting genetic algorithm II (NSGA-II) is a second generation MOEA, which uses fast nondominated sorting mechanism with $O(MN^2)$ complexity (M is the number of optimization objectives and N is the population size), elitism preservation strategy and crowding distance without external parameters to achieve the Pareto set for multiobjective optimizations. It has been applied to many optimizations in engineering due to its computational efficiency and stability.

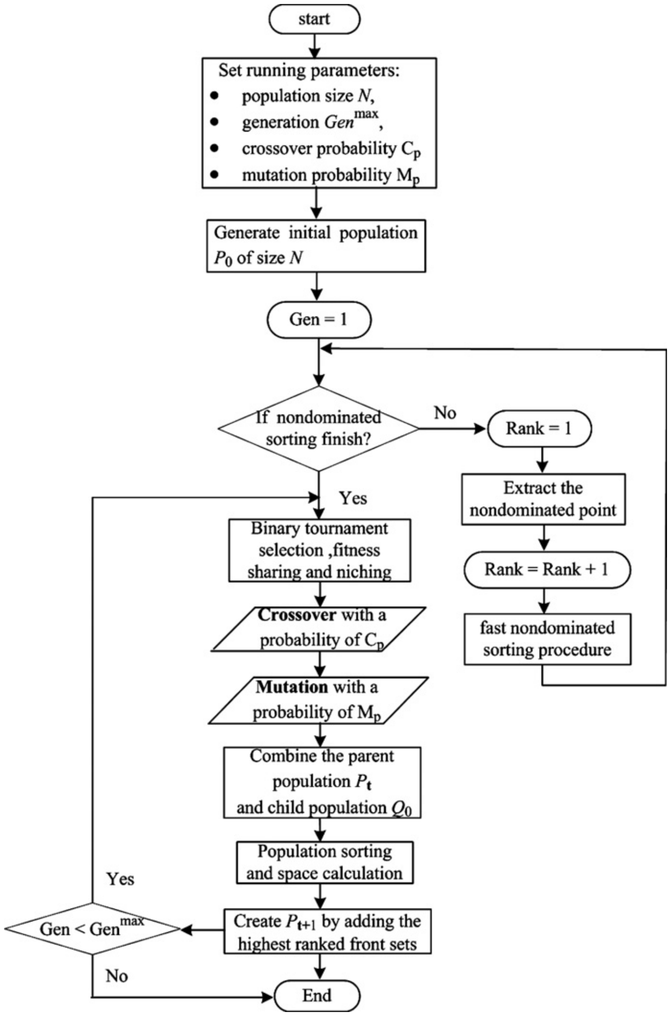


Fig. 4. Implementation framework of NSGA-II to the multiobjective optimization of product platform.

The implementation framework of the algorithm is illustrated in Fig. 4. The concept of Pareto-dominance is used in NSGA-II to assign fitness values to the sampling designs.

5. Case study

5.1. Optimization model of traction machine platforms

The design of a series of traction machines, which is the core component of elevator products [22] differ in output carrying capacity. The design variables include the velocity of elevator, moving wheels' diameter, electromotor rotation speed, gearbox ratios, steel cable diameter, worm teeth number, groove number etc. The traction machine platform is a typical scale-based product platform in which the derivative traction machine can be achieved by adjusting gearbox ratios and moving wheels' diameter. Its optimization objective is to design a family of traction machines that satisfies a variety of carrying capacity requirements with higher efficiency and lower input power by scaling a common traction machine platform around some scaling variables of the traction machine. The optimization model of traction machine platforms is illustrated as:

To optimize $F = (\mathrm{Min}(\mathrm{inputpower}), \mathrm{Max}(\mathrm{Efficiency}))$

$$\mathrm{Min}\, P = \frac{\pi(1-\varphi)Q}{58360iC}\frac{nD(100-4.2\sqrt{i})}{i(100-k_1\sqrt{i})} \tag{5.1}$$

$$\mathrm{MAX}\, \eta = \frac{100-k_1\sqrt{i}}{106}. \tag{5.2}$$

Table 1
Conditions of design variable.

