SYM-AM-19-086

PROCEEDINGS
OF THE
SIXTEENTH ANNUAL ACQUISITION RESEARCH SYMPOSIUM

THURSDAY SESSIONS
VOLUME II

Acquisition Research:
Creating Synergy for Informed Change

May 8–9, 2019

Published: April 30, 2019

Prepared for the Naval Postgraduate School, Monterey, CA 93943.

Implementing Set-Based Design in DoD Acquisitions

Norbert Doerry—is the Technical Director of the NAVSEA Technology Office (SEA 05T). In addition to leading special projects, Dr. Doerry facilitates the transition of technology from industry and academia into naval warships. He retired from active duty in the U.S. Navy as a Captain with 26 years of commissioned service. He currently is focused on developing Medium Voltage DC (MVDC) Integrated Power and Energy Systems (IPES) for future warships, institutionalizing Set-Based Design in the U.S. Navy, and facilitating the introduction of flexibility and modularity features in future U.S. warships. [norbert.doerry@navy.mil and doerry@aol.com]

Philip Koenig—is the Director of the NAVSEA Industrial and Economic Analysis Division (SEA 05C1). His current work focuses on projections of cost, capability, and capacity in the U.S. shipbuilding industrial base. His previous experience was in the Future Ship and Force Architecture Concepts Division (SEA 05D1), the Office of Naval Research, NSWC Carderock, Chevron Shipping Company, and Vickers Shipbuilding Group Ltd. He teaches shipbuilding courses at the University of British Columbia and at MIT. [philip.koenig@navy.mil and pckoenig@mit.edu]

Abstract

Set-based design (SBD) is a technical and managerial approach that is increasingly being used to improve quality and responsiveness in U.S. naval ship design projects. It was employed on the Ship-to-Shore connector, the Amphibious Combat Vehicle (ACV), and the Small Surface Combat Task Force, and is being applied in ongoing surface combatant and submarine design studies. In contrast to iterative point-based design approaches, SBD projects arrive at a design solution by systematically eliminating regions of the design space rather than by selecting a solution early and iterating it through a design spiral to make it work. This paper reviews the fundamentals of SBD and discusses implementation strategies to reduce technical, schedule, and market risk; accelerate design convergence; enable distributed design teams; and improve cost estimates. We discuss how SBD enables early identification and resolution of knowledge gaps, enabling quicker design progress. The role of SBD in organizational learning and the ability to re-use knowledge products across acquisition programs is highlighted.

Introduction

In the design of many types of complex engineering systems, requirements and technical attributes are subject to considerable uncertainty. In this environment, organizing and managing the design workflow and decision making process to ensure that the optimal design is produced is difficult. In the past, complex design projects have been run in a point-design-based paradigm, but that approach has some weaknesses. Set-based design (SBD) is a comparatively new method that has gained traction in recent years in naval ship early-stage design. It has been applied to ship-to-shore connectors (Mebane et al., 2011), amphibious combat vehicles (Burrow et al., 2014; Doerry et al., 2014), surface combatants (Garner et al., 2015), submarines (Parker et al., 2017), and other programs.

The SBD method is conventionally described as a process of generation and elimination. First, a range of possible design solutions is generated. Each is described in terms of a set of design variables. The ranges of each variable are combined to define an *n*-dimensional design space. Through a process of elimination, infeasible or highly dominated regions of the design space are discarded and the design space becomes more restricted.[1] Design decisions are deferred to the latest possible point in the project schedule, thus keeping the maximum extent of the design space available for consideration until the latest possible moment.

There are pitfalls that arise in applying this method; the way certain details are handled can determine the success (or otherwise) of the design outcome. For example, delaying decisions confers no intrinsic benefit of its own; value is created only when such a delay is designed to generate lead time to gain specific types of additional information needed to make a better decision. Otherwise, delay is merely procrastination, which reduces focus and dissipates momentum.

SBD Fundamentals

The SBD concept dates back to Toyota's approach to automotive design as described in benchmarking studies done in the 1990s (Ward et al., 1995a; Ward et al., 1995b). Sobek, Ward, and Liker (1999) set forth the general principles as follows:

1. Map the design space.
    a. Define feasible regions.
    b. Explore trade-offs by designing multiple alternatives.
    c. Communicate sets of possibilities.
2. Integrate by intersection.
    a. Look for intersections of feasible sets.
    b. Impose minimum constraint.
    c. Seek conceptual robustness.
3. Establish feasibility before commitment.
    a. Narrow sets gradually while increasing detail.
    b. Stay within sets once committed.
    c. Control by managing uncertainty at process gates.

Singer et al. (2017), working in naval ship design, characterized SBD as follows:

1. Communicating broad sets of design values,
2. Developing sets of design solutions,
3. Evaluating sets of design solutions by multiple domains of expertise,
4. Delaying design decisions to eliminate regions of the design space until adequate information is known, and
5. Documenting the rationale for eliminating a region of the design space.

Starting with a characterization of the design space that is large enough to ensure with high probability the inclusion of the best solution of a design problem, SBD systematically eliminates infeasible and highly dominated regions of that design space. SBD thus arrives at a design solution largely through a process of elimination. A region of the design space is infeasible if there is a high confidence that a solution to the design problem

[1] In the design space, feasible solutions are points (or regions encompassing many points) that satisfy the criteria of all design domains (disciplines) (e.g., hydrostatics, speed, range, military effectiveness, cost). Highly dominated regions of the design space are those in which there is another region that is superior by every metric.

does not exist within the region. A region is highly dominated if the key metrics of interest in another feasible region are all better, even when considering uncertainty.

