

A design optimization strategy for creating devices that traverse the Pareto frontier over time

Patrick K. Lewis · Vance R. Murray ·
Christopher A. Mattson

Received: 19 February 2010 / Revised: 3 June 2010 / Accepted: 25 July 2010 / Published online: 29 August 2010
© Springer-Verlag 2010

Abstract In some instances, the performance or function that is needed by a product naturally and predictably changes over time. Providing solutions that anticipate, account for, and allow for these changes is a significant challenge to manufacturers and design engineers. In this paper, a multiobjective optimization design method involving the strategic use of a series of optimization formulations is introduced to design products that adapt to changing needs by moving from one location on the Pareto frontier to another through the addition of a module. The design of a simple unmanned air vehicle is used to demonstrate implementation of the method, and results in the development of one aircraft platform and two module designs that adapt the aircraft to perform optimally for the particular mission at hand, thus optimally satisfying all three different mission profiles individually. The authors conclude that the developed method provides a new and general framework for selecting platform and module designs, and is capable of providing a set of designs based on predicted changes in needs.

Keywords Multiobjective optimization · Transient Pareto design · Modular design · Future needs

Nomenclature

b	UAV wing span, (m)
\bar{c}	Average cord length of UAV wing, (m)
$C_{L_{\max}}$	Coefficient of lift
$C_{T_{\max}}$	Coefficient of thrust

δ	Matrix dictating the desired progression that each module provides
D	Set containing all design variable values that describe all designs in a set of designs
D_m	Set containing all variable values of x_m and x_p
$D^{(i)}$	Design (i) of set D
D_{turn}	Distance traveled in a complete 360° UAV turn, (m)
\hat{D}_{turn}	Distance traveled per degree of turn, (m/°)
E	Surveillance elevation of UAV mission profile, (m)
g	Vector of inequality constraints
\hat{g}	Acceleration due to gravity, (m/s ²)
η_{\max}	Maximum load factor
h	Vector of equality constraints
θ	Degree of turn, (°)
L	Temperature lapse rate, (K/m)
L_m	Module wing extension lengths, (m)
L_t	Length of wing tip section, (m)
μ	Vector of design objectives
M	Molar mass of dry air, (kg/mol)
m_t	Total mass of UAV, (kg)
m_e	Mass of onboard equipment (i.e., cameras, batteries, computers), (kg)
m_f	Mass of UAV fuselage, (kg)
m_w	Mass of UAV wings, (kg)
n_d	Number of designs comprising the adaptive design set
n_g	Number of inequality constraints
n_h	Number of equality constraints
n_μ	Number of design objectives
$n_{\hat{\rho}}$	Number of additional objective constraints needed to define anticipated regions of interest
n_m	Number of modules desired
n_x	Number of design variables

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

n_{x_a}	Number of adjustable design variables
n_{x_m}	Number of module design variables
n_{x_p}	Number of platform design variables
$P^{(\alpha)}$	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
\hat{p}_0	Sea level standard atmospheric pressure, (Pa)
\hat{p}	Atmospheric pressure at E , (Pa)
ρ	Density of air at E , (kg/m^3)
ρ_w	Density of wing material, (kg/m^3)
R	Universal gas constant, ($\text{J}/(\text{mol}\cdot\text{K})$)
R_{turn}	Minimum radius of turn of UAV, (m)
S_{ref}	Reference wing area, (m^2)
T	Temperature of air at E , (K)
T_0	Sea level standard temperature, (K)
T_{lost}	Time lost in a 180° UAV turn, (s)
\hat{T}_{lost}	Time lost per degree of turn, ($\text{s}/^\circ$)
$\hat{T}_{\text{lost,max}}$	Maximum time lost allowed per degree of turn, ($\text{s}/^\circ$)
t_w	Equivalent thickness of a rectangular cross-section wing, (m)
V	Mission cruise velocity, (m/s)
x	Vector of design variables
x_a	Vector of non-platform design variables that are either scaled or discretely adjusted
x_m	Vector of non-platform design variables that characterize the design of modules
x_p	Vector of platform design variables

1 Introduction

The customer preferences related to a product's performance can naturally change over time (Trope and Liberman 2000). This change in preference is often linked to changes in customer needs. Products that adapt to these changes through the addition of modules reduce production costs and cater to customization and adaptation (Simpson 1998; Tseng et al. 1996). In situations where purchase costs are high and these changes occur rapidly, there is a need for products that are capable of adaptation and expandability (Li et al. 2008). Perhaps the most accessible example of this is the personal computer with its adaptive, easily changed memory (Simpson 1998). Providing solutions that anticipate, account for, and allow for changing customer needs and preferences is a significant challenge to manu-

facturers and design engineers. Product families are often used to address this challenge through product performance diversity, while still maintaining product commonality as seen by manufacturers (Tseng and Jiao 1998; Yearsley and Mattson 2008a). Two platforms for building product families are identified within the literature: Scale-based and Module-based product platforms (Yang et al. 2004; Yearsley and Mattson 2008a). Product family approaches are well-suited to provide a set of products that satisfy the variation in needs across *multiple market segments* (Yearsley and Mattson 2008c). A natural and needed extension to these methods is one that includes the effects of future needs and allows for a *single* device to traverse the Pareto frontier over time. The objective of the current paper is to present such an extension, and thereby optimally adjust to new and changing needs. We note that this paper considers the condition wherein customer needs and preferences change over time not the condition where predictive design models change over time. We also note, that for simplicity, *changing customer needs and preferences* are hereafter referred to as simply *changing needs*.

Introduced and presented in Section 3 of this paper is a multiobjective optimization-based method developed by the authors for designing module-based products. Similar to traditional module-based product family design approaches, the presented method uses commonality to identify an optimal product platform and module designs. However, the presented method differs from traditional module-based product family design approaches through the process used to obtain the platform and module designs. In other words, instead of using a traditional factorial problem (Fujita et al. 1999) to choose the modules, the current method identifies a platform and modules that satisfy anticipated changes in needs by constraining the search space to a series of anticipated regions of interest representing predicted future needs. Assuming these changes over time are known, and that the design space is fixed, the use of a multiobjective optimization method provides two key benefits: (1) The ability to leverage a set of non-dominated designs to enhance the selection of the platform variables (values remain constant for all product family members) and module variables (values characterize the modules)— x_p and x_m respectively. (2) The ability to balance the competing nature of present needs against future needs.

Based on the assumptions identified in the previous paragraph, it should be noted that the concept of multi-scenario (i.e., scenarios where the design objective space evolves over time), although not the topic of this paper, is a parallel area of research to that presented in this paper. We refer the interested reader to Wiecek et al. (2009) for additional information on this topic.

The remainder of this paper is presented as follows: A review of literature forming an enabling foundation for the

development of the method is included in Section 2. In Section 3, the theoretical development of the method is presented. In Section 4 the design of a simple unmanned air vehicle is used to demonstrate the method. Concluding remarks are provided in Section 5.

2 Literature survey

This section provides a review of previous research, and establishes a foundation for the general methodology of designing module-based products presented in Section 3 of this paper. The technologies that form an enabling foundation for the methodology are (i) multiobjective optimization and (ii) product modularity and adaptability. As the design of a UAV is included in Section 4 as an example implementation of the methodology presented in Section 3, we note that the review of pertinent UAV literature is included in that section.

