

A goal-programming enhanced collaborative optimization approach to reducing lifecycle costs for naval vessels

Dylan Temple¹ · Matthew Colette¹

Received: 25 March 2015 / Revised: 21 October 2015 / Accepted: 9 November 2015 / Published online: 6 January 2016
© Springer-Verlag Berlin Heidelberg 2015

Abstract Understanding the trade-offs involved in assessing lifetime cost for engineering systems requires understanding trends in various engineering disciplines that require significantly different analysis methods to efficiently explore. The corresponding design spaces can be flat, defined by weak minima, and thus difficult to understand using traditional optimization methods. This paper presents a new multi-disciplinary framework that uses a goal-programming enhanced multi-objective collaborative optimization (eMOCO) approach to facilitate the development of the spaces. In order to further increase its efficiency in discrete or flat spaces well-suited to evolutionary optimization a unique discipline level genetic algorithm is proposed. Naval vessels are an example of an engineering system that has a difficult design space with respect to lifetime cost, however, one where it is critical to understand. As these costs are increasing, they are becoming limiting factors in a vessel's operational life. Though they are so important, the interaction between different cost categories such as production and operation has not been explored in depth and is not always clear. Understanding the trade-offs between different aspects of a vessel's total ownership costs early in the design stage can aid in the production of new ships where they are minimized. The proposed framework is verified on mathematical problems, and then used to develop trade-spaces between resistance and production for a nominal naval combatant vessel. These trade-spaces show

both the knowledge gained by designers in understanding these trade-offs and the ability of the proposed eMOCO framework to develop them effectively.

Keywords Multi-disciplinary optimization · Naval design · Lifetime cost

1 Introduction

The lifetime cost of engineering systems are becoming an increasingly large burden to owners and operators and, therefore, an important factor to consider for designers. Optimization has been used in many applications to attempt to address reducing these costs over a systems operational life. Engineers who work on naval platforms are changing their design approach due to rising operational costs. In Stambaugh and Barry (2014) the authors outline large costs the United States Coast Guard is facing maintaining and operating their current ageing fleet. They posit that the economic burden of these costs are becoming the decisive factor that limits a vessel's service life. Given this state of naval design it is important to utilize the total ownership costs as a design driver. This overall cost includes various categories such as production, operation, and maintenance. These categories are often competing with one another and, therefore, it becomes important to understand their interactions and trade-offs. Unfortunately, gaining this understanding is a difficult task as the design space is difficult to explore with traditional multi-objective optimization algorithms even for simple problems as shown in Temple and Collette (2015). This work aims to allow for a greater understanding of this trade space by developing a new multi-disciplinary optimization framework that is specifically tailored to solve this unique and difficult problem.

✉ Dylan Temple
dylantem@umich.edu

¹ Department of Naval Architecture and Marine Engineering,
University of Michigan, 2600 Draper Drive, Ann Arbor,
MI 48104, USA

When analyzing the trade-offs in different categories of lifetime cost the problem is inherently multi-disciplinary. Operational costs such as fuel consumption are largely governed by hydrodynamics, whereas categories such as production or maintenance incorporate structural analysis. Multi-disciplinary optimization (MDO) has been used to design within this type of coupled, difficult space before. It has been used in the marine design field (Li et al. 2010; Hannapel 2012; Hart and Vlahopoulos 2009) for various applications. There are a wide range of architectures for MDO that approach solving these problems in different ways. A broad line to divide them by is monolithic and distributed architectures (Martins and Lambe 2013). Distributed architectures break the problem into system- and sub-system level problems in order to facilitate finding the coupled solution. Collaborative Optimization (CO) is a distributed multi-disciplinary optimization method that has been utilized heavily in engineering design optimization. Various papers have presented methods of using CO in a multi-objective environment to develop Pareto fronts (Aute and Azarm 2006; Ruiyi et al. 2011; Depince et al. 2005; Gunawan et al. 2003). The distributed nature of CO allows the sub-level problems to exploit the specific nature of their design spaces. The proposed framework will take advantage of this quality and make further enhancements that facilitate understanding trade-offs in lifetime costs.

There is a rich body of optimization work done in different design disciplines for naval systems. Researcher's have used optimization to reduce the drag on a ship's hull (Harries et al. 2001; Sariz 2006; Brenner and Sener 2011) and developed search methods to aid in the exploration of the design space (Kuhn et al. 2007; Zalek et al. 2008). Similarly work has been done to optimize ship's structures' for objectives such as production cost (Krol 1991; Mermiris et al. 2005; Yang and Hwang 2002) and maintenance (Chung et al. 2006; Junca and Sanchez-Silva 2013; Kim and Frangopol 2011). However understanding the interactions between these different disciplines and using optimization to find trade-offs is unexplored. One of the reasons for this is the mathematical difference between the design spaces. For example, when considering resistance and lifetime maintenance the hydrodynamic discipline is a continuous mathematical space wherein gradient based algorithms can be effectively used. The structural space, however, is often a discrete space with weak minima (Temple and Collette 2013b) that lends itself well to heuristic methods. Another is that the coupling between these disciplines is weaker than in other MDO engineering problems. The authors of this work have found that traditional multi-objective tools are unable to consistently

converge to the entire Pareto-front for this type of multi-disciplinary problem. Proposed here is a multi-objective multi-disciplinary optimization framework that uses two enhancements to facilitate solving these unique problems: a novel goal-programming decision making process (DSP) at the discipline level and a new locally-elitist single objective genetic algorithm in non-continuous sub-spaces. By utilizing these two approaches the enhanced MOCO (eMOCO) is able to efficiently and accurately resolve these difficult Pareto-fronts. This can aid designers in understanding these trade-offs in order to ensure that engineering decisions are being made that will minimize the lifetime cost of a vessel.

A multi-objective collaborative optimization (MOCO) framework has the benefit of being able to utilize different types of optimizers at a sub-system level to effectively design within each individual space. In order to take advantage of the distributed architecture of the MOCO it is enhanced using a goal-programming technique at the sub-system level to aid in minimizing local objectives. Goal programming is a decision making process (DSP) that has been used in engineering to give optimization algorithms a sense of the designers intent. It uses a hierarchy of objective functions, that are successively minimized, thus allowing the algorithm to minimize what the designer deems most important first. In the proposed enhanced MOCO (eMOCO) framework the goal programming is used to first minimize the interdisciplinary feasibility, as in a traditional MOCO, and then reduce local objective functions second. This aids in pushing the system level optimizer towards the global Pareto-front, and is specifically useful in solving problems with weak minima or flat regions as the goal-programming at the sub-system level helps push the population through them.

One of the issues that arises when using heuristic algorithms, especially those that are population based, in a distributed MDO architecture is that the computation time can quickly become intractable. To counter this a second enhancement is proposed for the sub-spaces that are discrete or flat in nature: a new variation on the single objective genetic algorithm (SOGA) called a locally-elitist genetic algorithm (LEGA). This LEGA allows the discipline optimizer to solve the sub-system problem in a single execution as opposed to a different one for each solution in the system level population, allowing for the discipline level problem to be solved much faster. These two enhancements can be implemented and used separately, however, they are designed in tandem to be used in solving this specific type of naval design problem. In order to show the framework's ability to solve the unique type of problem it has been designed for it is first tested on multi-objective and multi-

disciplinary test problems followed by a case study in the naval design field.

The subsequent sections of this paper will cover the eMOCO algorithm, then proceed to go over the various test problems used to verify its ability. It will conclude with a case study using a nominal hull form and structure for a standard naval combatant and a discussion of the results.

2 Enhanced MOCO framework (eMOCO)

Goal programming is a decision support process (DSP) that has been used within engineering optimization to handle competing objectives of varying importance. In goal-programming different objectives are satisfied in an order based on their relative importance. This work utilizes lexicographical goal-programming in which each objective is satisfied completely before the optimizer begins working on the next one. This formulation can be expressed as shown in Eq. 1

$$\begin{aligned} \text{minimize: } & F(x) = [d_1^+, d_2^+, \dots, d_n^+] \\ \text{subject to: } & g(x) \leq 0 \\ & f_i(x) + d_i^+ - d_i^- = b_i \quad \forall i = 1, \dots, n \end{aligned} \quad (1)$$

In Eq. 1 each objective, f_i is being minimized towards a goal, b_i , according to their respective priority. This manifests in the variables d_i^+ and d_i^- , which represent the positive or negative variance from the goal of the objective. In the lexicographical approach to goal-programming the constraints ensure that it does not increase the objective functions that have higher priority (Tamiz et al. 1998).

This work uses goal-programming in a novel approach to allow the sub-system problems to reduce the local objective functions while maintaining the inter-discipline feasibility. This lets them act as a local decision maker, ensuring that solutions passed back to the system-level optimizer represent areas of the design space that are locally optimal. By utilizing this technique, along with the inter-disciplinary feasibility metrics inherent to collaborative optimization, the system-level optimizer can efficiently resolve difficult Pareto-fronts involving sub-spaces of drastically different natures.

Collaborative Optimization (CO) separates a system-level problem and discipline-level ones by introducing copies of coupling and shared design vectors and passing them to the sub-system optimizers (Martins and Lambe 2013). The system-level optimizer then focuses on minimizing the objective vector, $f(x)$; while the sub-system problem then ensures all local constraints are satisfied while minimizing discrepancy between local design variables and the copies sent from the system-level optimizer.