The process of eliminating a region of the design space is called a set reduction. Early on, set reduction is generally accomplished by determining that a region is not feasible. While determining that a region of a design space is feasible requires a considerable amount of information because every domain (sometimes called a design discipline) must evaluate with high confidence that the region is feasible, determining that a region is not feasible only requires one domain to conclude with confidence that the region is not feasible. In this way, an SBD design process can proceed cumulatively as each domain adds new knowledge. Thus, in traditional point-based design (PBD) methods that concentrate on evaluation of the feasibility of a design concept, the activities of the many domains must be coordinated; the schedule is impacted by the slowest domain. The asynchronous nature of SBD relaxes the need for tight coordination among the domains, reducing the dependency of the project schedule on any one domain.

In SBD, focus is placed on identifying key knowledge gaps, conducting experiments and analyses to resolve the knowledge gaps, and deferring associated design decisions until the knowledge gap has been resolved. As described by Cloft, Kennedy, and Kennedy (2018), this was the method employed by the Wright brothers to beat all others in becoming the first to achieve heavier-than-air flight with a relatively small budget. The Wright brothers identified three knowledge gaps:

- "the construction of the sustaining wings"
- "the generation and application of the power required to drive the machine through the air"
- "the balancing and steering of the machine after it is actually in flight"

To close the gaps, the Wright brothers systematically performed experiments to understand the impact of different design options on each of the knowledge gaps. They constructed a wind tunnel to test hundreds of different wings and produce trade-off curves in a short time. Their newly gained understanding of wings enabled them to design an efficient propeller which in turn reduced the power required from the engine. Cloft, Kennedy, and Kennedy (2018) cite the Wright brothers' approach to engineering design, based on an organized approach to the obtaining and application of knowledge, as an early example of effective SBD.

Figure 1 illustrates an example of a process for set-reduction. Initially, the entire integrated design space is considered feasible because none of the regions have been shown to be not feasible. Domains 1 and 2 begin work to create new knowledge to determine what parts of the design space are feasible (green), not feasible or highly dominated (red), or uncertain (yellow) from the domain's perspective. Domain 2 illustrates a good practice of starting with low fidelity analysis that can quickly and inexpensively categorize much of the design space as feasible, not feasible, or highly dominated, but still leaves a considerable amount of the design space uncertain. Follow-on higher fidelity work, which takes longer and is more expensive, can concentrate on the uncertain region. As each Domain completes its analysis, its results are incorporated into the integrated design space as part of a set reduction. Note that since Domain 3 started after set reductions had taken place, it need not consider regions of the design space that had already been eliminated.

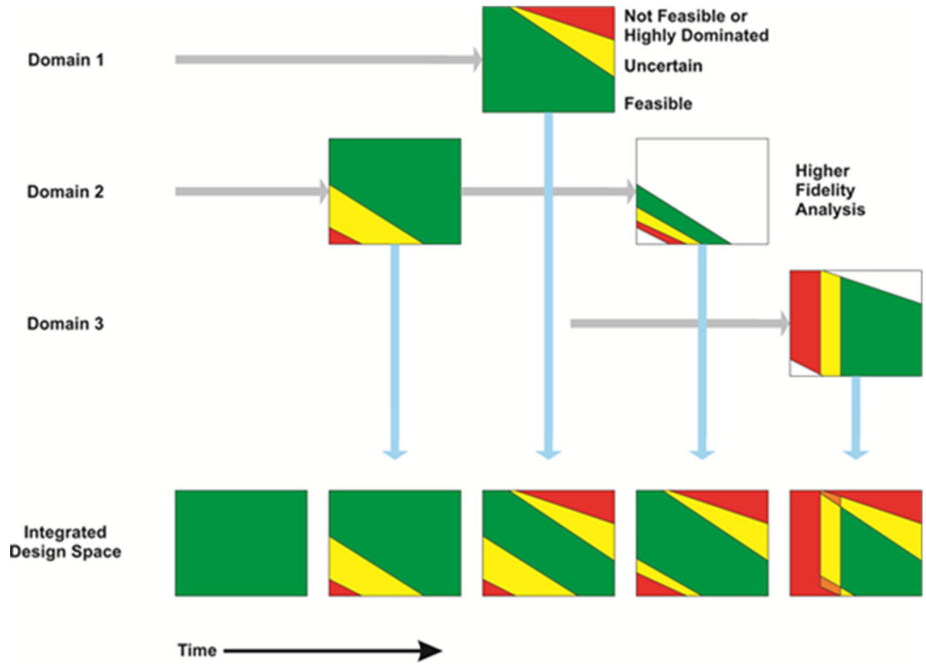

Figure 1. Representative Implementation of Set Reduction
(Singer et al., 2017)

The uncertainty of the analytic processes and test procedures should be well understood and considered in deciding to characterize a region of a design space feasible, not feasible or highly dominated. The goal is that for a given domain, no new information would result in a feasible region being considered not feasible or highly dominated, or a region not feasible or highly dominated being considered feasible. The uncertainty of the analytic process and test procedure results should be used to determine the boundaries of the remaining uncertain region from the perspective of the domain.

Figure 1 can also be used to distinguish between feasibility and viability. The green area in the integrated design space denotes feasibility. A region of the design space is feasible if analysis or testing to date has not shown that region to be uncertain or not feasible. Input from new domains could result in additional regions of the design space becoming uncertain or not feasible. A region is viable if all future analyses and testing (including verification testing) show that configurations exist that meet all requirements.