2.1 Multiobjective optimization

Performance or functional needs of a product, some of which are expressed as design objectives, are often competing and change over time. Thus, the ability of multiobjective optimization to balance the competing nature of present needs against future needs is fundamental within the presented design method (Balling 2000; Das 1999; Kasprzak and Lewis 2000; Li et al. 1998; Mattson and Messac 2003, 2005; Park and Grierson 1999; Tappeta et al. 2000; Wu and Azarm 2001). Figure 1 demonstrates the characterization of trade-offs between the objectives through the creation

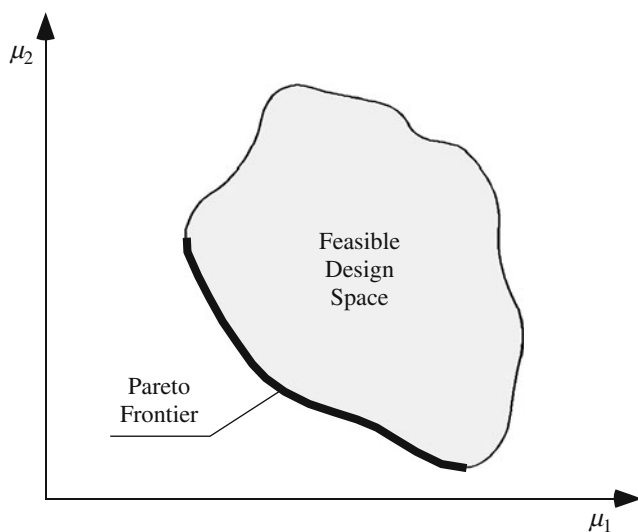


Fig. 1 A feasible design space (shaded) for objectives 1 and 2. The Pareto frontier (bold line) represents the most desirable set of solutions in the feasible space for this minimization-minimization problem

of a Pareto frontier—a set of non-dominated optimal solutions. Each solution comprising the frontier, graphically demonstrated in Fig. 1, is said to be *Pareto optimal*—no other designs better satisfy *all* design objectives (Belegundu and Chandrupatla 1999; Messac and Mattson 2002; Miettinen 1999; Steuer 1986). These Pareto solutions are generally sought because they indicate that objectives have been improved as much as possible without sacrificing another design objective’s performance. In addition, each solution represents the optimal balance of design objectives according to the product needs at a specific instance.

A generic multiobjective optimization problem (MOP) formulation yielding a set of optimal solutions—those belonging to the Pareto frontier—is presented as follows:

Problem 1 Generic multiobjective optimization problem statement

$$D := \{(x_1, x_2, \dots, x_{n_x})\} \tag{1}$$

x defined by:

$$\min_x \{ \mu_1(x, p), \mu_2(x, p), \dots, \mu_{n_\mu}(x, p) \} \quad (n_\mu \geq 2) \tag{2}$$

subject to:

$$g_q(x, p) \leq 0 \quad \forall q \in \{1, \dots, n_g\} \tag{3}$$

$$h_k(x, p) = 0 \quad \forall k \in \{1, \dots, n_h\} \tag{4}$$

$$x_{jl} \leq x_j \leq x_{ju} \quad \forall j \in \{1, \dots, n_x\} \tag{5}$$

where D is a set containing all values of x for each Pareto optimal design obtained through the evaluation of the MOP; μ_i denotes the i th generic design objective; x is a vector of design variables; and p is a vector of design parameters.

For multiobjective optimization approaches, the decision of which Pareto optimal solution is to be used comes through the inclusion of objective function parameters, and sometimes constraints that capture the product needs for a single instance in time. Similar to traditional multiobjective optimization approaches, the method presented seeks Pareto optimal solutions, but expands upon traditional approaches by selecting Pareto-optimal solutions, within anticipated regions of interest, based on the solutions ability to (i) be implemented by a module-based product, and (ii) expand/adapt to satisfy known changes in needs over time.

2.2 Product modularity and adaptability

Under the presented methodology (described in the next section), and as identified within the literature, there is a need for strategic module designs that make product platform designs progressively expandable (Gonzalez-Zugasti and Otto 2000; Magrab 1997; Mattson and Magleby 2001). To this end, previous work in the areas of product family and modular product design serve as a starting point (Gonzalez-Zugasti and Otto 2000; Mattson and Magleby 2001; Wie et al. 2001; Yearsley and Mattson 2008a, b, c). A module-based product family is a group of related products derived from independent functional or geometric units (Baguley et al. 2004; Pahl and Beitz 2007; Ulrich and Eppinger 2004) that differ through the addition or subtraction of modules (Pahl and Beitz 2007; Ulrich and Eppinger 2004; Yearsley and Mattson 2008b).

In the literature three types of modularity are identified: (i) *Slot-modular architecture*, (ii) *Bus-modular architecture*, and (iii) *Sectional-modular architecture* (Oyebode 2004; Padamat 2004; Ulrich and Eppinger 2004). A slot-modular architecture provides each module with a unique interface in order to eliminate improper assembly (Oyebode 2004; Padamat 2004; Ulrich and Eppinger 2004). Bus-modular architecture implements interfacing that is the same for all modules, thus making the platform design behave as a common connection platform for all modules (Ulrich and Eppinger 2004). Sectional-modular architecture is similar to bus-modular in that all modules contain the same interface, but in this architecture no single element is identified as the platform to which all modules attach (Oyebode 2004; Ulrich and Eppinger 2004). Building on these foundational elements, the presented method uses these definitions of modular architectures to specify the approach needed to develop module designs according to the a desired architecture type.

Recent developments in the literature show that a desirable product family can be identified from among the designs comprising the Pareto frontier (Yearsley and Mattson 2008a, c) obtained through the evaluation of an MOP (see Section 2.1). In module-based product family development, modularity is only one of many potential design objectives, and increased modularity may come at the cost of other objectives (Mattson and Magleby 2001). These previous developments evaluate and select product family members from among the set of Pareto designs by considering the design's unique performance *and* common features compared to other designs in the product family (a critical part of product family design). In addition, one method of identifying module and platform variables is accomplished through the use of Pareto-filtering methods that explore the effects of each variable on the objective space performance (Yearsley and Mattson 2008a, c).

Building on these approaches as a foundation, a selection criteria based on known changes in needs over time is added to the evaluation—to ensure that a progression from one design on the Pareto front to another can be done through the addition of a module.

3 Theoretical development

By its nature, a Pareto frontier contains many optimal, yet functionally different, designs representing all optimal product candidates. To satisfy changes in needs over time through the addition of modules requires the strategic selection of these Pareto-optimal designs based on their ability to facilitate adaptability. In this section, the development of a novel multiobjective optimization design method is introduced. It is enabled by the foundational literature surveyed in the previous section, and provides a Pareto-optimal product and module designs capable of satisfying changes in needs over time. Figure 2 illustrates the intent of the method to satisfy changing needs by selecting a Pareto-optimal product platform design which, through the addition of modules, expands to other Pareto-optimal designs. The first (μ_1) and second (μ_2) objectives are represented along the horizontal and vertical axis, respectively. From the figure it is seen that the platform design, shown as $P^{(1)}$, adapts to become $P^{(2)}$ through the addition of Module 1. Through this approach, the platform and subsequent modules, provide the desired product performance resulting from the changing needs as represented by $P^{(1)}$, $P^{(2)}$, $P^{(3)}$, and $P^{(4)}$. Figure 3 provides a flow chart that illustrates the five primary steps of the multiobjective optimization design

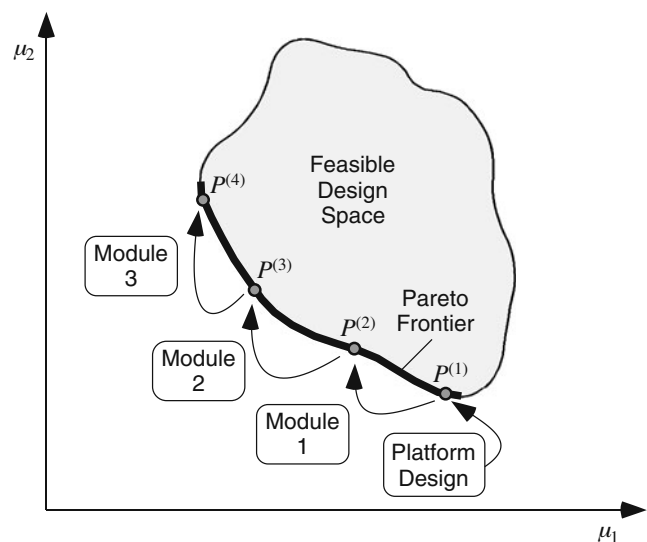


Fig. 2 Graphical representation of the intent of the proposed method to provide a product that expands from one Pareto-optimal design to another through the addition of modules