Item	Conditions of design variable
Electromotor rotation speed n	$1000 \text{ r/m} \le n \le 1400 \text{ r/m}$
Moving wheels' diameter D	$500 \text{ mm} \le D \le 1100 \text{ mm}$
Gearbox ratios i	$20 \le i \le 63$
Velocity of elevator v	$0.8 \le v \le 2.6 \text{ m/s}$
Steel cable diameter d	$d = \{8, 10, 11, 13, 16, 19, 22\}$
Worm teeth number z_2	$25 \le z_2 \le 100$
Balance variable ϕ	$0.4 \le \phi \le 0.6$
Groove number n_g	$4 \le n_g \le 7$

Subject to constraints:

$$M_d = \frac{39\pi(1-\varphi)Q}{2400iC}\frac{nD(100-4.2\sqrt{i})}{100i_{12}} \tag{5.3}$$

$$M_d \le 2100 \text{ N m} \tag{5.4}$$

$$i = \frac{z_2}{z_1} \tag{5.5}$$

$$D \ge \left(\le \frac{2.65Q}{i_1 n_g d} + \frac{\pi Hq}{4}\right)\frac{8\cos\gamma q(1+v)}{(\pi - 2\gamma - \sin\gamma)(12.5+4v)} \tag{5.6}$$

$$35 \le \frac{D}{d} \le 50 \tag{5.7}$$

where P is input power, η is the efficiency, M_d is braking torque, n is electromotor rotation speed, D is moving wheels' diameter, z_1 is worm gear number, z_2 is worm teeth number, i is gearbox ratios, v is the velocity of elevator, d is steel cable diameter, k_1 is geared characteristic coefficient, Q is output carrying capacity, γ is cut angle of traction wheel, n_g is groove number of traction wheel, H is the height of cabin, q is the unit weight of cable. The terminal voltage V_t is fixed at 220 V corresponding to standard household voltage, and cut angle of traction wheel γ is set to 42.50°, which helps to reduce the manufacturing and inventory cost. Worm gear number z_1 is set to 2, geared characteristic coefficient k_1 is set to 2.4 based on the common column worm gear pair, output carrying capacity Q is set within 400, 800, 1000, 1500, 2000, 3000 kg. The constraint conditions of each design variable are shown in Table 1.

5.2. Traction machine platform optimization by TMOPDM

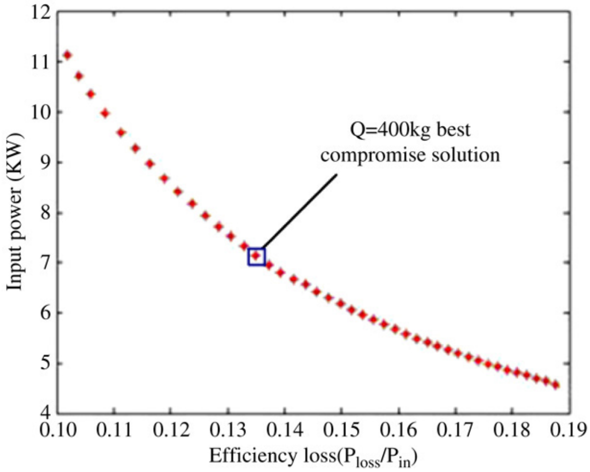
Based on the traction machine optimization model discussed above, NSGA-II is used to optimize every product in the traction machine series. The maximal efficiency ($\eta = P_{\text{out}}/P_{\text{in}}$) is transferred into the minimal efficiency loss ($\eta_{\text{loss}} = 1-\eta = P_{\text{loss}}/P_{\text{in}}$) to conform to the minimal input power objective. The population size of NSGA-II is 100 and the crossover and mutation probability are 0.9 and 0.05, respectively. Their distribution indexes are both 20. The maximum generation is 500.

All the experiments were run on a Pentium 4 (3.0 GHz) with 1 GB RAM using GNU/Linux (taking a CPU time of about 426 s). The results are shown in Fig. 5. Making two competing objectives input power vertically and efficiency loss horizontally, the Pareto-optimal front of a traction machine with different output power are listed in Fig. 5(a) (f). Due to the imprecise nature of human decisions, we adopt a fuzzy-based mechanism [19] which extracts a Pareto-optimal solution as the best compromise one. As illustrated in Fig. 5(a), by setting the lower and upper bounds of input power and efficiency loss as [4, 12] and [0.10, 0.19], respectively, the solutions in Pareto-optimal set are arranged according to their membership function, the best design solution for 400 kg traction machine takes the input power 7 kW and efficiency 0.865.

