Proceedings of the ASME 2008 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference
IDETC/CIE 2008
August 3-6, 2008, Brooklyn, New York, USA

DETC2008-49335

PRODUCT FAMILY COMMONALITY SELECTION THROUGH INTERACTIVE VISUALIZATION

Ritesh Khire, Jiachuan Wang, Trevor Bailey, Yao Lin
United Technologies Research Center, East Hartford, CT 06118

Timothy W. Simpson
Penn State University, University Park, PA 16802

ABSTRACT

High dimensionality and computational complexity are curses typically associated with many product family design problems. In this paper, we investigate interactive methods that combine two traditional technologies – optimization and visualization – to create new and powerful strategies to expedite high dimensional design space exploration and product family commonality selection. In particular, three different methods are compared and contrasted: (1) exhaustive search with visualization, (2) individual product optimization with visualization, and (3) product family optimization with visualization. Among these three, the individual product optimization with visualization methods appears to be the most suitable one for engineer designers, who do not have strong optimization background. This method allows designers to "shop" for the best designs iteratively, while gaining key insight into the tradeoff between commonality and individual performance. The study is conducted in the context of designing a UTC product using an in-house, system-level simulation tool. The challenges associated with (1) design space exploration involving mixed-type design variables and infeasibility, and those associated with (2) visualizing *product family design spaces* during commonality selection are addressed. Our findings indicate a positive impact on the company's current approach to product family design and commonality selection.

1 INTRODUCTION

In many technology-focused companies, engineering practice is evolving to a state where rigorous physics-based models are being used to analyze and verify product design performance and reliability. Increased product complexity and competitive pressure to accelerate product introductions to market have motivated large companies such as Boeing, Ford, General Motors, and United Technologies Corporation (UTC) to pursue rigorous analytical approaches to engineering design. At UTC, for instance, benchmarks have shown that a commitment to model-based design, analysis, and verification can provide as high as 30% savings in engineering costs, and also cut development time in half.

To complicate matters further, most companies now offer many of their products as variants within a large product family. A product platform concept, consisting of common components or subsystems across the family, is typically used to generate high profits [1, 2]. A typical product family consists of a set of products that have (1) some unique characteristics and (2) shared components and modules. As such, product family optimization inherits all of the idiosyncrasies involved with single product optimization (such as multiple objectives and mixed variable types), while adding coordination across the family. The latter involves additional intricacies such as making commonality decisions, significantly high dimensionality, and combined combinatorial-attribute decision making. This creates additional challenges (such as high computing requirements) when physics-based models are used for engineering design and verification of product families. We refer the reader to work by de Weck [3], which provides a thorough example of the types of models needed to translate product family design decisions into profitability for a company.

Typical steps in product family optimization include: (1) defining the product family, (2) formulating the product family optimization problem, (3) solving the product family optimization problem, and (4) evaluating the tradeoff between different product family design alternatives and making a final decision. In this paper, we discuss the challenges involved with formulating the product family design problem within an industrial setting. Thereafter, we compare and contrast three methods that focus on the last step of the aforementioned process, i.e., evaluating the tradeoff and making a final decision. While the arguments, challenges, methods, and results discussed in this paper are within the context of a specific problem, we assert that our findings are generalizable to industrial problems of similar complexity and of comparable levels of technical difficulty.

The next section reviews challenges in product family design and optimization as well as related work in this area. Section 3 introduces a UTC product family design problem used in this work. Section 4 presents, compare, and down-select between the three methods that we are investigating. Section 5 presents and discusses the results obtained using the down-selected method. Section 6 provides closing remarks.

2. RELATED WORK IN PRODUCT FAMILY DESIGN

As mentioned earlier, a typical product family consists of a set of products that share common components and/or modules. There are many advantages of sharing common components [4], including: (1) economies of scale, (2) reduced development time, (3) reduced SKU (Stock Keeping Units), (4) reduced manufacturing and service complexity, and (5) increased product quality due to less part variety. From a business perspective, these advantages can be tied to low cost products or increased profitability of a product line. However, these advantages must be carefully weighed against the potential disadvantages of commonality.

Perhaps the biggest drawback to commonality is the increased potential for the lack of product distinctiveness [5]. As more components are shared, it becomes increasingly difficult to differentiate between product variants in the market [6]. Moreover, individual product performance may degrade significantly due to commonality, resulting in the loss of market share [7].

Therefore, a product family designer must carefully balance the tradeoff between commonality and individual product differentiation. More than 40 optimization-based approaches have been proposed to help resolve this tradeoff [8]. These approaches can be generally categorized into one of three product family design strategies: (1) select the platform first and then optimize the platform and individual products, (2) optimize the individual products first and then select the platform that causes the minimum performance loss with maximum commonality savings, and (3) simultaneously select the platform while optimizing the platform and the individual products. There are advantages and disadvantages to each strategy [9]. Note that strategy (1) requires a priori selection of the platform. Typically, such selection is based on designer's experience and, often, no systematic processes are used during selection. On the contrary, the three methods presented in the current paper are based on strategies (2) and (3), which provide systematic platform selection processes, see Section 4.

Regardless of which particular strategy is employed, product family design optimization entails (a) exploration of product family design space and (b) ultimately making a decision regarding the appropriate level of commonality in the family, what we refer to as *commonality selection*. With regards to (a), involvement of multiple products significantly increases the dimensionality and complexity of product family design space, even for modest-sized product families. As shown in Figure 1, design space exploration of a product family involves not just one product model, but all the individual product models within a product family.

