Struct Multidisc Optim (2012) 45:467–478
DOI 10.1007/s00158-011-0735-7

RESEARCH PAPER

A method for developing systems that traverse the Pareto frontiers of multiple system concepts through modularity

P. K. Lewis · C. A. Mattson

Received: 10 May 2011 / Revised: 21 October 2011 / Accepted: 23 October 2011 / Published online: 16 December 2011

Abstract Natural changes in customer needs over time often necessitate the development of new systems that satisfy the new needs. In a previous work by the authors, a 5-step multiobjective optimization-based method was presented to identify systems that anticipate, account for, and allow for these changes by moving from one Pareto design to another through module addition. Recognizing the potential for changes in needs to exceed the limits of a single Pareto frontier, the present paper introduces important advancements that extend development to modules connecting multiple disparate system concepts. As such, the search for suitable system designs is extended from a Pareto frontier that characterizes one system concept to a Pareto frontier that characterizes a set of system concepts. An expanded methodology is described, and a tri-objective hurricane and flood resistant residential structure example is used to demonstrate the method. The authors conclude that the developed method provides a new methodology for selecting platform and module designs in the presence of multiple system concepts, and is capable of identifying a set of modular system designs that are well-suited to satisfy changing needs over time.

Keywords Multi-objective optimization · Pareto frontier traversing · Changing needs · Modular system design · s-Pareto frontier · Hurricane and flood resistant structures

P. K. Lewis · C. A. Mattson (✉)
Department of Mechanical Engineering, Brigham Young University, Provo, UT 84602, USA
e-mail: mattson@byu.edu

Nomenclature

δ	Matrix dictating the desired progression that each module provides.
D_{a}	Set containing all design variable values of x_{a} and x_{p}.
D_{m}	Set containing all design variable values of x_{m} and x_{p}.
g	Vector of inequality constraints.
h	Vector of equality constraints.
J	Aggregate objective function.
μ	Vector of design objectives.
n_{d}	Number of designs comprising the adaptive design set.
$n_{\hat{\mu}}$	Number of additional objective constraints needed to define anticipated regions of interest.
$P^{(\alpha)}$	Vector of design objective values of the base design of a module.
$P^{(\beta)}$	Vector of design objective values of the target design of a module.
$\bar{P}^{(i)}$	Vector of design objective values of a design when used with the i-th module.
$\Delta P^{(i)}$	Vector of the change in design objective values from the base design to $\bar{P}^{(i)}$.
p	Vector of design parameters.
x	Vector of design variables.
x_{a}	Vector of non-platform adjustable design variables.
x_{m}	Vector of non-platform design variables that characterize the design of modules.
x_{p}	Vector of platform design variables.

Subscripts, superscripts, and other indicators

$[\,]^{(i)}$	indicates current design/module.

$[\,]^{(k)}$ indicates current design concept.
$n_{[\,]}$ indicates the number of [ ].
$[\,]_l$ indicates the lower limit of [ ].
$[\,]_u$ indicates the upper limit of [ ].
$[\,]^*$ indicates the optimal value of [ ].

1 Introduction & background

The needed performance of a system tends to naturally change over time. When these changing needs would result in the future selection of designs that are beyond the feasible capabilities of a *single* system or concept, additional systems/concepts are often developed. Additionally, the need for user-level adaptation and expandability of systems can result from factors such as high system purchase costs and frequently occurring changes in needs (Li et al. 2008). Choosing whether to develop new systems as needs change, or to develop systems that anticipate need changes is a multifaceted decision.

One approach for achieving the needed system adaptability/expandability is through the development of reconfigurable modular systems (Ferguson et al. 2007, 2009). Although it can be difficult to develop modular systems that anticipate, account for, and allow for natural changes in needs to be met through the addition of performance changing modules, such systems capitalize on system commonality to reduce production costs, and cater to customization and adaptation (Simpson 2004; Tseng et al. 1996).

Based on these benefits of modular systems, the objective of this paper is to develop a method that identifies system designs that are high performing *and* well suited to facilitate system reconfigurability through module addition. In a previous work by the authors, a 5-step multiobjective optimization-based method was presented in response to this objective (Lewis et al. 2011). However, the formulations of this previous method are limited to developing modular systems capable of targeting selected points along a single Pareto frontier, which is the collection of non-dominated designs from a single system concept.

In this paper, an extended method that facilitates situations where changes in needs exceed the scope of a single Pareto frontier is introduced in this paper. This is the case when changing needs are best met by a *set* of disparate concepts, each one having its own Pareto frontier. The extended method presented herein expands the optimization formulations to include multiple disparate system concepts. Thus allowing the search for desirable system designs to expand from the Pareto frontier to a collection of non-dominated designs from a set of disparate system concepts (Mattson and Messac 2005).

1.1 Product family and modular system design methods

The approach used in this paper to identify platform system elements/variables is similar to those used in product family design approaches. Two common goals of product family design are to identify product platforms that maximize both variable commonality, and performance diversity in the identified product family (de Weck et al. 2003; Tseng and Jiao 1998). In contrast, the presented method seeks to identify platforms that facilitate modularity, even if variable commonality is not maximized. Additionally, the presented method also seeks to drive module-enabled performance as close to targeted future performance needs as possible, without regard to how diverse those future needs are.

It is important to observe that the presented method can be viewed as a type of product family design method. However, due to the differences in objectives, traditional product family design and the method presented herein will likely result in different platform and/or module designs. To illustrate this point, consider the development of treadle irrigation pumps for developing countries. Under a traditional product family design approach, a set of products is identified that maximize the performance diversity and variable commonality of the overall set. In addition, each product in the set is strategically designed to meet the needs of a particular market segment, while being built on a common platform to capitalize on economies of production scale. In contrast, the methodology presented in this paper identifies a single pump—not a set of pumps. The single pump identified is chosen based on its ability to transform from one performance state to another through the addition of modules, regardless of how diverse the desired performance states are. The objective of the method presented in this paper is to identify designs that are capable of transforming from one optimal state to another.

In the development of modular systems, the literature identifies four main architecture types: (i) slot modular (Ulrich and Eppinger 2004); (ii) sectional modular (Ulrich and Eppinger 2004); (iii) bus modular (Ulrich and Eppinger 2004); and (iv) type II modular (Strong et al. 2003). As described in Lewis et al. (2011), the presented method allows designers to leverage these existing developments in modular system classifications. In addition, research in the areas of adaptive systems (Khire and Messac 2008), flexible systems (Olewnik et al. 2004), and reconfigurable systems (Siddiqi and de Weck 2008) can also be used to assist in the identification of platform and module design concepts.