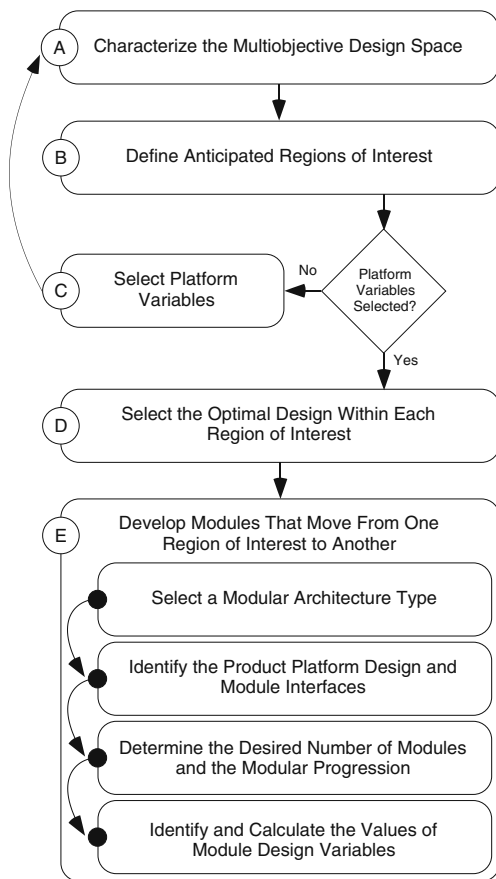


Fig. 3 Flow chart describing the five-step multiobjective optimization design method developed in this section

method developed herein. Each of these steps is described in the following sections.

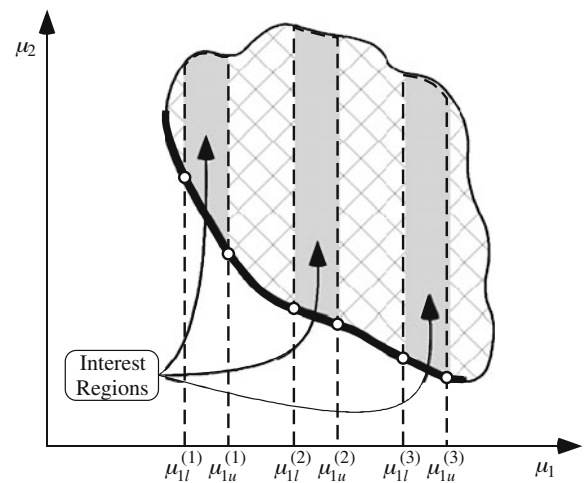
3.1 Characterize the multiobjective design space

The first step of the method explores the multiobjective design space to evaluate and characterize the effects of each design variable on the objective space performance, and is accomplished through the evaluation of an MOP as described in (1–5) of Section 2.1. Recalling that the result of the evaluation of an MOP as presented in Section 2.1 is a Pareto frontier, it should be noted that it is the responsibility of the designer to identify an appropriate Pareto frontier generation method based on the specifics of the design problem being addressed.

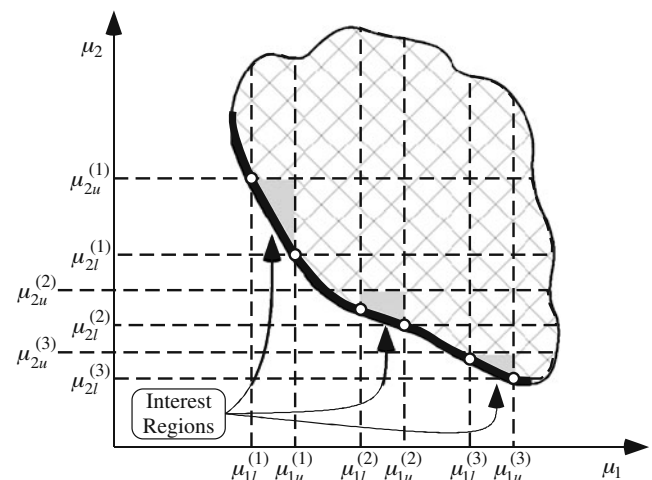
3.2 Define anticipated regions of interest

The second step of the method captures the predicted changes in needs over time and enhances the ability of an optimizer to select the designs that are optimal for adaptation, as illustrated in Fig. 2, by identifying designs within *Anticipated Regions of Interest*.

For each anticipated region of interest presented in Fig. 4, a new MOP, with a reduced design space, is defined by additional objective constraints based on known changes in needs. For example, for the left most region of interest in Fig. 4b the objective μ_1 is constrained by $\mu_{1,l}^{(1)} \leq \mu_1 \leq \mu_{1,u}^{(1)}$, where $\mu_{1,l}^{(1)}$ and $\mu_{1,u}^{(1)}$ are prescribed. The result is the bounding of the MOP to search the design space within the geometric shape of the anticipated region of interest. Further definition of the anticipated region of interest is unnecessary due to the function of a MOP of finding solutions along the Pareto frontier. For the examples presented in Fig. 4, the information capturing the changes in customer needs over time for each design in the set would be expressed as additional boundary constraints for the acceptable values of μ_1



(a)



(b)

Fig. 4 Representation of the construction of Anticipated Regions of Interest for known changes in needs for three intervals. The anticipated regions of interest in (a) provide inequality constraints for μ_1 . The anticipated regions of interest in (b) provide inequality constraints for μ_1 , and μ_2

and μ_2 . In the event that the anticipated region of interest restricts the optimizer to an infeasible space, a compromise in the acceptable range of the objectives for the infeasible region of interest is required, or a new design model must be considered which provides feasible solutions within the desired region of interest.

3.3 Select platform variables

The third step of the method uses the Pareto frontier within the regions of interest identified previously to identify those variables which are best suited as platform variables (x_p). This may be accomplished through the use of Pareto-filtering methods as described in Section 2.2 or any other suitable method. In cases where a concept of how the product is intended to expand through module addition has previously been identified (i.e., variables which are best suited as platform variables are already known), this step simplifies to the providing of that information for the remaining steps of the method. In addition, as is illustrated in Fig. 5, by selecting platform variables, it is likely that the Pareto frontier will shift. This shift represents a loss in the best possible performance due to the restricting of design variable values. To insure that the resulting shift in the Pareto frontier has not produced a shift that places 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.

3.4 Select the optimal design within each region of interest

The fourth step of the method is to develop the optimization routine used to select the optimal design in each anticipated

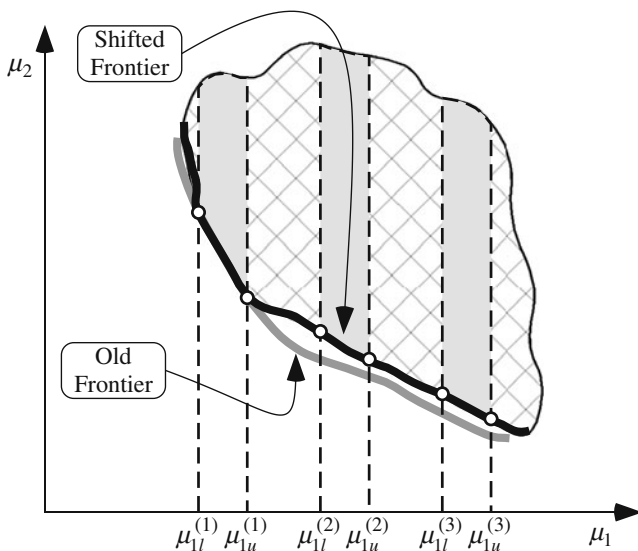


Fig. 5 Illustration of the expected shift in the Pareto frontier from Fig. 4 due to the selection of platform variables. The anticipated regions of interest presented in Fig. 4a are also shown

region of interest and identify the accompanying design variable values. The resulting optimal design set (D) containing all variable values is obtained through the following single-objective optimization formulation:

Problem 2a Optimization formulation for optimal adaptive product identification

$$D := \left\{ \left(x_{p,1}, x_{p,2}, \dots, x_{p,n_{x_p}}, x_{a,1}^{(i)}, x_{a,2}^{(i)}, \dots, x_{a,n_{x_a}}^{(i)} \right) \mid \forall i \in \{1, 2, \dots, n_d\} \right\} \quad (6)$$

$x_p, x_a^{(i)}$ defined by:

$$\min_{x_p, x_a^{(i)}} \left\{ \frac{1}{n_d} \sum_{i=1}^{n_d} J^{(i)} \left(x_a^{(i)}, x_p, p^{(i)} \right) \right\} \quad (7)$$

where:

$$J^{(i)} = w_1^{(i)} \cdot \mu_1(x_a^{(i)}, x_p, p^{(i)})^m + w_2^{(i)} \cdot \mu_2(x_a^{(i)}, x_p, p^{(i)})^m + \dots + w_{n_\mu}^{(i)} \cdot \mu_{n_\mu}(x_a^{(i)}, x_p, p^{(i)})^m \quad (n_\mu \geq 2) \quad (8)$$

subject to:

$$g_q^{(i)}(x_a^{(i)}, x_p, p^{(i)}) \leq 0 \forall q \in \{1, \dots, n_g^{(i)}\} \quad (9)$$

$$h_k^{(i)}(x_a^{(i)}, x_p, p^{(i)}) = 0 \forall k \in \{1, \dots, n_h^{(i)}\} \quad (10)$$

$$x_{a,j,l} \leq x_{a,j} \leq x_{a,j,u} \forall j \in \{1, \dots, n_{x_a}\} \quad (11)$$

$$x_{p,r,l} \leq x_{p,r} \leq x_{p,r,u} \forall r \in \{1, \dots, n_{x_p}\} \quad (12)$$

$$\mu_{y,l}^{(i)} \leq \mu_y^{(i)} \leq \mu_{y,u}^{(i)} \forall y \in \{1, \dots, n_\mu^{(i)}\} \quad (13)$$

where the adjustable variables (x_a) represent all non-platform design variables (variables that are either scaled or discretely adjusted); m is a compromise programming power (Chen et al. 1999); $w_1^{(i)}, w_2^{(i)}, \dots, w_{n_\mu}^{(i)}$ are weights associated with the local preference within each region of interest; the set D now represents the set of all design variable values of x_a and x_p obtained through the evaluation of the optimization problem; and the superscript (i) on p, g , and h indicate the possibility that parameters and constraints are different (non-constant) for each design in the set D . It is important to note that Problem 2a will result in a single solution within each region of interest.

From the optimization formulation presented in Problem 2a, it is seen that for each design—indicated by the superscript (i) —in the set D , the values of x_p are required to be the same for all $D^{(i)}$, while the values of x_a are not. In addition, the solution of Problem 2a will result in a set of designs that are located along the Pareto frontier within each region of interest.

Figure 6 is a representation of how the solution to Problem 2a for the set of anticipated regions of interest and the shifted Pareto frontier are used to identify the values of x_p and $x_a^{(i)}$. In addition, Fig. 6 shows how the intent of the proposed method to strategically select Pareto-optimal designs based on their ability to facilitate adaptability is satisfied through the implementation of Problem 2a.

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

By this step in the process, the set D 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—modules are designed in a manner that constrains them to provide a specified progression in product performance when added to a specific embodiment of the product while only using the variable values from set D . It should be noted that in order to do this, the designer must develop predictive design models for each of the modules that will be designed. In order to do this, complete this final step of the method, and obtain the module designs requires the following: (i) *Select a modular architecture*

type, (ii) *Identify the platform design and module interfaces*, (iii) *Determine the desired number of modules and modular progression*, and (iv) *Identify and calculate the values of module design variables*. Each of these four parts is briefly discussed.

Select a modular architecture type: Of the three types of modularity identified in the literature (see Section 2.2), *Slot-modular architecture* and *Bus-modular architecture* are best suited for implementation in the present method due to the use of platform designs. The decision of which architecture type to be used depends on the desired functionality of the product and modules as a whole.

Identify the product platform design and module interfaces: Prior to the identification of modules, one of the designs in set D must be identified as the product platform design. In order to facilitate adaptability, the platform design is generally identified as the design contained in D with the most commonality. In addition, the module interfaces must be specified according to the modular architecture type selected previously, and any other related interfacing design activities must be performed (i.e. begin defining the design model(s) describing the module(s)).

Determine the desired number of modules and the modular progression: With a knowledge of the modular architecture type that is desired, it is now possible to determine the number of modules (n_m) that are desired. The identification of n_m requires a knowledge of the manner in which the product is intended to expand. For the slot/

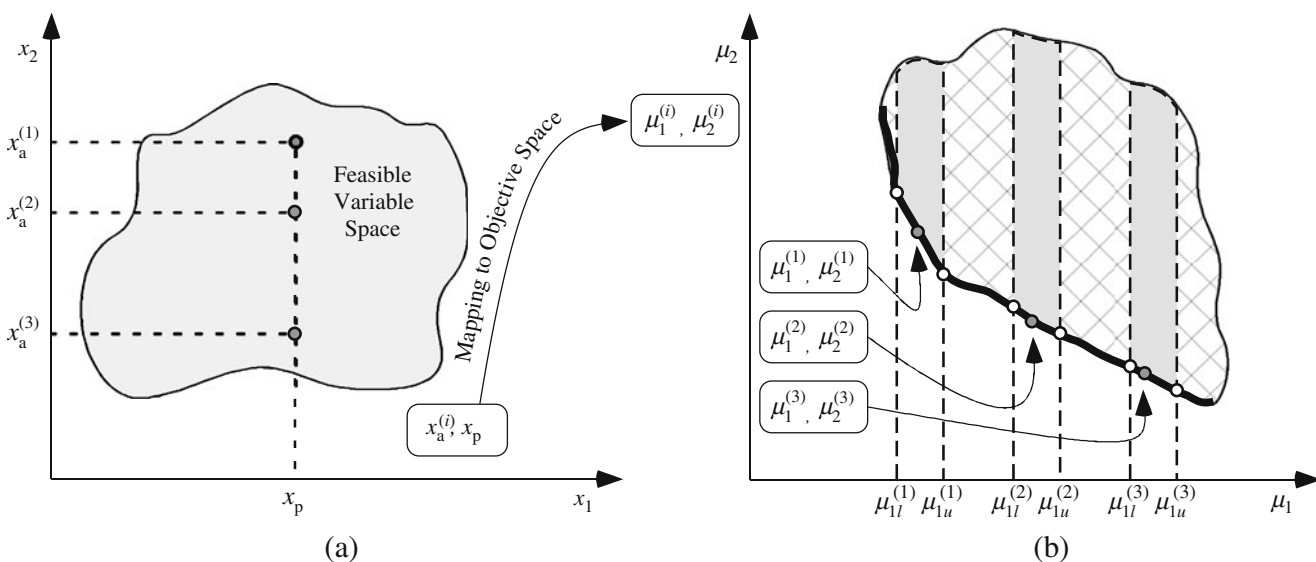


Fig. 6 Theoretical identification of the values of x_p and $x_a^{(i)}$ for the set of anticipated regions of interest and shifted Pareto frontier from Fig. 5 and the optimization problem formulation presented in Problem 2a

bus-modular cases the maximum and minimum values for n_m obtained for all possible module progression sequences are identified as follows.

$$n_{m,\max} = \sum_{n=1}^{n_d-1} n \tag{14}$$

$$n_{m,\min} = n_d - 1 \tag{15}$$

it follows that the selected value of n_m is an integer satisfying the condition:

$$n_{m,\min} \leq n_m \leq n_{m,\max} \tag{16}$$

Using the integer value of n_m that is desired, it is now possible to create an n_m -by-2 matrix (δ) dictating the desired progression from one design contained in set D to another. As a note, the first entry of the δ matrix ($\delta_{1,1}$) is generally the platform design from set D identified in the previous section. A generic construction of a δ matrix is presented as follows:

$$\delta = \begin{bmatrix} \alpha_1 & \beta_1 \\ \alpha_2 & \beta_2 \\ \vdots & \vdots \\ \alpha_{n_m} & \beta_{n_m} \end{bmatrix} \tag{17}$$

where α and β respectively refer to the starting and the ending designs of the set D that each module is bridging. This information is used in the final step to refer to the values of x_a needed to design each module.

