12th AIAA Aviation Technology, Integration, and Operations (ATIO) Conference and 14th AIAA/ISSM
17 - 19 September 2012, Indianapolis, Indiana

AIAA 2012-5444

Including Principles of Set-Based Design in Multidisciplinary Design Optimization

Shari E. Hannapel[1], Nickolas Vlahopoulos[2], and David J. Singer[3]
University of Michigan, Ann Arbor, MI, 48105

The traditional point-based design process is highly iterative and can be inefficient, particularly for for multidisciplinary problems. Set-based design constitutes a design space reduction process that offers improvements over the point-based approach. In set-based design, designers begin with a broad set of values for the design variables, then gradually narrow the sets as more information becomes available. The principles of set-based design are reflected in the development of a new multidisciplinary design optimization algorithm. The algorithm returns the optimal choice for the reduced design space, instead of a single specific value for each design variable. The new multidisciplinary design optimization algorithm inspired by set-based design was used in a multidisciplinary ship design application.

I. Introduction

THE conventional design approach is point-based and iterative. Designers begin by selecting a variety of possible solutions, then choosing one to investigate further. The single design is evaluated and modified as necessary, then re-evaluated and changed again, in an iterative process which continues until a satisfactory design has been found. This process can be inefficient because new designs must continually be evaluated. The problem becomes more complex for multidisciplinary problems, with multiple likely-conflicting discipline analyses, where different disciplines may even be the responsibility of different teams of engineers.

Set-based design is a design methodology that seeks to offer improvements over the traditional point-based design approach. Set-based design was popularized in a series of articles[1-3] discussing Toyota's success with set-based design practices. The fundamental idea behind set-based design is to utilize sets of values for the design variables so that engineers communicate about the design in terms of those sets, instead of points. Singer, Doerry, and Buckley[4] outline the following four features of set-based design:

1. Initially define a broad set of values for the design parameters.
2. Delay narrowing the sets to increase the amount of information available when making decisions.
3. Narrow the sets gradually as the design is improved.
4. Increase the design fidelity as the sets are narrowed.

In set-based design, engineers from different areas of the design (or disciplines) determine sets of feasible values for their own analysis, possibly also including preference information. Engineers share the set information to determine areas of feasible overlap, then gradually reduce the sets to focus on the feasible region.

In this paper, three advantages of set-based design are considered: (1) communicating with sets leads to less rework than point-based design, (2) delaying decisions means that decisions are better-informed, (3) working with sets and delaying decisions allows for better to handling of uncertainty during the design process.

The first advantage of set-based design is that it can reduce the amount of rework required in the design process. In point-based design, changes are made to the (single) design of interest when moving from one iteration to the next. It is unlikely that properties of the new design are the same as the previous iterations so the analyses for the design must be performed again with the new design characteristics. Therefore, a clear disadvantage of point-based design the constant rework and re-analysis[2]. In set-based design, engineers work with sets of values for the design

[1] Graduate Student, Department of Naval Architecture & Marine Engineering, 2600 Draper Dr., Ann Arbor, MI 48109, hannapel@umich.edu.
[2] Professor, Department of Naval Architecture & Marine Engineering, 2600 Draper Dr., Ann Arbor, MI, 48109, nickvl@umich.edu.
[3] Assistant Professor, Department of Naval Architecture & Marine Engineering, 2600 Draper Dr., Ann Arbor, MI, 48109, djsinger@umich.edu.

variables, and the sets are gradually narrowed as work progresses. Because the sets are being narrowed, no new designs are added to the design space[1], and the previous analysis performed on the larger set is still valuable for the reduced set.

The second advantage of set-based design is that design decisions are more informed because the decisions are delayed. The amount of information about a design increases with time, as more analyses are performed and requirements become more clear. Therefore, when design decisions are made early in the design process, there is less information available on which to base the decisions. This may lead to increased iterations with point-based design because it is highly unlikely that the first few iterations will be based on accurate information. One of the principles of set-based design is to delay decisions until more information is available, because designers are much more likely to make a good decision when they have more information[3].

Not only does utilizing sets and delaying decisions improve decision-making, but it also improves the process' response to uncertainty. By using sets of values for the design variables, the design maintains flexibility in an uncertain environment[3]. The advantage over the point-based method is expressed concisely by Lee[5]: "... designers can represent sets of design possibilities instead of guessing one design if there are uncertainties." With set-based design, small changes due to uncertainty do not necessarily push the design into an unfeasible region or require rework.

A relatively small number of analytical formulations for set-based design have become available in the literature. Wang and Terpenny[6] developed an evolutionary design synthesis procedure that handles populations of designs with the principles of genetic algorithms. The method utilizes fuzzy set theory to handle modeling inaccuracies, and principles of set-based design are included in the evolutionary procedure for generating and selecting designs.

Nahm and Ishikawa[7] developed a set-based design method which uses interval arithmetic to map from the design space to the performance space. The authors created an aggregated preference and robustness index that measures the designer's preference, and that includes robustness for handling uncertainty in the preference metrics. Shahan and Seepersad[8] developed a set-based design technique where Bayesian networks are used indicate the regions of interest in the design space for each discipline, and the networks are shared to communicate information about the design.

Madhavan et al.[9] developed a set-based design method where a compromise decision support problem (a mathematical model for multiobjective decision-making) is developed for each discipline. The disciplines calculate and share target values for coupled parameters, then the disciplines use the target values to generate Pareto optimal solutions. Malak, Aughenbaugh, and Paredis[10] developed an approach for conceptual design that handles imprecision, or the lack of specific knowledge for the design. The authors implemented features from multi-attribute utility theoryand set-based design.

Finally, Avigad and Moshaiov[11] developed a computational approach for multiobjective problems to incorporate the set-based design concept of intentionally delaying decisions about the design. The authors utilized a tree representation of the design space, where different branches represent different possible decisions, and the trees are pruned as decisions are made.