The system- and sub-system level problems can be seen in Eqs. 2 and 3:

$$\begin{aligned} \text{minimize: } & f(x) = (f_0, f_1, \dots, f_N) \\ \text{with respect to: } & x = (x_0, \hat{x}_1, \dots, \hat{x}_N, \hat{y}) \\ \text{subject to: } & g_0(x) \geq 0 \\ & J_i^* = \|\hat{x}_{0i} - x_0\|_2^2 + \|\hat{x}_i - x_i\|_2^2 + \|\hat{y}_i - y_i\|_2^2 = 0 \text{ for } i = 1, \dots, N \end{aligned} \quad (2)$$

$$\begin{aligned} \text{minimize: } & J_i = \|\hat{x}_{0i} - x_0\|_2^2 + \|\hat{x}_i - x_i\|_2^2 + \|\hat{y}_i + y_i\|_2^2 \\ \text{with respect to: } & \hat{x}_{0i}, x_i \\ \text{subject to: } & g_i(\hat{x}_{0i}, x_i, y_i) \geq 0 \end{aligned} \quad (3)$$

In Eq. 2 x_0 represents the vector of variables shared between disciplines, x_i represent design variables unique to the i^{th} discipline, and y_i represents coupling variables from discipline i to a subset of disciplines $j \neq i$. The coupling variables may be specific design variables themselves or, more often, state vectors that are calculated based on the other design variables. Variables marked with a $\hat{}$ represent copies passed between the system and sub-system problems. By having local copies of each variables within the various problems; the algorithms can work in parallel to reduce local objectives and constraints. The inter-disciplinary feasibility function, J_i for each discipline ensures that, when the system-level optimizer converges, each discipline agrees on the optimal design vectors.

2.1 Goal programming at the discipline level

Due to the distributed nature of the MOCO architecture it is possible to utilize the sub-system optimizers in the development of the Pareto-fronts. By enhancing the MOCO with the goal-programming DSP outlined above at a sub-system level the local objective function value can be reduced. The sub-system populations passed back to the system level optimizer then represent design vectors that are minimized with respect to their disciplines, facilitating the movement of the Pareto-front towards the optimum. This leads to the revised sub-system problem formulations in Eq. 4:

$$\begin{aligned} \text{minimize: } & F(x) = [d_{iJ}^+, d_{if_1}^+, \dots, d_{if_N}^+] \\ \text{with respect to: } & \hat{x}_{0i}, x_i \\ \text{subject to: } & g_i(\hat{x}_{0i}, x_i, y_i) \geq 0 \\ & J_i(\hat{x}_0, x_i, y_i) + d_{iJ}^- - d_{iJ}^+ = 0 \\ & f_{i_1}(\hat{x}_0, x_i, y_i) + d_{if_1}^- - d_{if_1}^+ = 0 \\ & \vdots \\ & f_{i_N}(\hat{x}_0, x_i, y_i) + d_{if_N}^- - d_{if_N}^+ = 0 \end{aligned} \quad (4)$$

The system level problem remains as shown in Eq. 2. The vector of objective functions is reduced while satisfying any shared constraints; and ensuring that there is inter-disciplinary feasibility. However, at a sub-system level a lexicographical goal-programming approach is used to accomplish two goals:

1. Minimize the discipline infeasibility for the local design variables
2. Reduce any local objective functions

Note that the formulation shown in Eq. 4 assumes the local objective functions are being minimized towards zero; however, this could be changed to reflect the nature of whatever problem is being solved. This means the sub-system problem first minimizes the local J function; then, secondarily, reduces the local objectives. This can aid the system-level optimizer in resolving difficult Pareto-fronts by ensuring that discipline design variables are pushed towards the front. Note that N number of discipline objectives can be minimized during the goal-programming routine. This allows the engineer using the algorithm to set a hierarchy to different objectives, allowing their engineering experience and intuition to play a part in the outcome of the optimizer.

Since the interdisciplinary feasibility functions can be optimized with any type of algorithm, the sub-system problems can utilize optimization techniques tailored to the local objectives. For instance, if the space is discontinuous or multi-modal a gradient-free algorithm such as a genetic algorithm or particle swarm optimizer can be used; however, if it is C_1 continuous a more effective gradient-based method can be used. Thus, the system-level can employ a multi-objective genetic algorithm to resolve the full-Pareto front, yet, the specific nature of each discipline can still be exploited by the sub-system problem to aid the system-level optimizer. This also allows various optimizers to be implemented at a sub-system level at the designers will with relative ease depending on the problem being solved.

2.2 Locally elitist genetic algorithm (LEGA)

In order for the goal-programming enhanced MOCO to effectively optimize non-continuous sub-spaces; a gradient-free algorithm must be utilized at a sub-system level. In current literature sub-system genetic algorithms have been developed for a MOCO framework by using a design-of-experiments method to sample points in the local vicinity around each point in the system-level population. These sample points are then used as the initial population for a single-objective genetic algorithm (SOGA). This means that for every candidate solution in the system-level a full genetic algorithm must be performed. While

this approach is conceptually intuitive, it is also computationally expensive. It would be advantageous if a single execution of a genetic algorithm could be done at the sub-system level.

However, each candidate solution in the system-level population will have different values of \hat{x}_i , x_0 , and \hat{y}_i . Since these vectors do not change throughout the discipline-level optimization they will be referred to as *static vectors*. Since the static vectors are fundamental to i^{th} sub-system's J_i function, the discipline-level problem associated with each candidate solution is actually slightly different. This makes it difficult to use the standards evolutionary operators such as crossover; since each solution represents a slightly different optimization problem based on its location in the design space. The problem lies in that if the standard operators are used the discipline-level SOGA will quickly move each solution to whichever candidate is nearest to the feasible domain, drastically diminishing the genetic diversity in the population. In order to avoid this the diversity of the static vectors within a population must be maintained throughout the genetic algorithm. This can be done by defining new crossover and selection functions that are elitist *locally* with respect to solutions with identical static vectors; however, highly non-elitist with respect to the entire population. The crossover function can be seen in Algorithm 1:

Algorithm 1 Crossover For LEGA

Require: Two parent solutions, x_1 and x_2 , with corresponding state vectors, y_{i_1} and y_{i_2} , discipline-specific copies, \hat{x}_{i_1} and \hat{x}_{i_2} , and shared copies, x_{0_1} and x_{0_2}

- 1: **function** LEGA-CROSSOVER($x_1|y_1, x_2|y_2$)
 - 2: $\theta_1 \leftarrow (x_{0_1}, \hat{x}_{i_1}, y_{i_1})$
 - 3: $\theta_2 \leftarrow (x_{0_2}, \hat{x}_{i_2}, y_{i_2})$
 - 4: $c_1|_{\theta_1}, c_2|_{\theta_1}, c_1|_{\theta_2}, c_2|_{\theta_2} \leftarrow \text{crossover}(x_1|_{\theta_1}, x_2|_{\theta_2})$
 - 5: $w_1 \leftarrow \min(\min(f(c_1|_{\theta_1}), f(c_2|_{\theta_1})), f(x_1|_{\theta_1}))$
 - 6: $w_2 \leftarrow \min(\min(f(c_1|_{\theta_2}), f(c_2|_{\theta_2})), f(x_2|_{\theta_2}))$
 - 7: **return** w_1, w_2
-

In Algorithm 1 the static vectors of each parent solution are stored in: θ_1 and θ_2 . A crossover method is then used to produce two children from the parents; and each candidate child solution is given both parents θ vectors, producing four unique solutions. All solutions associated with a given θ vector are then compared; and the winner from these comparisons is added to the child population. This ensures that the a strong-candidate solution corresponding to *each* static θ vector is added to the child population; thus helping to preserve genetic diversity with respect to the static vectors.

In order to ensure the diversity of the static vectors remains from each generation to the next; a selection

method is also utilized that focuses on maintaining this diversity.

Algorithm 2 Selection of continuing population for LEGA

Require: Population of children W and parents, P

1. **function** LEGA-SELECTION(W, P)
2. **for all** $c \in W$ **do**
3. **for all** $x \in P$ **do**
4. **if** c is offspring of x **then**
5. $x \leftarrow \min(f(c), f(x))$
6. **return** P

Algorithm 2 will only ever compare a child candidate solution (the population of which is generated using Algorithm 1) to one of the parent solutions that produced it. Therefore, even if a small number of static variable vectors make up the mating pool, or child pool, the diversity of static vectors will not degenerate from one generation to the next. This creates an algorithm that, from a population perspective, is highly non-elitist; since poor solutions can remain in the population to preserve the diversity of static vectors, however, it is quite elitist locally in improving the solutions attached to each of the static vectors.

The primary reason this must be done is that the discipline level SOGA is seeking to improve the entire system-level population in a single run; meaning that the entire final population must be significant to the system-level optimizer, not only a single optimum solution. While this approach is not as intuitive as designing an entire SOGA run around each system-level individual, it functions well within the MOCO framework and saves significantly on computational effort.

3 Test problems

In order to show the ability of the enhanced MOCO to solve multi-objective and multi-disciplinary engineering problems multiple test cases will be analyzed. First a set of standard multi-objective problems that have been used to test algorithms such as the NSGA-II will be used to show the eMOCO’s ability to find the solution to basic multi-objective problems. Then an MDO test problem has been used to show the ability of the eMOCO to solve a multi-disciplinary problem more efficiently than a traditional MOCO.