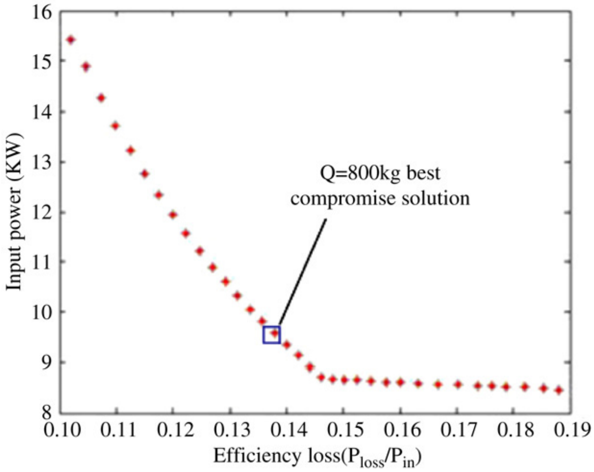
Based on the acquired traction machine design variables, the parameter's mean (m) and standard deviations (δ) are listed in Table 2 with the change rate ($\delta/m\%$). If the standard deviation of a design variable is found to be very small relative to its mean value, it indicates that this parameter has very little contribution to achieving the range of performance, and it is then taken as a common platform parameter, while, the set of design parameters that has significant variations in the result are used as the non-platform variables (or scale factors) in the second stage of the TMOPDM. As shown by the sign A, we consider the design variables whose change rate was less than 10% as platform common parameters for this problem. The performance parameters of the product family is shown in Table 2

The next stage of the TMOPDM is to instantiate the individual traction machines of the product family. While the common platform variables determined from stage I are fixed as constant parameters, the to-be-identified variables are five non-platform variables. According to the small-scale derivative traction machine optimization model, the achieved scaling variables and the resulting performances are listed in Table 3.

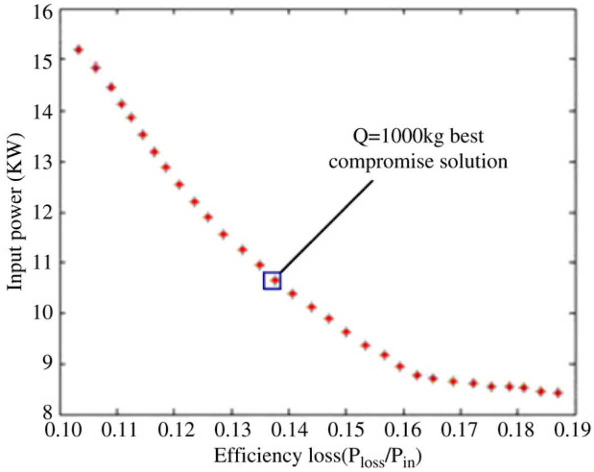
As is shown, with the output carrying capacity improving in the traction machine series, moving wheels' diameter, gearbox ratios, velocity of elevator, electromotor rotation speed and gearbox ratios decrease or increase, respectively. As a result, traction machines which differ in output carrying capacity can be designed through changing the five non-platform



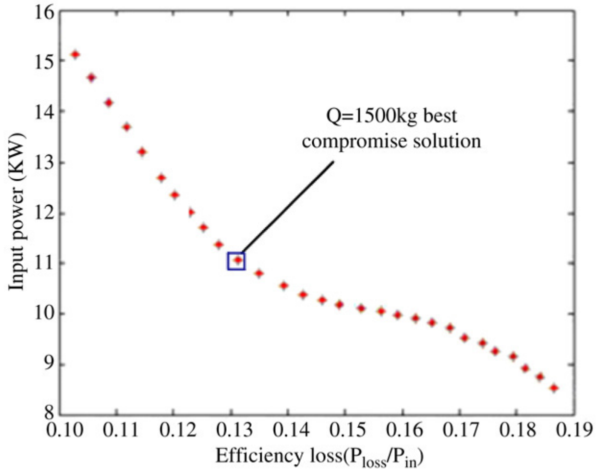
(a) $Q = 400$ kg.



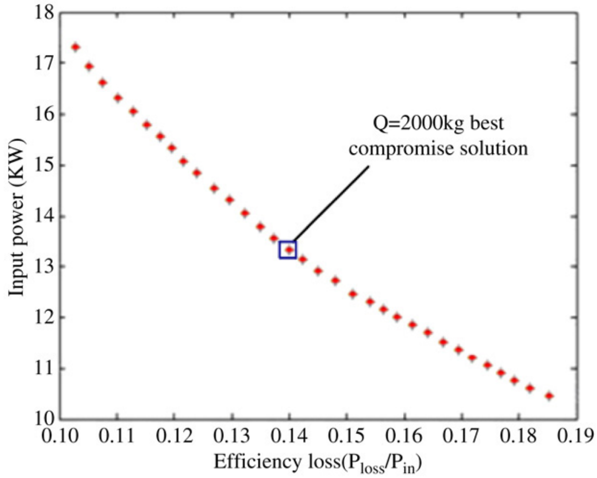
(b) $Q = 800$ kg.