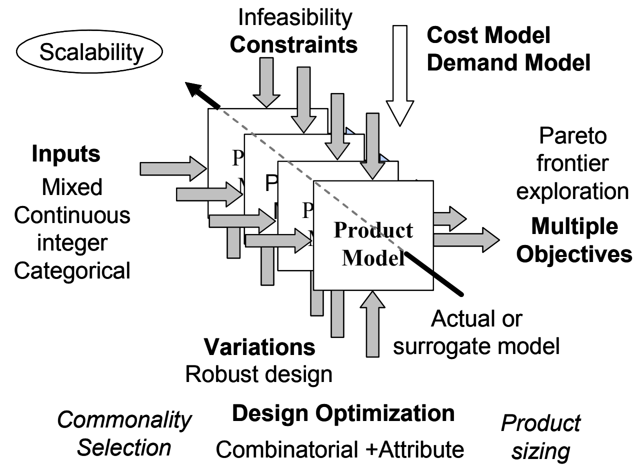


Figure 1 Challenges in Product Family Optimization: Design Space Exploration

Product family optimization consists of both combinatorial optimization (to select the common components and modules that embody the platform) and parametric optimization (to optimize both the platform and non-platform design variables). Working with mixed-type design variables and multiple conflicting design objectives exacerbates the problem further, as does having "black box" simulations with implicit constraints that yield many infeasible solutions within the design space. This makes the design space discontinuous such that it is difficult to apply traditional optimization techniques to search the design space.

Once the design space has been explored to the best extent possible, then designers can proceed with (b), namely, determining the appropriate level of commonality for the product family. The challenges associated with commonality selection are: (1) it involves the aforementioned tradeoff between commonality and individual product performance, which needs human preference guidance, and (2) there are multiple valid solutions for each individual product [10], and commonality selection needs to examine all of the data and make sound judgment.

What we have seen and experienced in practice, however, is that designers – already leery about enforcing commonality – are all the more hesitant to trust the optimization results, and rightly so given the challenge in formulating an accurate optimization problem that reflects the subjectivity involved in the tradeoff between commonality and individual performance. This reluctance also stems from (1) an innate, albeit unfounded, belief that any commonality will adversely affect the product's performance combined with (2) the inability to view the tradeoffs that are occurring within the design space. It is one thing to visualize how an individual product performs relative to a known baseline design, but it is much more challenging to visualize how an entire family of products performs. Work in this area has been very limited to date [10], and this motivated our current study, namely, to promote commonality, designers must be able to visualize the tradeoffs that are occurring in the product family. We propose to take this one step further by putting designers "in the loop" during optimization process as shown in Figure 2, leveraging *recent research to support "Design by Shopping" approaches to engineering design [11, 12, 13, 14].*

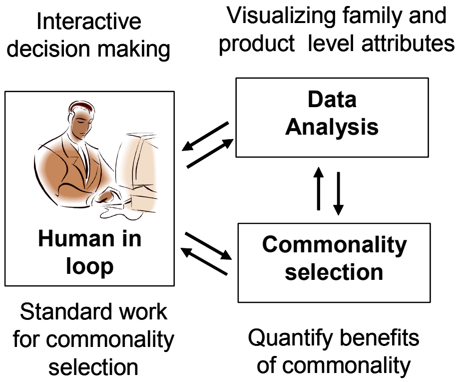


Figure 2 Challenges in Product Family Optimization: Commonality Decision Making

An intuitive approach to make commonality decisions is to present all of the possible solutions to designers, and allow them to make commonality decisions interactively. However, this is easier said than done given the high dimensionality and sheer amount of data associated with a product family design. The three methods investigated in Section 4 provide three different approaches for exploring the design space and then visualizing it before making commonality decisions. Before introducing the methods, however, we first introduce the UTC product family design problem that has motivated our work.

3 UTC PRODUCT FAMILY DESIGN PROBLEM

Figure 3 depicts the scope of the UTC product family problem. Because of the proprietary nature of the products being designed, we restrict ourselves to terminologies such as UTC product and UTC product family. The family consists of 11 product variants that are each defined by three continuous and two categorical variables. The overall UTC product family design problem includes a total of 33 continuous and 22 categorical variables, 23 objectives (2 objectives/unit plus a commonality index), and more than hundred constraints. The constraints are internal to the system-level product simulation model that was developed in-house to support physics-based design and analysis. The challenges involved in solving the UTC product family design problem are discussed next before being used to illustrate, compare, and contrast the interactive product family design methods in Section 4.

3.1 UTC Product Design Space and Optimization Algorithm

For the UTC product family problem, we consider five design variables for each product, X_1 to X_5, of which three are continuous and two are categorical. Unlike continuous variables, the numerical values of categorical variables do not have any physical significance. For example, refrigerant temperature is a continuous variable, as its value indicates hotness or coldness. On the other hand, a compressor model number is a categorical variable, as it does not necessarily contain any physical meaning.

Handling continuous and categorical variables simultaneously poses computational challenges for any optimization environment. Either mixed-integer nonlinear programming (MINLP) or non-gradient based methods can potentially be used to solve such an optimization problem, and MINLP formulations for product family design are being investigated [15]. Success of MINLP approaches in finding optimal solutions typically depends on a number of factors, such as: starting point, convexity of design space, and continuity and infeasibility associated with the design space. The UTC product family design problem entails a significant level of (1) discontinuity and (2) infeasibility, making it inappropriate for MINLP methods. As such, most of the product family optimization problems are discontinuous in nature due to discrete choices of platform and non-platform variables. As for the infeasibilities, for UTC product family, they arise from non-convergence of the system-level simulation model itself – an area of ongoing investigation and future research. Our experience suggests that when the existing UTC product model is randomly sampled within the design space, at best 40% of the samples yield feasible designs. If we extend this observation to the UTC family of 11 products, the feasibility of the design space would be $(0.4)^{11}$ = 0.004%. To put than in perspective, the initial sample size of GA algorithm will have to be at least 25000 to get one feasible solution. This significant infeasibility poses additional challenges to the design space exploration process, impacting the choice of the algorithm used for optimization.

To overcome these discontinuities and infeasibilities, we selected a non-gradient-based evolutionary algorithm for the UTC product family problem. In particular, we use the Non-Dominated Sorting Genetic Algorithm (NSGA-II), a popular multi-objective genetic algorithm (GA) developed by Deb [16]. NSGA-II is robust to discontinuities in the design space and is capable of searching for global optima. An important feature of the NSGA-II is its ability to explore Pareto frontiers, and it has been used by many researchers to solve product family optimization problems successfully [8]. The objectives for the UTC problem are discussed next.