In this paper, the principles of set-based design are used in the formulation of a multidisciplinary design optimization (MDO) algorithm which studies the design variables in terms of sets. The techniques and advantages of set-based design are used to develop a novel MDO algorithm and the mathematical formulation of the new MDO algorithm is described. The new set-based design MDO algorithm is applied to a ship design analysis, and results are presented to demonstrate the effectiveness of the algorithm.

A. Problem Definition for Multidisciplinary Design Optimization

To ensure clear and consistent notation, a general MDO problem is defined as follows. Let there be n design variables contained in the vector $\mathbf{x}$, and for each design variable denote the allowable range of values as

$$x_i \in [x_{i,LB}, x_{i,UB}] \quad i = 1, \ldots, n \tag{1}$$

where the subscripts LB and UB indicate lower bounds and upper bounds, respectively. The space defined by the bounds on the design variables is denoted χ.

Let there be p disciplines in the MDO problem; each discipline has one objective function that depends on the values of the design variables, denoted $f_i(\mathbf{x})$, $i = 1, \ldots, p$. Let there be m_i constraints in discipline i contained in the vector $\mathbf{g}_i(\mathbf{x})$. Then the goal of the multidisciplinary design optimization is to solve

$$\min_{\mathbf{x}\in\chi} f_i(\mathbf{x}) \qquad i = 1, \ldots, p \tag{2}$$

$$\text{subject to} \quad \mathbf{g}_i(\mathbf{x}) \leq \mathbf{0}$$

The goal of the MDO algorithm is to solve a problem of the form of Eq. (2).

II. Multidisciplinary Design Optimization Algorithm

The new MDO algorithm inspired by the principles of set-based design is presented in this section. The new algorithm allows for greater flexibility when dealing with evolving requirements compared to a single-point optimization. The principles of set-based design which are included in the MDO algorithm are identified in the following sections and then transformed into mathematical statements for the optimization algorithm. Information is also presented from five areas of the algorithm: system design variables, objective function scaling, flexibility in constraints, the system optimization statement, and the discipline level optimization statement.

A. System Design Variable Definition

In the MDO problem of Eq. (2), there are n design variables contained in the vector $\mathbf{x}$, and p disciplines with p corresponding objective functions $f_i(\mathbf{x})$ which are contained in the vector $\mathbf{f}(\mathbf{x})$. The vector of constraints $\mathbf{g}_i(\mathbf{x})$ applies to discipline i. The allowable ranges for the design variables are denoted χ; the ranges for the design variables can be defined with lower and upper bounds:

$$x_i \in [x_{i,LB},\ x_{i,UB}] \quad i = 1, \ldots, n \tag{3}$$

The range of allowable values for x_i can be viewed as the set of values between the lower bound $x_{i,LB}$ and the upper bound $x_{i,UB}$.

As described in the previous section, one of the principles of set-based design is to describe the design variables by sets which change while the design progresses. Because the bounds on the design variables change during the optimization, the bounds on the design variables at any state are defined as

$$x_i \in [x_{i,min},\ x_{i,min} + \Delta x_i] \quad i = 1, \ldots, n \tag{4}$$

This expression indicates that the new lower bound on x_i is $x_{i,min}$ and the new upper bound on x_i is $x_{i,min} + \Delta x_i$, where Δx_i is the width of the interval for x_i. While it may seem simpler to define a maximum for x_i, instead of the sum $x_{i,min} + \Delta x_i$, the purpose of this formulation is to easily track the width of the interval.

Figure 1 illustrates the change in the bounds on the design variables for a simple case with only two design variables. The figure shows the original design space as the large outer rectangle, and the new design space is the smaller, lightly shaded region. The new design space is defined by the dashed lines which indicate the new ranges on the design variables.

The purpose of the new MDO algorithm is to use the principles of set-based design to motivate the design optimization. This means that the design space is changed through the optimization process; the design space is defined using the sets for the design variables defined in Equation (4). Therefore, the choice for the design variables in the system optimization statement are not the design variables $\mathbf{x}$, but the variables which define the sets: $x_{i,min}$ and Δx_i.

Additionally, before beginning the optimization, scaling of the design variables is important for accurate performance of the optimizer. The system design variables are scaled to take values between zero and one, denoted $z_{i,min}$ and Δz_i.

In summary, the system level optimization includes $2n$ design variables: $z_{i,min}$ and Δz_i ($i = 1, \ldots, n$), or as vectors $\mathbf{z}_{min}$ and $\Delta\mathbf{z}$; these design variables describe the size and location of

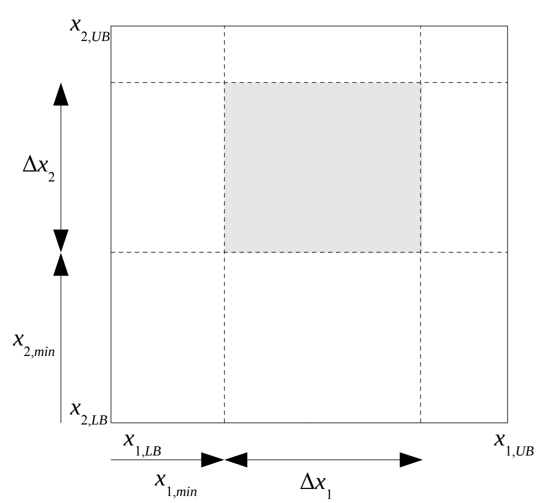

Figure 1. Illustration of the reduction in size of the design space by changing the ranges of the design variables.

the new design space. Furthermore, the system level design variables describe the design space in terms of sets for the design variables, which is a fundamental requirement for set-based design.