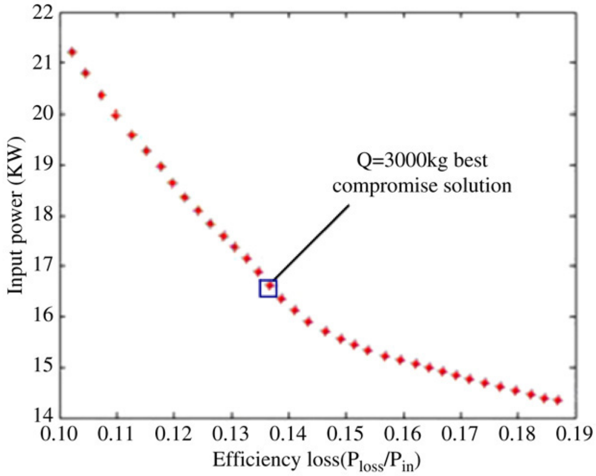
(c) $Q = 1000$ kg.



(a) $Q = 1500$ kg.



(b) $Q = 2000$ kg.



(c) $Q = 3000$ kg.

Fig. 5. Pareto-optimal front and best compromise solution for each traction machine.

Table 2
The mean, standard deviation and change rate of traction machine design variables.

Parameters	Mean m	Standard deviation δ	Change ratio δ/m (%)	Common parameter
D	806	208	25.8	
n	1250	135	10.8	
d	15	1.3	8.66	A
d_2	65	12.6	19.38	
i	36.56	9.97	27.27	
v	1.83	0.39	21.31	
ϕ	0.55	0.04	7.27	A
n_g	5	0.25	5	A

Table 3
Scaling variables and performances of derivative traction machine.

Scaling parameters					Performances		
D	n	z_2	i	v	Q (kg)	P (kW)	η
556	1120	49	24.5	1.42	400	7.1	0.847
712	1220	60	30.0	1.86	800	9.7	0.882
754	1250	73	36.5	2.17	1000	10.5	0.892
806	1320	80	40.0	1.83	1500	11.2	0.896
912	1340	89	44.5	1.93	2000	13.4	0.890
1052	1370	97	48.5	2.06	3000	16.2	0.862

Table 4
Comparison of performances among the design of traction machine by TMOPDM, benchmark groups and VBPDM.

TMOPDM		Benchmark		VBPDM		TMOPDM vs benchmark		TMOPDM vs VBPDM	
P (kW)	η	P (kW)	η	P (kW)	η	P (%)	η (%)	P (%)	η (%)
7.1	0.847	7.31	0.821	7.38	0.816	−2.74	+3.17	−3.79	+3.80
9.7	0.882	9.67	0.869	9.81	0.873	+0.31	+1.47	−1.12	+1.03
10.5	0.892	10.76	0.873	10.67	0.865	−1.49	+2.18	−1.59	+3.12
11.2	0.896	11.53	0.886	11.68	0.882	−2.86	+1.13	−1.24	+1.59
13.6	0.890	14.01	0.876	14.76	0.879	−2.93	+1.60	−1.08	+1.25
16.2	0.862	16.47	0.857	16.52	0.852	−1.64	+0.58	−1.94	+1.17
Average time (s)						Average performance			
412.36		535.47		451.79		−1.89	+1.71	−1.79	+1.99

variables while keeping the platform common parameters constant. As a result, the manufacturing cost will be reduced and the productivity will be improved based on the optimized scale-based traction machine platform.

5.3. *Comparison of the TMOPDM against previous approaches*

The traction machine platform developed by TMOPDM is compared to the benchmark group optimizing and the platform according to VBPDM. All the traction machine families meet their design and performance constraints, but differ in their response for input power, efficiency and computational complexity. Table 4 lists the percentage difference of input power and efficiency from the TMOPDM to benchmark and the TMOPDM to VBPDM. We can summarize from Table 4 that the product family obtained by TMOPDM is slightly better than the benchmark group with average lower input power (−1.89%) and slightly higher efficiency (+1.71%) besides the improvement of computational complexity. It is also better than the VBPDM in terms of achieving lower input power (−1.79%) and higher efficiency (+1.99%) with the same commonality level

6. Conclusion

The success of the resulting product family often relies on properly resolving the tradeoff between increasing commonality across the family and performance loss compared to individual design. In this paper, we describe the optimization model of a scale-based product platform and present a two-stage multiobjective optimization-based platform design methodology (TMOPDM), which aims to satisfy a range of performance requirements using the smallest variation of the product designs in the family, while multiobjective genetic algorithms are applied to the product platform optimization in two stages respectively to generate uniformly Pareto optimal set in the design space with better performances. Finally, the design of a family of traction machines is used as an example to compare the proposed method to VBPDM and benchmark groups. Results also show that the commonality-performance Pareto front contains solutions with generalized commonality and better performances.

