Evaluating Fragility of Interdependent Design Spaces to Quantify the Risk of Space Reduction Decisions in Set-Based Design

Joseph Van Houten
Doctoral Prospectus
University of Michigan, Ann Arbor, MI, USA
Department of Naval Architecture & Marine Engineering

Doctoral Committee:
Matthew Collette (Chair), Associate Professor, University of Michigan
David Singer, Associate Professor, University of Michigan
Seymour Spence, Associate Professor, University of Michigan
Austin Kana, Assistant Professor, Delft University of Technology

Abstract

The litany of decisions made over the course of any marine vessel design have significant impacts on the design's outcome. Iterative approaches, such as point-based design (PBD), make precise decisions on vessel characteristic values as necessitated by the sequence in which interdependent design activities are performed, but these decisions make PBD susceptible to inefficient rework cycles. To reduce the frequency and magnitude of rework, vessel characteristic assignments often coincide with previous designs of similar vessels, even if those designs are suboptimal. Contrarily, convergent approaches, such as set-based design (SBD), allow for more ambiguous definitions of vessel characteristics by making decisions that remove unfavorable values while keeping ranges of values open. With this approach, SBD depends less on experience to reduce variable sets because that experience is supplemented by various sampling and modeling tools unique to each discipline. However, each discipline cannot thoroughly sample their design space, and it is common for modeling and requirement changes to transpire over the course of a design. Design managers need to reduce design spaces to meet project timelines and budgets, but in doing so, they risk having unexplored areas or design changes reveal important information that would have impacted those reductions. Design managers currently lack any method that quantifies and regulates this risk. A novel way to do so may be through the creation of a framework and metrics rooted in Information Theory. Before developing a strategy to assess the risk of space reductions, this work explores the sequential decisions that make up a conventional PBD approach through a simple polynomial model. The formalization of designer influence and consequential rework path properties relating to decisions made earlier in a sequence prompts a shift in approaching design through an iterative method to a convergent SBD method. Background research is conducted on entropy metrics from Information Theory that can evaluate design space fragility, and the creation of a framework with these metrics is proposed to quantify the risk of space reductions. Plans are set to test the framework and metrics with a new SBD problem and to develop a multidisciplinary adaptive sampling approach that alleviates high-risk reductions.

1. Introduction

1.1. Motivation

Managing the unforeseen impacts of decisions is incredibly challenging in marine design due in large part to the interdependence of its design activities. Each discipline of a design team (i.e. stability, seakeeping, structures, etc.) not only has to make decisions on vessel characteristics that satisfy their own analyses, they must also coordinate those decisions such that they do not hinder the success of other disciplines. Moreover, the transfer of information between designers and design activities is a dynamic process [17], so designers generally bear the consequences of their decisions long after they are made. For an *iterative* point-based design (PBD) approach, those decisions may lead to inefficient rework cycles, making it commonplace for design teams to model new concepts after previous vessels [15]. This accumulation of experience dictating design variable decisions not only becomes an asset for PBD teams, it becomes a necessity. In stark contrast, set-based design (SBD) is a newer and widely studied *convergent* design approach that, in a sense, builds experience on its own. SBD encourages disciplines to maintain sets of design options that are eliminated during periods of collaboration instead of a single design that is left backtracking through analyses whenever hard-set, vessel characteristics fail [23]. Unfortunately, design managers lack a method to gauge the risk of those decisions in regards to the vulnerability of remaining designs when faced with new, possibly conflicting, information. This work starts by investigating if the consequences of iterative PBD design decisions can be reasonably abated, before shifting attention to determining how to quantify and plan for the risk accompanying convergent SBD decisions.

Aside from relying on experience, the negative impacts that past decisions have on later analyses for iterative approaches, like PBD, cannot be mitigated by simply making more conscientious decisions. Instead, those decisions can be made less arduous by carefully considering the order in which design activities are performed. When talking about his design spiral, Evans admits, “The radial lines of the diagram represent the salient considerations of the designer arranged, it is believed, in the logical order most conducive of rapid convergence on the ultimate, refined and balanced solution indicated by the inner closed circle” [7]. Evans is aware that the selected order of design activities (or design path) plays a role in design convergence, but he leans on experience and general intuition to dictate that order. More concrete strategies have developed to logically order complex design activities such as the program review evaluation technique (PERT), critical path method (CPM), precedence diagramming method (PDM), and Petri nets ([6], [17]). Although to date, the primary tool used is the Design Structure Matrix (DSM) because of its ability to capture coupled and iterative relationships between design activities [6]. As discussed in Steward’s 1981 paper [25], this tool identifies known interdependent relationships between design activities to order them such that the amount of needed feed-forward information is reduced. In other words, Steward ensures design activities receive as much of their input information as possible before beginning. Others, such as Eppinger et al. [6], have built off of Steward’s matrices with the belief that all design activities are not created equal; they suggest other factors to consider as part of the DSM process, like analysis execution time and variance of analysis results. In its simplest form, DSM will not fix PBD because the ideal order of design activities depends on more factors than just interdependent relationships. Determining which factors are most influential is an ongoing discussion.