Identify and calculate the values of module design variables: The identification of module designs first requires that the development of the design model(s) describing the module(s) be completed, and those variables that are best suited to characterize the modules be identified—module variables (x_m). This identification of module variables can be performed using the same methods described previously for identifying platform variables (i.e., Pareto filtering, concept generation, or any other suitable method). In cases where the variables that are best suited as module variables for manufacturing a modular product are known, this process of variable identification simplifies to the providing of that information for the module design routine presented below. Using this information, and the information provided in δ and D , a generic constrained module design routine is presented below.

Problem 2b Optimization problem formulation for constrained module design

$$D_m := \left\{ \left(x_{p,1}, x_{p,2}, \dots, x_{p,n_{x_p}}, x_{m,1}^{(i)}, x_{m,2}^{(i)}, \dots, x_{m,n_{x_m}}^{(i)} \right) \mid \forall i \in \{1, 2, \dots, n_m\} \right\} \tag{18}$$

x_m is defined by:

$$\min_{x_m} J^{(i)} = \left\| P^{(\beta)} - \bar{P}^{(i)} \right\| \tag{19}$$

where:

$$\alpha = \delta_{i,1} \tag{20}$$

$$\beta = \delta_{i,2} \tag{21}$$

$$\bar{P}^{(i)} = P^{(\alpha)} + \Delta P^{(i)} \tag{22}$$

defined by:

$$P^{(\alpha)} = \left(\mu_1(x_a^{(\alpha)}, x_p, p^{(\alpha)}), \mu_2(x_a^{(\alpha)}, x_p, p^{(\alpha)}), \dots, \Delta\mu_{n_\mu}(x_a^{(\alpha)}, x_p, p^{(\alpha)}) \right) \quad (n_\mu \geq 2) \tag{23}$$

$$P^{(\beta)} = \left(\mu_1(x_a^{(\beta)}, x_p, p^{(\beta)}), \mu_2(x_a^{(\beta)}, x_p, p^{(\beta)}), \dots, \Delta\mu_{n_\mu}(x_a^{(\beta)}, x_p, p^{(\beta)}) \right) \quad (n_\mu \geq 2) \tag{24}$$

$$\Delta P^{(i)} = \left(\Delta\mu_1(x_m^{(i)}, x_p, \hat{p}^{(i)}), \Delta\mu_2(x_m^{(i)}, x_p, \hat{p}^{(i)}), \dots, \Delta\mu_{n_\mu}(x_m^{(i)}, x_p, \hat{p}^{(i)}) \right) \quad (n_\mu \geq 2) \tag{25}$$

where D_m is the set of values and variables of x_p and x_m for each module design; $P^{(\alpha)}$ and $P^{(\beta)}$ are vectors containing the design objective values of the base (α) and target (β) designs; $\bar{P}^{(i)}$ is a vector containing the design objective values of design α when used in conjunction with the i th module; $\Delta P^{(i)}$ is a vector containing the change in design objective values from design α to $\bar{P}^{(i)}$; and x_m represents the value(s) and variable(s) that characterize ΔP .

In examining Problem 2b it is seen that for each design in the set D_m , the values of x_p are the same as those contained in set D . Also, if the variables contained in x_a are geometric (i.e., lengths, widths, heights), x_m represents the change of the geometric values of the variables that produce the desired $\Delta P^{(i)}$. If the variables contained in x_a are non-geometric (i.e., technology selection, hardware selection, software selection), x_m provides the information needed to create the bridge between x_a from design α to design β and provide the desired $\Delta P^{(i)}$.

With completion of the constrained module design process, a product capable of adapting to changes in needs over time through the addition of modules is achieved. In

addition, each iteration of the product obtained through the addition of modules provides the optimal performance according to the objectives provided in Problem 2a (see Section 3.4).

In the following section, the development of a simple, modular unmanned air vehicle is provided as an example of the implementation of the method described in Section 3.

4 Example: Unmanned Air Vehicle (UAV)

The example that follows shows the application of the multiobjective optimization design method in the creation of a small (wingspan (b) 2.5 m) modular UAV. In addition, the ability of the method developed in the previous section to provide platform and module designs capable of satisfying three different operating conditions and parameters representing changing customer needs over time is shown.

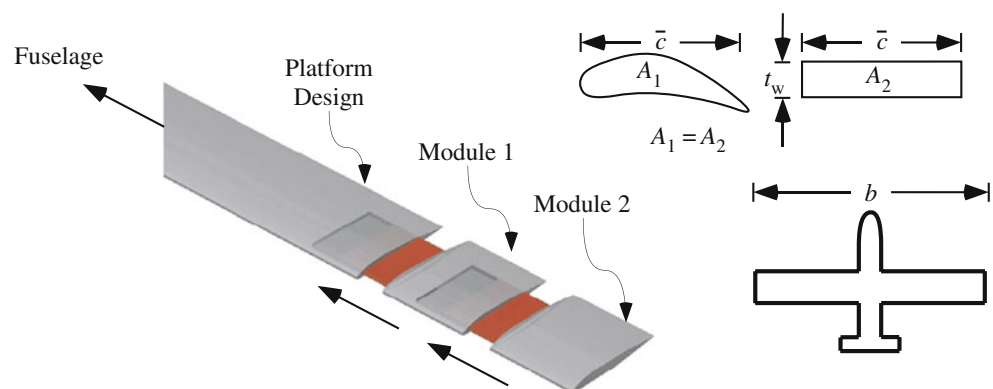
Others have shown the benefits of using multiobjective optimization based methods in UAV design. For example, in Gonzalez et al. (2007), a framework in which optimization problems can be analyzed along with the use of evolution strategies incorporating concepts of multiobjective optimization, hierarchical topology, asynchronous evaluation of candidate solutions, and parallel computing are implemented in bi-objective (drag and weight) UAV wing plan-form optimization examples. In Jun et al. (2006) a new method of collaborative optimization (multi-level decomposed methodology for a large-scale multidisciplinary design optimization) is presented and implemented in the design of an aircraft wing. Rajagopal and Ganguli (2008) demonstrate the use of a Multiobjective Genetic Algorithm (MOGA) in the bi-objective (endurance and wing weight) optimization of a low speed, long endurance UAV. In Viswamurthy and Ganguli (2007, 2009) optimization techniques are used to obtain the Pareto frontier for a bi-objective (hub vibration and flap control power) optimization of the location of trailing-edge flaps on a helicopter.

In each case, as seen in these articles, multiobjective optimization has extended the designer's capabilities and made newer and better designs achievable. The developments presented in the present paper also extend the capabilities of the designer and make newer and better designs achievable. Specifically, the method enables the designer to anticipate and plan for changing needs, and to subsequently design modular UAV's capable of being optimally adjustable to the needs of differing mission profiles.