1.2 Multiobjective optimization

Fundamental within the methodology presented in this paper is the need to characterize/balance competing objectives or goals, where the potentially competing nature of

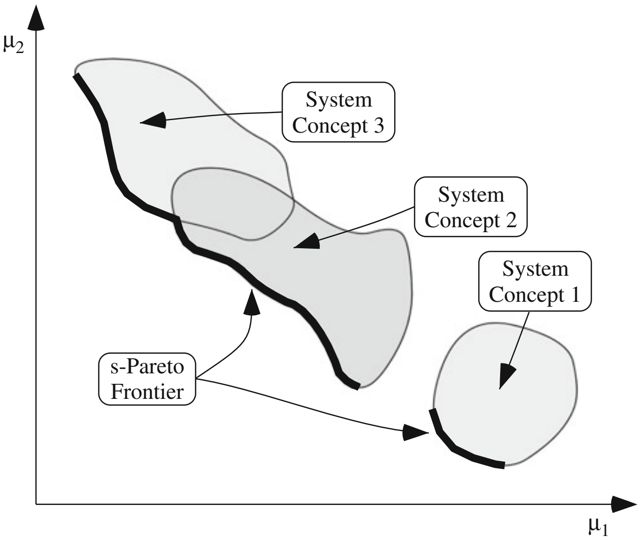

Fig. 1 A feasible design space (*shaded*) for objectives μ_1 and μ_2. The s-Pareto frontier (*bold line*) represents the set of non-dominated solutions in the feasible space for this bi-objective minimization-minimization problem with three possible system concepts

present and future system needs are one example. Multiobjective optimization is a well-known, well-accepted, means to quantify tradeoffs between competing design objectives (Frischknecht et al. 2011; Kasprzak and Lewis 2000; Messac 1996), and as such forms a fundamental building block for the methodology presented in Section 2 of this paper. One pertinent application of multiobjective optimization in the context of this paper is that of identifying a set of non-dominated designs—Pareto frontier (Gurnani and Lewis 2008; Messac and Mattson 2004; Todoroki and Sekishiro 2008). Figure 1 illustrates the meaning of non-dominance in the presence of multiple system concepts (bold line) for the minimization of two objectives (μ_1 and μ_2). The feasible regions of each system concept are also illustrated in the figure as shaded regions. In Mattson and Messac (2003), the term *s-Pareto* was adopted to refer to the combined Pareto frontier resulting from a *set* of disparate concepts. As can be seen in the figure, the s-Pareto frontier comprises all non-dominated designs because there are no other feasible designs from any other concept that are better in all objectives.

The s-Pareto frontier can be obtained numerically using the following generic multiobjective optimization problem (MOP). The formulation yields a set of optimal solutions, $D := \{(x_1^{(k)*}, x_2^{(k)*}, ..., x_{n_x^{(k)}}^{(k)*})\}$, that belong to the s-Pareto frontier.

Problem 1: Generating s-Pareto solutions

$$\min_{k} \left\{ \min_{x^{(k)}} \left\{ \mu_1^{(k)}\left(x^{(k)}\right), \ldots, \mu_{n_\mu}^{(k)}\left(x^{(k)}\right) \right\} \quad (n_\mu \geq 2) \right\} \quad (1)$$

Subject to $g_q^{(k)}(x^{(k)}, p^{(k)}) \leq 0,\ q \in \{1, 2, ..., n_g^{(k)}\}$; $h_v^{(k)}(x^{(k)}, p^{(k)}) = 0,\ v \in \{1, 2, ..., n_h^{(k)}\}$; and $x_{jl}^{(k)} \leq x_j^{(k)} \leq x_{ju}^{(k)}$. Where k denotes the k-th system concept; $\mu_i^{(k)}$ denotes the i-th generic design objective; $x^{(k)}$ is a vector of design variables for the k-th system concept; and $p^{(k)}$ is a vector of design parameters for the k-th system concept. Note that this MOP yields a set of s-Pareto solutions.

Within the literature, s-Pareto identification methods include directly evaluating the said MOP, Pareto filters that find s-Pareto solutions among sets of Pareto optimal solutions (Di Barba 2001; Mattson et al. 2004), eliminating non-Pareto and locally Pareto solutions with Pareto filters (Cheng and Li 1998; Mattson and Messac 2003; Messac and Mattson 2004), and combinations of these methods. In particular, the method presented in Mattson and Messac (2003) for generating an s-Pareto frontier by reducing the Pareto frontiers from disparate system concepts into a single s-Pareto frontier has direct application to the balancing of the tradeoffs of a set of system concepts needed within the proposed method.

In the context of this paper, another pertinent application of multiobjective optimization is that of identifying a single solution from among the s-Pareto solutions. The decision of which Pareto-optimal solution is to be used requires that changes in objective function parameters, and sometimes constraints, over time be included in the multiobjective method implemented. This is accomplished by evaluating an optimization formulation that selects system designs within a series of anticipated regions of interest representing system needs or preferences for different instances in time. As such, the presented method expands upon existing optimization methods by making selections based on the solution's ability to (i) facilitate development of a module-based system, and (ii) satisfy known changes in needs over time through expandability/adaptability.

The remainder of this paper presents the theoretical development of the proposed method extensions in Section 2. In Section 3 a tri-objective hurricane and flood resistant residential structure example is used to demonstrate the method. Concluding remarks are provided in Section 4.

2 Modular system design method development

This section provides a methodology for identifying system designs that can move from one optimal position on the s-Pareto frontier (Mattson and Messac 2003) to another through module addition. To satisfy changing needs over time through module addition requires s-Pareto designs to be strategically selected based on their ability to facilitate adaptability across disparate system concepts. Figure 2

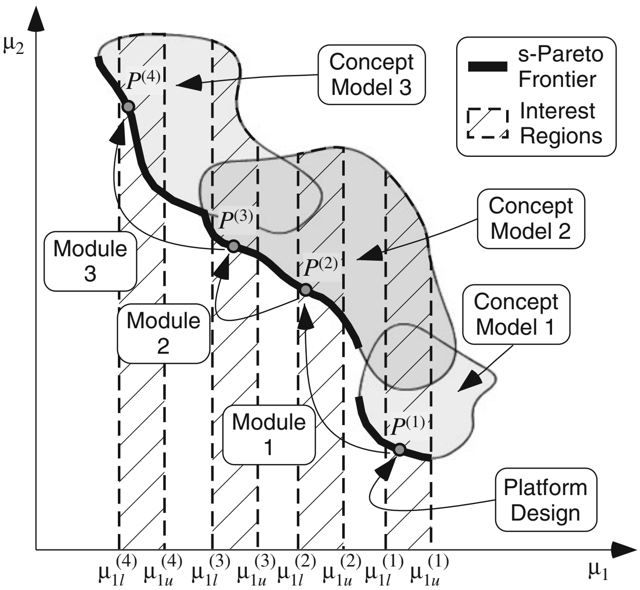

Fig. 2 Graphical representation of the intent of the proposed expanded method to provide a system that expands from one *s-Pareto-optimal* design to another through the addition of modules. Also notice that the designs that it can adapt to are within designer defined regions of interest

illustrates the intent of the method to select an s-Pareto optimal platform design that, through the addition of modules, becomes other s-Pareto designs. For example, the figure shows that the platform design, labeled $P^{(1)}$, adapts to become $P^{(2)}$ through the addition of Module 1. In like manner, the subsequent target designs identified as $P^{(3)}$, and $P^{(4)}$ are also achieved through Modules 2 and 3 respectively.