During design, while the feasibility of any one configuration can be determined based on analysis and testing performed to date, the viability of the configuration cannot be determined with confidence because the complete set of analysis and testing will not have been performed. If, however, the set of feasible configurations that correspond to a design space region are different enough from one other such that the probability that all of the configurations currently evaluated as feasible prove not to be viable is very small, then we can conclude that a viable configuration exists in that feasible design space region. Identifying the configurations within a feasible design space region that are viable or not feasible becomes the objective of future work.

Cost Estimating

Extending ship concept design cost estimating processes to the SBD environment is a work in progress. In SBD, the design variables defining the integrated design space do not always completely define a configuration; they are generally those that have a strong interaction between two or more domains. Design parameters that do not have a strong interaction, but are only an influence within a single domain, are typically treated independently by the domain teams. Hence a single point in the *n*-dimensional design space (with specified design variables) may reflect a large number of configurations corresponding to the multitude of combinations of individual domain design parameters that can be mapped to that single point.

Since cost estimating is one of the design domains, cost drivers should be design variables. However, practical difficulties arise due to the many design parameters, not all of whose cost implications are well enough understood to be incorporated into cost estimating relationships. See Cooper and Koenig (2018) for a discussion of this issue. Furthermore, there are some cost variables whose implications are not yet adequately built into the ship design solution generation process. An example of the latter would be industrial base capacity utilization, which is a very sensitive driver of naval ship cost. Work remains to be done to develop methods for incorporating that (and other) cost drivers into the design set generation process.

Figure 2 depicts an integrated design space consisting of a set of configurations intended to meet a specific set of requirements. The y axis is associated with one design variable with a hard constraint that separates feasible points (blue) from points that are not feasible (red). If all the feasible configurations were to prove viable, the best cost to assign to this point in the integrated design space would be the least expensive point (blue point furthest to the left); this configuration achieves the stated goals at the lowest cost. However, this feasible configuration may not prove viable once additional analysis and testing is conducted. Hence this cost is a lower bound with considerable cost risk. A higher cost estimate for a point in the integrated design space is associated with more configurations with a cost estimate equal to or below the higher cost estimate. For some cost above the lower bound, the probability will likely be low that all of the feasible configurations with a cost estimate below the specified cost are shown to be not viable. The lowest cost where this condition is met should be used as the cost estimate for that particular point in the integrated design space. Doerry (2015) details a method based on a diversity metric for determining this cost point.

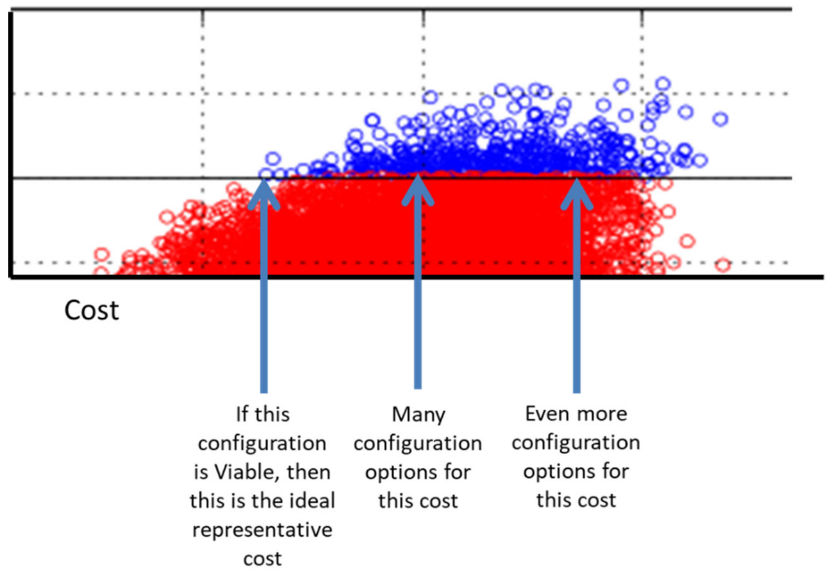

Figure 2. Cost Estimating in SBD

This process evaluates the cost for a configuration meeting a specific set of requirements based on the set of individual configuration cost estimates and not on the cost estimate of a particular configuration. If all the feasible configurations share one or more common failure modes, and the particular set of requirements associated with the design space is of great interest, then work and analysis should be performed to resolve whether the failure modes are failures or not.

The difference between estimating the cost of a single configuration and the cost associated with a group of possible configurations can be illustrated with an options analysis. SBD inherently incorporates the concept of an option. An option is the right or ability to do something in the future for a specified cost, but not the obligation to do so. The cost of acquiring an option is compared to the potential value it will bring in the future when more information is available to make a decision.

As an example, consider a project with designs for two configurations, one which includes widget A and one with widget B. Widget A costs $1,000 and is certain to work. Widget B costs $300, but there is only a 70% chance that it will work as planned. If it does not, there will be an estimated $2,000 of rework. The cost estimate for the configuration with Widget B may incorporate the $300 to account for Widget B and assume the change-order pool will be sufficient to cover the possibility that Widget B does not work. Alternately the cost estimate could include the rework:

$$C_B = 0.70 \times \$300 + (1 - 0.70) \times (\$2{,}000 + \$300) = \$900 \quad (1)$$

where the minimum cost would be $300 and the maximum $2,300. If, however, preserving the option to install Widget A or B is incorporated at a cost of $100, the cost estimate would be

$$C_{A\ or\ B} = 100 + 0.70 \times \$300 + (1 - 0.70) \times \$1{,}000 = \$610 \quad (2)$$

where the minimum cost would be $400 and the maximum $1,100. Without the option, Widget B would likely be selected because its expected cost of $900 is less than the expected cost of $1,000 for Widget A. Preserving the option to use either Widget A or Widget B reduces the down-side risk as compared to option B alone ($1,100 instead of

$2,300) as well as the expected cost ($610 instead of $900 for Widget B alone or $1,000 for Widget A alone).