For the scale of aircraft addressed in this example, design traditionally involves the optimization of performance objectives for a set of operating conditions and mission parameters, traditionally expressed through a mission profile. Drive for the development of this modular UAV stems from the losses in performance of the aircraft due to changes in operating conditions and parameters when the aircraft is used for missions other than it was designed for. To overcome this disparity, a concept wing design for a modular UAV is developed (see Fig. 7) to provide the ability to optimally expand the design of the aircraft (see Fig. 8) between missions through the addition of modules. The intent of this concept is not to provide details on the manner or mechanics of how these modules would be attached, but to simply provide an idea of how a UAV could be optimally expanded through modules.

Inspection of Figs. 7 and 8 reveal the platform variables best suited for manufacturing this concept design as the average chord length (\bar{c}). Assumptions made in this example are as follows: (1) During surveillance operations of the mission profiles, the UAV flies at constant altitude. (2) The aircraft has sufficient thrust for a sustained turn (Nigam and Kroo 2008). (3) The coefficients of lift ($C_{L_{max}}$) and thrust ($C_{T_{max}}$) are constant and equal 1.2 and 0.1 respectively (Nigam and Kroo 2008). (4) The UAV is being designed for remote location surveillance operations where useful data is not captured when the aircraft executes a turn. Additionally, the operator must carry all needed equipment. Therefore, the design objectives for this problem are to minimize the surveillance time lost per degree of UAV turn (\hat{T}_{lost}) and the total mass of the UAV (m_t). (5) Density (ρ_w) of the wing

Fig. 7 Schematic of a concept modular UAV wing design that provides the ability to optimally satisfy different mission profiles



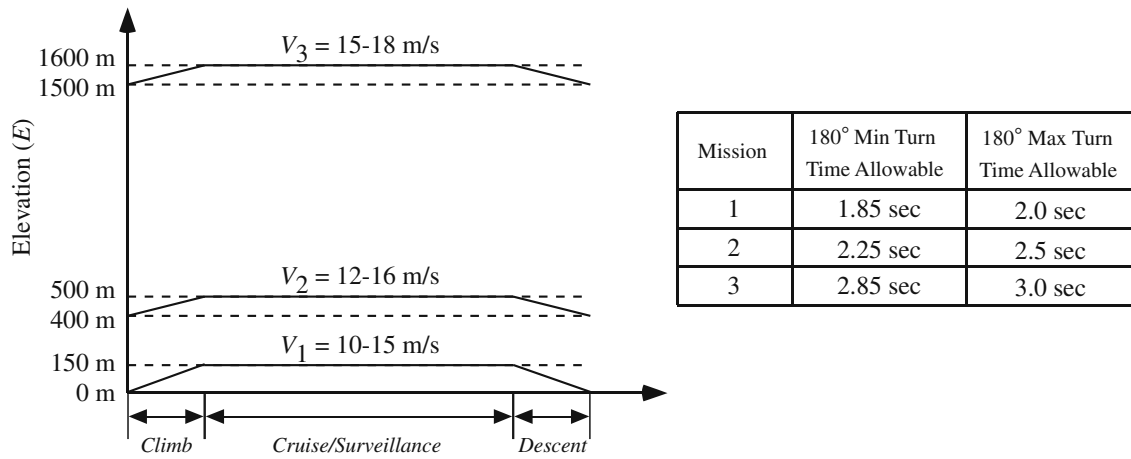


Fig. 8 Schematic of the three mission profiles used in the UAV example

material does not change with changes in b . As a note, the wing material is assumed to be 1.9 lb EPP foam. Current design approaches increase wing density where the wing connects to the fuselage as the wings lengthen to provide more strength (Wakayama and Kroo 1995). In the concept presented in Fig. 7, this final assumption requires that the connection between the fuselage and wings be designed for the maximum wing span possible. The mass associated with this connection is accounted for in m_f . The complete formulation of the single-objective optimization problem and identification of anticipated regions of interest for this example is as follows.

Problem 3a UAV example—optimization problem formulation

$$D := \{(x_p, x_a^{(i)}) \mid \forall i \in \{1, 2, 3\}\} \tag{26}$$

x_p, x_a defined by:

$$\min_{x_a^{(i)} x_p} \left(\frac{1}{3} \right) \sum_{i=1}^3 \left(\hat{T}_{\text{lost}}(x_a^{(i)}, x_p, p^{(i)}) + m_t(x_a^{(i)}, x_p, p^{(i)}) \right) \tag{27}$$

where:

$$x_p = \{\bar{c}\} \tag{28}$$

$$x_a^{(i)} = \{ V^{(i)} \quad b^{(i)} \} \tag{29}$$

$$p^{(i)} = \{ m_e \quad m_f \quad \rho_w \quad t_w \quad T_0 \quad L \quad \dot{p}_0 \quad R \quad M \quad E^{(i)} \} \tag{30}$$

subject to:

$$\hat{T}_{\text{lost}}^{(i)} - \hat{T}_{\text{lost, max}}^{(i)} \leq 0 \tag{31}$$

$$\hat{T}_{\text{lost, min}}^{(i)} - \hat{T}_{\text{lost}}^{(i)} \leq 0 \tag{32}$$

$$V_l^{(i)} \leq V^{(i)} \leq V_u \tag{33}$$

$$1.1\text{m} \leq b^{(i)} \leq 2.5\text{m} \tag{34}$$

$$0.09\text{m} \leq \bar{c} \leq 0.17\text{m} \tag{35}$$

where:

$$\hat{T}_{\text{lost}}^{(i)} = \frac{\hat{D}_{\text{turn}}^{(i)}}{V^{(i)}} \tag{36}$$

$$m_t^{(i)} = m_e + m_f + m_w^{(i)} \tag{37}$$

with supporting equations:

$$m_w^{(i)} = \rho_w S_{\text{ref}}^{(i)} t_w \tag{38}$$

$$S_{\text{ref}}^{(i)} = \bar{c} b^{(i)} \tag{39}$$

$$T^{(i)} = T_0 - LE^{(i)} \tag{40}$$

$$\dot{p}^{(i)} = \dot{p}_0 \left(1 - \frac{LE^{(i)}}{T_0} \right)^{\left(\frac{\dot{g}M}{RL} \right)} \tag{41}$$

$$\rho^{(i)} = \frac{\dot{p}^{(i)} M}{RT^{(i)}} \tag{42}$$

$$\eta_{\text{max}}^{(i)} = \frac{\rho^{(i)} (V^{(i)})^2 S_{\text{ref}}^{(i)} C_{L\text{max}}}{2m_t^{(i)} \hat{g}} \tag{43}$$

Table 1 Values of the constant parameters of $p^{(i)}$ needed to evaluate Problem 3a

m_e (kg)	m_f (kg)	ρ_w (kg/m ³)	t_w (m)	T_0 (K)	L (K/m)	\dot{p}_0 (Pa)	R (J/(mol·K))	M (kg/mol)
0.25	1.5	30.435	0.06	288.5	0.0065	101325	8.31447	0.0289644

$$R_{\text{turn}}^{(i)} = \frac{(V^{(i)})^2}{\hat{g}\sqrt{(\eta_{\text{max}}^{(i)})^2 - 1}} \tag{44}$$

$$D_{\text{turn}}^{(i)} = \frac{\pi R_{\text{turn}}^{(i)} \theta}{180} \tag{45}$$

$$\hat{D}_{\text{turn}}^{(i)} = \frac{D_{\text{turn}}^{(i)}}{\theta} \tag{46}$$

$$T_{\text{lost}}^{(i)} = \frac{D_{\text{turn}}^{(i)}}{V^{(i)}} \tag{47}$$

where each of the terms in the preceding equations is defined in the Nomenclature section of this paper.