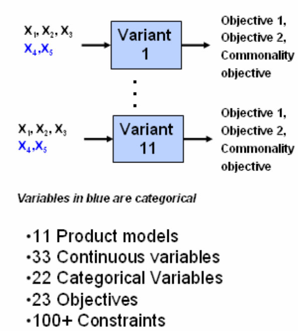


Figure 3 Scope of UTC Product Family Design Problem

3.2 Objectives for the UTC Product Family

As seen in Figure 3, each UTC product family involves multiple objectives. Specifically, each product has two objectives, with a preference for maximizing Objective 1 and minimizing Objective 2. This creates 22 objectives for the family, with the 23rd objective being commonality index within the product family (see Section 3.3).

Handling multiple instances of similar objectives poses a unique challenge for product family design. Aggregating similar objectives into a single objective seems to be an intuitive way of tackling the problem at hand. However, aggregation poses some critical computational and practical challenges, such as: (1) loss of individuality, (2) handling different scales of objectives (e.g., value of Objective 1 for one of the product variants may be in tens, whereas that for another variant may be in thousands), and (3) handling the relative importance between different units.

Based on our observations, these critical challenges have been handled based on ad-hoc rules, which can potentially be a significant source of sub-optimality. Decomposition-based optimization strategies can be implemented to alleviate some of these challenges, but at the expense of added computational complexity and coordination cost [9]. As we shall see in the next section, visualization-aided decision making framework appears more effective than simple aggregation of objectives.

3.3 Commonality Selection

Typically, an engineering system consists of a number of possible commonality choices (components and modules). As such, comprehensive product family design involves exploring potential commonality options for all these components and modules. For this study, we will focus on selecting only one common feature among different UTC products; refer to as the commonality variable.

Quantifying the benefits of commonality is important for making commonality decisions; however, quantifying the benefits of commonality is extremely difficult in practice. Hence, commonality indices are typically used as surrogates to qualify the resulting cost savings. A typical commonality index is a function of the number of components / assemblies / manufacturing processes that are common across different products in the family [17]. Khajavirad and Michalek [18] argue that the commonality index (CI) introduced by Martin and Ishii [19] captures the tooling cost savings of component commonality better than any other commonality metric. Using this index as our starting point, and given our limited focus on commonality variable, our commonality index for this product family optimization problem is expressed as a percent of the number of different commonality variable values used in the family – see Section 4.3 for more detail. *Interestingly, in Section 4.2, we show the effectiveness of the visualization aided commonality selection procedure that does not rely on traditional commonality indices.*

4. INTERACTIVE VISUALIZATION METHODS

In this section, we discuss the three methods that we used to solve the UTC product family problem. Table 1 summarizes the aspects of each method. We discuss each method in detail in the sub-sections that follow.

Table 1: Summary of Product Family Optimization Methods

	Design Space Exploration	Commonality Selection	Advantage	Disadvantage
Method 1	DOE sampling	Interactive visualization	Simple, interactive,	No optimality Computationally intensive,
Method 2	Individual product optimization	Interactive visualization	Interactive, Pareto Optimal	Limited variety of product families
Method 3	Product family optimization	Formulated in optimization	Unlimited variety in product families	Computationally intensive, Not interactive

4.1 Method 1 (Exhaustive + Visualization)

Figure 4 shows the steps used in Method 1: exhaustive sampling followed by visualization. In this method, the design space is explored exhaustively (and separately) for each product using a large number of sample points. The number of sample points typically depends on factors such as: (1) available time, (2) computational resource, (3) number of design variables, and (4) prior knowledge of the design space. Based on our experience, the first two factors dominated the UTC product family design problem.

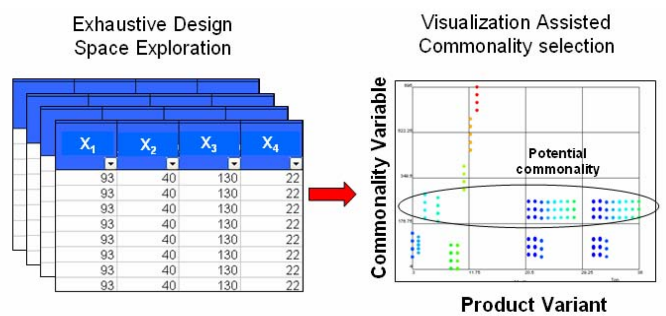


Figure 4 Exhaustive Product Family Design

After exhaustive sampling, the input and output data for each product is assembled into a single file location (typically an Excel spreadsheet or a text document) for product family design. In particular, the product family data is processed to enable commonality decisions using any of a variety of filtering tools or sorting techniques. For example, when selecting the value of commonality variable, the designer can successively filter different commonality variable values to identify UTC products that share a particular variable value.

According to our experience, such text-based filtering is unfriendly and time-consuming for designers. They prefer visual tools to do this processing interactively – such techniques are simpler, more user friendly, and allow the designer to explore larger design spaces with ease. Also, we found that the quality (tradeoff between the objectives) of the design selected using visualization technique is frequently superior to that obtained from the aforementioned text-based filtering techniques. Since the specific visualization techniques are also used in Method 2, they are discussed in the next section after introducing Method 2.

In summary, the advantages of Method 1 are that it is easy to use, and it retains many aspects of the engineer's current design practice. While this latter point may seem counter-intuitive, it is critical when considering adoption of new techniques in current practice. Disadvantages of this method are that exhaustive sampling many not uniformly sample the objective space and the computational expense of the simulation model may limit the number of samples that can be obtained. Both of these may lead to designs that have poor tradeoff between design objectives, creating an opportunity to utilize optimization to search the design space more efficiently as advocated in Method 2 as discussed next.

4.2 Method 2 (Individual Opt + Visualization)

Method 2 is similar to Method 1 except that the exhaustive design space exploration used in Method 1 is replaced by systematic optimization of individual products in Method 2 to search the design space more efficiently. Figure 5 shows the integration of the UTC simulation model within Engineous' iSight environment [20] for performing individual optimization. As discussed earlier, the UTC product design problem includes categorical variables, which are mapped to arithmetic variables before sending them to the system level model, as shown in Fig. 5. Next, we discuss the use of visualization for exploring *product family design space*.