This inherent incorporation of options within SBD is one of its strengths. Expected costs can be reduced at the same time schedule delays due to rework can be avoided. Incorporating Widget B without incorporating the probability of rework would make it a program risk. Incorporating the option to use Widget A or Widget B effectively transforms the risk associated with Widget B into an opportunity. The value of this opportunity can be incorporated into the cost estimate.

Implementation Strategies

While the basic concepts of SBD are known, how to implement SBD for a particular design problem is not always clear. Key issues include

1. Defining the design problem
2. Organization of the design team
3. Specifying the design domains or disciplines
4. Identifying the variables that define the design space
5. Setting the initial boundaries of the design space
6. Establishing feasibility metrics
7. Establishing dominance metrics
8. Determining the types of analyses needed and scheduling them
9. Making a design choice once the design space has been narrowed to that which is feasible from the perspective of all domains

Defining the Design Problem

A design problem should be defined specifically enough to enable the design team to focus its efforts, but not so specific to require redefinition, as knowledge is gained during design activities. Many times, a set of requirements is provided by the customer, but these requirements may not be firm. Typical reasons for requirements not being finalized include (Singer et al., 2017) the following:

a. Some known requirements may require study to determine appropriate values or measures.
b. Some specified requirements may be relaxed once the cost impact is fully understood.
c. The need for some requirements may not be known because of a lack of understanding of the design space.
d. The need for some requirements may not be known because of evolving exogenous factors.

One of the first tasks of a design team should be to clearly define the initial set of requirements and characterize the uncertainty of these requirements. The uncertainty of requirements can be evaluated as part of a requirements risk review (Singer et al., 2017).

Where a requirement has uncertainty, it should be bounded within a range as part of the requirements risk review. For these requirements, the work and timeframe necessary to establish the threshold requirement should be defined. The system design must be affordably flexible to handle the range of requirement values until the requirement is finalized. Note that the requirement may never be finalized or may change over the product's service life, in which case a modularity or flexibility based approach towards meeting the requirement may be required.

Organization of the Design Team

Figure 3 shows one way of organizing a design team for accomplishing SBD in product development (such as preliminary design for a ship). The stakeholder board includes those with vested interests in the product's technical characteristics, schedule, cost and contribution to an overall portfolio of products. Often, the stakeholder board approves major set reductions and if necessary, selects the final configuration from the remaining feasible design space. One of the values of SBD is in helping the stakeholder board understand the design space and gain an organizational consensus on the way forward. Because SBD starts with broad boundaries for the design space and systematically eliminates regions of the design space based on evidence, the impact of late "did you consider X, Y, or Z?" questions is minimal because the answer will generally be "Yes, we considered X, Y, and Z and eliminated them for the following reasons …" In a traditional PBD, the design team either expends additional (and probably unplanned for) resources to address X, Y, and Z, or risks the political consequences of ignoring the interests of a stakeholder.

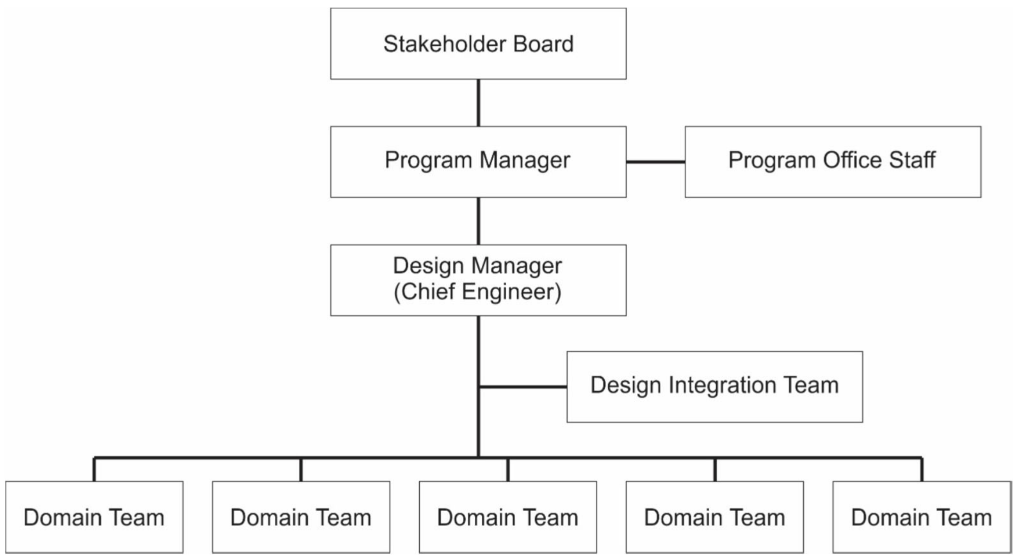

Figure 3. Design Team Organization
(Singer et al., 2017)

It is not unusual for stakeholders to have their own favorite solution (sometimes called a pet rock) prior to the start of the design effort. Ideally these pet rocks fall within the initial design space; hence, if they are eliminated, they are eliminated based on solid data and on consensus of the overall stakeholder board. In some cases, when presented the data, stakeholders will themselves advocate for the set reduction that eliminates their own pet rock. This is in contrast to a traditional PBD, which rarely includes all of the stakeholder's pet rocks in its initial set of configurations. Even if a pet rock is included and then eliminated because another configuration is evaluated as "optimal," the pet rock owner may not be satisfied because of a disagreement in the formulation of the optimization utility function. Note that SBD does not require the formulation or use of a utility function.