3.1 Multi-objective test problems

Two mathematical problems are used to show the algorithm’s ability to resolve different Pareto-fronts. These can be seen in Eqs. 5 and 6. They are called CONSTR and

TNK respectively. Both of these are taken from Deb et al. (2002) and have been used in the validation of many multi-objective optimization algorithms.

$$\begin{aligned} \text{minimize: } & f(x) = (x_1, \frac{1+x_2}{x_1}) \\ \text{subject to: } & g_1(x) = -x_2 + 9x_1 - 1 \geq 0 \\ & g_2(x) = \frac{x_2 + 9x_1 - 6}{6} \geq 0 \end{aligned} \tag{5}$$

$$\begin{aligned} \text{minimize: } & f(x) = (x_1, x_2) \\ \text{subject to: } & g_1(x) = x_1^2 + x_2^2 - 1 - 0.1 \cos(16 \arctan(\frac{x_1}{x_2})) \geq 0 \\ & g_2(x) = \frac{-(x_1 - 0.5)^2 - (x_2 - 0.5)^2 + 0.5}{0.5} \geq 0 \end{aligned} \tag{6}$$

The problem in Eq. 5 is driven largely by the second objective function and first constraint; whereas the problem in Eq. 6 is clearly driven entirely by the constraints.

In order to test the enhanced MOCO using these problems they were divided into two disciplines each; where x_1 is the discipline variable for the first discipline, x_2 is the discipline variable for the second, and there are no shared variables. Equation 5 can be transformed into the following system-level and sub-system problems:

System Level:

$$\begin{aligned} \text{minimize: } & f(x) = (\hat{x}_1, \frac{1+\hat{x}_2}{\hat{x}_1}) \\ \text{subject to: } & \|\hat{x}_1 - x_1\|_2^2 + \|\hat{x}_2 - x_2\|_2^2 = 0 \end{aligned} \tag{7}$$

Discipline 1:

$$\begin{aligned} \text{minimize: } & F(x) = [d_{1_j}^+, d_{1_f}^+] \\ \text{subject to: } & g_1(x) = -\hat{x}_2 + 9x_1 - 1 \geq 0 \\ & g_2(x) = \|\hat{x}_1 - x_1\|_2^2 + d_{1_j}^+ - d_{1_j}^- = 0 \\ & g_3(x) = x_1 + d_{1_f}^+ - d_{1_f}^- = 0 \end{aligned} \tag{8}$$

Discipline 2:

$$\begin{aligned} \text{minimize: } & F(x) = [d_{2_j}^+, d_{2_f}^+] \\ \text{subject to: } & g_1(x) = \frac{x_2 + 9\hat{x}_1 - 6}{6} \geq \\ & g_2(x) = \|\hat{x}_2 - x_2\|_2^2 + d_{1_j}^+ - d_{1_j}^- = 0 \\ & g_3(x) = \left(\frac{1+x_2}{\hat{x}_1}\right) + d_{2_f}^+ - d_{2_f}^- = 0 \end{aligned} \tag{9}$$

By breaking Eq. 5 into Eqs. 7, 8 and 10 it can be solved using the enhanced-MOCO algorithm. The goal-programming is lexicographical in approach; this means that the sub-system optimizers will first minimize the disciplinary infeasibility as much as possible; and then focus

on reducing the local objective function; without increasing the value of J_i . This allowed the algorithm to ensure the inter-disciplinary infeasibility, and also allow the sub-system optimizers to reduce the objectives specific to their subspaces.

The problem shown in Eq. 6 can be broken down similarly as seen in Eqs. 11, 12, and 14.

System Level:

minimize: $f(x) = (\hat{x}_1, \hat{x}_2)$
 subject to: $\|\hat{x}_1 - x_1\|_2^2 + \|\hat{x}_2 - x_2\|_2^2 = 0$ (11)

Discipline 1:

minimize: $F(x) = [d_{1_j}^+, d_{1_f}^+]$
 subject to: $g_1(x) = x_1^2 + \hat{x}_2^2 - 1$
 $-0.1 \cos\left(16 \arctan\left(\frac{x_1}{\hat{x}_2}\right)\right) \geq 0$
 $g_2(x) = \|\hat{x}_1 - x_1\|_2^2 + d_{1_j}^+ - d_{1_f}^- = 0$
 $g_3(x) = x_1 + d_{1_f}^+ - d_{1_f}^- = 0$ (12)

Discipline 2:

minimize: $F(x) = [d_{2_j}^+, d_{2_f}^+]$
 subject to: $g_1(x) = \frac{-(\hat{x}_1 - 0.5)^2 - (x_2 - 0.5)^2 + 0.5}{0.5} \geq 0$ (13)
 $g_2(x) = \|\hat{x}_2 - x_2\|_2^2 + d_{1_j}^+ - d_{1_f}^- = 0$
 $g_3(x) = x_2 + d_{2_j}^+ - d_{2_f}^- = 0$ (14)

It is important to note that the division of this problem into two disciplines is arbitrary. This means it is not expected for the enhanced-MOCO to outperform more traditional algorithms, however, it is important that the algorithm is able to resolve these Pareto-fronts.

The Pareto-fronts for these problems using both the enhanced-MOCO and a traditional genetic algorithm (in this case an implementation of the NSGA-II (Deb et al. 2002)) are shown in Figs. 1 and 2. In both cases the system level optimizer evolved the initial population of 50 individuals through 50 generations. Both discipline optimizers used to produce the results are the LEGA presented above; however, similar ones were found using an SQP algorithm at the discipline level.

Figures 1 and 2 show that the eMOCO is able to resolve both pareto fronts as well or better than a traditional NSGA-II. The TNK problem is a constraint-dominated function with a highly non-linear Pareto-front, while the

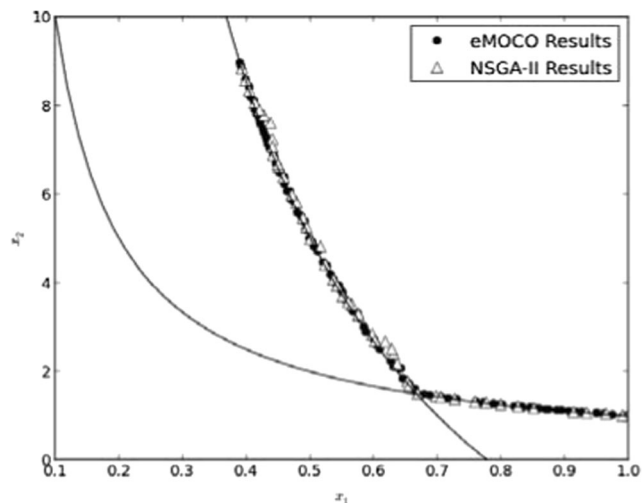


Fig. 1 CONSTR Problem - solved with eMOCO and NSGA-II

CONSTR problem is objective-dominated in one region and constraint-dominated in another. For both these problems the eMOCO performs well.

3.2 Multi-disciplinary test problem

In order to test the algorithm on an intentionally multi-disciplinary problem a test problem based on one shown in Ruiyi et al. (2011) is used. The problem involves three variables and two disciplines; each of which has a state variable that is passed to the other. An outline of the problem can be seen in Fig. 3.

The problem involves one shared variable, two discipline variables, and two state variables that are passed between

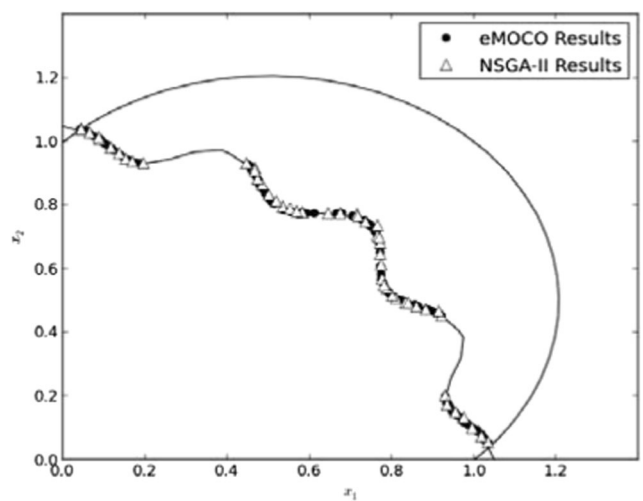


Fig. 2 TNK Problem - solved with eMOCO and NSGA-II

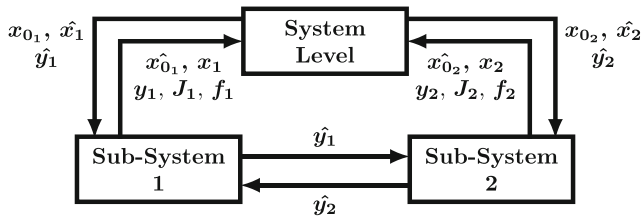


Fig. 3 Basic data flow for the two-discipline example

the disciplines. The system and discipline formulations are shown in in Eq. 15

System Level:

$$\text{minimize: } f_1(x) = x_1 + \sqrt{x_2} + \frac{y_2^2 - y_1^3}{10^4} + 150,$$

$$f_2(x) = x_2^2 + x_3 + y_1 + e^{-y_1}$$

with respect to: $x_1, \hat{x}_2, \hat{x}_3, \hat{y}_1, \hat{y}_2$

$$\text{subject to: } \sum_{i=1}^3 \|x_i - \hat{x}_i\|_2^2 + \sum_{i=1}^2 \|y_i - \hat{y}_i\|_2^2 = 0 \tag{15}$$

Discipline 1:

$$\text{minimize: } F(x) = [d_{1_j}^+, d_{1_f}^+]$$

with respect to: \hat{x}_1, x_2, y_1

$$\begin{aligned} \text{subject to: } & (\hat{x}_1^2 + x_2 - 0.2\hat{y}_2) - y_1 = 0 \\ & \|x_1 - \hat{x}_1\|_2^2 + \|x_2 - \hat{x}_2\|_2^2 + \dots \\ & \dots \|y_1 - \hat{y}_1\|_2^2 + d_{1_j}^+ + d_{1_f}^- = 0 \\ & f_1(x) + d_{1_f}^+ - d_{1_f}^- = 0 \end{aligned} \tag{16}$$