Acknowledgements

This work was supported by the National Natural Science Foundation of China. The authors thank the reviewers of this paper for their helpful comments.

References

[1] M.H. Meyer, A.P. Lehnerd, The Power of Product Platforms: Building Value and Cost Leadership, Free Press, New York, 1997.
[2] T.W. Simpson, R.A. Maier, F. Mistree, Product platform design: Method and application, Research in Engineering Design 13 (2001) 2–22.
[3] C. Conner, D.J. Kroon, F. Mistree, A product variety tradeoff evaluation method for a family of cordless drill transmissions, in: 1999 ASME Design Technical Conference, paper no. DETC99/DAC-8625, Las Vegas, Nevada.
[4] A. Messac, M.P. Martinez, T.W. Simpson, Effective product family design using physical programming, Engineering Optimization 34 (2002) 245–261.
[5] R.U. Nayak, W. Chen, T.W. Simpson, A variation-based method for product family design, Engineering Optimization 34 (2002) 65–81.
[6] G. Hernandez, T.W. Simpson, J.K. Allen, Robust design of families of products with production modeling and evaluation, Journal of Mechanical Design 123 (2001) 183–190.
[7] Z.H. Dai, J.S Michael, Effective product family design using preference aggregation, Intelligent Manufacturing 18 (2007) 97–113.
[8] M.H. Wang, A systematic approach for the robust design of scalable product platforms, Engineering Manufacture 220 (2006) 1983–1995.
[9] B.S. D'Souza, T.W. Simpson, A genetic algorithm based method for product family design optimization, in: The ASME Design Engineering Technical Conference, vol. 2, 2002, pp. 681–690.
[10] C.A. Coello, D.A. Van Veldhuizen, G.B. Lamont, Evolutionary Algorithms for Solving Multiobjective Problems, Kluwer Academic Publishers, Boston, MA, 2002.
[11] E. Zitzler, L. Thiele, An evolutionary algorithm for multiobjective optimization, the Strength Pareto Approach. TIK-Report 43, Swiss Federal Institute of Technology, 1998.
[12] J. Horn, N. Nafpliotis, A niched Pareto genetic algorithm for multiobjective optimization, in: Frist IEEE Conference on Evolutionary Computation, 1994, pp. 82–87.
[13] N. Srivans, K. Deb, Multiobjective optimization using nondominated sorting in genetic algorithms, Evolutionary Computation 2 (1995) 221–248.
[14] L. Manuel, P.T. Devi, B. Paechter, Multi-objective optimisation of the pump scheduling problem using SPEA2, in: 2005 IEEE Congress on Evolutionary Computation, 2005, pp. 435-442.
[15] M. Kim, T. Hiroyasu, M. Miki, SPEA2+: Improving the performance of the strength pareto evolutionary algorithm, Parallel Problem Solving from Nature (2004) 742–751.
[16] K. Deb, A fast and elitist multiobjective genetic algorithm: NSGA-II, IEEE Transactions on Evolutionary Computation 6 (2002) 182–197.
[17] T.W. Simpson, Product platform design and optimization: Status and promise, Artificial Intelligence for Engineering Design 18 (2004) 3–20.
[18] D.C. Montgomery, Design and Analysis of Experiments, John Wiley & Sons, New York, 1997.
[19] T.W. Simpson, W. Chen, Use of the robust concept exploration method to facilitate the design of a family of products, simultaneous engineering, Methodologies and Applications (1999) 247–278.
[20] K. Chakraborthy, S.B. Vinay, R. Nair, Rolling element bearing design through genetic algorithms, Engineering Optimization 35 (2003) 649–659.
[21] K. Mitra, R. Gopinath, Multiobjective optimization of an industrial grinding operation using elitist nondominated sorting genetic algorithm, Chemical Engineering Science 59 (2004) 385–396.
[22] C.L. Yang, Design of Traction Machine in Elevator, Mechanism Industry Press, Beijing, 2004.