B. Objective Function Scaling

The MDO problem includes *p* objective functions which may have different units of measurement and vastly different magnitudes. Additionally, some objective functions may vary greatly throughout the design space, while others exhibit less variation. Therefore, in order to include any evaluation and comparison of the different objective functions, they must be scaled, or normalized. The objective functions are normalized as[12]:

$$F_i(\mathbf{x}) = \frac{f_i(\mathbf{x}) - f_i^*}{f_i^m - f_i^*} \tag{5}$$

where F_i is the normalized form of objective function f_i. f_i^* is the minimum of f_i when considering only discipline i, or

$$f_i^* = f_i(\mathbf{x}_i^*) = \min_{\mathbf{x}} f_i(\mathbf{x}) \tag{6}$$

$$\text{subject to} \quad \mathbf{g}_i(\mathbf{x}) \leq \mathbf{0}$$

f_i^m indicates the maximum value of f_i, and f_i^m can be approximated according to[12]

$$f_i^m = \max_j f_i(\mathbf{x}_j^*) \qquad i \neq j \tag{7}$$

Typically, the objective functions are conflicting (otherwise the problem is not a true multidisciplinary problem); then f_i^m is close to the worst performance that may be encountered in objective i.

Equation (5) yields values for the normalized objective function F_i in the range between 0 and approximately 1. The discipline optimum $\mathbf{x}_i^*$ gives the best possible performance for f_i, and at that point F_i is 0. The value of F_i can be interpreted as an amount of compromise in objective f_i; small values of F_i indicate little compromise, while values close to 1 indicate that the current point is far from f_i^*.

C. Flexibility in Constraints

When an engineer configures a real design, it may not be preferable (or possible) to select a design that exactly satisfies all of the constraints. Instead, it may be beneficial to relax a constraint if it allows for a significant improvement of the objective function. For example, consider a single-objective optimization problem where the objective is to maximize the speed of a vehicle, subject to a constraint for the maximum cost of the vehicle. For a real design problem, the designer may be willing to compromise on the cost constraint; if a small violation in the cost constraint (say, 1%) allows for a 20% increase in speed, the designer could choose to allow the slightly higher cost. One way to handle compromise in the constraints would be to determine the maximum allowable compromise, and simply shift the constraint by that amount. However, this does not ensure that the compromise in the constraint will correspond to a *significant* improvement in the objective function.

A method to handle the compromise in the constraints and the corresponding changes in the objective functions is included in the system level optimization of the new MDO algorithm. A new design variable denoted $\boldsymbol{\varepsilon}$ is introduced where each element in $\boldsymbol{\varepsilon}$ corresponds to a constraint in the MDO problem:

$$g_j(\mathbf{x}) \leq \varepsilon_j \tag{8}$$

This expression indicates that the original constraint g_j can be violated by as much as ε_j. However, because ε_j is a design variable, not a constant, the amount by which the constraint can be violated will vary during the optimization.

Furthermore, it is possible that flexibility is not desired in some constraints. These constraints are denoted $\mathbf{h}_i$ for discipline i and must satisfy

$$\mathbf{h}_i(\mathbf{x}) \leq \mathbf{0} \tag{9}$$

D. System Level Optimization Statement

The purpose of the system level optimization is to coordinate the discipline optimizations. The system optimization problem has $2n + m$ design variables, $\mathbf{z}_{min}$, $\Delta\mathbf{z}$, and $\boldsymbol{\varepsilon}$ (when n is the number of design variables for the original problem and m is the total number of constraints with flexibility). The system level optimization statement is:

$$\min_{\mathbf{z}_{min},\Delta\mathbf{z},\boldsymbol{\varepsilon}} \exp\left(\Delta z_1 \cdot \ldots \cdot \Delta z_n\right) + \sum_j \exp\left(\frac{\varepsilon_j}{\varepsilon_{j,\max}}\right) + \sum_{i=1}^{p} \exp\left(F_i\left(\mathbf{x}_i^{\text{new}}\right)\right) \tag{10}$$

$$\text{subject to} \quad g_j(\mathbf{x}_i^{\text{new}}) \le \varepsilon_j \quad \text{for all } j \text{ and } i = 1, \ldots, p$$

$$\mathbf{h}_k(\mathbf{x}_i^{\text{new}}) \le \mathbf{0} \quad \text{for } i = 1, \ldots, p \text{ and } k = 1, \ldots, p$$

The first term in the system objective function describes the size of the design space (the hypervolume $\Delta z_1 \cdot \ldots \cdot \Delta z_n$). The first term is used reduce the size of the design space as much as possible, because minimization of this term reduces the size of the design space. Therefore, this term implements the principle of set-based design to gradually reduce the size of the design space.

The second term in the system objective includes the compromise values $\boldsymbol{\varepsilon}$. During the optimization, the compromise values allow the constraint j to be violated by as much as ε_j. However, the amount of compromise should be as little as possible, thus the system objective function minimizes the compromise values. The summation over index j represents the summation over all constraints which allow for compromise; this allows for the possibility that some constraints cannot be relaxed.

The third term includes the effects of the discipline objective functions. The term includes the function F_i evaluated at the point $\mathbf{x}_i^{\text{new}}$, where $\mathbf{x}_i^{\text{new}}$ denotes the current value for the design variables returned by the discipline i optimization (details of the discipline level optimization are given in the following section). The purpose of the third term is to evaluate the current performance of the (normalized) objective functions. Because of the normalization, F_i takes values close to zero when $\mathbf{x}_i^{\text{new}}$ is close to $\mathbf{x}_i^*$; therefore, minimization of the sum of F_i seeks to minimize the discipline objective functions.

The system objective function is composed of a sum of terms that represent a multiobjective problem (the first term describes the size of the design space, the second term describes the constraint flexibility, and the third term describes the improvement in the disciplines' objective functions). Objective functions which are stated as a sum of other functions sometimes are not well-behaved, because the Pareto front for the corresponding multiobjective problem is not necessarily convex. In Eq. (10), the system objective function evaluates the exponential of each term. The purpose of this choice is to make the Pareto front convex[13], which can improve the performance of the optimization.

Finally, the system optimization statement includes the constraints g_j and $\mathbf{h}_k$ from all disciplines and the constraints are evaluated at each of the new discipline optima i; that is, every constraint is checked at each discipline optimum $\mathbf{x}_i^{\text{new}}$. The constraints ensure that all of the new discipline optima are feasible for all disciplines, or that all (new) discipline optima lie within the common feasible space. This enforces the concept of set-based design that designs must be found in the intersection of the sets. The solution of the system optimization returns the optimal values $\mathbf{z}_{min}^*$ and $\Delta\mathbf{z}^*$. The values describe the reduced design space, not a specific choice for the original design variables $\mathbf{x}$. This is in agreement with the set-based design perspective of viewing the design space in terms of sets instead of point designs.