In examining Fig. 2, it should be noted that the original method developments in Lewis et al. (2011) would be represented by $P^{(2)}$ being a platform design, and Module 2, which scales of the system design from $P^{(2)}$ to $P^{(3)}$ on the same frontier. As such, the identification of modules that span different concepts (Modules 1 and 3 in Fig. 2), and are therefore not simply scaling the system design, represent the resulting method extensions presented in this section.

Figure 3 provides a flow chart that represents the six primary steps of the expanded multiobjective optimization design method developed herein. Each of these steps is described below. It is important to note that the titles of steps A–C and E–F are similar to those of the original single-concept method presented in Lewis et al. (2011). However, with the exception of Step B, each of these steps requires new and essential extensions to enable the method to identify systems capable of traversing an s-Pareto frontier.

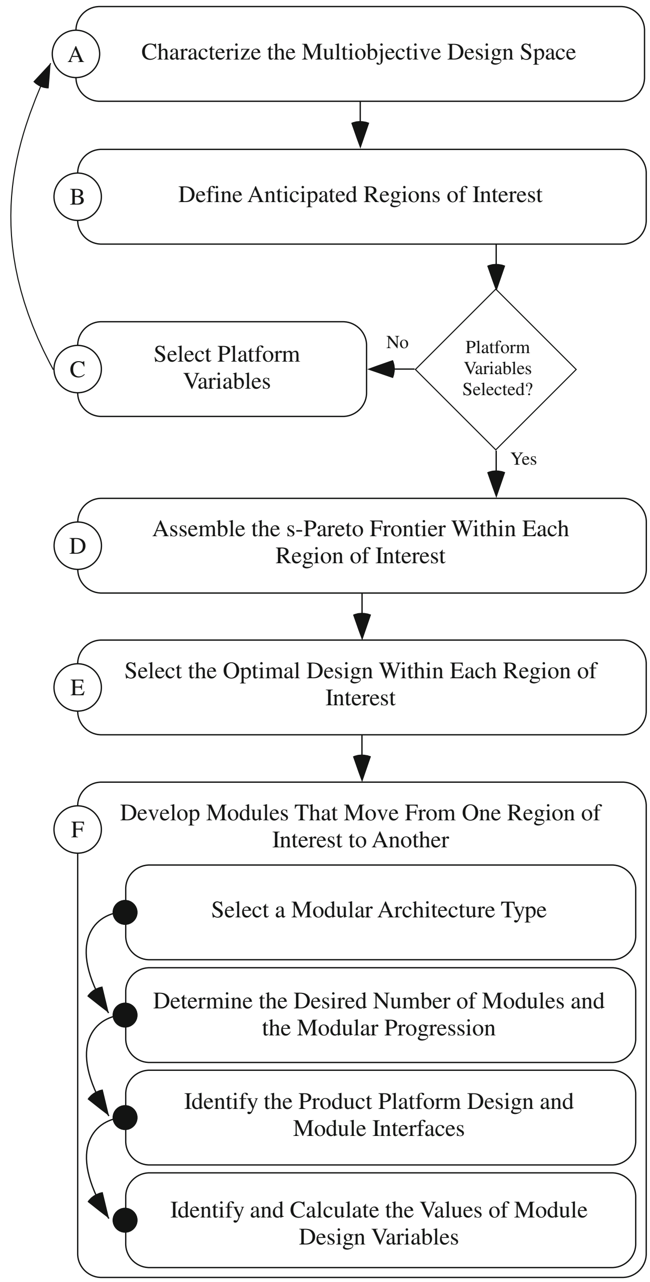

Fig. 3 Flow chart describing the six-step multiobjective optimization design method developed in this section

2.1 Characterize the multiobjective design space

The first step of the method is to explore the multiobjective design space to evaluate and characterize the effects of each design variable on the objective space performance. In addition, as seen in Fig. 2, when multiple system concepts are needed to satisfy the future system needs, the expanded function of this step of the method requires that a MOP for each system concept be evaluated.

2.2 Define anticipated regions of interest

A key idea of the presented method is that changes in the desired system performance are equivalent to changes in the desired values of one or more design objectives. To that end, the second step of the method captures the predicted changes in system needs over time, and represents them as *Anticipated Regions of Interest* of the multiobjective design space. These designer-specified regions enhance the ability of an optimizer to select the designs that are optimal for adaptation, by limiting the search to those regions. It is important to note that these regions represent predicted future needs regarding objective performance. To maintain simplicity in the graphical presentation of Fig. 2, we have shown regions of interest involving only one objective (μ_1). However, it is expected that regions of interested would be specified for as many of the objectives as the designer sees fit.

In considering the identification and characterization of future needs, it is observed that such needs fall into two classes. The first class involves needs that are known or reasonably predicted (e.g., cost reductions over time, availability of increased CPU speeds, better gas mileage). The second class involves needs that are not yet known or not reasonably predicted. Note that this paper does not attempt to address the complexities of identifying this second class of unknown needs. However, some potential methods that could be used in identifying/characterizing future needs are: (i) to identify external drivers of change and then identify components that are likely to change in response (Martin and Ishii 2002), and (ii) to gather information by adapting common methods for identifying current needs (e.g. focus groups, surveys, observation). In terms of the method presented herein, it will be assumed that future needs are known or can be reliably characterized as anticipated regions of interest.

2.3 Select platform variables

The third step of the method uses the disparate Pareto frontiers within the anticipated regions of interest to identify the design variables that are best suited as platform variables (x_{p}). In addition, by selecting platform variables, it is likely that the Pareto frontier of the system concepts will shift. To ensure that the resulting shift has not placed an anticipated region of interest in what is now infeasible space, steps A and B of the method must be repeated as shown in Fig. 3.

Identifying platform variables across multiple concept models can be noticeably more difficult than doing the same for a single concept model. The following guidelines for selection can ease this difficulty: (i) platform variables should be common across all concept models, (ii) platform variables should be related to concept features that do not need to change over time, and (iii) platform variables should cause minimal variation along the Pareto frontiers within many (if not all) anticipated regions of interest. As described in Yearsley and Mattson (2008), one approach to identify minimal variation is to calculate the standard deviation of each common design variable for all concept-specific Pareto points within the regions of interest. However, any other suitable method may be used.

It is observed that as the number of concepts and regions of interest increases, the likelihood of identifying a suitable platform decreases. This is due to the reduced likelihood of identifying system concepts that all lend themselves to facilitating modularity across the identified regions of interest. As such, the ability of the method to provide good solutions is dependent on: (i) the designers selection of concepts; and (ii) the level to which the concepts lend themselves to modularity. In situations where a good platform cannot be identified for a given set of system concepts and regions of interest, the designer should ask the following question: In order to progress the design, is there a subset of concepts that a platform can be identified for?