Discipline 2:

$$\text{minimize: } F(x) = [d_{2_j}^+, d_{2_f}^+]$$

with respect to: \hat{x}_1, x_3, y_2

$$\begin{aligned} \text{subject to: } & (\hat{x}_1 + x_3 + \sqrt{\hat{y}_1}) - y_2 = 0 \\ & \|x_1 - \hat{x}_1\|_2^2 + \|x_3 - \hat{x}_3\|_2^2 + \dots \\ & \dots \|y_2 - \hat{y}_2\|_2^2 + d_{2_j}^+ + d_{2_f}^- = 0 \\ & f_2(x) + d_{2_f}^+ - d_{2_f}^- = 0 \end{aligned} \tag{17}$$

In Eqs. 15 through 17 x_1 is a shared variable, x_2 is specific to discipline 1 and x_3 is specific to discipline 2. Both disciplines have state variables y_1 and y_2 that are dependent on each other and the entire vector of design variables. The objective function also involve both disciplines and their respective state variables. This is an example problem that is somewhat representative of the naval design situations this

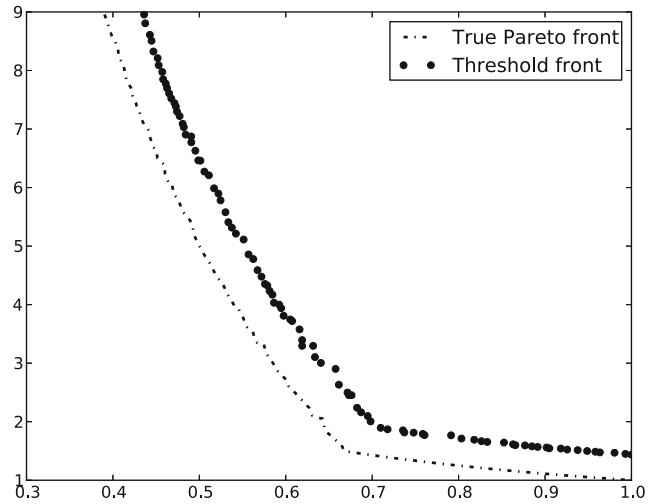


Fig. 4 An example of a true Pareto front and threshold front

algorithm is being developed for. The design vector and, more specifically, the number of state variables are small, yet they are critical to resolving the front.

In order to show the benefit of using the enhanced MOCO this problem will be solved using both a traditional MOCO and the eMOCO.

3.2.1 Convergence measurement

Though the MDO example is more complex than the problems shown in the previous section, it is still mathematically simple enough that both the enhanced MOCO and a traditional one can solve it. Thus, it is necessary to have a measurement of convergence in order to analyze the speed at which they solve the problem. Measuring the speed of convergence in a multi-objective problem is not as straight forward or standardized as in single-objective optimization, however, different approaches have been proposed. In Collette and Siarry (2005) a convergence speed metric was proposed based on a threshold frontier. In this metric a true Pareto front is offset by a vector, v as seen in Fig. 4.

In the metric proposed in Collette and Siarry (2005) convergence is measured as the portion of a population that falls between the true Pareto front and threshold front at a given generation. This work proposes a similar metric to compare the convergence of the two different MOCO algorithms. A metric such as the threshold front is useful, however, it is possible to report unrealistic convergences where a candidate population is clustered in a small portion of the Pareto-front. Therefore a metric is proposed that combines a threshold measurement with a Pareto-span metric. The

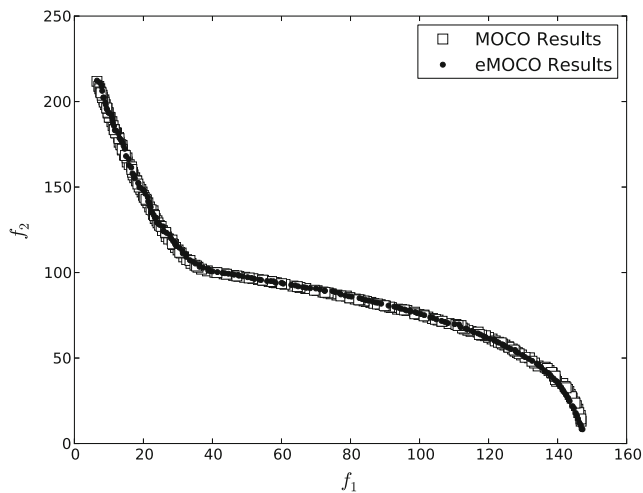


Fig. 5 The results for the MDO test problem for both the traditional and enhanced MOCO

convergence at the i^{th} generation, denoted as $\hat{e}(i) \in (0, 1)$, is calculated using Eq. 18.

$$\hat{e}(i) = \frac{1}{2} \left(\frac{W_{TH}(S_i)}{|S_i|} + \frac{f_{F_1}(x)_{max} - f_{i_1}(x)_{min}}{f_{F_1}(x)_{max} - f_{F_1}(x)_{min}} \right) \quad (18)$$

In Eq. 18 $W_{TH}(S)$ is the number of individuals in the i^{th} solution set that fall under the threshold front and $|S_i|$ is the size of the i^{th} solution set. $f_F(X)$ is the threshold (or ‘true’) Pareto front, and $f_i(x)$ is the Pareto frontier during the i^{th} generation. The maximum value $\hat{e}(i)$ can take is 1.0. Both the eMOCO and traditional MOCO are run until the Pareto front reaches a convergence value of 0.95.

The first term in this metric measures the amount of the population below the threshold frontier, where the second term measures the span of the current front relative to the final one. By utilizing both these concepts a more complete picture of convergence can be determined.

The Pareto front found using a traditional MOCO (tMOCO) and the eMOCO can be seen in Fig. 5. A population size of 200 was used for all runs.

Figure 5 shows that both algorithms converge to the same Pareto front. This is expected as the problem is a relatively straight-forward MDO example. However, it is important to examine the speed at which both algorithms converge. In order to compare the two algorithms 100 runs of each were done and the average generations to convergence was examined. These values can be seen in Table 1.

Table 1 Average generations to converge for the traditional (tMOCO) and enhanced (eMOCO) MOCO over 100 samples

Algorithm	Average Generations
tMOCO	148.9
eMOCO	109.1

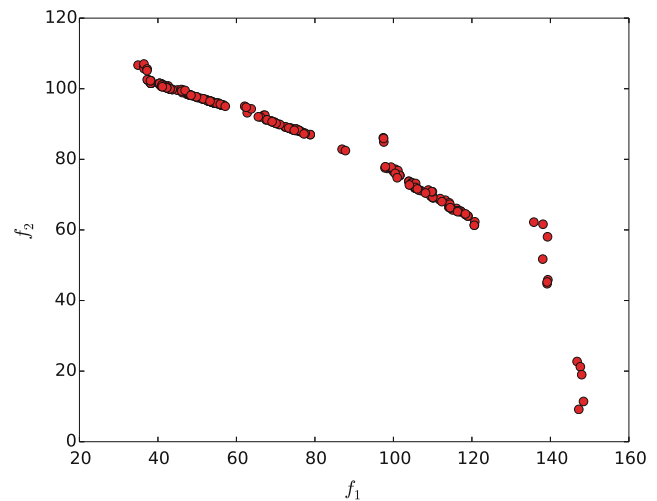


Fig. 6 Generation 20 for the enhanced MOCO

In order to ensure the validity of the mean shift seen in Table 1 Welch’s T-Test was performed and the null hypothesis was rejected with a p -value of 0.00653, showing that the two samples of test runs are statistically different. Table 1 shows that the enhanced MOCO is able to consistently solve the test problem in less generations than the traditional one. In fact, it shows that even on this simple example problem the enhanced MOCO is able to converge in 27 % less generations than a traditional one over 100 sample runs. This represents a substantial increase in speed; especially for engineering problems where objective calculation can be computationally expensive.

The eMOCO is able to converge to the final Pareto front faster than the traditional MOCO due to its ability to avoid clustering in easier to resolve areas of the front. This is a result of the addition of goal-programming to the subsystem problem. With goal programming, the disciplines are also pushing resolve the Pareto front from a local perspective. This increases the quality of the return points at the system level, allowing a more evenly spaced and better resolved front to emerge. Figures 6 through 11 show this improved front by contrasting the Pareto fronts for the eMOCO and traditional MOCO at generations 20, 30, and 50 of the algorithm.