When design teams lack the needed experience or find that design activities are far too complex and interconnected to optimally order them, they should consider adopting a convergent design approach. Convergent approaches give designers the opportunity to make more conscientious design commitments by exploring and comparing many designs. Gumina discusses many of these convergent options in great detail in his dissertation [10]. The Method of Controlled Convergence (MCC) considers several different design alternatives, ranks them against each other based on pre-established criteria, and gradually eliminates or merges designs until the single, superior option is down-selected. Multiple Objective Decision Making (MODM) techniques, such as multidisciplinary optimization (MDO), are similar to MCC, except they rely on computer codes and algorithmic feedback from tested points to guide convergence to a single point. Design of experiments (DOE) and Response Surface Methods (RSM) explore design spaces of many different inputs in a structured manner and then develop curve and surface approximations of tested points to guide a designer searching for optimality. While effective strategies for delaying premature decisions until variable trade-offs are explored, they sacrifice some of their robustness by focusing so heavily on exploiting optimal design spaces too soon. These approaches are at risk of committing to designs commonly found along the boundaries of perceived feasible regions, only to become encumbered by changes in requirements or analyses later. Doerry argues set-based designers can avoid this fate because they do not focus on locating optimal designs along the edges of feasible regions too early in the design process while their understanding of the problem is still growing. In his words, SBD instead "seeks to gain sufficient information to eliminate the highly infeasible and the highly dominated solutions" [4].

As discussed, SBD allows designers of multiple disciplines to operate independently of each other while forming sets of perceived preferences for different vessel characteristics. Ward et al. talk at length about SBD in what they call the Second Toyota Paradox [28]. In their report, they acknowledge the seeming inefficiencies that are expected to accompany delayed design decisions and communicated sets of preferred characteristics; however, they go on to discuss how it has actually become a contributing factor of Toyota's superiority in manufacturing efficiency. Variable sets are developed by testing out discrete points within each discipline's design space, and those sets are communicated to a design manager tasked with eliminating areas of their design spaces. SBD neither looks for a single, optimal design nor holds a designer limited to tested design points or "optimal" designs along feasible boundaries; SBD helps designers build their understanding by testing many points in their design space to remedy a lack of experience. When design spaces cannot be further reduced, designers may choose to increase the fidelity of their analyses so new information can be gathered for the continuation of the process.

At its core, SBD is a human-centered process. By allowing disciplines to communicate their preferences independently and then leaving reduction decisions in the hands of a design manager, the process is *supplemented* by multi-fidelity, discipline-specific programs but *driven* by knowledgeable designers. Designers present reduction requests and information supporting them to a design manager, and then the design manager has to integrate requests with other disciplines while judging if the supporting information warrants the reduction. Shields and Singer insist, "...solutions can only be eliminated if no new information would change the outcome" [20]. New information can describe feedback from unexplored areas of a design space as well as feedback following changes to analysis tools or design requirements. Exploring a design space to the point of absolute certainty that no new information will change the outcome of a reduction decision is unreasonable because design managers have to balance the responsibility of delaying reduction decisions with the obligation of meeting project timelines and budgets. Instead, design managers must tolerate a certain amount of risk in their reduction decisions. Shields and Singer expand on this risk, stating, "Elimination decisions create a low-risk knowledge structure. Only making decisions when the supporting knowledge is well-understood and is unlikely to change leaves stable knowledge to be further developed" [20]. Set-based design managers lack the means to both evaluate the risk of reduction and determine a suitable amount of risk worth enduring to keep up with project-based time and budget constraints.