E. Discipline Level Optimization Statement

The purpose of the discipline level optimization statement is to call the discipline analyses and calculate the new design point $\mathbf{x}_i^{\text{new}}$ for each discipline i. At each iteration in the optimization, discipline optimizations are performed, but with the bounds on the design variables determined by the current values of $\mathbf{z}_{min}$ and $\Delta\mathbf{z}$. The discipline optimization statement is

$$\min_{\mathbf{x} \in \chi(\mathbf{z}_{\min},\Delta\mathbf{z})} f_i(\mathbf{x}) \tag{11}$$

$$\text{subject to} \quad g_j(\mathbf{x}) \le \varepsilon_j \quad \text{for all } j \text{ in discipline } i$$

$$\mathbf{h}_i(\mathbf{x}) \le \mathbf{0}$$

The difference between this discipline optimization statement and the independent discipline optimization in Eq. (6) is the allowable ranges of the design variables, χ. For the discipline level optimization statements, χ is a function of the current values of the system design variables $\mathbf{z}_{min}$ and $\Delta\mathbf{z}$. Therefore, the discipline optimization is performed over the reduced design space described by $\mathbf{z}_{min}$ and $\Delta\mathbf{z}$, so the results for the discipline level optimization will be different from the discipline optimization of Eq. (6).

The discipline level optimization also includes the flexibility in the constraints, $\boldsymbol{\varepsilon}$. The optimization statement of Eq. (11) requires that the constraints g_j for discipline i are satisfied with the appropriate level of flexibility given by the corresponding values of ε_j. The values for $\boldsymbol{\varepsilon}$ are the current values from the system level optimization and they will change during the system optimization process. The discipline optimization statement also includes the constraints for discipline i without flexibility, $\mathbf{h}_i$.

The discipline level optimization with the new variable bounds $\chi(\mathbf{z}_{min}, \Delta\mathbf{z})$ returns the optimum values for the design variable values $\mathbf{x}_i^{new}$ and objective function value $f_i(\mathbf{x}_i^{new})$. The purpose of the discipline optimization is to locate the the discipline optimum within the current design space, defined by the sets at the system level; this ensures that the algorithm takes into account the discipline optima (or viewed as discipline preference), which is part of the set-based design approach.

III. Ship Multidisciplinary Design Optimization Application

The MDO algorithm using set-based design was applied to a complex ship design problem, using simulation software for performance evaluation. Fortran codes were used for conducting resistance, maneuvering, and seakeeping computations based on mathematical models from the literature. The results from the set-based design MDO algorithm are compared with results from a point-based multiobjective optimization.

A. Ship MDO Problem Definition

A tanker hull that was readily available from the website of the commercial naval architecture software MaxSurf was selected for this study. The design variables for ship model are length L, length-to-beam ratio L/B, and beam-to-draft ratio B/T. Ratios are selected as design variables instead of the dimensions (such as beam) because the mathematical models used for the discipline analyses have requirements on the ratios in order to be valid. The lower and upper bounds on the design variables ensure that the values for the length, length-to-beam ratio, and beam-to-draft ratio meet the requirements for the discipline analyses. The allowable ranges for the design variables are given in Table 1; the table also includes the values for the parent hull as a reference.

Table 1. Ranges for the design variables.

	Lower Bound	Upper Bound	Parent Hull
L (m)	250.0	320.0	315.0
L/B	5.60	8.00	6.20
B/T	2.25	3.75	3.00

While it is possible to use MaxSurf's parametric transformation capabilities to generate new hull forms based on the values of the design variables, including MaxSurf within the optimization loop was considered infeasible since manual operations are required for using the program. Instead, parametric models were developed to evaluate all of the information necessary to run the discipline analysis codes; the following section presents the parametric modeling information. The remainder of the section defines the MDO problem, which includes three disciplines: resistance, maneuvering, and seakeeping.

B. Hull Modeling

The codes for the discipline analyses require a variety of input information that describes the hull form. In order to run the discipline analyses, parametric models for the inputs were developed so that the necessary inputs could be calculated from the three design variables.

Several of the discipline analysis codes have requirements on the value for the block coefficient C_B, where

$$C_B = \frac{\Delta}{L\,B\,T} \tag{12}$$

All disciplines include the block coefficient constraints:

$$0.56 \le C_B \le 0.87 \tag{13}$$

C. Resistance Discipline

The objective function for the resistance discipline is the hull resistance, which is evaluated using the powering prediction method of Holtrop and Mennen[14-15]. The resistance discipline includes the two constraints on the block coefficient from Eq. (13). Then the resistance discipline optimization problem is

$$\min_{\mathbf{x}} \; R_{TOTAL} \tag{14}$$

$$\text{subject to} \quad \begin{array}{c} 0.56 - C_B \leq 0 \\ C_B - 0.87 \leq 0 \end{array}$$

where R_{TOTAL} denotes the total resistance of the hull, and **x** is the vector containing the three design variables L, L/B, and B/T.

D. Maneuvering Discipline

The second discipline is the maneuvering performance, which is evaluated using the Maneuvering Prediction Program (MPP)[16], which uses the methods of Clarke, Gedling, and Hine[17]. The objective function of the maneuvering discipline is the tactical diameter, which is to be minimized. The maneuvering discipline includes the constraints on the block coefficient from Eq. (13). The maneuvering discipline also includes a constraint on the stability criterion C, which is required to be positive. Then the maneuvering discipline optimization problem is

$$\min_{\mathbf{x}} \; D_T \tag{15}$$

$$\text{subject to} \quad \begin{array}{c} 0.56 - C_B \leq 0 \\ C_B - 0.87 \leq 0 \\ -C \leq 0 \end{array}$$

where D_T is the tactical diameter of the ship.