Figures 6 and through 11 highlight this by showing how clustering is avoided by utilizing the goal-programming approach. Figure 9 shows that the traditional MOCO clusters in areas of the Pareto front that are easier to resolve before slowly spreading out to fill out the entire optimal front. Conversely, Figures 6 through 8 show that the enhanced MOCO is able to initially develop a Pareto front with much less clustering. The reduction in clustering allows the new algorithm to fill in the entire front faster while avoiding the pitfalls of clustering in an evolutionary algorithm. Clustering reduction also allows portions of the Pareto front that

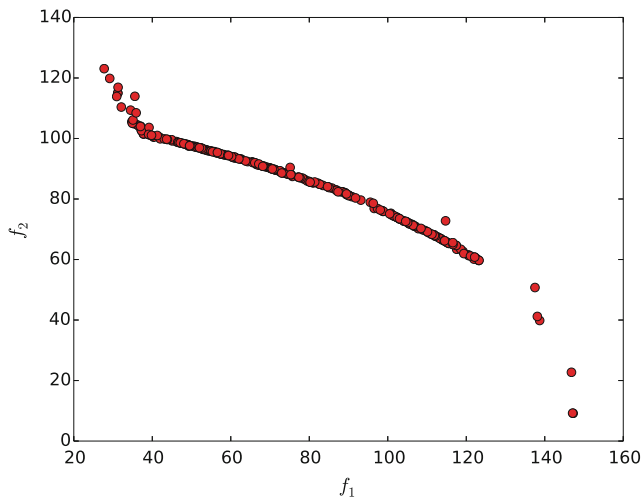


Fig. 7 Generation 30 for the enhanced MOCO

may be more difficult to resolve to be found earlier in the algorithm. Previous results have shown that for the lifetime naval cost algorithm, finding the full extent of the Pareto front can be a challenge for the optimizer (Temple and Collette 2015). Thus, when designers are seeking to visualize the entire tradespace, the enhanced MOCO helps to ensure that the full Pareto front is returned.

4 Naval design case study

In order to show the ability of the enhanced MOCO to develop the trade-spaces between different facets of lifetime cost a nominal DTMB-5145 is optimized for both resistance and production. A simplified version of the structure found in Ashe et al. (2009) and hull form found in Benedetti et al. (2007) are used. This is the type of problem that the

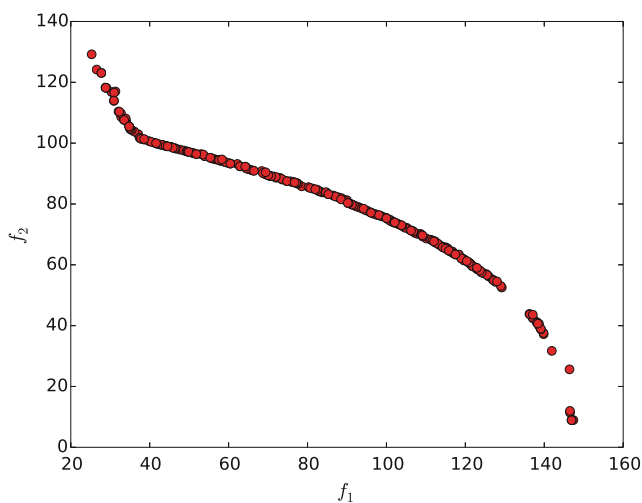


Fig. 8 Generation 50 for the enhanced MOCO

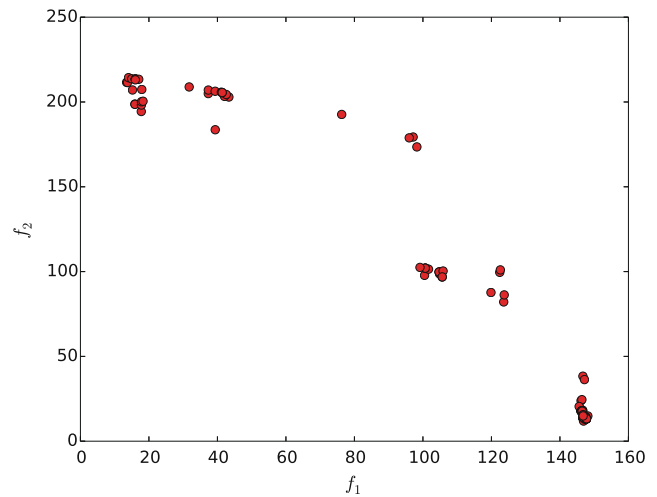


Fig. 9 Generation 20 for the traditional MOCO

resistance-production trade-space represents; as these two objectives are competing and coupled, however, the coupling is small between the objectives themselves. It is also a relationship representing the cost during two drastically different phases of the vessel’s life (production and operation); this is difficult to understand intuitively, making it a good test case for this framework.

4.1 Resistance calculation

The resistance experienced by a ship’s hull, R_T , can be considered the sum of a wave component, R_W , and a frictional component, R_F . The frictional component is often estimated using one of many frictional lines, that are a function of the length-based Reynolds number. The wave resistance is more difficult to analyze and is often estimated using either a simple regression or high fidelity CFD programs.

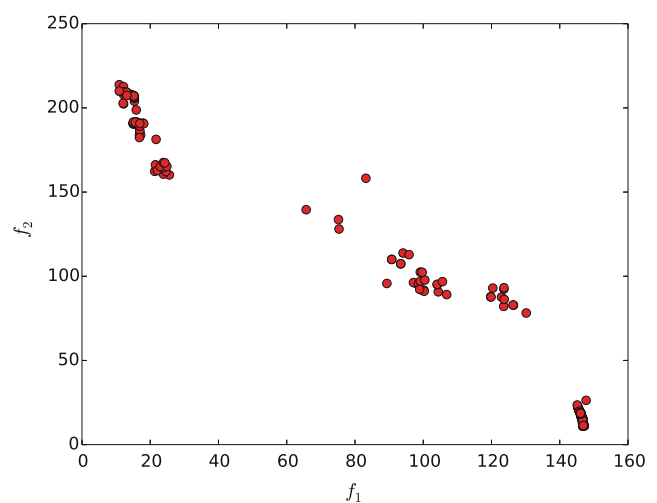


Fig. 10 Generation 30 for the traditional MOCO

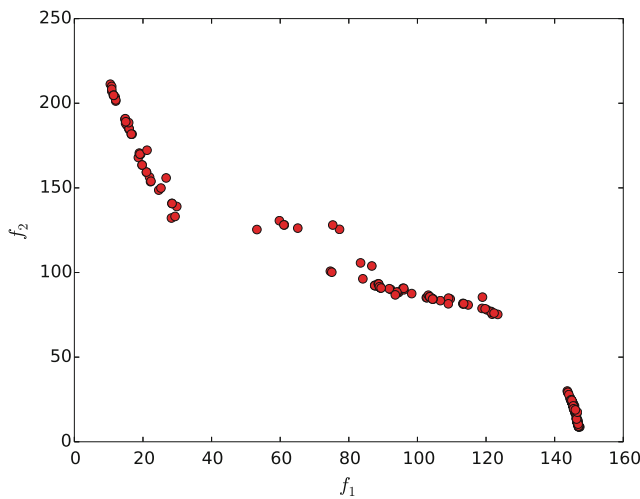


Fig. 11 Generation 50 for the traditional MOCO

In this work resistance is calculated using a combination of thin ship theory and the ITTC-57 friction line.

Thin ship theory, using Michell’s integral, is an effective way to estimate the hull’s wave resistance. Other work, Maki et al. (2008), Tuck (1989), and Tuck and Lazauskas (2008), has shown that the estimate is acceptable to understand the basic trends in resistance; which is the accuracy needed for the early-stage optimization work being done here. It can also do this relatively fast, allowing the optimization framework to avoid the costly use of higher fidelity estimates such as CFD. Thin ship theory uses Eq. 19 to calculate the wave resistance at speed U .

$$R_w = \pi \rho U^2 \int_{-\frac{\pi}{2}}^{\frac{\pi}{2}} |A(\theta)| \cos^3(\theta) d\theta \tag{19}$$

In this equation ρ is the density of salt water, θ is the heading angle of waves in the vessel’s wake, and $|A(\theta)|$ is known as the ’complex wave amplitude function. This can be calculated using Eq. 20.

$$|A(\theta)| = \frac{2}{\pi} \nu \sec^3(\theta) \iint_{cp} \frac{\partial \eta}{\partial x} e^{\nu \sec^2(\theta)(z+ix \cos(\theta))} dz dx \tag{20}$$

This is an integral along the centerplane of the vessel based on the derivative of the slope of the hull in the longitudinal coordinate, $\frac{\partial \eta}{\partial x}$, and a given wave heading angle. The variable ν is calculated based on the forward speed of the vessel and the acceleration due to gravity, g , using Eq. 21. This value is the theoretical critical wave number of the ship’s wake in infinitely deep water.

$$\nu = \frac{g}{U^2} \tag{21}$$

By integrating just over the ship’s centerplane the theory assumes a long, thin vessel. While this is an approximation

for most naval vessels, it is sufficiently accurate to understand trends in the naval design space during the early stages of a ship’s design.

The total resistance can be considered a sum of the wave resistance and frictional resistance. In order to estimate the frictional resistance the ITTC friction line, calculated as a function of Reynolds number as seen in Eq. 22, is used. This frictional line was developed in the 1957 international towing tank conference and is widely used by naval architects as an estimate for the frictional resistance a ship experiences.

$$C_{F,ITTC-57} = \frac{0.075}{(\log_{10}(R_n - 2))^2} \tag{22}$$

Equation 22 calculates the coefficient of friction which can be used to determine the frictional resistance using Eq. 23.

$$R_f = \frac{1}{2} C_{F,ITTC-57} \rho S_W \tag{23}$$

Where S_W is the wetted surface area of the vessel.

Using these equations to estimate the wave and frictional resistance a value for total resistance at the vessel’s design speed can be calculated.