This work is motivated by presenting design managers with a method for quantifying the risk associated with space reduction decisions. This method will introduce perturbations (that mirror reasonable design changes) to the remaining design space of a proposed reduction and then assess its fragility (or lack of robustness) in handling the new information that accompanies the perturbations. The fragility can then be contextualized against project-based constraints to postulate acceptable risk levels worth enduring at different stages of the design process. Before discussing the aspects of Information Theory that can be leveraged to formalize this process, work was carried out to weigh the impacts that past decisions have on later analyses of a conventional, PBD approach, and whether those negative impacts can be better managed. A simple polynomial model was used to realistically model interactions between complex marine design activities, eventually introducing the ideas of designer influence and consequential rework for consideration when effectively organizing analyses and the subsequent sequence of decisions. Results from the point-based, polynomial model simulations form the reasoning for transitioning to a set-based way of thinking. After SBD and its drawbacks pertaining to the fragility of space reduction decisions are explained in more detail, a new problem is introduced to promote the creation of a framework that addresses these drawbacks. This paper concludes with major contributions I will strive to make during my PhD and a tentative timeline for accomplishing them.

1.2. Research Scope

Set-based design managers do not currently have a way to assess the risk of desired design space reductions. For the purposes of this study, reduction risks will be relative to new information from design changes that would expose the fragility of present information that is used to rationalize a reduction decision. Design managers also lack context on how to handle high-risk reductions as well as an acceptable amount of risk worth sustaining with respect to impending timelines and budgets. By analyzing the informational content upon which disciplines form their space reduction *requests*, design managers can ensure their space reduction *decisions* are as likely as possible to remain valid regardless of impending design changes. Keeping these research gaps in mind, the following questions are being proposed to direct the remainder of my graduate studies:

1. How can design managers track uncertainty associated with the behavior of unexplored areas of interdependent design spaces when making a space reduction?

2. To what extent can design managers quantify reduction risk across multiple disciplines by comparing design space informational content before and after a proposed reduction?

3. How might design managers leverage *present* information to suggest where continued exploration would elicit the most beneficial *new* information to alleviate a high-risk space reduction?

4. What is an acceptable level of space reduction risk to tolerate relative to project-based time and budget constraints?

2. Simulating Design Paths

Prior to formulating those SBD-related research questions, a PBD model was created to replicate and observe the impacts that sequential decisions have on designers of successive analyses of a conventional marine design approach. The model consists of different polynomial equation sequences receiving numerical inputs and producing numerical outputs to better understand the significance of a chosen design path. Design paths were explored with polynomial equations to keep the study's focus on the paths themselves, rather than the intricacies of different marine design analyses, but these equations are intended to be analogous to actual marine design activities. The variables of the polynomial model mirror vessel characteristics and performance metrics, and the model's bounds reflect design requirements and industry standards. As the model and paths are analyzed in more detail, further analogies to marine design will become prevalent.

All of the variables, bounds, and equations are completely original and created for the purposes of this model. Those bounds and equations are defined as follows:

- $1 \leq x_1 < 5$
- $1 \leq x_2 < 5$
- $0 \leq x_3 < 8.5$
- $0.5 \leq x_4 < 6$
- $-2 \leq x_5 < 9$
- $1.5 \leq x_6 < 10$
- $-3.5 \leq x_7 < 4$
- $-3.5 \leq x_8 < 2.5$

- $-4 \leq x_9 < 3$
- $1 \leq x_{10} < 10$

Analysis 1:

$$x_1 + x_2 = x_3 \tag{2.1}$$

Analysis 2:

$$x_3 - x_4^2 = x_5 \tag{2.2}$$

$$2x_2 + x_4 = x_6 \tag{2.3}$$

Analysis 3:

$$x_5 + \sqrt{x_6} - x_7 = x_8 \tag{2.4}$$

Analysis 4:

$$x_8 - 2(x_9 + x_1^3) = x_{10} \tag{2.5}$$

The standard path (*Path 1*) is depicted in Figure 2.1. In this path, outputs of earlier analyses neatly flow into the inputs of the next analysis, and there is no situation where a variable acts as an input before being calculated as an output.