The responsibilities of a program manager and program office staff do not change whether the design is conducted using SBD or point-based methods. In many programs, the program manager concentrates on external interfaces such as the stakeholder board, Congress, and Department of Defense (DoD) organizations as well as program management activities such as contract management and financial management.

The director of the overall design effort is known as the chief program engineer or design manager. The design manager, supported by the design integration team, develops the overall plan for conducting SBD, provides tasking to each of the domain teams, coordinates domain team activities, presents major set-reductions to the stakeholder board to concur with set-reductions, documents set-reductions, manages the requirements, and manages the integrated design space. The design manager is also responsible for the integration and production of the specifications for the following detail design and construction contract. These specifications describe either the final solution or remaining design space.

Each domain, or design discipline, will have its own domain team. The exact number and definition of the domains and domain teams will vary somewhat project to project. Ideally the majority of the members of any one domain team would be co-located, but the collection of design teams need not be co-located. In some cases, it may be beneficial for one or more members of a domain team to also be a member of the design integration team to facilitate overall communication and coordination.

Specifying the Design Domains or Disciplines

Domains, or design disciplines, are typically defined based on the structure of the design organization. In many design organizations, design team members are provided by functional organization to form a large project team. In other design organizations, a small centralized design integration team assigns design tasks to the functional organizations; the functional organizations may not provide dedicated team members. SBD can function in both design organization constructs.

Another consideration for determining the boundaries of a domain is the ability of the domain to work independently and in parallel with the other domains. A design structure matrix (Eppinger & Browning, 2012) may prove useful for capturing the relationships between proposed domains and determining the degree of coupling among them. Ideally, a domain would require few key design variables to analyze.

When SBD is implemented for the design of a product, the domains should include not only those that define the product, but also those that evaluate the product. For preliminary ship designs (Technology Maturation and Risk Reduction), the definition domains typically are aligned with the traditional design disciplines:

- Hull
- Propulsion
- Electric plant
- Auxiliary systems
- Habitability
- Communications systems
- Weapon systems and combat system
- Aviation
- Arrangements
- Topside design

The evaluation domains typically are defined for assessments that have strong dependencies on multiple definition domains. Assessments that are strongly dependent on a single definition domain are typically accomplished by the definition domain. Typical preliminary ship design evaluation domains include the following:

- Weight management
- Signatures
- Producibility
- Cost
- Survivability
- Operational effectiveness
- Reliability, maintainability, and availability
- Human systems integration and manpower assessments
- Environmental, safety, and occupational health compliance
- Requirements management/traceability

For a ship concept study, conducted as part of the Material Solution Analysis, there may be only one “Ship Design” definition domain, and the evaluation domains could be limited to cost, survivability, operational effectiveness, and requirements management. Because many evaluation domains are not considered, a concept study should not result in a point design, but rather a design space which can be further reduced during preliminary design. The design space should be diverse in that it includes a variety of design approaches and/or features such that the likelihood that the un-evaluated domains will render the entire design space not feasible is small. Doerry (2015) provides methods for calculating diversity metrics for a design space. In SBD, Requirements Management should include tracking the uncertainty of requirements over time.

Identifying the Variables That Define the Design Space

While there are many thousands of design decisions that must be made to fully define a complex product, many of these decisions have impact entirely within one definition domain. On the other hand, some design decisions have significant ramifications across multiple definition and evaluation domains. The design variables associated with these significant cross-domain impacts should be used to define the overall product design space. The impact of design variables with small cross-domain impacts should be captured in uncertainty analysis; evaluation domains should consider the range of these small impact design variables when establishing the region of the design space categorized as uncertain.

In some domains, it may prove advantageous to apply SBD recursively within the boundaries of the domain.

For many products, the design manager and the leaders of each domain will collectively have sufficient insight to identify the design variables to use for defining the product design space. A design structure matrix (Eppinger & Browning, 2012) may prove useful for capturing the relationships among the domains.

Setting the Initial Boundaries of the Design Space

The initial range of values for design variables should be broad enough so that the resulting design space includes the global optimal solution to the design problem. Of course, if one doesn’t know which combination of design variable values results in the global optimal solution, then it is hard to have confidence that any restricted range will encompass the optimum. The way out of this dilemma is to take advantage of constraints, requirements, and the expertise of the domains.

For many acquisitions, the constraints with the greatest impact on design space boundaries are time, cost, and technical maturity. For example, immature technologies, such as those with a low Technology Readiness Level (TRL; Office of the Assistant Secretary of Defense for Research and Engineering [OASD(R&E)], 2011) that cannot realistically transition to mature products in time to support the acquisition can be safely eliminated. The consideration and elimination of these technologies should be documented.