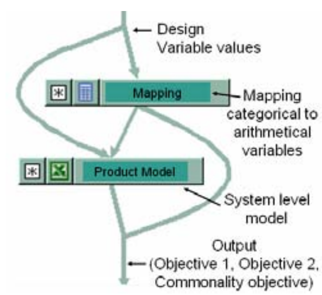


Figure 5 Optimization of Individual Product

4.2.2 Interactive Visualization

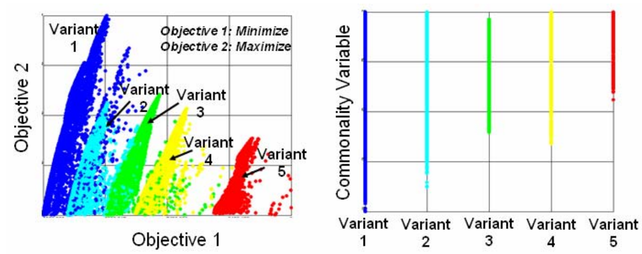
After individual optimization of all the products is completed, multi-dimensional data visualization is used to perform commonality selection. For this work, we employ the Applied Research Laboratory's Trade Space Visualizer (ATSV) [12, 21, 22, 23, 24], a Java-based application that displays multidimensional data using glyph, histogram, scatter, scatter matrix, and parallel coordinate plots. The ATSV is developed entirely in Java, making it cross-platform compatible, and offers linked views, brushing (filtering), preference shading, and Pareto filtering to help designers "shop" for the best design, which in this case, is the best product family given a selected level of commonality.

In the past, ATSV has been used primarily for single product optimization. The uniqueness of this paper includes extending the use of ATSV for product family optimization. As such, assembling data from individual optimizations of different products into a single file is a part of customizing ATSV for product family optimization. An important outcome of the current research is the identification of product family-specific capabilities for ATSV, such as data pre-processing for aggregating individual product data.

Figure 6 shows the design space for five variants in the UTC product family. Specifically, Figure 6(a) shows Objective 1 vs. Objective 2 space for different UTC product variants, while Figure 6 (b) shows available commonality variable values for different variants. Interestingly, Figure 6(b) suggests that commonality variable can potentially be shared by all five product variants by setting its value high.

This shows the simplicity and effectiveness of multi-dimensional data visualization in exploring the design space for a complete product family. Also, visualization (Fig. 6) has provided high-level information to the designer regarding possible commonality choices in the family.

Next, we use the commonality variable data shown in Figure 6(b) for making commonality selections. The impact of commonality selection is evaluated using the Objective 1 vs Objective 2 data shown in Figure 6(a).



(a) Efficiency vs Cost (b) Commonality vs variants

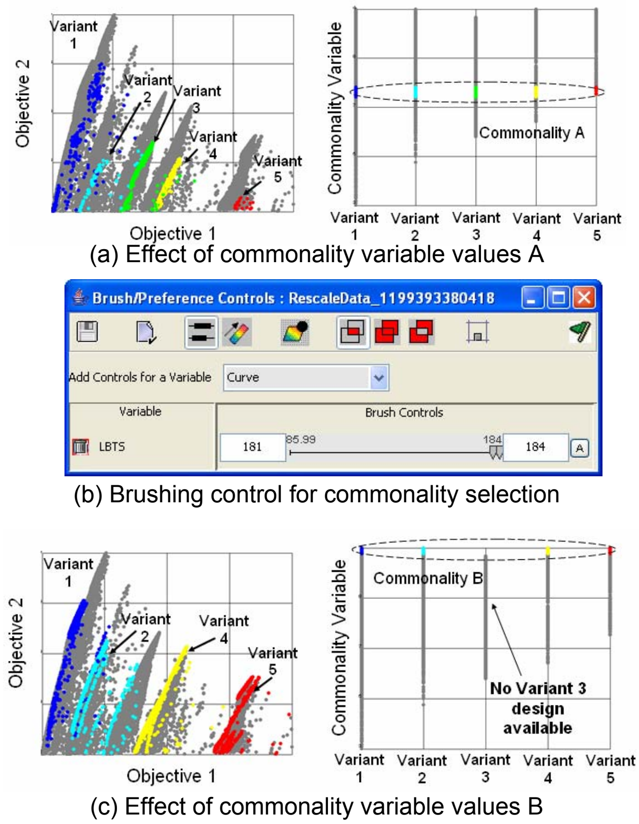
Figure 6 Product Family Design Space Obtained from Individual Optimization

4.2.2 Commonality Selection with Visualization

Figure 7 (a) and (c) show the effect of commonality variable values A and B on Objective 1 and Objective 2. Also, Figure 7 (b) shows the brushing controls in ATSV, which act as a sliding data filter to allow users to evaluate different commonality selections.

In Figure 7 (b), the brushing is set for the commonality variable. As the brush slides from left to right, the designs that have the corresponding commonality variable values are shown in colors. On the other hand, the designs with commonality variable values different from that selected by the brush are automatically turned grey, as seen in Figure 7 (a) and (c). Thus, by simply sliding the brush controller, the designer can visually evaluate the effect of change in the commonality variable value on the two objectives for all the products in the family

We further explain the commonality selection with the help of Figure 7 (a) and (c). By setting the brush at Commonality A, we generate Figure 7 (a). We can observe that commonality variable value A can be made common between all variants of the UTC products units. Additionally, we can also observe that commonality variable value A results in designs that are on the Pareto frontiers of the most UTC products. On the other hand, commonality variable value B does not result in any feasible design for product variant 3. Hence, commonality variable value B cannot be made common for the entire product family.



(a) Effect of commonality variable values A

(b) Brushing control for commonality selection

(c) Effect of commonality variable values B

Figure 7 Effect of Different Commonality Selection

Thus we observe that by combining individual product optimization with interactive visualization, we have developed a powerful method that is simple yet effective in making commonality selection. Additional advantages of Method 2 include an improved ability to find designs with better tradeoff resolution (all objectives show simultaneous improvement) over Method 1, added flexibility while still being moderately easy to use. A disadvantage of Method 2 is the post-processing of the individual product data to aggregate it for visualization. Also, there is a conflict of interest in that the optimization drives towards Pareto optimality for the individual products, while commonality selection may force you away from these individual Pareto frontiers for the benefit of the family. Solutions obtained from the individual optimization may not always be most appropriate for commonality selection, which is why visualization is all the more critical at this stage of the product family design process.