If the answer to this question is yes, then the designer has two options: (i) Use the identified subset of concepts and continue with the method (subset must contain solutions in all identified regions of interest); or (ii) Alter/replace any concept(s) to incorporate the identified platform or a different platform. If the answer to this question is no, then there is no modular design that can satisfy the identified needs. As such, the designer has two options: (i) Refine the regions of interest; and/or (ii) Alter/replace the considered concepts to enable the identification of a common platform.

2.4 Assemble the s-Pareto frontier within each region of interest

The fourth step of the method identifies the platform-constrained s-Pareto optimal solutions from the various system concepts within each anticipated region of interest. Notice that because the platform-constrained Pareto frontier of each system concept was obtained in the previous step of the method, the current step may be easily accomplished through the use of Pareto-filtering methods as described in Section 1.2. Alternatively, the s-Pareto frontier can be generated directly based on the chosen platform variables and regions of interest using (1). For additional details regarding s-Pareto frontier generation, such as how to handle concept-specific objectives, we refer the interested reader to Mattson and Messac (2003).

2.5 Select the optimal design within each region of interest

The fifth step of the method implements an optimization routine to select one design within each region of interest that will be used in the final step of the method as targets for the platform and module combinations developed. The resulting optimal design set $D_a := \{(x_p^*, x_a^{(i)*}) \mid \forall i \in \{1, ..., n_d\}\}$, containing all variable values of x_p^* and $x_a^{(i)*}$, is obtained through the following optimization formulation:

Problem 2a: s-Pareto optimal adaptive system identification

$$\min_{x_p, x_a^{(i)}} \left\{ \sum_{n=1}^{n_d} w^{(i)} J^{(i)} \left(x_a^{(i)}, x_p \right) \right\} \tag{2}$$

where:

$$J^{(i)} \left(x_a^{(i)}, x_p \right) = \min_k \left\{ \min_{x_a^{(k)},\ x_p} J^{(k)} \left(x_a^{(k)}, x_p \right) \right\} \tag{3}$$

subject to:

$$g_q^{(k)} \left(x_a^{(k)}, x_p, p^{(k)} \right) \le 0 \ \forall q \ \in \left\{ 1, ..., n_g^{(k)} \right\} \tag{4}$$

$$h_v^{(k)} \left(x_a^{(k)}, x_p, p^{(k)} \right) = 0 \ \forall v \ \in \left\{ 1, ..., n_h^{(k)} \right\} \tag{5}$$

$$x_{a,j,l} \le x_{a,j}^{(k)} \le x_{a,j,u} \ \forall j \ \in \left\{ 1, ..., n_{x_a}^{(k)} \right\} \tag{6}$$

$$x_{p,r,l} \le x_{p,r} \le x_{p,r,u} \ \forall r \ \in \left\{ 1, ..., n_{x_p} \right\} \tag{7}$$

$$\mu_{y,l}^{(k)} \le \mu_y^{(k)} \le \mu_{y,u}^{(k)} \ \forall y \ \in \left\{ 1, ..., n_{\hat{\mu}}^{(k)} \right\} \tag{8}$$

where the adjustable variables (x_a) represent all non-platform design variables (variables that are either scaled or discretely adjusted) for each system concept; k, $1 \le k \le n_c$, denotes the k-th system concept; $w^{(i)}$ are weights associated with the local preference within the i-th region of interest; $J^{(i)}$ and $J^{(k)}$ are aggregate objective functions for the i-th region of interest and j-th concept respectively; and the superscript (k) on p, g, and h indicate that parameters and constraints can be different (non-constant) for each system concept. It should be noted that the introduction of the superscript k in Problem 2a captures the extension of the presented method from a single system concept (see Lewis et al. 2011) to multiple concepts. As such, if $n_c = 1$, Problem 2a reduces to the original formulation presented in Lewis et al. (2011).

Figure 4 is a graphical representation of how the solution to Problem 2a for a set of three anticipated regions of interest and the corresponding s-Pareto frontiers are used to identify the values of x_p and $x_a^{(i)}$. In addition, Fig. 4 illustrates the outcome of evaluating Problem 2a, which is the identification of a single system design $(\mu_1^{(i)}, \mu_2^{(i)})$ within each region of interest defined by (8).

Considering the performance (in design objective space) of each design identified through Problem 2a, the final step of the method identifies the variables and values of the module specific design variables (x_m) that target the performance of the identified designs in D_a.

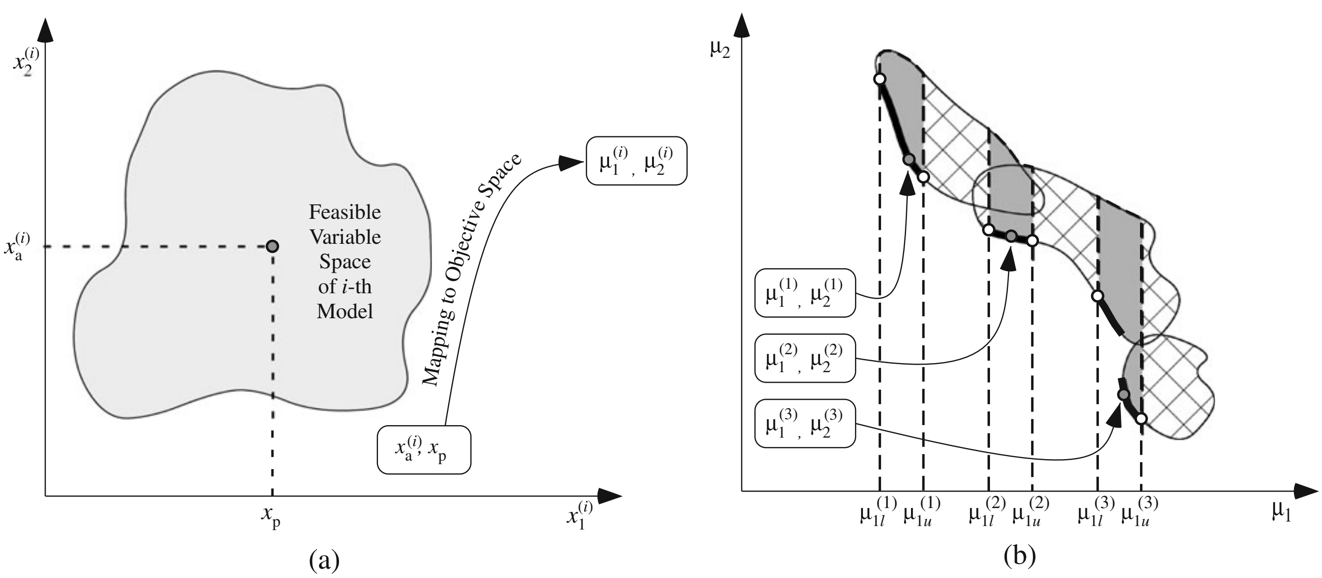

Fig. 4 Theoretical identification of the values of x_p and $x_a^{(i)}$ for a set of three anticipated regions of interest and s-Pareto frontier from the MOP formulation presented in Problem 2a