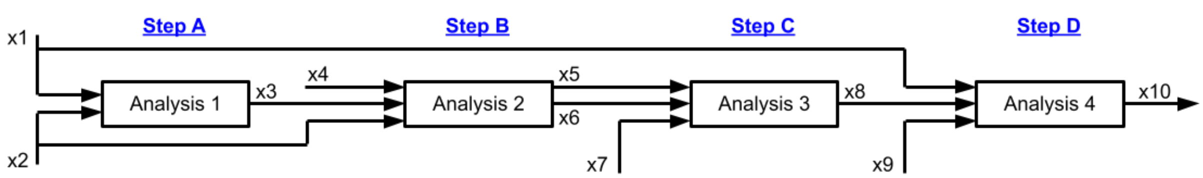

Figure 2.1: Sequence for *Path 1* of the polynomial model

The other three paths assessed have the following analysis sequences:

- *Path 2*: 2-3-1-4
- *Path 3*: 3-4-1-2

- *Path 4*: 4-1-2-3

These paths are discontinuous and represent instances where the outputs of earlier analyses do not seamlessly flow into the inputs of the next analysis. Figures for these paths will be provided but depend on the model's assumption of whether analyses can be solved bidirectionally, unidirectionally, and/or in combination.

Two sets of assumptions were utilized and explored in separate conference papers; the bidirectional and combined analysis assumptions were followed for the 6$^{\text{th}}$ International Conference on Maritime Technology and Engineering (MARTECH 2022) in Lisbon, Portugal [26], and the unidirectional and non-combined analysis assumptions were followed for the 15$^{\text{th}}$ International Symposium on Practical Design of Ships and Other Floating Structures (PRADS 2022) in Dubrovnik, Croatia [27]. The assumptions between the conferences build on each other in an attempt to gradually and wholly reflect sequential design activities observed in marine design. To keep the focus on managing *complex* design activities, only results from the unidirectional model will be discussed here.

2.1. Unidirectional Model

The polynomial model for PRADS 2022 restricted analyses to being solved in one direction (inputs-to-outputs) and not in combination. The assumptions complicated the polynomial model but made it reflective of complex design activities. For example, it is unreasonable to think that a finite element analysis (FEA) can be solved from strains and stresses to loads and material characteristics in the same way as it is normally solved from loads and material characteristics to strains and stresses. It is also unreasonable to expect an FEA analysis to be solved in combination with other unrelated design programs.

Figure 2.1 is representative of *Path 1* under these assumptions, while Figures 2.2 - 2.4 visualize *Paths 2*, *3*, and *4*. In these figures, a variable may be assigned an input value before its output value is calculated, as signified by the red variables. With the new unidirectional