4.3 Method 3 (Product Family Opt + Visualization)

Within the context of the UTC product family design problem, we have also applied two product family-based optimization techniques: (1) all-in-one approach and (2) decomposition-based approach [9]. As the name implies, the all-in-one approach takes all the inputs and outputs of each individual product optimization problem, and combines them into a large optimization problem. A typical all-in-one optimization problem uses commonality index as an additional objective that ties all the products together.

Formulation of the all-in-one approach for the UTC product family design problem is shown in Figure 8. Typically, an all-in-one problem scales the number of design variables, constraints and objectives based on the number of product variants. This scaling increases the complexity of the all-in-one optimization problem. The two objectives, Obj1 and Obj2, shown in Figure 8 are obtained from aggregating corresponding objectives from each product. Commonality index (CI), in this case is simply the number of distinct values taken by the commonality variable across the product family. The CI ranges from 1 to k, k indicates each product uses a unique commonality variable value, while 1 implies that all the products share the same commonality variable value.

For all-in-one approach, visualization techniques can also be applied to help designers make commonality decisions. Figure 9 shows an illustrative scatter plot (Obj1 vs. Obj2) for a commonality study. We want to maximize Obj1 while at the same time minimize Obj2. Each point on Figure 9 represents a possible solution from the all-in-one optimization. Without a visualization tool, the designers were not able to differentiate between the commonality associated with each design. By color coding the solutions according to CI, the designers were able to see the clustering of the solutions based on different commonality levels. In Figure 9, the color code from 1 to 7 represents different number of commonality variable values needed for the UTC product family. The figure indicates that using 3 values for the commonality variable, which corresponds to the green color, makes a reasonable balance between Obj1 and Obj2. Using only one commonality variable value across the whole product family increases the obj2, see Figure 9 - dark blue solutions.

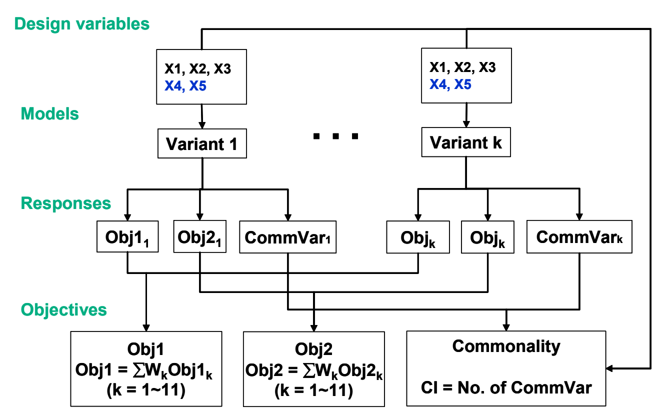


Figure 8 All-in-one Product Family Optimization

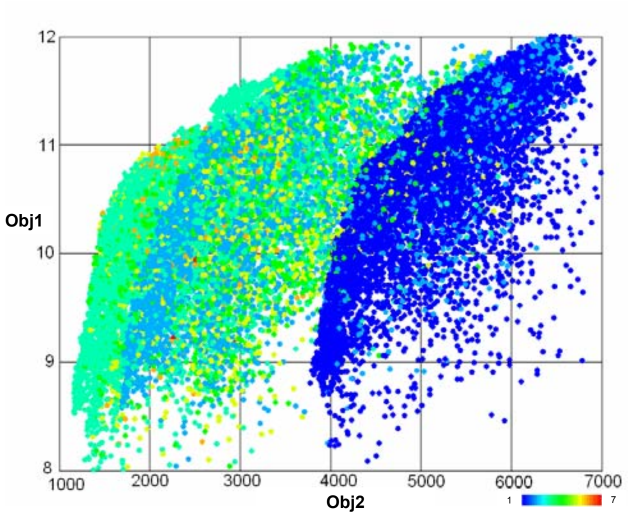


Figure 9 Results from Product Family Optimization

There are disadvantages associated with all-in-one approach. Since the approach lumps all outputs together, it does not considers individual performance explicitly, making it difficult to tradeoff performance among individual products. At the same time, typically, it is difficult to scale the problem formulation as the number of product variants increases. Consequently, the decomposition-based approach is an alternative approach to platform family optimization advocated by Khajavirad, et al. [9]. This approach considers each individual product performance explicitly, which makes it flexible to accommodate individual product specific evaluation criteria. Khajavirad, et al. [9] also hypothesized that the decomposition approach is likely to explore global optimality efficiently than the all-in-one approach.

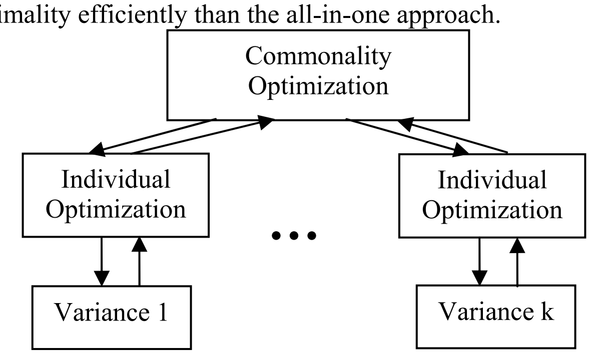


Figure 10 Decomposition-based Product Family Optimization

As shown in Figure 10, in the decomposition-based approach, the product family optimization problem is decomposed into two levels: (1) commonality optimization and (2) individual optimization. The commonality optimization determines the optimal platform configuration, while each individual optimization explores design space for each product variant. First, the commonality optimization problem communicates commonality decisions to individual optimizations. Second, an individual optimization problem typically uses the commonality decision as additional constraints, to search for a design that optimizes product performance. Finally, the individual optimization problems communicate corresponding product performance back to the commonality optimization problem.

Typically, product family-based optimization approaches should be able to handle multiple commonality assessments concurrently, for example, potential commonalities for more that one component. According to our experience, these approaches also require less data post-processing.

In the current phase of the research, we have developed optimization formulations for both, all-in-one and decomposition, approaches. Results shown in Figure 9 were obtained by implementing this formulation on a single processor computer, which were found to significantly inadequate from computational perspective. In the future, we will implement both the formulations on a parallel computing facility, which is needed to solve the UTC product family problem.