2.6 Develop modules that move from one region of interest to another

By this step in the process, the set D_a now contains all variable values that can be used to develop the module designs. Developing these designs is now a matter of constrained module design. To complete this final step of the method and obtain the module designs requires the following:

1. *Select a modular architecture type* Identify the desired functionality of the platform and modules as a whole, and select an applicable architecture (Otto and Wood 2001; Ulrich and Eppinger 2004).
2. *Identify the system platform design and module interfaces* The target system identified through Problem 2a with the most commonality to the other target designs is selected as the platform. Module interfaces are then defined based on the selected architecture type (i.e., begin developing concept module designs).
3. *Determine the desired number of modules and modular progression* Identify the number of modules to be developed, and the corresponding module progression matrix (δ) defining the intended module progression sequences (i.e., a module connecting $P^{(1)}$ and $P^{(2)}$ in Fig. 2 would correspond to a row entry of [1 2] in δ). Additional information on the form and definition of δ is provided in Section 3.2 of this paper in the context of an example. It should be noted that in situations where the desired modular progression bridges two or more different concepts, module concepts must now be developed that are capable of creating the needed connections between concepts.
4. *Identify and calculate the values of module design variables* Complete the development of all module concepts, and determine the optimal values of the module specific design variables ($x_m^{(i)*}$) that will enable the combined platform and modules to obtain the target system performances identified in Problem 2a.

For additional information on parts 1–4 see Lewis et al. (2011). To complete part 4, the set $D_m := \{x_p^*, x_m^{(i)*}) \mid \forall i \in \{1, ..., n_m\}\}$, containing all variable values of $x_m^{(i)*}$ and x_p^* , is identified using the formulation for constrained module optimization below. It should be observed that x_p^* does not appear in (9) as a variable in the minimization. As such, the values of x_p^* shown in D_m and (13)–(15) are the values identified in Step E and are fixed in (9).

Problem 2b: Formulation for constrained module optimization

$$\min_{x_m^{(i)}} \; J^{(i)} = \left|\left| P^{(\beta)} - \bar{P}^{(i)} \right|\right| \tag{9}$$

where:

$$\alpha = \delta_{i,1} \tag{10}$$

$$\beta = \delta_{i,2} \tag{11}$$

$$\bar{P}^{(i)} = P^{(\alpha)} + \Delta P^{(i)} \tag{12}$$

defined by:

$$P^{(\alpha)} = \left(\mu_1|_{x_a^{(\alpha)*}, x_p^*}, \; \dots \; , \mu_{n_\mu}|_{x_a^{(\alpha)*}, x_p^*} \right) \; (n_\mu \geq 2) \tag{13}$$

$$P^{(\beta)} = \left(\mu_1|_{x_a^{(\beta)*}, x_p^*}, \; \dots \; , \mu_{n_\mu}|_{x_a^{(\beta)*}, x_p^*} \right) \; (n_\mu \geq 2) \tag{14}$$

$$\Delta P^{(i)} = \Big(\Delta\mu_1(x_m^{(i)}, x_p^*, \hat{p}^{(i)}), \; \dots \; \dots \; , \Delta\mu_{n_\mu}(x_m^{(i)}, x_p^*) \Big) \; (n_\mu \geq 2) \tag{15}$$

where D_m is the set of values and variables of x_p^* and x_m^* for each module design; vectors $P^{(\alpha)}$ and $P^{(\beta)}$ characterize the objective space performance of the base (α) and target (β) designs; vector $\bar{P}^{(i)}$ represents the objective space performance of design α when used in conjunction with the i-th module; vector $\Delta P^{(i)}$ represents the change in objective space performance from design α to $\bar{P}^{(i)}$; and x_m represents the value(s) and variable(s) that characterize ΔP. It should be noted that this formulation enables the variables of x_m to be different for each module designed (See (9)).

With completion of the constrained module design process, a system capable of adapting to changes in needs over time through the addition of modules is achieved. In addition, each iteration of the system obtained through the addition of modules provides the optimal performance according to the objectives provided in Problem 2a (see Section 2.5).

In the following section, a tri-objective hurricane and flood resistant residential structure example is provided to demonstrate the implementation of the method described in Section 2.

3 Hurricane and flood resistant modular residential structure example

Over the past 30 years, the frequency and intensity of tropical cyclones has steadily increased (Fay et al. 2003). In Latin America alone, approximately 2.5 million people were made homeless by cyclones and other natural disasters between 1990 and 1999 due in large part to the quality of local infrastructure (Fay et al. 2003). As a result, international entities, including the World Bank, have called for the development of infrastructure solutions capable of withstanding these increasingly occurring disasters (Fay et al. 2003).

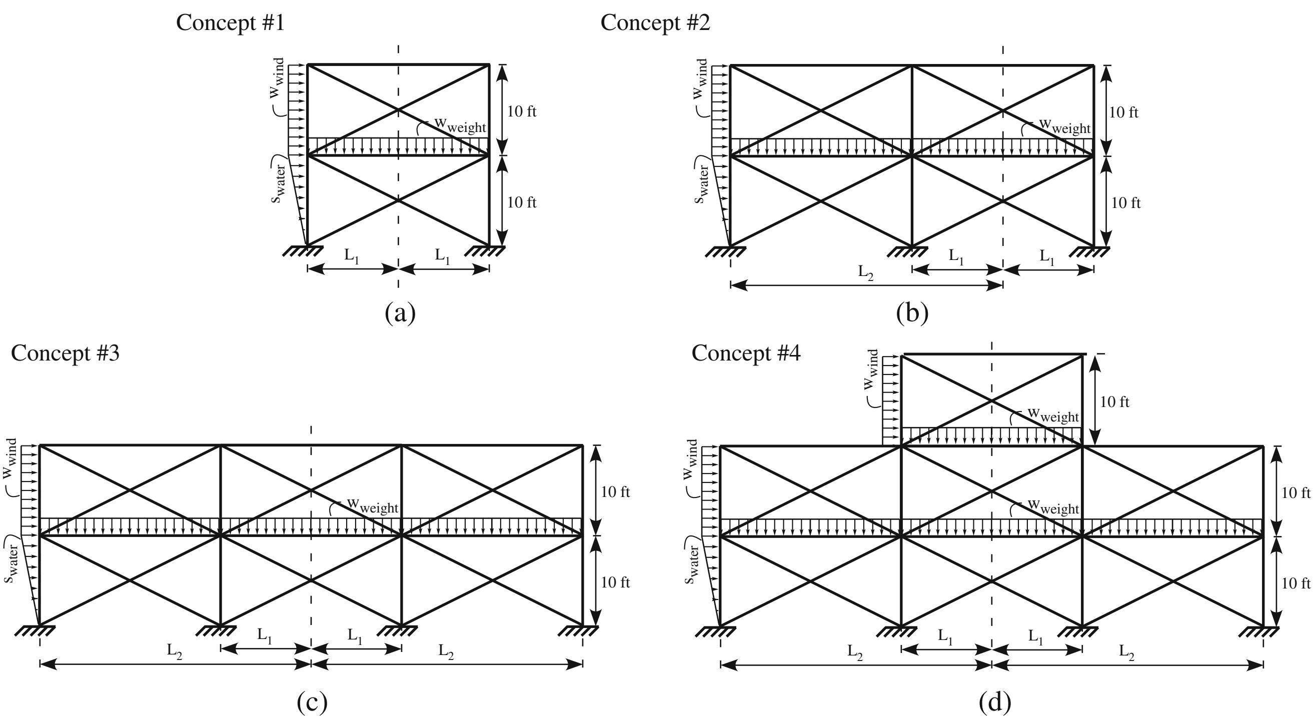

Fig. 5 Graphical summary of four hurricane and flood resistant residential structure concepts and assumed plane frame loading conditions