Values of the elevation (E), $\hat{T}_{\text{lost, max}}$, and $\hat{T}_{\text{lost, min}}$, along with the lower (V_l) and upper (V_u) limits of the mission cruise velocities for the different designs presented in Tables 1 and 2 are obtained from the mission profiles presented in Fig. 8. Values of T_0 , L , \dot{p}_0 , R , and M presented in Table 1 come from the 1976 International Standard Atmosphere document (ISO 1975). The variable t_w represents an equivalent thickness of the wings—approximates the wing cross sectional area as a rectangle (See Fig. 7). Equations used to evaluate the unmanned air vehicle’s objective space performance (see (36) and (37)) are derived from equations presented in Nigam and Kroo (2008, see (43) and (44) above). Equations used to calculate the density of air (see (41–43) above) as a function of E are obtained using the ideal gas law assumption (Shelquist Engineering 2009). Evaluation of Problem 3a was performed using a Multiobjective Genetic Algorithm, and complete results indicating the variable and objective values of each design are presented in Table 3.

Through the evaluation of Problem 3a above, the set D now contains all variable values needed to develop the module designs (see Table 3). Prior to developing the module designs, information on the type, number, and desired

Table 2 Values of the non-constant objective limits, adjustable variable limits, and parameters of $p^{(i)}$ needed in Problem 3a to define the anticipated regions of interest and obtain the i th design of set D

i	Adjustable variable limits		Objective limits		Parameters E (m)
	V_l (m/s)	V_u (m/s)	$\hat{T}_{\text{lost, min}}$ (s/°)	$\hat{T}_{\text{lost, max}}$ (s/°)	
1	10	15	1.85/180	2.0/180	150
2	12	16	2.25/180	2.5/180	500
3	15	18	2.85/180	3.0/180	1,600

progression of modules that are to be used to obtain the Pareto-optimal designs contained within set D is needed. Using the information provided in Fig. 7, it can be seen that a Bus-modular approach was selected for this example. Examination of the nature of the x_a variables reveals that the differences in the variable b for each design in D is geometric, and therefore the design with the most commonality is the design with the smallest length of b ($D^{(3)}$). Using this information, the desired number of modules to be developed (n_m) is chosen to be two, and the δ matrix is constructed in the following equation:

$$\delta = \begin{bmatrix} 3 & 2 \\ 2 & 1 \end{bmatrix} \tag{48}$$

Formulation of a constrained module design routine of the form presented in Problem 2b is presented as follows.

Problem 3b UAV example—constrained module design

$$D_m := \left\{ \left(x_{p,1}, x_{p,2}, \dots, x_{p,n_{xp}}, x_{m,1}^{(i)}, x_{m,2}^{(i)}, \dots, x_{m,n_{xm}}^{(i)} \right) \mid \forall i \in \{1, 2\} \right\} \tag{49}$$

x_m is defined by:

$$\min_{x_m} J^{(i)} = \left\| P^{(\beta)} - \bar{P}^{(i)} \right\| \tag{50}$$

defined by:

$$\bar{P}^{(i)} = P^{(\alpha)} + \Delta P^{(i)} \tag{51}$$

$$P^{(\alpha)} = \left(\hat{T}_{\text{lost}}^{(\alpha)}, m_t^{(\alpha)} \right) \tag{52}$$

$$P^{(\beta)} = \left(\hat{T}_{\text{lost}}^{(\beta)}, m_t^{(\beta)} \right) \tag{53}$$

$$\Delta P^{(i)} = \left(\Delta \hat{T}_{\text{lost}}^{(i)}, \Delta m_t^{(i)} \right) \tag{54}$$

Table 3 Variable and objective values obtained through evaluation of Problem 3a for the i th design of set D

i	Variables			Objectives		
	\bar{c} (m)	V (m/s)	b (m)	m_t (kg)	\hat{T}_{lost} (s/°)	$\Rightarrow T_{\text{lost}}$ (s)
1	0.17	15	2.4635	2.5147	0.0103	1.8617
2	0.17	16	1.8259	2.3168	0.0127	2.2923
3	0.17	18	1.3857	2.1802	0.0159	2.8708

Table 4 Variable values of the selected platform design needed in the evaluation of Problem 3b

\bar{c} (m)	b (m)	V (m/s)
0.17	1.3857	18

where:

$$x_m^{(i)} = \{L_m^{(i)}, V^{(\beta)}\} \tag{55}$$

$$x_p = \{\bar{c}\} \tag{56}$$

$$\alpha = \delta_{i,1} \tag{57}$$

$$\beta = \delta_{i,2} \tag{58}$$

$$\begin{aligned} \Delta \hat{T}_{lost}^{(i)} &= \hat{T}_{lost} \left((b^{(\alpha)} + 2L_m^{(i)}), V^{(\beta)}, \bar{c}, \hat{p}^{(\beta)} \right) \\ &\quad - \hat{T}_{lost} \left(b^{(\alpha)}, V^{(\alpha)}, \bar{c}, \hat{p}^{(\alpha)} \right) \end{aligned} \tag{59}$$

$$\begin{aligned} \Delta m_t^{(i)} &= m_t \left((b^{(\alpha)} + 2L_m^{(i)}), V^{(\beta)}, \bar{c}, \hat{p}^{(\beta)} \right) \\ &\quad - m_t \left(b^{(\alpha)}, V^{(\alpha)}, \bar{c}, \hat{p}^{(\alpha)} \right) \end{aligned} \tag{60}$$

$$L_m^{(i)} = 0.5(b^{(\beta)} - b^{(\alpha)}) \tag{61}$$

The variable values of the platform design needed to evaluate Problem 3b are presented in Table 4. The results of the evaluation of Problem 3b are presented in Table 5.

Results of the evaluation of Problem 3b above (see Table 5 for complete summary) provide the variable values needed to describe the module designs. With completion of the constrained module design process, a UAV capable of adapting to three different mission profiles through the addition of modules is achieved. Figure 9 provides a graphical representation of the Pareto frontier for the 3 regions of interest defined in Problem 3a, along with the objective values for the solutions to Problem 3a (indicated by the symbol “o”) and Problem 3b (indicated by the symbol “x”). Confidence in the representation of the Pareto frontier provided in Fig. 9 is two-fold: (1) A comparison of the desired boundary points of each region of interest provided in Table 2 and the end points of the Pareto frontier

Table 5 Variable values of the module designs (i) obtained through evaluation of Problem 3b

i	L_m (m)	V (m/s)
1	0.2201	16
2	0.3188	15

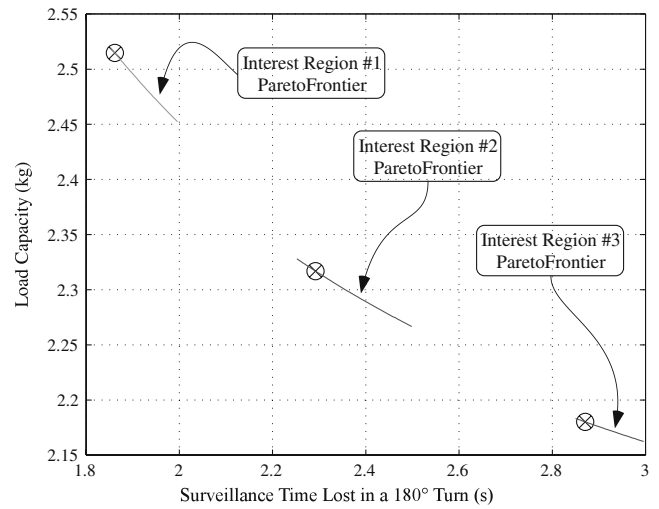


Fig. 9 Graphical representation of the Pareto frontier for the three regions of interest defined in Problem 3a, along with the plotted solutions to Problem 3a (indicated by the symbol “o”) and Problem 3b (indicated by the symbol “x”). The plot shows that each iteration of the UAV obtained through the addition of modules provides the desired Pareto-optimal performance from Problem 3a

within each interest region of Fig. 9 reveals that the obtained boundary points match those provided in Table 2. (2) The solid line approximation of the Pareto frontier in each region of interest was obtained by connecting straight lines to a set of 60 Pareto points within each region of interest which were distributed between the corresponding boundary points. In addition, Fig. 9 illustrates the ability of each UAV iteration obtained through the addition of modules to provide the desired Pareto-optimal performance according to the objectives, parameters, and constraints provided in Problem 3a, and thus satisfy the intent of the design method.