4.4 Comparison of Methods: Our Experience

This investigation of three methods had a dual purpose: (1) design a UTC product family and (2) determine the extent to which designers would embrace/accept any of the proposed methods. As noted in Section 2, designers have been hesitant when it comes to commonality selection. Our tenet was that having tools to visualize (1) the product family data, and (2) tradeoffs from sharing common components, would help overcome these fears. Although we cannot present specifics of the product family due to its proprietary nature, we share our experiences and observations at a relatively high level. Table 2 summarizes our findings, and specific aspects of each method are discussed in the ensuing paragraphs. We note that the comparison of the three methods is based on our experience with UTC product family only. Although because of the limited scope of our study, we do not make any firm conclusions in the current paper, we expect that our findings should provide insight into the advantages and disadvantages of each method.

Method 1:

As shown in Table 2, Method 1 faired surprisingly well in every category, yet it did produce poor quality of designs (poor tradeoff resolution between different objectives). As such, the designs obtained from Method 1 were substantially inferior to those found using Method 2. Results indicate that the search strategy – exhaustive sampling – used in Method 1, failed to explore the product family design space sufficiently. Increasing the number of samples is an intuitive approach to improve design space exploration. Unfortunately, such an increase is ad-hoc, and may not always ensure improved design space exploration. Consequently, Method 1 was less attractive compared to Method 2.

Table 2: Comparison of Methods 1, 2, and 3*

	Method 1	Method 2	Method 3
Optimal Solution	Worst	Best	Worst
Ease of use	Best	Moderate	Worst
Interactive	Best	Best	Moderate
Suitable for implementation	Best	Moderate	Worst
Training required	Best	Moderate	Worst
Versatility to PF types	Moderate	Moderate	Best
Robustness	Best	Moderate	Moderate

* *Reporting current state, work is still in progress*

Method 2:

According to our experience, Method 2 was found to be the most attractive of all three, which was unexpected. As shown in Table 2, Method 2 was found to be the best method (and the only so far) for obtaining designs with superior tradeoff resolution (all objectives show simultaneous improvement compared to other methods), which is the underlying premise of this research. Because of the visualization–aided commonality selection, Method 2 is very interactive, which proved to be its biggest strength. However, Method 2 does require (1) availability of optimization tool and (2) a formal training to the designers in the use of such tools. On the positive side, for our application, Method 2 does not require specialized computational facilities (e.,g parallel computing). Interestingly, once the optimization was complete, designers were able to identify promising commonality options in a relatively short time (few minutes) using the visualization tools.

As we recall, Method 2 uses NSGA-II as an optimization algorithm. Typically, NSGA-II requires customizing some of its parameters, such as population size and number of generations. For the UTC product family design problem, significant convergence infeasibility required a population size of 300 and at least 25 generations to generate a uniform Pareto frontier for all product variants. Since the simulation models were not computationally expensive, we could execute them overnight, providing new data in the morning. More computationally intensive simulations will require other strategies (e.g., approximations, surrogate models) when using individual product optimization as advocated in Method 2.

In this paper, we demonstrated Method 2 on a three objective problem (objective 1, objective 2, and commonality). At times, the product family may involve more than three objectives, in which case more visualization windows than Figure 11 will have to be investigated simultaneously. Typically, the decision making complexity increases rapidly with the number of objectives in any optimization problem, and so will in the case of Method 2. However, the visualization assisted interactive decision making aspect of Method 2 is expected to lower the complexity of decision making in multi-objective design space. Extending Method 2 to problems with multiple objectives in the future would be important to understand decision making complexities associated with this method.

Overall, Method 2 was found to be the most suitable because it is (1) simple yet effective for commonality selection, (2) capable of finding product family designs, and (3) is suitable for implementing in an environment where designers have limited knowledge of formal optimization techniques.

Method 3:

Method 3 is the most computationally intensive method of all three based on our experiences to date. Significant increase in computational expenses can be attributed to the following factors.

The convergence infeasibility of the entire product family is nearly 100% for the all-in-one approach, as all of the products in the family are handled simultaneously in a single optimization problem. Thus, for this UTC problem, Method 3 needs substantially larger population and generation sizes. To date, population sizes of 500 running for more than 200 generations have failed to yield an optimal solution or a solution with better tradeoff resolution than other methods. It is important to note that the computational cost of Method 3 is a magnitude higher than that of Method 2. In the case of the decomposition strategy, it involves solving 11 individual optimization problems in a single iteration of commonality optimization. The computational cost of decomposition appears to be even higher than all-in-one approach. Also, problem formulation and algorithm settings require specialized training, which does not bode well for user adoption.

Pareto Band:

An interesting off-shoot of Method 2 has been the identification of a new concept, which we refer to as the Pareto Band (see Figure 12). In particular, when reviewing the product families resulting from Method 2, we observed that the product family solutions typically lie in a “band” around individual Pareto frontiers, and typically not on the frontiers. The width of this band indicates the tradeoff between commonality and performance – the larger the band, the larger the tradeoff as shown in Figure 12. The concept is similar to the *design bandwidth* idea advocated by Claesson and Berglund [25] However, it works in reverse in that it is driven by the range of solutions that is obtainable in the objective space versus the range of bandwidth one has in the design space.

We are continuing to investigate this finding in more detail to understand its implications better, with the most notable being a potentially new objective for product family optimization, namely, targeting a width of this band to balance the tradeoff between commonality in the family and the individual product performance. As such, it is our tenet that the Pareto Band approach may find its application in other fields as well, such as robust optimization of a single product.

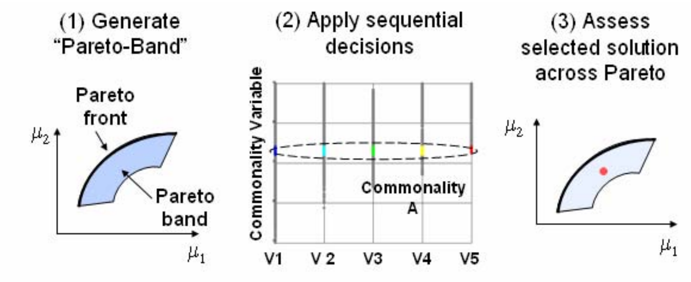


Figure 12 Pareto Band Approach

5 IMPLICATIONS OF COMMONALITY SELECTION PROCESS

After reviewing all three methods for the UTC product family design problem, the individual product optimization + visualization method appears to be the most suitable one for adoption with UTC's business units. The details of the commonality decision making process is shown in Figure 13 and elaborated as follows.