One way to identify the boundaries is to start with the high priority requirements such as Key Performance Parameters (KPPs), Key System Attributes (KSAs), and Additional Performance Attributes (APAs), as defined in DoD (2015). Next, have the domains use ideation methods to develop sets of approaches for achieving these high priority requirements. Shah, Vargas-Hernandez, and Smith (2003) list a number of ideation methods as well as provide metrics for evaluating the number and variety of alternative ways of meeting an objective. An initial assessment of feasibility may be useful to eliminate options that cannot meet constraints. Documentation of this initial set-reduction is key to enable rapid reassessment of an eliminated solution approach should a constraint be relaxed or new, unanticipated information is obtained.

Based on the combined sets of approaches from all the domains, each domain should be able to translate them into a proposed initial set of boundaries. For some domains, the approaches will impact derived requirements (such as electrical and cooling demand). These boundaries should incorporate uncertainty as evaluated by the domains.

Establishing Feasibility Metrics

Early on, many immature technologies and products can be eliminated if they clearly cannot support the acquisition schedule, even if moderate delays in the acquisition schedule are accommodated. The evaluation of immaturity should be based on conversations with the industrial base or other hard evidence. Assuming a product will not be available because it currently is not available may result in a premature set-reduction. If an emerging technology has substantial benefit but cannot meet current schedule constraints, this should be conveyed to the customer to determine if delaying the schedule is warranted, or whether modularity and flexibility features should be incorporated to enable technology insertion when it is ready.

Physics based modeling and simulation should be employed as much as possible. Singer et al. (2017) define a Feasibility Element to be the output of analysis expressed as one of three values:

1. Feasible: high confidence that the configuration is feasible with respect to the analysis
2. Uncertain: low confidence that the configuration is either feasible or not feasible
3. Not Feasible: high confidence that the configuration is not feasible

Initially, low fidelity modeling can be used by each domain to classify the design space into feasible (green), uncertain (yellow), and not feasible (red) regions from their perspective. The integration team combines the design space evaluations from the different domains to create an integrated design space based on the following rule set (Singer et al., 2017):

1. Feasible: All feasibility elements are feasible.
2. Uncertain: All feasibility elements are either feasible or uncertain, with at least one uncertain.
3. Not Feasible: At least one feasibility element is not feasible.

As the design progresses, compound integration risk can be captured by considering regions where more than “n” feasibility elements are uncertain as Not Feasible under the assumption that the likelihood that all of the uncertain feasibility elements will eventually prove feasible is low.

Using the three colors to indicate the feasibility assessment values helps considerably in visualizing the impact of set-reductions. As regions of the design space that are red are eliminated as part of a set-reduction, each of the domains can concentrate of the remaining regions within their domain design spaces that are evaluated as uncertain. In this way, higher fidelity modeling can be focused on the regions of uncertainty rather than over the entire design space.

As the design progresses and the design space is better understood, the uncertainty associated with constraints can be reduced based on discussions with the customer. These constraints will further restrict the feasible region of the integrated design space.

In some cases, the lack of time or resources may require assessment of feasibility values for a particular feasibility element to be made qualitatively based on expert input. Documenting the rationale for the expert assessment is critically important to developing a recovery strategy if the assessment later is determined to be incorrect. Where possible, the uncertain region should be explored with quantitative analysis, and the feasible and not feasible regions selectively verified through quantitative analysis.

Establishing Dominance Metrics

One of the advantages of delaying decisions in SBD is that one can identify and pick the lowest cost option for which one has confidence will work. In point-based methods, options are often selected early when both feasibility and cost are not known with any degree of certainty. Within SBD, as more is known of the cost and feasibility of options, certain options can be eliminated because although they will work, other solutions will with high probability also work and will also cost less. A set-reduction can therefore be made based on dominance if the set-reduction does not have a significant impact on either the risk of feasibility or on the projected cost.

Determining the Types of Analyses Needed and Scheduling Them

Early on, priority should be given to analyses that can quickly and inexpensively eliminate as much of the integrated design space as possible. Regions eliminated need not be analyzed by other domains, thereby reducing the amount of work required. For example, Garner et al. (2015) reported that logic and initial appraisals led to the quick elimination of nearly 96% of the initially defined design space. The remaining analyses could focus on the remaining 4%, confident that the “best solution” did not reside in the eliminated 96%.

If a possibility exists that a feasible design space does not exist at all, testing limiting conditions may be of great value to prevent costly analysis of a concept that is fatally flawed. For example, during the concept exploration of the Amphibious Combat Vehicle (ACV), Burrow et al. (2014) reported that a baseline study was conducted to see if an ACV could be devised that met less than acceptable performance requirements at a reasonable price. One of the purposes of this study was to ensure that it made sense to conduct the more detailed and expensive analysis. If the unacceptable performance was not feasible, or if its cost was excessive, then it didn’t make sense to continue the study. Any additional capability would cost more, and achieving feasibility would be more difficult. As it turned out, the unacceptable performance was feasible and not at an unreasonable cost; further analyses continued.

For many domains, there is great value in initially using fast, low fidelity, but well understood, models to screen the remaining design space with high confidence into feasible, not feasible, and uncertain regions. Higher fidelity modeling can then focus on resolving the remaining uncertain regions that have not been eliminated by other domains.

Some domains rely heavily on model testing. Ideally these model tests should not be used in a confirmation role as is typical in PBD, but rather to validate digital simulation models that are scalable across the remaining design space. The choice of parameters for the model should be based on maximizing contributions to model validation and not to ensure a point design meets program requirements. Maximizing learning usually has greater value than simple requirements verification. Furthermore, because the model parameters do not depend on the final outcome, their parameters can be chosen early, enabling the fabrication and testing of the models to occur earlier, thereby enabling earlier application of the testing insights gained.