4.2 Production cost

The production cost is estimates using costing equations from Rahman and Caldwell (1995) where the estimated cost to produce a structure, C_P , is calculated using Eq. 24.

$$C_P = \sum_{i=1}^n \sum_{j=1}^9 C_{ji} \tag{24}$$

In Eq. 24 C_{ji} is the cost of the j_{th} construction component for the i_{th} panel in the structure. The nine different components of construction that are used are as follows:

- C_1 = cost of materials for hull plates
- C_2 = cost of materials for longitudinal stiffeners
- C_3 = cost of materials for longitudinal frames
- C_4 = cost of welding for longitudinal stiffeners
- C_5 = cost of welding for transverse frames
- C_6 = cost of intersections between longitudinal stiffeners and transverse frames
- C_7 = cost of preparation of brackets and joints
- C_8 = cost of electricity and electrodes
- C_9 = fabrication cost of longitudinal stiffeners and transverse frames

$C_1, C_2,$ and C_3 are computed from the total weight of each material present in the structure, multiplied by a cost per pound for the material. C_4 and C_5 are calculated based

on the total meters of welding required to fabricate the structure, and a cost per meter of weld. C_6 and C_7 scale with the total number of intersections between stiffeners running orthogonally, such intersections require careful fit-up and bracketing. A fixed cost per intersection is assumed. C_8 and C_9 address welding consumables and additional costs if custom profiles are used in place of standard steel mill stock. As many of these variables vary over time, shipyard location and shipyard automation level, approximate values for western shipyards in the late 2000's were used. The resulting costs were then normalized by a reference design, and the final Pareto fronts presented as a percentage of the reference design cost. Further details and complete equations for the calculation of each of these components can be found in the original paper introducing the method (Rahman and Caldwell 1995).

4.3 Optimization

The two objective functions shown here are prime examples of design spaces whose nature differ significantly. Various researchers have used fast gradient-based optimizers to reduce drag as calculated by Michell's integral successfully and efficiently. However, the production cost is non-smooth due to the nature of the ship's structure. This makes it well suited for an evolutionary based optimizer such as the genetic algorithm. Therefore; developing tradespaces between these two objectives lends itself well to the enhanced MOCO presented in this work.

The optimization will take advantage of a transformation method developed in Temple and Collette (2012) to alter the shape of the vessel. The transformation method defines a spline across the centerplane of the vessel dictated by control points $\hat{T}_1, \dots, \hat{T}_N$. This splines sets distances to move stations defined in the table of offsets longitudinally fore and aft. The scantlings are defined using a structural discretization based on stiffened panels within various functional locations of the hull. This optimization method is developed and presented in Temple and Collette (2013a).

This leads to the following multi-disciplinary problem being developed.

SystemLevel :

$$\begin{aligned}
 &\text{minimize: } f(x) = (R_L(x), C_P(x)) \\
 &\text{with respect to: } \hat{x}_H = \hat{T}_{11}, \dots, \hat{T}_{1N} \\
 &\quad \hat{x}_S = (t_p, t_w, t_f, h_w, b_f)_i \text{ for } i = 1 \text{ to } n_f \\
 &\text{subject to: } \sum_{i=1}^N \|x_{H_i} - \hat{x}_{H_i}\|_2^2 + \sum_{j=2}^5 \sum_{i=1}^{n_f} \|x_{S_{j_i}} - \\
 &\quad x_{\hat{S}_{j_i}}\|_2^2 = 0 \tag{25}
 \end{aligned}$$

HydroDiscipline :

$$\begin{aligned}
 &\text{minimize: } F(x) = [d_{H_j}^+, d_{H_f}^+] \\
 &\text{with respect to: } T_{l_1}, \dots, T_{l_N} \\
 &\text{subject to: } |T_{new} - T| \leq 2 \\
 &\quad \sum_{i=1}^N \|x_{1_i} - \hat{x}_{1_i}\|_2^2 + d_{H_j}^+ + d_{H_f}^- = 0 \\
 &\quad R_T(x_H) + d_{H_f}^+ - d_{H_f}^- = 0 \tag{26}
 \end{aligned}$$

StructureDiscipline :

$$\begin{aligned}
 &\text{minimize: } F(x) = [d_{S_j}^+, d_{S_f}^+] \\
 &\text{with respect to: } (t_p, t_w, t_f, h_w, b_f)_i \text{ for } i = 1 \text{ to } n_f \\
 &\text{subject to: } W_S - 1.5W_{S_l} \geq 0 \\
 &\quad S_{M_V} - S_{M_{V_l}} \leq 0 \\
 &\quad U_{C_{S_p}} - U_{C_{S_{p_l}}} \leq 0 \forall p \\
 &\quad S_{M_p} - S_{M_{p_l}} \leq 0 \forall p \\
 &\quad d(x_i, x_j) < 0.06 \forall j, i \\
 &\quad \sum_{j=2}^5 \sum_{i=1}^{n_f} \|x_{2_{j_i}} - \hat{x}_{2_{j_i}}\|_2^2 + \dots \\
 &\quad \dots d_{S_j}^+ + d_{S_j}^- = 0 \\
 &\quad C_P(x_S) + d_{S_f}^+ - d_{S_f}^- = 0 \tag{27}
 \end{aligned}$$

In this formulation the system level optimizer is minimizing production steelwork cost and the resistance at the vessel's design speed while ensuring that interdisciplinary feasibility is maintained. To transform the shape of the hull the method developed in Temple and Collette (2012) is used. This method defines a transformation vector across the centerplane of the vessel defined by the set of N control points, $T_l = \{T_{l_1}, \dots, T_{l_N}\}$. This transformation vector is used to represent a function, $f_T(x)$, gives the optimizer a distance to perturb stations of the vessel's hull. A station of a ship's hull is a cross sectional slice taken in the transverse direction at a given point on its longitudinal axis. Thus, the optimizer will take cross sectional slices through the ship's hull at given points along its centerline axis and move those stations by the distance dictated by $f_T(x)$. The movement of these stations will create a transformed hull form. The optimizer can change the control points that dictate the shape of $f_T(x)$ and, through this, control the shape of the transformed vessel.

The hydrodynamic discipline problem is minimizing the difference between its design variables and the corresponding system level variables as a primary goal and the resistance as a secondary one. As the ship's hull is transformed the displacement remains constant and, thus, the draft is altered. This discipline constraint ensures that the draft does not change more than 2 meters.

The ship's structure is controlled by altering the scantlings of the stiffened panels that comprise it. The structure is initially decomposed into 'functional locations' that represent regions of the structure that have similar stress regions based on their location in the ship. These locations are then broken down into individual panels composed of a plate and number of stiffeners. The optimizer can control the thickness of the plate, t_p , the thickness of the stiffener web, t_w , and the thickness of the stiffener flange, t_f . It can also control the height and breadth of the flange, h_w and b_w . These scantlings can be altered for each panel within each of the n_f functional locations defined for the ship's structure. Since each functional location witnesses similar stress ranges, all the panels within a given location are identical for the purpose of this work.

The constraints satisfied here are that the weight does not increase more than 50 % and that the section modulus of the entire structure is not reduced. Additionally for each stiffened panel, the ultimate compressive strength of panel and its section modulus in bending must also not be reduced. This discipline also ensures that the components of the different panels do not come within 6mm of each other. The section modulus of the entire vessel and the corresponding panel ultimate compressive strength, UCS , are used together as an estimate of the global strength of the vessel. The UCS of each panel is calculated based on a regression of physical panel collapse tests developed by Paik and Duran (2004). This approach uses non-dimensional elastic column and plate slenderness ratios as has been shown suitable for marine-type panels.

The hydrodynamic discipline uses an SQP (sequential quadratic programming) optimizer included in the Python-based optimization suite pyOpt (Perez et al. 2012) This algorithm is a gradient based optimizer that is effective at minimizing functions that are C_1 continuous with well defined minima. The structural discipline uses the LEGA described in Section 2.2. This algorithm has been specifically designed for sub-problems such as this that are not continuous and may be multi-modal in objective space with weak minima. It is also designed to minimize computation time by only requiring one run for an entire population.

To demonstrate the ability of the enhanced MOCO to develop these Pareto fronts in the naval design space a nominal DTMB5145 naval combatant is used as the initial design. The hull form can be found in Benedetti et al. (2007) and a bodyplan is shown in Fig. 12. A nominal midship structure used is taken from Ashe et al. (2009) and can see in Fig. 13. For this example the transformation vector, T_l will be controlled by 5 points and the structure is broken down into 7 functional locations. This means there are 40 design variables at the system level. This type of problem can be time consuming to solve, even when using a simplified version such as presented here. Due to this the eMOCO

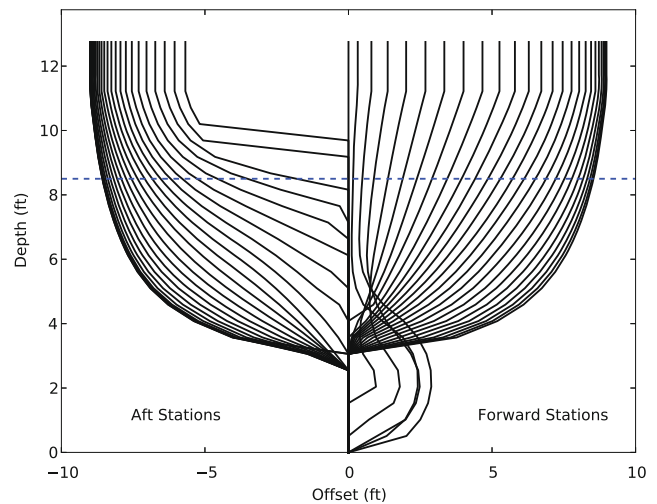


Fig. 12 Bodyplan for nominal naval combatant

has been designed to allow for parallelization across multiple cores in Python. While formal computational timing of multiple replicate runs was not made, this example shown here represented close to 1500 hours of CPU time across 200 cores.