Step 1: Build individual product models using appropriate simulation tools.

Step 2: Integrate each model with an optimization tool.

Step 3: Conduct individual product multiple-objective optimization. In this work NSGA-II is chosen as the optimization algorithm.

Step 4: Post process results from individual optimization such that the final data file is ready for interactive commonality selection.

Step 5: Apply data filtering and visualization to make commonality selection.

Step 6: Generate alternative concepts based on the desired level of commonality and performance.

Step 7: Compare performance of the new concepts against current baseline designs.

The designers typically need to iterate between Steps 5 and 7 until they are satisfied with the commonality selection and the resulting product family.

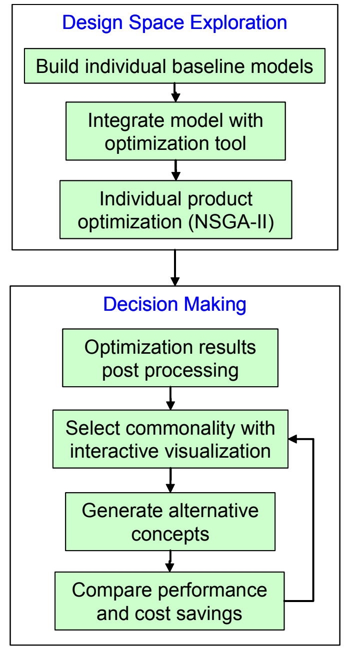


Figure 13 Commonality Decision Making Process

By working closely with designers and applying the commonality selection standard work for the UTC product family design, we have identified three new product concepts. The current baseline design uses four commonality variable values, while the new product concepts have either two or three commonality variable values, as shown in Figure 14. In Figure 14, Ci represents i^{th} coil and Ci+Ci indicates that the i^{th} coil has been used twice.

Varients	Concept 1 (2 components)	Concept 2 (2 components)	Concept 3 (2 components)	Baseline (4 components)
1	C1	C3	C3	C5
2				
3				C4
4		C4	C4	
5				
6	C2	C3 + C3	C3 + C3	C1
7	C1 + C1			
8				
9	C2 + C2	C4 + C4	C2 + C2	C2
10				
11				

Figure 14 Commonality concepts Selected Using Method 2

The cost savings per year of the new product concepts relative to the current baseline designs are shown in Figure 15. Please note that the cost savings are only from material cost savings, and further cost savings are expected from reduction of design time, qualification tests, supplier volume discount, inventory management, etc.

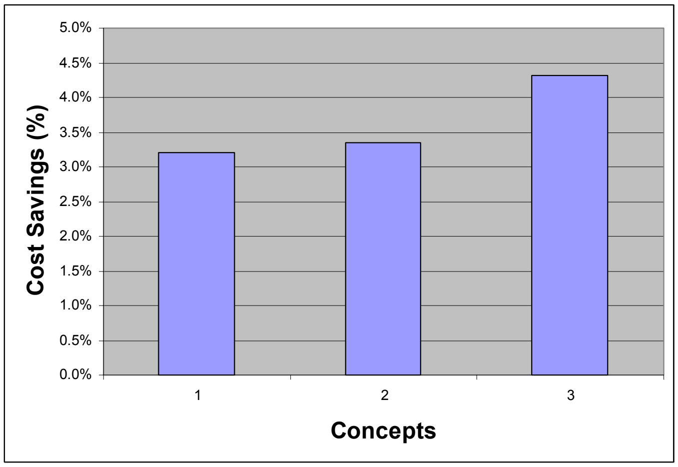


Figure 15 Estimated Yearly Cost Savings for UTC Product Family

6 CONCLUDING REMARKS

In this paper, we presented the product family optimization research conducted at the United Technologies Research Center, and its application to a UTC product family. Recognizing the challenges associated with product family design with regard to design space exploration and commonality selection, this paper exploits combined optimization-visualization strategies to facilitate product family design. Three different methods are compared and contrasted, and recommendations are given to support UTC's current product family design.

This paper recognizes gaps occurring in many engineering design practices, i.e., the designer's experience and adoption of design methods. Many times designers rely solely on their own domain knowledge to make design decisions, rather than seeking help from optimization or other advanced design methods. One of the rationales for such gaps is that designers typically do not have visibility inside the result generation process of black-box optimization methods. Without the capability to view solutions and visualize the tradeoffs, they are often leery of trusting such results.

The individual product optimization and interactive visualization method (Method 2) proposed in this paper attempts to bridge this gap by giving designers freedom to interactively make commonality selection and, perhaps more importantly, visualize its effect on individual product performance. As such, the method attempts to visually present the effect of designers' commonality selection on two key entities: (1) the gain from commonality and (2) performance losses incurred in individual products.

To realize complete benefits of the individual product optimization and visualization method, the following improvements are warranted in the future: (1) the visualization tool, which requires substantial data possessing at present, needs to be customized for visualizing product family design space; (2) at times, the absolute benefits and corresponding performance losses of commonality decisions are not obvious until the designer post-process the concepts. Development of additional subroutines is required that will eliminate the need for post-processing outside the framework of method 2; and (3) the interactivity only happens after the design spaces for all the products have been explored by optimization techniques. It would be worthwhile to explore the impact of providing the designers access to optimization assisted design exploration process.

We expect that, by gaining such access, the designer might be able to guide the design space exploration process towards regions of interest. However, on a cautious note, such an approach will (1) put additional burden on the designer, and (2) not expose the designer to the entire design space prior to decision making, and could potentially constrain the exploration in a narrow design space. As such, the current ATSV offers visual steering capabilities to let the user guide the exploration process for a single product [24]. However, further development work is needed to enable above mentioned interactivity for the entire product family. Potentially, such interactivity during individual optimization can also be integrated within the decomposition-based approach, which could potentially line-up with our goal of developing completely interactive product family optimization approaches.