Making a Design Choice Once the Design Space Has Been Narrowed to That Which Is Feasible From the Perspective of All Domains

The end game for SBD depends on the acquisition strategy and to some degree on the views of the stakeholder board. One possible outcome is a specification for the next acquisition stage that defines the remaining feasible design space. Industry is allowed to propose a configuration of their choosing that resides in the feasible design space. The government then selects the proposal using traditional source selection criteria.

Another possible outcome is to let the stakeholder board negotiate among itself to pick a single point or smaller set within the remaining feasible design space. This outcome recognizes that the optimal solution from typical utility functions may not be acceptable to enough stakeholders. In the end, it is enough for the stakeholders to form a consensus on what the single point or smaller set is, without having come to an agreement as to why the result should be chosen. Different stakeholders may support the same outcome for very different reasons. This outcome doesn't preclude using utility functions and traditional optimization techniques to help the stakeholder board better understand the remaining feasible design space.

Another outcome is to analyze the remaining risks and select a region of the feasible design space that is robust to the consequences of the risks being realized or not. Conduct additional analyses of this region, while at the same time conduct work to resolve the risks. As risks are resolved (i.e., determine that the consequence will or will not happen with certainty), adjust the boundaries of the selected region of the design space accordingly. In this way, the design progresses with a high degree of risk tolerance.

Organizational Learning

If one of the goals of a design endeavor is to minimize the cost and amount of time to complete the design, then a logical approach is to have conducted as much of the analyses as possible prior to the start of the design. If previous work enables an immediate set reduction, then convergence to a final solution can happen faster. While opportunistic applications of previous work should always be pursued, even more benefit can be obtained by instantiating formal organizational learning techniques. Companies such as Toyota which have implemented effective organizational learning have been able to reduce product development time even when the complexity of their products has increased (Cloft et al., 2018). These techniques can include the following:

1. *Document Set Reductions*. Since SBD calls for good documentation for set-reductions, if the generalized knowledge and the resulting rationale for set reduction from a previous study is still valid for a current study, then the set reduction can occur with little or no additional work. In this way, there is great value in making generalizable conclusions within a set-reduction and properly documenting the assumptions and conditions associated with these conclusions. This documentation must be accessible to future design teams.
2. *Conduct pre-studies to characterize the design space*. Often studies are conducted prior to the start of a design activity to develop point designs to understand the “art of the possible.” Unfortunately, the conclusions that can be drawn from these point designs are limited to the assumptions and tasking of the particular study, which can differ considerably from the current study. Studies of greater value provide more general insight that is intended to be applied to future studies instead of attempting to provide recommendations based on analysis of one or a few point designs. Historical examples of this generalized knowledge include the development of standardized series such as the Taylor Standard Series for hull resistance predictions (Gertler, 1954) and NACA wing section series for lift and drag predictions for foils (Abbott & Von Doenhoff, 1959). Other historical examples include the accumulation and publication of data such as the Hoerner (1965) manuscript on fluid-dynamic drag and the Hoerner and Borst (1985) manuscript on fluid-dynamic lift. Within the Naval Sea Systems Command, this type of knowledge is captured in design practices and criteria manuals (DPCs) which were previously called design data sheets (DDSs). The key is that these documents capture knowledge and insight rather than documenting a particular solution. Understanding the reasons for why potential solutions should be avoided is just as valuable (if not more so) than being presented with recommended solutions (where the recommendation may depend on many unstated assumptions). Tasking statements for pre-studies must emphasize the desired goal is generalized insight rather than point recommendations for a specific notional design. A process should exist for incorporating the knowledge gained from the pre-studies into the applicable DPCs or equivalent documents.
3. *Capture feedback from production and operations.* One of the challenges with documentation such as DPCs is keeping them up to date with lessons learned once the design has transitioned to the shipbuilder for detail design and construction. The value of capturing this critical information was recognized by Toyota. In 1995, Ward et al. reported that Toyota engineers would document in their lessons-learned books the positive and negative aspects of their designs once they transitioned to manufacturing. This insight enabled the designers to improve their future designs with respect to manufacturability without constant interactions with manufacturing engineers. Similarly, feedback from the operators should also be captured in lessons-learned documents.
4. *Capture knowledge in algorithms and data sets for design tools*. Automated design tools are very useful for systematically exploring a design space. These design tools must reflect in their algorithms acceptable design criteria and practices that necessarily evolve as technology advances and more is learned about a given discipline. Furthermore, most design tools require validated data sets to function. Since ship designs don’t occur frequently, the data associated

with each ship design should be captured for re-use on following ship designs. This process needs to be well thought out, resourced, and institutionalized.

5. *Ensure the design workforce is trained and understands the design space, design tools, and supporting data sets*. One way of accomplishing this has been proposed by Jons and Wynn (2009) as part of Continuing Collaborative Concept Formulation (C3F). As Jons and Wynn observed in 2009,

> Continuous concept formulation forges an effective ship design and warfare analysis community, shortens the time to respond to emerging requirements, and produces system cost estimates based on solid engineering. Collaboration enables rapid ways and means tradeoffs for a broad set of possible future environments.