5 Concluding remarks

This paper addresses an important limitation of current methods of module-based product design in accounting for natural changes in performance and functional needs over time. In response to this limitation, a multiobjective optimization design method has been developed and demonstrated in this paper. The presented approach involves the strategic use of a series of optimization formulations that ultimately result in modular products that can adapt to changing needs by moving from one design on the Pareto frontier to another through the addition of a module. While more traditional approaches focus on changes in needs across market segments, the present approach enables the design of a new kind of product that is based on the natural changes over time of one segment. The method, as presented in this paper, is broadly applicable to diverse applications.

As illustrated, in the case of a UAV, the method can be used to provide designs based on known changing mission profiles. For other diverse cases, such as developing inexpensive income-generating platforms that can be augmented over time to increase income generation capacity for the impoverished people of developing countries, the method is still capable of providing designs based on predicted changes in needs. This method, combined with traditional approaches of product design and development of product families, also provides a new framework for the selection of platform and module designs.

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

References

- Baguley P, Bramall D, Maropoulos P, Page T (2004) The automation of sustainability via re-use, modularization, and fuzzy logic. In: Design and manufacturing for sustainable development 2004. Professional Engineering Publishing
- Balling RJ (2000) Pareto sets in decision-based design. *Engineering Valuation and Cost Analysis* 3:189–198
- Belegundu A, Chandrupatla T (1999) Optimization concepts and applications in engineering. Prentice Hall, NJ
- Chen W, Wiecek MM, Zhang J (1999) Quality utility—a compromise programming approach to robust design. *ASME J Mech Des* 121
- Das I (1999) A preference ordering among various pareto optimal alternatives. *Struct Multidisc Optim* 18(1):30–35
- Fujita K, Sakaguchi H, Akagi S (1999) Product variety deployment and its optimization under modular architecture and module commonalization. In: ASME design engineering technical conferences—design for manufacturing. Las Vegas, NV, DETC99/DFM-8923
- Gonzalez LF, Periaux J, Damp L, Srinivas K (2007) Evolutionary methods for multidisciplinary optimization applied to the design of UAV systems. *Eng Optim* 39(7):773–795
- Gonzalez-Zugasti J, Otto K (2000) Modular platform-based product family design. In: Proceedings of the ASME design automation conference. DETC2000/DAC-14238
- ISO (1975) International standard atmosphere. International Organization for Standardization (ISO), ISO 2533:1975
- Jun S, Jeon YH, Rho J, Lee DH (2006) Application of collaborative optimization using genetic algorithm and response surface method to an aircraft wing design. *JMST* 20(1):133–146
- Kasprzak EM, Lewis KE (2000) An approach to facilitate decision tradeoffs in pareto solution sets. *Engineering Valuation and Cost Analysis* 3:173–187
- Li Y, Fadel GM, Wiecek MM (1998) Approximating pareto curves using the hyper-ellipse. In: Proceedings of the 7th AIAA/USAF/NASA/ISSMO symposium on multidisciplinary analysis and optimization. AIAA-98-4961
- Li Y, Xue D, Gu P (2008) Design for product adaptability. *Concurr Eng* 16(3):221–232
- Magrab E (1997) Integrated product and process design and development. CRC Press
- Mattson CA, Magleby SP (2001) The influence of product modularity during concept selection of consumer products. In: Proceedings of the ASME DETC'01, ASME 2001 design engineering technical conferences, and computers and information in engineering conference. DETC2001/DTM21712
- 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. *OPTE: Optimization and Engineering* 6(1):85–115
- Messac A, Mattson CA (2002) Generating well-distributed sets of Pareto points for engineering design using physical programming. *Optimization and Engineering* 3(4):431–450. Kluwer
- Miettinen KM (1999) Nonlinear multiobjective optimization. International series in operations research & management science. Kluwer
- Nigam N, Kroo I (2008) Control and design of multiple unmanned air vehicles for a persistent surveillance task. In: 12th AIAA/ISSMO multidisciplinary analysis and optimization conference. Victoria, British Columbia, AIAA-2008-5913
- Oyebode A (2004) Modularity and quality. In: Erixon G, Kenger P (eds) Proceedings from the 2nd seminar on development of modular products. Campus Framtidsdalen, Dalarna University, Sweden, pp 19–24
- Padamat M (2004) Methods for modularisation. In: Erixon G, Kenger P (eds) Proceedings from the 2nd seminar on development of modular products. Campus Framtidsdalen, Dalarna University, Sweden, pp 121–126
- Pahl G, Beitz W (2007) Engineering design: a systematic approach. Springer
- Park KW, Grierson DE (1999) Pareto-optimal conceptual design of the structural layout of buildings using a multicriteria genetic algorithm. *Comput-Aided Civil Infrastruct Eng* 14:163–170
- Rajagopal S, Ganguli R (2008) Conceptual design of long endurance UAV using kriging approximation and multi-objective genetic algorithm. *Aeronaut J* 112(1137):653–662
- Shelquist Engineering (2009) Air density and density altitude calculations. http://wahiduddinnet/calc/density_altitude.htm. Accessed 5 Nov 2009
- Simpson TW (1998) Product platform design and optimization: status and promise. *Artif Intell Eng Des Anal Manuf* 18(1):3–20
- Steuer RE (1986) Multiple criteria optimization, theory computations and applications. Wiley, New York
- Tappeta RV, Renaud JE, Messac A, Sundaraj GJ (2000) Interactive physical programming: tradeoff analysis and decision making in multidisciplinary optimization. *AIAA J* 38(5):917–926
- Trope Y, Liberman N (2000) Temporal construal and time-dependent changes in preference. *J Pers Soc Psychol* 79(6):876–889
- 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. *CIRP Ann (Manufacturing Technology)* 45(1):153–156
- Ulrich K, Eppinger S (2004) Product design and development, 4th edn. McGraw-Hill
- Viswamurthy SR, Ganguli R (2007) Optimal placement of trailing edge flaps for helicopter vibration reduction using response surface methods. *Eng Optim* 39(2):185–202
- Viswamurthy SR, Ganguli R (2009) Optimal placement of piezoelectric actuated trailing-edge flaps for helicopter vibration control. *J Aircr* 26(1):244–253
- Wakayama S, Kroo I (1995) Subsonic wing planform design using multidisciplinary optimization. *J Aircr* 32(4):746–753
- Wie MV, Greer J, Campbell M, Stone R, Wood K (2001) Interfaces and product architecture. In: Proceedings of the ASME design theory and methodology conference. DETC2001/DTM-21689
- Wiecek M, Blouin VY, Fadel GM, Engau A, Hunt B, Singh V (2009) Multi-scenario multi-objective optimization in engineering design, vol 618, XV edn. Springer lecture notes in economics and mathematical systems. LNEMS volume of MOPGP

- Wu J, Azarm S (2001) Metrics for quality assessment of a multi-objective design optimization solution set. *ASME J Mech Des* 123:18–25
- Yang TG, Beiter KA, Ishii K (2004) Product platform development: an approach for products in the conceptual stages of design. In: 2004 ASME international mechanical engineering congress and RDD expo. IMECE2004-62171
- Yearsley JD, Mattson CA (2008a) 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
- Yearsley JD, Mattson CA (2008b) Product family design using a smart pareto filter. In: 46th AIAA aerospace sciences meeting and exhibit. Reno, Nevada, AIAA 2008-909
- Yearsley JD, Mattson CA (2008c) Product family member and platform identification with concurrent variable and objective space smart Pareto filtering. In: 4th AIAA multidisciplinary design optimization specialist conference. AIAA-2008-2220