In order to test the ability of the eMOCO to find lifetime cost trade-offs in the naval design space it will be used, along with a traditional optimization method to find a Pareto-front between steelwork production cost and resistance. The lower resistance will lead to less operational costs, and historically has been the driver behind most naval design optimization. However, by considering multiple cost aspects the designer can focus on reducing total ownership costs from both a structural and hydrodynamic perspective. This is similar in complexity to the problem presented in

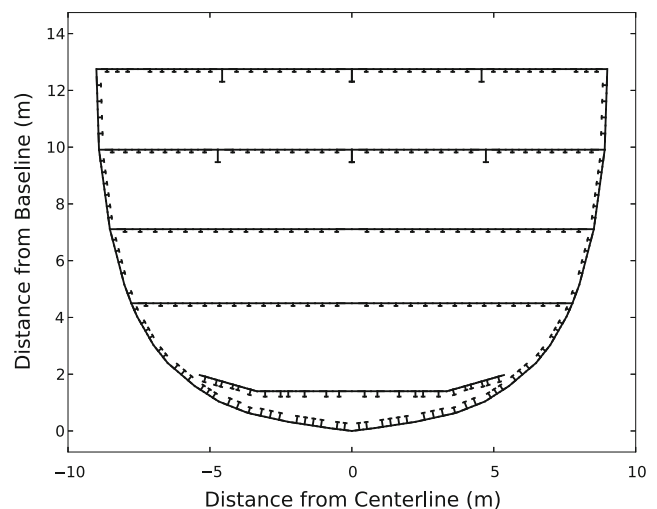


Fig. 13 Internal structure for nominal combatant

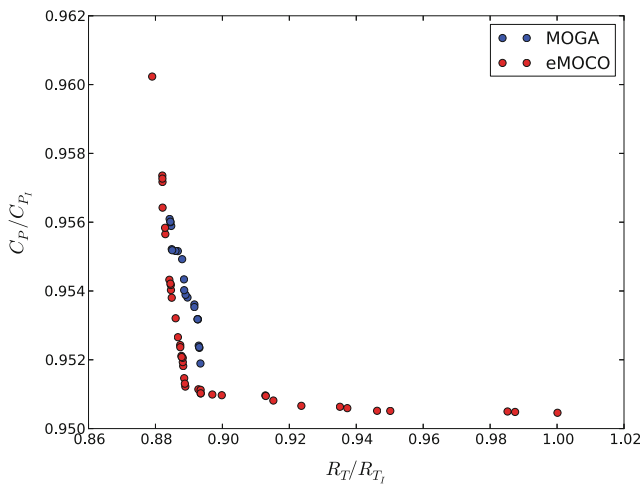


Fig. 14 Pareto front for nominal naval combatant

Temple and Collette (2015), albeit with a structural production metric. In the aforementioned work a traditional optimization algorithm was unable to find the complete Pareto front, and specifically struggled to resolve the resistance portion.

4.4 Results

The Pareto front found using both the enhanced MOCO and an NSGA-II can be seen in Fig. 14. In order to produce the results the system level optimizer evolved 200 individuals over 200 generations. This was sufficient time for each Pareto-front to converge to a point where it was no longer advancing through objective space.

In this figure both objectives are normalized by their respective values for the initial hull. As can be seen both optimizers are able to produce fronts with design solutions that dominate the original design with respect to both objective functions. However, the results here show a similar trend as the author’s previous work - namely, that traditional optimization routines cannot resolve the full front.

Even for this highly simplified test case for the enhanced MOCO the MOGA is only able to resolve the production end of the Pareto front, showing no solutions along the resistance side of the front. This is in line with the results found in Temple and Collette (2015). Meanwhile, the eMOCO is able to resolve solutions in both respective regions. The ability of the eMOCO to ensure that clustering does not reduce the integrity of the Pareto front shown in Figs. 6 through 11 is evident here as well. The NSGA-II has clustered in a small, and relatively insignificant, portion of the Pareto front, while the eMOCO is able to fully resolve the small ‘knee’ region. This gives the designer a better understanding of the design space and ensures that the information garnered from the Pareto-front is truly aiding them in reducing

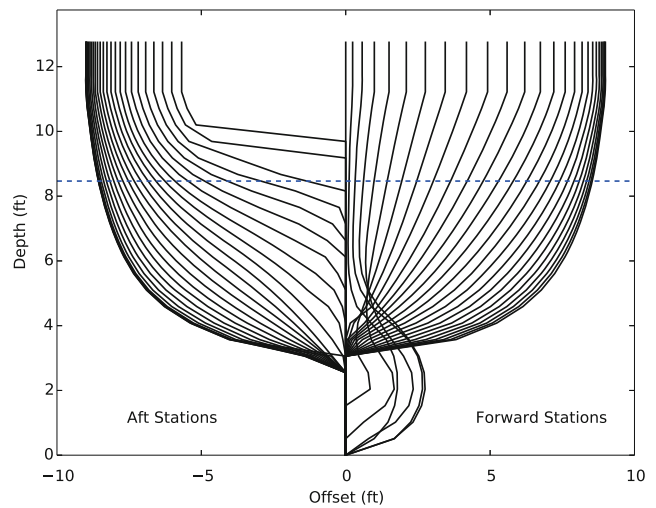


Fig. 15 Hull form for vessel with highest resistance

lifetime ownership costs. Given that the purpose of this type of optimization is to understand trends in the design space, capturing that space is critical. When only small portions of it are resolved it valuable information can be missed. For instance: the ends of the Pareto front (the portions missed by traditional optimization) are examined in Figs. 15 through 17.

From Fig. 15 it can be seen that in the structure that is cheapest to produce structure the bulbous bow of the original vessel is simplified. Along the Pareto front - as the knee

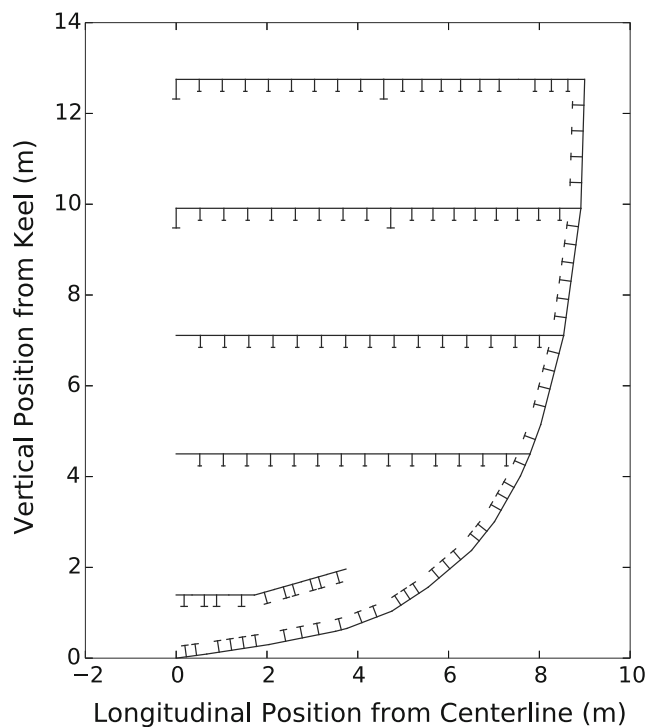


Fig. 16 Structure for vessel with highest resistance

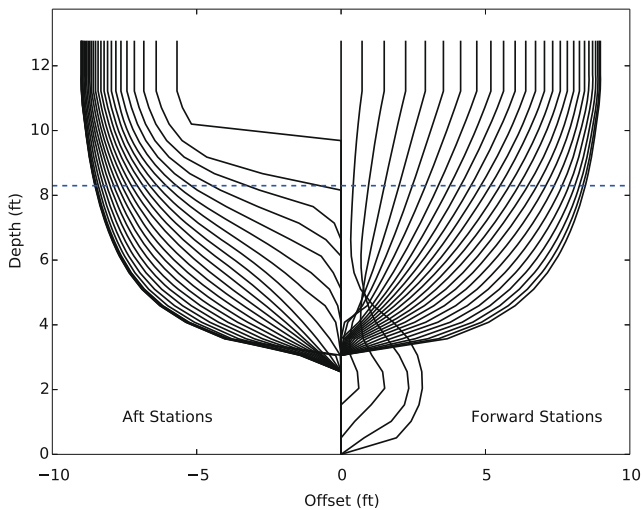


Fig. 17 Hull form for vessel with highest production cost

is approached - the bulbous bow becomes wider and the stern shallower - reducing the overall resistance. This can be seen in Figs. 17 and 18 where the resistance is minimized. Along the edges of the Pareto front between these points and the 'knee' the evolution of these trends can be seen. It is this information that is critical to the engineer - as understanding the trends in both the hydrodynamic and structural space and, specifically, how they influence each other is vital to

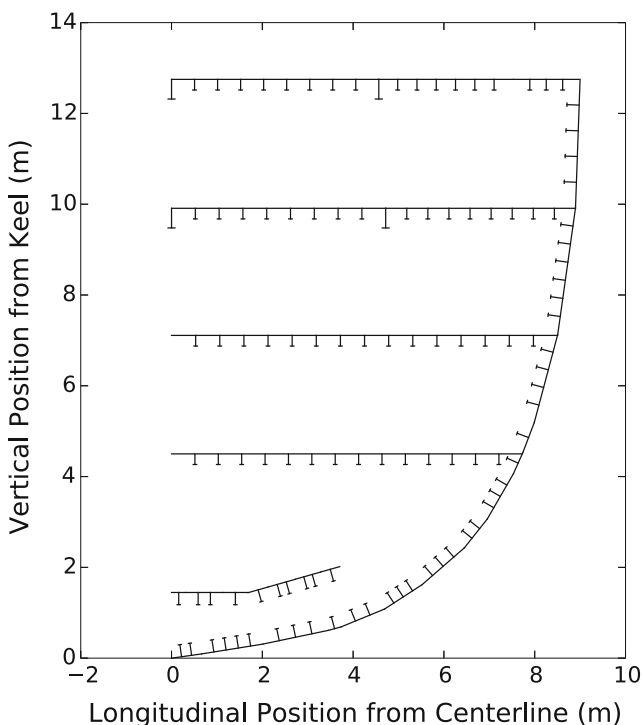


Fig. 18 Structure for vessel with highest production cost

reducing the overall ownership costs. These differences in the converged Pareto-front are due to the eMOCO's goal-programming implementation that allow the discipline-level optimizer to locally reduce their objective functions in areas of the design space that are problematic for the system-level. There are regions of the objective space with large flat areas or weak minima that are difficult to move a population through when the larger, multi-disciplinary problem is being considered. By allowing the discipline level optimizers, that are focused on a problem that is smaller in scope, to nudge the population through these areas, the system-level is able to converge to a better final Pareto-front. This shows that the enhancements developed for the eMOCO allow it to overcome the specific difficulties of the early-stage naval design problem.