E. Seakeeping Discipline

The third discipline is the seakeeping performance, which is evaluated using the Seakeeping Prediction Program (SPP)[16] that is an implementation of the SCORES Program[18]. The objective function of the seakeeping discipline is a metric which represents the maximum combined motion:

$$M_{\max} = \max(\text{RMS heave}) + \frac{L}{2}\max(\text{RMS pitch}) + \frac{B}{2}\max(\text{RMS roll}) \tag{16}$$

where roll is evaluated at a 30 degree heading, heave is evaluated in beam seas, and pitch is evaluated at a 120 degree heading; the sum in Eq. (16) does not represent the actual motion of any part of the ship, but a multiobjective metric for the ship motion. The seakeeping discipline includes the constraints on the block coefficient from Eq. (13). The seakeeping discipline also includes a stability constraint on the metacentric height GM, where an approximation for the GM (as calculated by SPP) is required to be greater than the US Coast Guard GM requirement, GM_{USCG}. Then the seakeeping discipline optimization problem is

$$\min_{\mathbf{x}} \; M_{\max} \tag{17}$$

$$\text{subject to} \quad \begin{array}{c} 0.56 - C_B \leq 0 \\ C_B - 0.87 \leq 0 \\ \text{GM}_{\text{USCG}} - \text{GM} \leq 0 \end{array}$$

F. Cost Estimate

An additional constraint was introduced on the cost of the ship. A simple linear cost model based on the recommendations of Parsons[19] is used to evaluate the cost of the ship, where the subscript "0" indicates properties of the parent hull:

$$\text{cost}(\mathbf{x})=\left[1+1.01\frac{(L-L_0)}{L_0}+0.58\frac{(B-B_0)}{B_0}+0.40\frac{(T-T_0)}{T_0}+0.22\frac{(C_B-C_{B0})}{C_{B0}}\right] \tag{18}$$

Based on this cost formulation, the cost of the parent hull is 1 and variations can be viewed as percentage increases or decreases in cost from the parent hull (for example, the cost of another hull may be 1.02, which indicates a 2% increase in cost). A cost constraint is defined so that the cost of the new hull must be less than or equal to the cost of the parent hull:

$$g_{\text{cost}}(\mathbf{x}) = \text{cost}(\mathbf{x}) - 1 \leq 0 \tag{19}$$

Flexibility was only considered in the cost constraint; no flexibility was included in the other four constraints because the they help ensure that the discipline analyses run properly (for example, the seakeeping analysis code will error if the ship has a negative GM due to flexibility introduced in the corresponding constraint).

G. MDO Problem Summary

The disciplines can be optimized individually by solving the independent optimization statements in Eqs. (14), (15), and (17). Results for the independent discipline optimizations are given in Table 2; the results show that the optima occur at different points for the different disciplines, which is expected for a multidisciplinary problem.

Table 2. Independent discipline optimization results.

	Design Variables $\mathbf{x}^*$			Objective Function f_i^*
	L (m)	L/B	B/T	
Discipline 1	320.00	6.8889	2.2500	4.2301×10^9 N
Discipline 2	303.18	5.6000	3.5633	869.78 m
Discipline 3	314.22	5.7716	3.7291	22.178 m

Additionally, the constraints at the three discipline optima were evaluated and the results are summarized in Table 3. The Discipline 1 optimization statement includes only the constraints on the block coefficient, so it is not surprising that the optimum violates the constraint for GM. The conflicts between the different disciplines illustrate that this ship design problem comprises a good MDO example because compromise must be achieved to ensure that all discipline optima are feasible.

Table 3 also includes the cost constraint evaluated at the discipline optima; the cost constraint was not enforced during the discipline optimizations and all three discipline optima violate the cost constraint. Therefore, significant compromise is expected during the multidisciplinary analysis for either the cost constraint or the location of the final design space. Figure 2 summarizes the structure of the set-based design MDO problem.

Table 3. Independent discipline optimization constraint values.

	Discipline 1 Optimum	Discipline 2 Optimum	Discipline 3 Optimum
$g_1 = 0.56 - C_B$	-1.47×10^{-1}	-3.10×10^{-1}	-3.09×10^{-1}
$g_2 = C_B - 0.87$	-1.63×10^{-1}	-1.96×10^{-11}	-1.25×10^{-3}
$g_3 = -C$	-2.33×10^{-5}	-3.00×10^{-7}	-2.00×10^{-6}
$g_4 = \text{GM}_{\text{USCG}} - \text{GM}$	1.61	-6.66	-7.83
$g_{\text{cost}} = \text{cost} - 1$	0.075	0.034	0.062

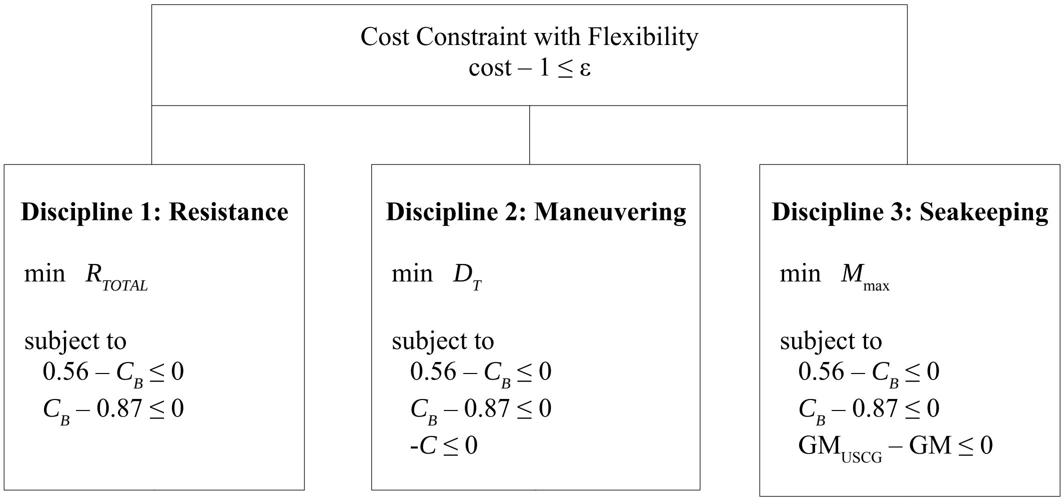

Fig. 2. Summary diagram of the ship design MDO problem.