Future work also includes developing robust platform optimization approaches that account for system variability, e.g., uncertain operating conditions, such that the product family designs are less sensitive to these variations. Development of robust platform optimization method would require combining robust design methods with platform optimization approaches.

ACKNOWLEDGEMENTS

Dr. Simpson acknowledges support from the National Foundation under NSF Grant No. CMMI-0620948. Any opinions, findings, and conclusions or recommendations presented in this paper are those of the authors and do not necessarily reflect the views of the NSF.

REFERENCES

[1] Simpson, T. W., Siddique, Z. and Jiao, J., Eds., 2005, *Product Platform and Product Family Design: Methods and Applications*, New York, Springer.

[2] Simpson, T. W., Marion, T. J., de Weck, O., Holtta-Otto, K., Kokkolaras, M. and Shooter, S. B., 2006, "Platform-Based Design and Development: Current Trends and Needs in Industry", *ASME Design Engineering Technical Conferences - Design Automation Conference*, Philadelphia, PA, ASME, Paper No. DETC2006/DAC-99229.

[3] de Weck, O., 2005, "Determining Product Platform Extent", *Product Platform and Product Family Design: Methods and Applications*, Simpson, T. W., Siddique, Z. and Jiao, J., New York, Springer, 241-301.

[4] Collier, D. A., 1981, "The Measurement and Operating Benefits of Component Part Commonality," *Decision Sciences*, 12(1), 85-96.

[5] Robertson, D. and Ulrich, K., 1998, "Planning Product Platforms," *Sloan Management Review*, 39(4), 19-31.

[6] Miller, S., 1999, VW Sows Confusion With Common Pattern for Models --- Investors Worry Profits May Suffer As Lines Compete, Wall Street Journal. New York, A.25.

[7] Lutz, R. A., 1998, *Guts: The Seven Laws of Business that Made Chrysler the World's Hottest Car Company*, New York, John Wiley.

[8] Simpson, T. W., 2005, "Methods for Optimizing Product Platforms and Product Families: Overview and Classification", *Product Platform and Product Family Design: Methods and Applications*, Simpson, T. W., Siddique, Z. and Jiao, J., New York, Springer, 133-156.

[9] Khajavirad, A., Michalek, J. and Simpson, T. W., 2007, "A Decomposed Genetic Algorithm for Solving the Joint Product Family Optimization Problem", *3rd AIAA Multidisciplinary Design Optimization Specialist Conference*, Honolulu, HI, AIAA, AIAA-2007-1876.

[10] Valliyappan, S. and Simpson, T. W., 2006, "Visualization Strategies to Support Product Family Design Optimization", *11th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Portsmouth, VA, AIAA, AIAA-2006-6949.

[11] Balling, R., 1999, "Design by Shopping: A New Paradigm?" *Proceedings of the Third World Congress of Structural and Multidisciplinary Optimization (WCSMO-3)*, Buffalo, NY, University at Buffalo, 295-297

[12] Stump, G., Yukish, M., Simpson, T. W. and Harris, E. N., 2003, "Design Space Visualization and Its Application to a Design by Shopping Paradigm", *ASME Design Engineering Technical Conferences - Design Automation Conference*, Chicago, IL, ASME, Paper No. DETC2003/DAC-48785.

[13] Winer, E. H. and Bloebaum, C. L., 2001, "Visual Design Steering for Optimization Solution Improvement," *Structural Optimization*, 22(3), 219-229.

[14] Eddy, J. and Lewis, K., 2002, "Multidimensional Design Visualization in Multiobjective Optimization", *9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Atlanta, GA, AIAA, AIAA-2002-5621.

[15] Khajavirad, A. and Michalek, J., 2007, "A Single-Stage Gradient-Based Approach for Solving the Joint Product Family Platform Selection and Design Problem Using Decomposition", *ASME Design Engineering Technical Conferences - Design Automation Conference*, Las Vegas, NV, ASME, Paper No. DETC2007/DAC-35611.

[16] Deb, K., 2001, *Multi-Objective Optimization Using Evolutionary Algorithms*, John Wiley & Sons, New York.

[17] Thevenot, H. J. and Simpson, T. W., 2006, "Commonality Indices for Product Family Design: A Detailed Comparison," *Journal of Engineering Design*, 17(2), 99-119.

[18] Khajavirad, A. and Michalek, J., 2007, "An Extension of the Commonality Index for Product Family Optimization", *ASME Design Engineering Technical Conferences - Design Automation Conference*, Las Vegas, NV, ASME, Paper No. DETC2007/DAC-35605.

[19] Martin, M. V. and Ishii, K., 1997, "Design for Variety: Development of Complexity Indices and Design Charts", *Advances in Design Automation*, Sacramento, CA, ASME, Paper No. DETC97/DFM-4359.

[20] Koch, P. N., Evans, J. P. and Powell, D., 2002, "Interdigitation for Effective Design Space Exploration using iSIGHT," *Structural and Multidisciplinary Optimization*, 23(2), 111-126.

[21] Stump, G., Yukish, M. and Simpson, T. W., 2004, "The Advanced Trade Space Visualizer: An Engineering Decision-Making Tool", *10th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, Albany, NY, AIAA, AIAA-2004-4568.

[22] Stump, G., Yukish, M., Simpson, T. W., Harris, E. N. and O'Hara, J. J., 2004, "Trade Space Exploration of Satellite Datasets Using a Design by Shopping Paradigm", *IEEE Aerospace Conference*, Big Sky, MT, IEEE, IEEE-1039-04.

[23] Yukish, M., Stump, G. M. and Lego, S., 2007, "Visual Steering and Trade Space Exploration", *2007 IEEE Aerospace Conference*, Big Sky, MT, IEEE, IEEE-2007-1188.

[24] Stump, G., Lego, S., Yukish, M., Simpson, T. W. and Donndelinger, J. A., 2007, "Visual Steering Commands for Trade Space Exploration: User-Guided Sampling with Example", *ASME Design Engineering Technical Conferences - Design Automation Conference*, Las Vegas, NV, ASME, DETC2007/DAC-34684.

[25] Claesson, A. and Berglund, F., 2005, "Design Bandwidth", *2005 Innovations in Product Development Conference - Product Families and Platforms: From Strategic Innovation to Implementation*, Cambridge, MA