One implemented concept for addressing this challenge within residential markets is to elevate the living space above the expected flood levels (Favela 2009). Building on this idea, this section presents a tri-objective implementation of the method described in Section 2, involving four hurricane and flood resistant residential structure design concepts. Representing these concepts as plane frames, a graphical summary of each concept and the assumed loading conditions is provided in Fig. 5. It is recognized that the concepts presented in Fig. 5 could be presented as parameterized versions of each other. However, in order to illustrate the ability of the method to account for multiple system concepts, each is analyzed with a distinct concept model.

3.1 Plane frame analysis model assumptions

In order to analyze the concept plane frame models represented in Fig. 5, the following assumptions are made:

1. Column sections are selected from among 28 AISC sections ranging from W12X336 to W12X16.
2. Beam sections are selected from among 57 AISC sections ranging from W33X169 to W18X35, all with width between 0.1524 and 0.3048 m.
3. Cross brace sections are selected from among 35 AISC hollow square sections ranging from HSS6X6X5/8 to HSS18X6X1/4, all with width of 0.1524 m.
4. Topology optimization that maintains symmetry in concepts 3 and 4 is performed for all cross braces shown in Fig. 5.
5. The uniformly distributed wind load (w_{wind}) is 9.46 kN/m.
6. The uniformly distributed weight load (w_{weight}) is 10.95 kN/m.
7. The triangular distributed flood water load (s_{water}) is 6.34 kN/m.
8. The allowable inter-story drift is 0.01016 m.
9. All joints connected to ground are assumed to be fixed.
10. The connections for shared beam joints are constrained to be the same for displacements and rotations.
11. The safety factor for buckling is 2.
12. The cost to material volume ratio of the structure is 1.
13. The allowable normal stress (σ_{allow}) is 206842.72 kN/m^2 for column and beam sections, and 190295.3 kN/m^2 for cross brace sections.
14. The allowable shear stress (τ_{allow}) is 137895.15 kN/m^2 for column and beam sections, and 126863.53 kN/m^2 for cross brace sections.
15. Half of the width of the first bay (L_1) can be values in 0.1524 m increments between 2.286 and 4.572 m.
16. The width of the second bay plus half of the width of the first bay (L_2) can be values in 0.1524 m increments between 6.858 and 13.716 m.

Table 1 Objective limits for the i-th anticipated region of interest

Region	Objective limits					
i	$C_l^{(i)}$	$C_u^{(i)}$	$\hat{\sigma}_{\max_l}^{(i)}$	$\hat{\sigma}_{\max_u}^{(i)}$	$A_l^{(i)}$	$A_u^{(i)}$
1	6500	9000	0	1	41.81	55.74
2	9600	12500	0	1	139.35	153.29
3	13000	17000	0	1	236.90	250.84
4	18000	25000	0	1	278.71	306.58

Table 2 Objective values of the optimal design selected within the i-th region of interest (column 1) obtained through Step E of the method. The design concept corresponding to the design selected within each region is equal to the i-th region (see column 1)

Region/Concept	Optimal objective values		
i,k	$C^{(i)*}$	$\hat{\sigma}_{\max}^{(i)*}$	$A^{(i)*}$
1	5969	0.6483	41.81
2	9489	0.4440	144.93
3	13009	0.3975	256.41
4	17102	0.6433	298.22

17. Columns and beams on the same level all have the same respective cross section.
18. Cross braces in the same bay and level have the same cross section.
19. The structures are assumed to extend 9.144 m out of plane (used to calculate livable area).

3.2 Method implementation

Prior to implementing the method described in Section 2, the objectives are identified as (1) *minimize* cost (C); (2) *minimize* the max stress to allowable stress ratio ($\hat{\sigma}_{\max} = \sigma_{\max}/\sigma_{\text{allow}}$); and (3) *maximize* the livable area of the building (A in m^2) Using these objectives, the bounds of four anticipated regions of interest in terms of these objectives are provided in Table 1.

Figure 6 provides a graphical illustration of the candidate s-Pareto target designs identified within each region of interest through Step D of the method. These results were obtained using a custom Multiobjective Genetic Algorithm (population = 500, blend crossover rate = 0.4, mutation rate = 0.2, and number of generations = 1200) tailored to search all regions of interest simultaneously. Analysis of the optimization objectives (C, $\hat{\sigma}_{\max}$, and A) for the plane frame structures were performed using a matrix stiffness computer program developed in Balling (2009).

Results of the objective values for the optimal design set proceeding from Step E of the method are presented in Table 2. It should be noted that the selected designs do not represent platform and module designs. Instead, they represent the non-modular system designs chosen by the method to be the best suited for conversion into platform and module designs in the remaining steps of the method. To obtain these results, the regional weights required by (2) are $w^{(i)} = \{0.6, 0.5, 0.2, 0.1\}$. These weights were selected based on the assumption that aggregate objective function values for designs selected in the first two regions of interest are more highly emphasized then in the last two regions. Although, it is noted that for this example the sensitivity of the resulting target designs to the values of $w^{(i)}$ is very low. In addition, the aggregate objective function required

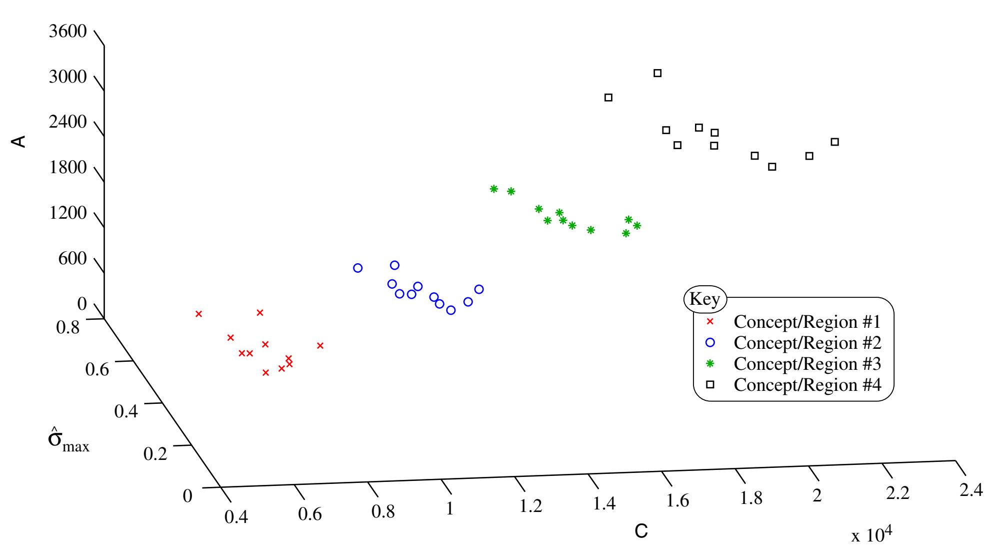

Fig. 6 The candidate s-Pareto target designs within each region of interest identified through Step D of the method