H. Results from the Set-Based Design MDO Analysis

When implementing the set-based design MDO for the ship design example, the cost constraint from Eq. (19) was included at the system level. The system optimization optimization statement for the ship design problem is

$$\min_{\mathbf{z}_{min}, \Delta \mathbf{z}, \varepsilon} \quad \exp\left(\Delta z_1 \cdot \Delta z_2 \cdot \Delta z_3\right) + \exp\left(\frac{\varepsilon}{\varepsilon_{\max}}\right) + \sum_{i=1}^{3} \exp\left(F_i\left(\mathbf{x}_i^{\text{new}}\right)\right) \tag{20}$$

$$\text{subject to} \quad g_{\text{cost}}(\mathbf{x}_i^{\text{new}}) \leq \varepsilon \quad \text{for } i = 1, 2, 3$$

$$\mathbf{h}_k(\mathbf{x}_i^{\text{new}}) \leq \mathbf{0} \quad \text{for } k = 1, 2, 3 \text{ and for } i = 1, 2, 3$$

The maximum relaxation of the constraint $\varepsilon_{\max}$ was set to 0.1.

The results for the optimization are tabulated in Table 4. The upper half of the table shows the bounds on the new, reduced design space. The lower half of the table shows the properties for the discipline optimizations when performed in the reduced design space. The resulting value for the relaxation of the cost constraint is ε = 0.046.

The results are also shown in Figure 3, which shows the original design space and the reduced design space indicated by dashed lines. The individual discipline optima are marked with small dots and the discipline optima within the reduced space are marked with stars.

Table 4. Results from the set-based design inspired MDO algorithm.

Reduced design space results			
	Initial	Optimal	
L min	250.00	302.12	
L max	320.00	314.86	
L/B min	5.60	6.66	
L/B max	8.00	7.02	
B/T min	2.25	2.69	
B/T max	3.75	3.08	
Properties of the discipline optima in the new design space			
	Discipline 1	Discipline 2	Discipline 3
L^*	314.86	309.99	310.23
L/B^*	6.66	6.66	6.66
B/T^*	2.69	2.69	2.70
f_i^*	8.1882×10^9	995.30	25.566

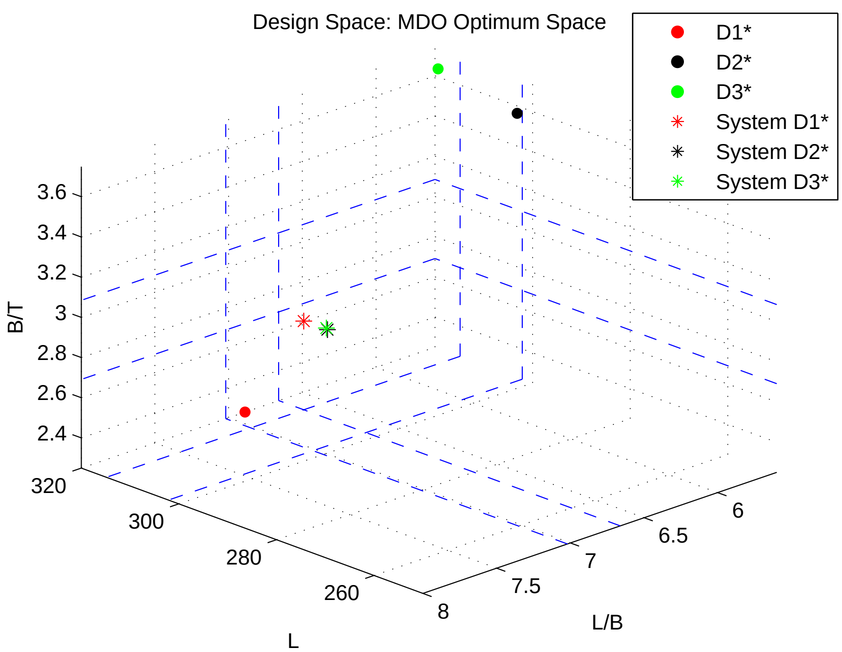

Fig. 3. Reduced design space returned by the set-based design MDO algorithm.

The relaxation in the constraint returned by the optimization was $\varepsilon = 0.046$. Relaxation of the cost constraint was expected because as shown in the single discipline optimizations (Table 3), much of the original design space did not satisfy the cost constraint, including the individual discipline optima. The results are valuable because the algorithm seeks to allow only the minimum relaxation necessary. Even though ε_{max} was 0.1, the new algorithm was able to find a suitable solution without having to relax the constraint to the maximum amount.

The set-based design MDO algorithm returns a reduced portion which represents preferred region of the original design space. The reduced space can be used as the design space for conducting a point-based optimization at later stages of the design process when the requirements are better defined.

I. Comparison to Point-Based Design

The results of the new set-based design MDO algorithm were compared to results from a single point design multiobjective optimization. The purpose of the comparison is to illustrate the advantages of the set-based design MDO algorithm by modeling flexibility in the constraints. For the comparison, three different point-based optimizations were performed:

1. Point-based optimization without the cost constraint in the original design space.
2. Point-based optimization with the relaxed cost constraint in the original design space.
3. Point-based optimization without the cost constraint in the reduced design space.

Case 1: The ship MDO problem was studied using a point-based optimization approach (a weighted sum of the objective functions) in the original design space. The point-based optimization included all of the disciplines' constraints but did not include the cost constraint. The optimization statement is:

$$\min_{\mathbf{x}\in\chi} \sum_{i=1}^{3} F_i(\mathbf{x}) \tag{21}$$

$$\text{subject to} \quad \mathbf{h}_{\mathrm{MO}}(\mathbf{x}) \le \mathbf{0}$$

where χ is the original design space defined in Table 1. F_i are the normalized objective functions, where the three disciplines objective functions (R_{TOTAL}, D_T, and $M_{\max}$) are scaled according to Eq. (5). The vector $\mathbf{h}_{\mathrm{MO}}$ contains the constraints from all three disciplines (a total of four constraints), but not the cost constraint. Omitting the cost constraint represents a situation where the design is initially driven by technical requirements and cost enters the decision-making process at a later stage.