As more information about the vessel becomes known and the design phase is completed only small portions of this front, for example the area in the 'knee', will be important. Future work could look at further accelerating the optimization process by focusing in on this region and not resolving the entire front. In the current work, the entire front is significant as it indicates both topology of the trade space - a very narrow knee region, and the penalty that would be paid by optimizing for one objective in isolation. Additionally, the front is better resolved than the MOGA front as noted, which shows the increased power of the proposed algorithm. The proposed enhanced collaborative optimization framework is able to resolve these Pareto-fronts and reveal this information.

5 Conclusion

The impact of lifetime ownership cost is becoming increasingly important to the operators and owners of large naval vessels. The cost to maintain a vessel and keep her operational is becoming a limiting factor in its possible service life. This makes it desirable to design new vessels with the aim to reduce these costs to ensure they can continue to operator as they need to. To do this it is important to understand the trade-offs between different categories of lifetime cost and, especially, the impact of the interactions between them on design. Unfortunately traditional multi-objective optimization routines are not capable of reliably resolving these trade-offs.

To address this the author's have proposed a multi-objective collaborative optimization framework with two enhancements to facilitate solving this unique problem: a goal-programming optimization at the discipline level and a new type of single objective genetic algorithm to more effectively search the structural sub-space. The eMOCO has been verified on simple multi-objective and multi-disciplinary test problems and been shown to be capable

of solving them. It also been shown to be able to resolve the trade-space for a simple naval design example. In even this simplified example the traditional optimizer is unable to resolve this front, showing the need for this type of algorithm to solve this problem.

When exploring more complex implementations of this problem, especially if maintenance cost was included, the traditional optimizer would stumble further and the benefit of an enhanced MOCO would become greater. Using this framework to explore and understand the trade-offs between production, maintenance, and operation costs could reveal trends in the design space that could aid designers in finding vessels whose total lifetime costs were minimized, and whose service lives were robust to changes in the latter years of operation.

References

- Ashe G, Cheng F, Kaeding P, Kaneko H, Dow R, Broekhuijsen J (2009) Committee V.5 Naval Ship Design. In: 17th International Ship and Offshore Structures Congress, vol 2
- Aute V, Azarm S (2006) A genetic algorithms based approach for multidisciplinary multiobjective collaborative optimization. American Institute of Aeronautics and Astronautics
- Benedetti L, Boucasse B, Broglia R, Fabbri L, Gala FL, Lugni C (2007) Pmm model test with ddg51 including uncertainty assessment. Tech. Rep. 14, INSEAN
- Brenner M, Sener B (2011) Design for energy efficiency: Simulation-driven design and hydrodynamic optimization for greener ships and EEDI compliance. In: 1st International Symposium on Naval Architecture and Maritime. Istanbul
- Chung HY, Manuel L, Frank KH (2006) Optimal inspection scheduling of steel bridges using nondestructive testing techniques. *J Bridg Eng* 11(3):305–319
- Collette Y, Siarry P (2005) Three new metrics to measure the convergence of metaheuristics towards the pareto frontier and the aesthetic of a set of solutions in biobjective optimization. *Comput Oper Res* 32:773–792
- Deb K, Pratap A, Agarwal S, Meyarivan T (2002) A fast and elitist multiobjective genetic algorithm: NSGA-II. *IEEE Trans Evol Comput* 6(2):182–197
- Depince P, Rabeau S, Bennis F (2005) Collaborative optimization strategy for multi-objective design. In: Proceedings of Computers and Information in Engineering Conference
- Gunawan S, Azarm S, Wu J, Boyars A (2003) Quality-assisted multi-objective multidisciplinary genetic algorithms. *AIAA J* 41(9):1752–1762
- Hannapel SE (2012) Development of Multidisciplinary Design Optimization Algorithms for Ship Design Under Uncertainty. Ph.D. thesis, The University of Michigan
- Harries S, Valdenazzi F, Abt C, Viviani U (2001) Investigation on optimization strategies for the hydrodynamic design of fast ferries. In: 6th International Conference on Fast Sea Transportation, Southampton
- Hart CG, Vlahopoulos N (2009) An integrated multidisciplinary particle swarm optimization approach to conceptual ship design. *Struct Multidiscip Optim* 41(3):481–494
- Krol WP (1991) Midship Section Design Using a Bilevel Production Cost Optimization Scheme. *Journal of Ship Production* 7(1):29–36
- Junca M, Sanchez-Silva M (2013) Optimal maintenance policy for permanently monitored infrastructure subjected to extreme events. *Probabilistic Engineering Mechanics* 33:1–8
- Kim S, Frangopol DM (2011) Inspection and monitoring planning for RC structures based on minimization of expected damage detection delay. *Probabilistic Engineering Mechanics* 26(2):308–320
- Kuhn J, Chevalier K, Schlageter E, Scragg C, Wyatt D (2007) Use of Linear Programming and Basis Functions for Hull-Form Optimization. In: 9th International Conference on Numerical Ship Hydrodynamics
- Li F, Li G, Sun G, Luo Z, Zhang Z (2010) Multi-disciplinary optimization for multi-objective uncertainty design of thin walled beams. *Computers Materials and Continua* 19(1):37
- Maki K, Doctors L, Scher R, Wilson W, Troesch A (2008) Conceptual Design and Hydrodynamic Analysis of a High-Speed Sealift Adjustable-Length Trimaran. In: Transactions of SNAME
- Martins J, Lambe AB (2013) Multidisciplinary Design Optimization: A Survey of Architectures. *AIAA Journal* 51(9):2049–2075
- Mermiris G, Das P, Moatsos I, Winkle I (2005) Towards a Method for the Optimisation of Midship Section in Terms of Production Cost in Preliminary Ship Design. In: Proceedings of the International Conference on Offshore Mechanics and Arctic Engineering, vol 1B, pp 981–990
- Paik JK, Duran A (2004) Ultimate strength of aluminum plates and stiffened panels for marine application. *Mar Technol* 41(3):108–121
- Perez RE, Jansen PW, Martins JRRA (2012) pyOpt: A Python-based object-oriented framework for nonlinear constrained optimization. *Structures and Multidisciplinary Optimization* 45(1):101–118
- Rahman MK, Caldwell JB (1995) Ship Structures: Improvement by Rational Design Optimisation. *Int Shipbuild Prog* 42(429)
- Ruiji S, Liangjin G, Zijie F (2011) Multi-objective collaborative optimization based on evolutionary algorithms. *J Mech Des* 133(104):502
- Sariz E (2006) An optimization approach for fairing of ship hull forms. *Ocean Eng* 33(16):2105–2118
- Stambaugh K, Barry C (2014) Naval Ship Structure Service Life Considerations. *Nav Eng J* 126(3):103–117
- Tamiz M, Jones D, Romero C (1998) Goal programming for decision making: An overview of the current stat-of-the-art. *Eur J Oper Res* 111:569–581
- Temple D, Collette M (2013) Optimization of structural design to minimize lifetime maintenance cost of a naval vessel. pp 525–532
- Temple D, Collette M (2013) Optimum lifetime maintenance schedule for naval vessels subjected to fatigue and corrosion. In: Proceedings of the 12th International Symposium on Practical Design in Shipbuilding
- Temple D, Collette M (2015) Understanding the trade-offs between producibility and resistance for differing vessels and missions. *Journal of Ship Design and Production* 31(2):1–12
- Temple DW, Collette M (2012) Multi-Objective Hull Form Optimization to Compare Build Cost and Lifetime Fuel Consumption. In: The 11th International Marine Design Conference, pp 391–403
- Tuck EO (1989) The Wave Resistance Formula of J.H. Michell (1898) and its Significance to Recent Research in Ship Hydrodynamics. *J Aust Math Soc* 30
- Tuck EO, Lazauskas L (2008) Drag on a ship and michells integral. In: 22nd Int. Congress of Theoretical and Applied Mechanics, Adelaide, South Australia
- Yang JM, Hwang CN (2002) Optimization of Corrugated Bulkhead Forms by Genetic Algorithms. *J Mar Sci Technol* 10(2):146–153
- Zalek S, Beck R, Parsons M (2008) Nonlinear hullform transformation for use with design optimization. In: Proceedings of the 2008 Grand Challenges in Modeling and Simulation Conference, Simulation series, vol 40, pp 227–238. Edinburgh