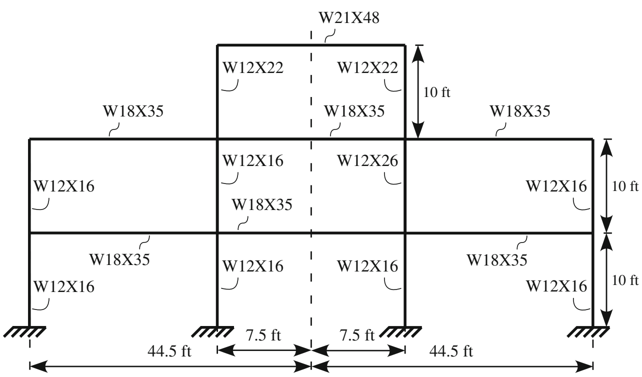

Fig. 7 Graphical representation of the optimal topology, along with the values of L_1, L_2, and the selected cross sections corresponding to the fourth region of interest

by (3) is identified in (16) as a Substitute Objective Function (Cheng and Li 1996; Messac 2000).

$$J^{(k)} = \prod_{j=1}^{n_\mu} \left(\frac{\mu_{j,\max}^{(i)} - \mu_i^{(k)}}{\mu_{j,\max}^{(i)} - \mu_{j,min}^{(i)}} \right) \tag{16}$$

where $\mu_{j,\max/\min}^{(i)}$ is the maximum/minimum non-shifted s-Pareto value of the j-th objective within the i-th region of interest.

The cross-brace topology, along with the values of L_1, L_2 (See Fig. 5), and the selected cross sections is presented in Fig. 7. Since the design selected in the fourth region of interest also represents the designs from the other regions, this design is the only one represented in Fig. 7.

Prior to developing the module designs, information on the type, number, and desired progression of modules that are to be used to obtain the s-Pareto designs contained identified in Table 2 is needed. For this example, a slot-modular approach was selected for the modular architecture (Lewis et al. 2011; Ulrich and Eppinger 2004). By examining the concepts presented in Fig. 5, it is observed that the design that is common to all other concepts will correspond to concept/region one (i.e., $D_a^{(1)}$). Using this information, and the assumption that the customer preference is to progress sequentially from one region to the next, the desired number of modules to be developed (n_m) is chosen to be three. In addition, the matrix δ (see Section 2.6), defining the desired modular progression, is constructed in (17). Columns 1 and 2 in δ respectively refer to the starting and target design in D_a for each module (rows in δ). Note that the method is not constrained by the assumption to progress sequentially from one region to the next. For this example it has been assumed as such to simplify presentation and illustration of this example.

Table 3 Variable values of the module designs (i) obtained through evaluation of a constrained module design routine of the form presented in Problem 2b

Module	Module variables			
i	L_1^* (m)	L_2^* (m)	S_c	S_b
1	11.2776	–	–	–
2	–	11.2776	–	–
3	–	–	W12X22	W21X48

$$\delta = \begin{bmatrix} 1 & 2 \\ 2 & 3 \\ 3 & 4 \end{bmatrix} \tag{17}$$

Results from evaluating a constrained module design routine of the form presented in Problem 2b are presented in Table 3. The modular variables identified in Table 3 are: (i) the length of the beam sections for module 1 (L_1), (ii) the length of the beam sections for module 2 (L_2), (iii) the AISC cross-section of the top level columns (S_c), and (iv) the AISC cross-section of the top level beam (S_b). All other variables defining the module designs are identified within the selected platform design.

3.3 Discussion of results

With completion of the constrained module design process, a modular residential structure capable of expanding through the addition of three different modules is achieved. To illustrate the selected platform and three module designs identified in Table 3, Fig. 8 is provided.

It should be remembered that the anticipated regions of interest identified in Step B of the method (see Table 1) represent what the customer/designer wishes to achieve over time. As such, the identified solutions are deemed to be good for three reasons: (i) the designs identified in Step E of the method (see Table 2) are within the designer-defined regions of interest; (ii) the selected platform variables enabled the identified solutions to be located on the original Pareto frontiers of each concept (shift from s-Pareto frontier to accommodate modularity was minimized); and (iii) modules that enable the system to traverse the identified s-Pareto solution set using the selected platform were identified.

Due to the optimization formulations found in Steps E and F, these results are not surprising. The formulation in Step E serves to ensure that identified designs are within the

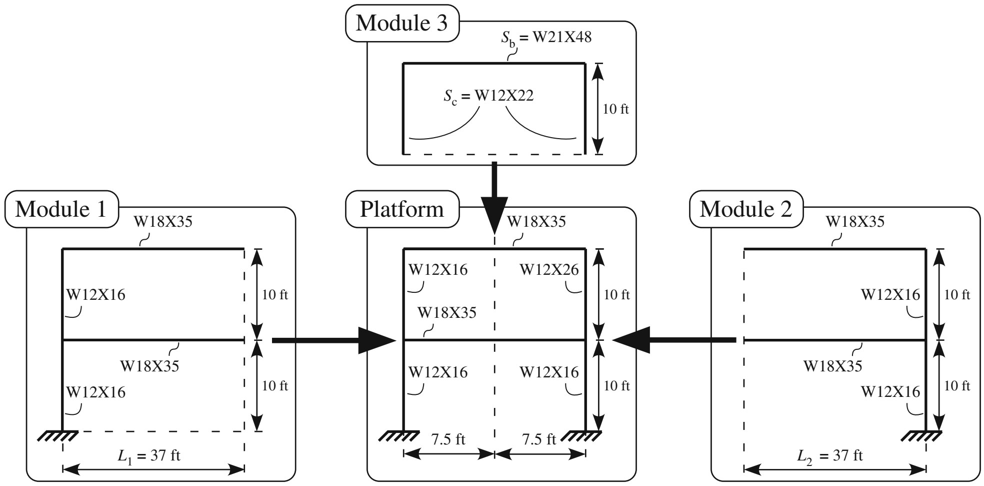

Fig. 8 Graphical representation of the selected platform and three module designs identified in Table 3

regions of interest, while the formulation of Step F ensures that the modules developed are as close to the s-Pareto designs selected in Step E as possible.