The results of the point-based optimization are shown in Table 5 in the row labeled "Case 1," and the results are plotted with comparison to the individual discipline optima in Figure 4 (the plot is rotated to the same angle as Figure 3 for comparison).

Table 5. Comparison of results for multiple point-based optimizations.

Case	L^*	L/B^*	B/T^*	f_1^*	f_2^*	f_3^*	$g_{cost}(\mathbf{x}^*)$
1	314.91	5.6000	3.7473	11.963×10^9	989.2	23.296	0.081
2	314.89	5.6000	3.7471	11.972×10^9	989.0	23.295	0.081
3	314.00	6.6609	2.7000	8.999×10^9	1,057	26.917	0.029

The point-based solution satisfies the four constraints from the disciplines. The performance of the three discipline objective functions at the point-based optimum is inferior to the individual optima, as expected. However, upon visual inspection (Figure 4) the multiobjective solution may not be a good candidate for a solution because it is located in a corner of the design space; this does not illustrate much compromise between all three disciplines, and leaves almost no room to adjust the design should any changes in requirements occur.

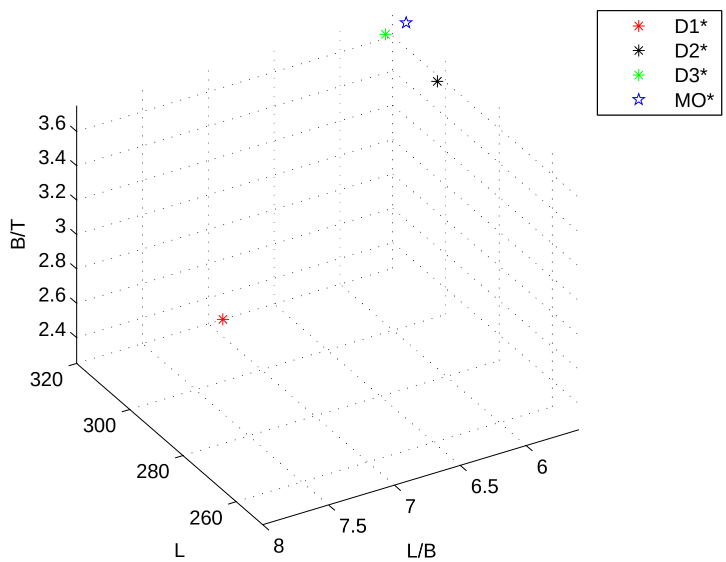

Fig. 4. Point-based optimization results.

Additionally, the cost constraint was not considered in the multiobjective optimization. Without accounting for cost, neither the discipline optima nor the multiobjective optimum satisfy the cost constraint; the value of the cost constraint is 0.081 at the multiobjective optimum. Starting from this specific design point, it would be difficult to approach the problem with the cost constraint because none of the current solutions are feasible.

Case 2: The ship MDO problem was next solved using a point-based optimization including the cost constraint in the original design space. The cost constraint was relaxed according to ε_{max} to model the situation that the designer has decided that the constraint may be violated by as much as ε_{max}; this is the same statement made for the set-based design MDO, except that the set-based design MDO algorithm can reduce the constraint relaxation during the optimization. The optimization statement is:

$$\min_{\mathbf{x}\in\chi} \sum_{i=1}^{3} F_i(\mathbf{x}) \tag{22}$$

$$\text{subject to} \quad \mathbf{h}_{MO}(\mathbf{x}) \le \mathbf{0}$$

$$g_{cost}(\mathbf{x}) \le \varepsilon_{max}$$

where $\mathbf{h}_{MO}$ contains all of the constraints from the disciplines. This optimization problem represents the situation where the designer has encountered a new, strict constraint during the point-based design process and the constraint has been relaxed in order to locate feasible solutions.

The results for the second point-based optimization are shown in Table 5 in the row labeled "Case 2." The solution is almost identical to the solution for Case 1; this result is expected because the solution to Case 1 yielded a cost constraint violation of 0.081, which is less than $\varepsilon_{max} = 0.1$. Addition of the cost constraint with the maximum relaxation of 0.1 has no effect on the solution because it is not active, and therefore the design from Case 2 has not incorporated any changes due to the cost requirements when compared to Case 1.

Case 3: To demonstrate the benefits of the set-based design MDO algorithm, a point-based optimization was conducted using the reduced design space determined by the set-based design MDO algorithm. The optimization statement is

$$\min_{\mathbf{x}\in\chi_S} \sum_{i=1}^{3} F_i(\mathbf{x}) \tag{23}$$

$$\text{subject to} \quad \mathbf{h}_{\text{MO}}(\mathbf{x}) \leq \mathbf{0}$$

where χ_S is the reduced design space located by the set-based design MDO algorithm (listed in Table 4). The cost constraint was not included in the optimization statement. The results of the multiobjective optimization are summarized in Table 5 in the row labeled "Case 3;" the results are clearly very different compared to the previous two cases. The results are plotted in Figure 5 along with contours of the cost constraint.

As Figure 5 shows, the discipline optima and the multiobjective optimum are closely grouped within the new design space; they are also located very close to the cost constraint boundary. Even though the cost constraint was not included in the multiobjective optimization, the value of the cost constraint at the multiobjective optimum is 0.029; not only is this a smaller constraint violation compared to the previous multiobjective optimization, but 0.029 is also less than the relaxation of the constraint from the MDO solution (ε = 0.046). Thus the single-point multiobjective optimization solution achieved by operating within the reduced design space satisfies the relaxed cost constraint even though the cost constraint was not included in the multiobjective optimization. This is accomplished because the cost constraint was accounted for when determining the reduced design space by the set-based design MDO algorithm.