Compared to point-based design (PBD) methods where a baseline concept is chosen early and modified over time, SBD promises to arrive at better designs quicker without a cost penalty. Singer et al. (2017) list the following benefits of SBD:

1. Rework is minimized because decisions are delayed until there is sufficient knowledge to make robust decisions. This is in contrast to other design methods where decisions are made early based on the best (but incomplete) information available at that time.
2. Decisions are made based on a good understanding of the overall design space, not just on the analysis of one or two options.
3. Decisions can be made on partial information. If one domain of expertise finds a region of the design space to be infeasible, that region is infeasible independent of what other domains discover.
4. The different domains of expertise can work semi-autonomously. This enables design teams that are geographically dispersed. Additionally, the overall schedule is less likely to slip if one domain of expertise is late.
5. New information, including changing requirements, can be more readily incorporated into the design process. Good documentation of set reduction decisions can quickly identify the impact (if any) of new information.
6. With the right organization, tools, and experienced workforce, the design process can be accomplished faster than traditional designs.
7. Because options are not selected until proved feasible, the end product should have less technical risk as compared to traditional designs.

Conclusions

If properly implemented, SBD can improve design decisions and the quality of designs in less time than conventional PBD. This paper described the basic method and how it can be applied to design problems. It highlighted a number of points that should be considered in planning and executing SBD. With the information provided, a design team should be able to successfully plan and execute an SBD based design process.

References

Abbott, I., & Von Doenhoff, A. (1959). *Theory of wing sections*. New York, NY: Dover.

Burrow, J., Doerry, N., Earnesty, M., Was, J., Myers, J., Banko, … Tafolla, T. (2014, October 23). *Concept exploration of the amphibious combat vehicle.* Presented at the SNAME Maritime Convention, Houston, TX.

Cloft, P., Kennedy, M., & Kennedy, B. (2018). *Success is assured: Satisfy your customers on time and on budget by optimizing decisions collaboratively using reusable visual models.* Productivity Press.

Cooper, D., & Koenig, P. (2018, November 14–15). Cost estimating concepts and approaches for modular adaptable ship designs. ASNE Design Sciences Series Symposium: Modular Adaptable Ships, Washington, DC.

Office of the Assistant Secretary of Defense for Research and Engineering (ASD[R&E]). (2011, April). *Technology readiness assessment (TRA) guidance*. Washington, DC: Author.

DoD. (2015, February 12). *Manual for the operation of the joint capabilities integration and development System (JCIDS)* (including errata as of 18 Dec 2015). Washington, DC: Author.

Doerry, N. (2015, March 4). *Measuring diversity in set-based design*. Presented at ASNE Day 2015, Arlington, VA.

Doerry, N., Earnesty, M., Weaver, C., Banko, J., Myers, J., Browne, D., Hopkins, M., & Balestrini, S. (2014, September 25). *Using set-based design in concept exploration*. Presented at the SNAME Chesapeake Section Technical Meeting, Army Navy Country Club, Arlington, VA.

Eppinger, S., & Browning, T. (2012). *Design structure matrix methods and applications*. Cambridge, MA: MIT Press.

Garner, M., Doerry, N., MacKenna, A., Pearce, F., Bassler, C., Hannapel, S., & McCauley, P. (2015, November 3–7). *Concept exploration methods for the small surface combatant*. Presented at the World Maritime Technology Conference 2015, Providence, RI.

Gertler, M. (1954, March). *A reanalysis of the original test data for the Taylor Standard Series* (Report 806). Washington, DC: Navy Department, The David W. Taylor Model Basin.

Hoerner, S. (1965). *Fluid-dynamic drag.* Published by the author.

Hoerner, S., & Borst, H. (1985). *Fluid-dynamic lift*. Published by Liselotte Hoerner.

Jons, O., & Wynn, S. (2009, April 8–9). *Continuing collaborative concept formulation (C3F): A proposal to improve ship acquisition and to invigorate naval engineering*. Presented at ASNE Day 2009, National Harbor, MD.

Mebane, W., Carlson, C., Dowd. C., Singer, D., & Buckley, M. (2011, September). Set-based design and the ship to shore connector. *Naval Engineers Journal*, *123*(3), 79–92.

Parker, M., Garner, M., Arcano, J., & Doerry, N. (2017, June 14–15) *Set-based requirements, technology, and design development for SSN(X)*. Presented at

Warship 2017: Naval Submarines & UUVs, The Royal Institution of Naval Architects, Bath, UK.

Shah, J., Vargas-Hernandez, N., & Smith, S. (2003, March). Metrics for measuring ideation effectiveness, *Design Studies*, *24*(2), 111–134.

Singer, D., Doerry, N., & Buckley, M. (2009). What is set-based design? *ASNE Naval Engineers Journal*, *121*(4), 31–43.

Singer, D., Strickland, J., Doerry, N., McKenney, T., & Whitcomb, C. (2017). *Set-based design*. SNAME T&R, 7–12.

Sobek, D., Ward, A., & Liker, J. (1999, Winter). Toyota's principles of set-based concurrent engineering. *Sloan Management Review,* 67–83.

Ward, A., Liker, J., Christiano, J., & Sobek, D. (1995a). The second Toyota paradox: How delaying decisions can make better cars faster. *Sloan Management Review*, *36*, 43–61.

Ward, A., Sobek, D., Christiano, J., & Liker, J. (1995b). Toyota, concurrent engineering, and set-based design. In J. K. Liker, J. E. Ettlie, & J. C. Campbell (Eds.), *Engineered in Japan: Japanese technology management practices* (pp. 192–216). New York, NY: Oxford University Press.

ACQUISITION RESEARCH PROGRAM
GRADUATE SCHOOL OF BUSINESS & PUBLIC POLICY
NAVAL POSTGRADUATE SCHOOL
555 DYER ROAD, INGERSOLL HALL
MONTEREY, CA 93943

www.acquisitionresearch.net