4 Concluding remarks

This paper has presented an optimization-based method to address an important limitation of current methods of module-based system design—accounting for significant, natural changes in system needs over time. The method builds on elements of multiobjective optimization, product family design, and modular product design found in the literature. Additionally, the presented methodology builds upon fundamental elements presented in a previous work by the authors by introducing important advancements that extend the optimization formulations to include multiple disparate system concepts. Therefore, the method presented in this paper enables the development of systems that can modulate from one optimal position on the s-Pareto frontier to another. Further, the presented method enables the adaptability and reconfigurability of such designs to be based on the natural changes in system needs over time.

The method, as presented in Section 2 of this paper, is broadly applicable to diverse applications, one of which is a detailed and involved pump design developed by the authors (Lewis et al. 2010). In the present paper, a tri-objective hurricane and flood resistant residential structure example demonstrated the ability of the method to select a set of system designs that facilitate the development of platform and module designs. Specifically, from this example it is seen that, through a series of multiobjective optimization routines, as detailed by the presented method, has emerged a new methodology for selecting platform and module designs based on predicted changes in needs, even in higher dimensional design situations.

One of the fundamental assumptions of the presented method is that the changes in customer needs over time are known. Building on the methods presented herein, future developments will include the identification of additional methods for determining and quantifying future needs. Recognizing the existence of uncertainty related to identified future needs (Martin and Ishii 2002), additional work will also include the incorporation of uncertainty analysis methods in the selection of platform and module designs. The focus of these methods will be to account for variations in customer perception, available market data, material properties, manufacturing precision, system health deterioration due to failure/wear, and other sources.

Acknowledgments We would like to recognize the National Science Foundation Grant CMMI-0954580 for funding this research.

References

Balling RJ (2009) Computer Structural Analysis, vol I. BYU Academic Publishing

Cheng FY, Li D (1996) Multiobjective optimization of structures with and without control. J Guid Control Dyn 19(2):392–397

Cheng FY, Li D (1998) Genetic algorithm development for multiobjective optimization of structures. AIAA J 36(6):1105–1112

de Weck OL, Suh ES, Chang D (2003) Product family and platform portfolio optimization. In: ASME IDETC/CIE 2003, DETC03/DAC-48721

Di Barba P (2001) A fast evolutionary method for identifying non-inferior solutions in multicriteria shape optimization of a shielded reactor. COMPEL: Int J for Computation and Maths. in Electrical and Electronic Eng 20(3):762–776

Favela VD (2009) Houses put to flood and hurricane test. Inter Press Service http://ipsnews.net/news.asp?idnews=48303

Fay M, Ghesquiere F, Solo T (2003) Natural disasters and the urban poor. En breve 32:1–4

Ferguson S, Siddiqi A, Lewis K, de Weck OL (2007) Flexible and reconfigurable systems: Nomenclature and review. In: ASME international design engineering technical conference & computers and information in engineering conference, DETC2007/DAC-35745

Ferguson S, Kasprzak E, Lewis K (2009) Designing a family of reconfigurable vehicles using multilevel multidisciplinary design optimization. Struct Multidiscipl Optim 39(2):171–186

Frischknecht BD, Peters DL, Papalambros PY (2011) Pareto set analysis: local measures of objective coupling in multiobjective design optimization. Struct Multidiscipl Optim 43(5):617–630. doi:10.1007/s00158-010-0599-2

Gurnani AP, Lewis K (2008) Using bounded rationality to improve decentralized design. AIAA J 46(12):3049–3059. doi:10.2514/1.35776

Kasprzak EM, Lewis KE (2000) An approach to facilitate decision tradeoffs in pareto solution sets. J Eng Valuation and Cost Anal 3:173–187

Khire RA, Messac A (2008) Selection-integrated optimization (SIO) methodology for optimal design of adaptive systems. ASME J Mech Des 130(10):101,401–101,413. doi:10.1115/1.2965365

Lewis PK, Murray VR, Mattson CA (2010) Accounting for changing consumer needs with s-Pareto frontiers. In: The 13th AIAA/ISSMO multidisciplinary analysis optimization conference

Lewis PK, Murray VR, Mattson CA (2011) A design optimization strategy for creating devices that traverse the Pareto frontier over time. Struct Multidiscipl Optim 43(2):191–204. doi:10.1007/s00158-010-0555-1

Li Y, Xue D, Gu P (2008) Design for product adaptability. Concurr Eng 16(3):221–232

Martin MV, Ishii K (2002) Design for variety: developing standardized and modularized product platform architectures. Res Eng Des 13(4):213–235. doi:10.1007/s00163-002-0020-2

Mattson CA, Messac A (2003) Concept selection using s-Pareto frontiers. AIAA J 41(6):1190–1204

Mattson CA, Messac A (2005) Pareto frontier based concept selection under uncertainty, with visualization. OPTM: Optim and Eng 6(1):85–115

Mattson CA, Mullur AA, Messac A (2004) Smart pareto filter: Obtaining a minimal representation of multiobjective design space. Eng Optim 36(4):721–740

Messac A (1996) Physical programming: Effective optimization for computational design. AIAA J 34(1):149–158

Messac A (2000) From dubious construction of objective functions to the application of physical programming. AIAA J 38(1):155–163

Messac A, Mattson CA (2004) Normal constraint metthod with guarentee of even representation of complete pareto frontier. AIAA J 42(10):2101–2111

Olewnik A, Brauen T, Ferguson S, Lewis K (2004) A framework for flexible systems and its implementation in multiattribute decision making. J Mech Des 126(3):412–419

Otto K, Wood K (2001) Product design: techniques in reverse engineering and new product development. Prentice Hall

Siddiqi A, de Weck OL (2008) Modeling methods and conceptual design principles for reconfigurable systems. ASME J Mech Des 130(10):101,102–101,116. doi:10.1115/1.2965598

Simpson TW (2004) Product platform design and optimization: Status and promise. Artif Intell Eng Des Anal Manuf 18(1):3–20

Strong MB, Magleby SP, Parkinson AR (2003) A classification method to compare modular product concepts. In: ASME international design engineering technical conference & computers and information in engineering conference, DETC2003/DTM-48661, pp 657–668. doi:10.1115/DETC2003/DTM-48661

Todoroki A, Sekishiro M (2008) Modified efficient global optimization for a hat-stiffened composite panel with buckling constraint. AIAA J 46(9):2257–2264. doi:10.2514/1.34548

Tseng MM, Jiao J (1998) Design for mass customization by developing product family architecture. In: ASME international design engineering technical conference & computers and information in engineering conference, DETC98/DFM-5717

Tseng MM, Jiao J, Merchant ME (1996) Design for mass customization. Annals of CIRP—Manuf Technol 45(1):153–156

Ulrich K, Eppinger S (2004) Product design and development, 4th edn. McGraw-Hill

Yearsley JD, Mattson CA (2008) Interactive design of combined scale-based and module-based product family platforms. In: Proceedings of 12th AIAA/ISSMO multidisciplinary analysis and optimization conference, AIAA-2008-5819