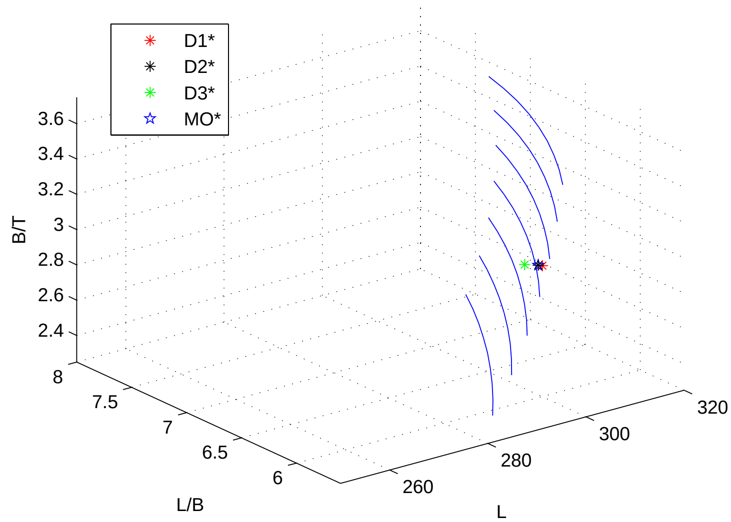

Fig. 5. Multiobjective optimization results for the new design space with contours of the cost constraint. *The constraint is satisfied on the left side of the contours.*

IV. Conclusion

This paper presents a new MDO algorithm developed using the principles of set-based design. The mathematical formulation is discussed and the performance of the algorithm is demonstrated through a ship design application. The multidisciplinary ship design application includes resistance, maneuvering, and seakeeping disciplines along with a simple cost estimate. The new set-based design MDO algorithm is used to identify a reduced design space.

Additionally, a multiobjective optimization is performed within the reduced design space located by the MDO algorithm, and the results are preferable to the results of a similar multiobjective optimization performed within the original design space. This ship design application illustrates that is it is valuable to first utilize a space-reducing technique (using sets to describe the design variables) before approaching a problem with a single point-based optimization. Furthermore, incorporating flexibility in the constraints of the set-based design MDO allows the optimization to handle a problem with very strict constraints in a rational manner and minimize the relaxation introduced in the constraints.

References

[1]Ward, A., Liker, J. K., Cristiano, J. J., and Sobek, D. K. II, "The second Toyota paradox: How delaying decisions can make better cars faster," *Sloan Management Review*, Vol. 36, No. 3, 1995, pp. 43-61.

[2]Liker, J. K., Sobek, D. K. II, Ward, A. C., and Cristiano, J. J., "Involving suppliers in product development in the United States and Japan: Evidence for set-based concurrent engineering," *IEEE Transactions on Engineering Management*, Vol. 43, No. 2, 1996, 165-178.

[3]Sobek, D. K. II, Ward, A. C., and Liker, J. K., "Toyota's principles of set-based concurrent engineering," *Sloan Management Review*, Vol. 40, No. 2, 1999, pp. 67-83.

[4]Singer, D. J., Doerry, N., and Buckley, M. E., "What is set-based design?" *Naval Engineers Journal*, Vol. 121, No. 4, 2009,

pp. 31-43.

[5]Lee, J., "Set-based design systems for stampings and flexible fixture workspaces," Ph.D. Dissertation, Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI, 1996.

[6]Wang, J., and Terpenny, J., "Interactive evolutionary solution synthesis in fuzzy set-based preliminary engineering design," *Journal of Intelligent Manufacturing*, Vol. 14, 2003, pp. 153-167.

[7]Nahm, Y. E., and Ishikawa, H., "Novel space-based design methodology for preliminary engineering design," *International Journal of Advanced Manufacturing Technology*, Vol. 28, 2006, pp. 1056-1070.

[8]Shahan, D., and Seepersad, C. C., "Bayesian networks for set-based collaborative design," *Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, DETC2009/85741, ASME, New York, NY, 2009.

[9]Madhavan, K., Shahan, D., Seepersad, C., Hlavinka, D. A., and Benson, W., "An industrial trial of a set-based approach to collaborative design," *Proceedings of the ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, DETC2008/49953, ASME, New York, NY, 2008.

[10]Malak, R. J. Jr., Aughenbaugh, J. M., and Paredis, C. J. J., "Multi-attribute utility analysis in set-based conceptual design," *Computer-Aided Design*, Vol. 41, 2009, pp. 214-227.

[11]Avigad, G., and Moshaiov, A., "Set-based concept selection in multi-objective problems involving delayed decisions," *Journal of Engineering Design*, Vol. 21, No. 6, 2010, pp. 619-646.

[12]Marler, R. T., and Arora, J. S., "Survey of multi-objective optimization methods for engineering," *Structural and Multidisciplinary Optimization*, Vol. 26, 2004, pp. 369-395.

[13]Athan, T. W., and Papalambros, P., "A note on weighted criteria methods for compromise solutions in multi-objective optimization," *Engineering Optimization*, Vol. 27, 1996, pp. 155-176.

[14]Holtrop, J., and Mennen, G. G. J., "An approximate power prediction method," *International Shipbuilding Progress*, Vol. 29, No. 335, 1982, pp. 166-170.

[15]Holtrop, J., "A statistical re-analysis of resistance and propulsion," *International Shipbuilding Progress*, Vol. 31, No. 363, 1984, pp. 272-276.

[16]Parsons, M. G., Li, J., and Singer, D. J., "Michigan conceptual ship design software environment – User's manual," University of Michigan Department of Naval Architecture and Marine Engineering, Report No. 338, Ann Arbor, MI, 1998.

[17]Clarke, D., Gedling, P., and Hine, G., "The application of manoeuvring criteria in hull design using linear theory," *Transactions RINA*, Vol. 124, 1982, pp. 45-68.

[18]Raff, A. I., "Program SCORES – Ship structural response in waves," Ship Structure Committee, Report SSC-230, 1972.

[19]Parsons, M. G., "Parametric design," *Ship design and construction*, edited by T. Lamb, Society of Naval Architecture and Marine Engineers, Jersey City, NJ, 2003, pp. 11-1 – 11-48.