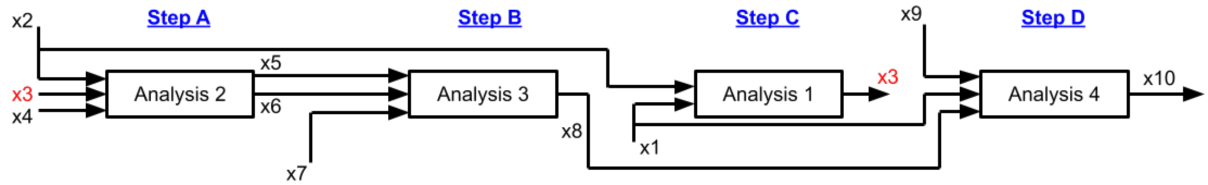

Figure 2.2: Unidirectional sequence for *Path 2* of the polynomial model

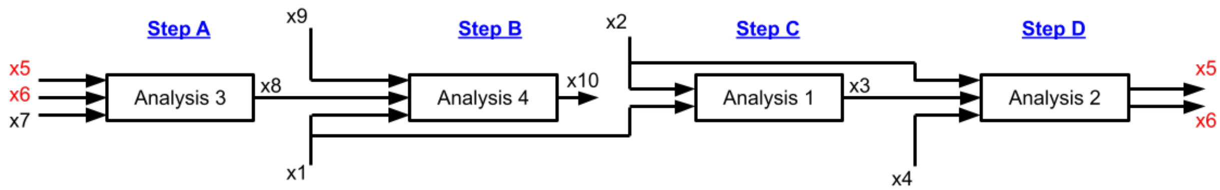

Figure 2.3: Unidirectional sequence for *Path 3* of the polynomial model

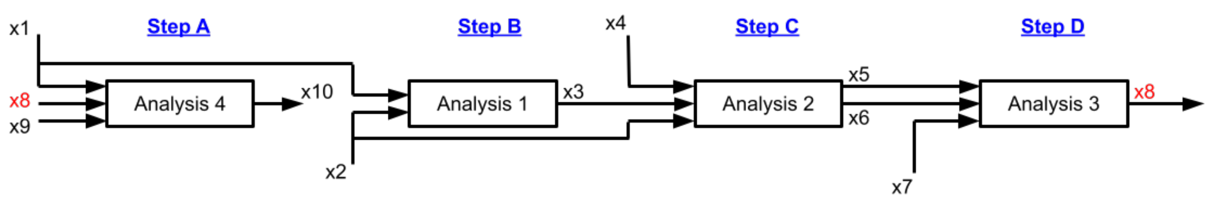

Figure 2.4: Unidirectional sequence for *Path 4* of the polynomial model

assumption, there is potential for these input and output values of the same x-variable to differ. In these instances, iteration is required to ensure those values match. Built-in rework is defined as iteration resulting from the logistics of the sequence (as described with the red variables), while consequential rework is defined as iteration resulting from any design variables not meeting their required bounds. These simulations only incorporate built-in rework because they are observational rather than success-driven.

The scale of built-in rework varies between the two simulation sets described in the next subsection. The two types considered are being called analysis loops and restart loops.

Analysis loops define rework that cycles back from the outputs to the inputs of the same analysis; in *Path 4* of the polynomial model, an analysis loop would consist of cycling back from the outputs to the inputs of *Analysis 3* because the calculated x_8 value differs from its previously assigned value in *Analysis 4*. A designer would perform an analysis loop in hopes of not having to interfere with work completed and variable assignments made earlier in the sequence. Restart loops define rework that cycles back from the outputs to the inputs at the start of the sequence; in *Path 4* of the polynomial model, a restart loop would consist of cycling back from the outputs of *Analysis 3* to the inputs of *Analysis 4* because the x_8 values remain in conflict with one another. A designer would perform a restart loop after having made several attempts at eliminating any variable conflicts with analysis loops and resorting to starting the sequence over with a better idea of more cooperative variable values. Analysis loops avoid revisiting previous steps in the sequence, making it a less foreboding type of rework than restart loops.

Just as *Path 1* is the only sequence without the potential for built-in rework, it is the only sequence where the designer has influence over at least one x-variable in each analysis. The extent of designer influence correlates to the number of variables that a designer is free to manipulate in each analysis; these variables are referred to as independent variables to differentiate them from input variables. Table 2.1 outlines the independent x-variables of each path. All independent variables are input variables, but not all input variables are independent variables. An example of an input variable that is not an independent variable is the x_7 variable of *Path 4*. In this path, x_8 is assigned an input value at the start of the sequence, but it is also calculated at the end of the sequence. If those two values do not match, then *Analysis 3* will undergo analysis loops where designers change the only input value unique to their analysis: x_7. So while designers can exercise flexibility in x_7's value assignment for other paths, its purpose in *Path 4* is to negate any x_8 value conflicts. The table is nearly equivalent to the one provided in the paper for PRADS 2022 aside from one small difference in *Step C* of *Path 3* [27]. For PRADS 2022, this step was reasoned as having no independent variables because *Step D* only has one variable (x_4) to negate potential conflicts for two output variables (x_5 and x_6). Since that paper, x_2 has been changed to an independent variable because the designer in *Path 3* technically has freedom to manipulate this value without revisiting it with an analysis loop; however, the designer's value assigned for x_2 might make it impossible for x_4 alone to rid any x_5 and x_6 value conflicts, leading to less preferred restart loops.

Table 2.1: Independent variables of each path

Path	Step A	Step B	Step C	Step D
1	x_1, x_2	x_4	x_7	x_9
2	x_2, x_3, x_4	x_7	None	x_9
3	x_5, x_6, x_7	x_1, x_9	x_2	None
4	x_1, x_8, x_9	x_2	x_4	None

Experience is also introduced to the simulations of the unidirectional model, but only after executing an initial set of inexperienced runs through each path. Instead of assigning each input variable a uniform random value within its required bounds, mean and standard deviation data collected from successful runs of the inexperienced simulations are used to assign a normal random value to each input variable. Modeling experience in this manner imitates a designer using a database of similar designs to guide the current design. Figure 2.5 visualizes the algorithm used to incorporate analysis loops, restart loops, and experience.

The following subsections provide more detail on the simulation setup and results of the unidirectional polynomial model. The results not only shed light on the implications of path selection on design success, they also expound on the concepts of designer influence and consequential rework affecting path selection.

2.1.1. *Simulation Setup*

A series of Monte Carlo simulations are used to see how likely each path's sequence is to meet the x-variable bounds. Monte Carlo simulations were chosen because they capture the randomness of possible independent variable value combinations across each path. For

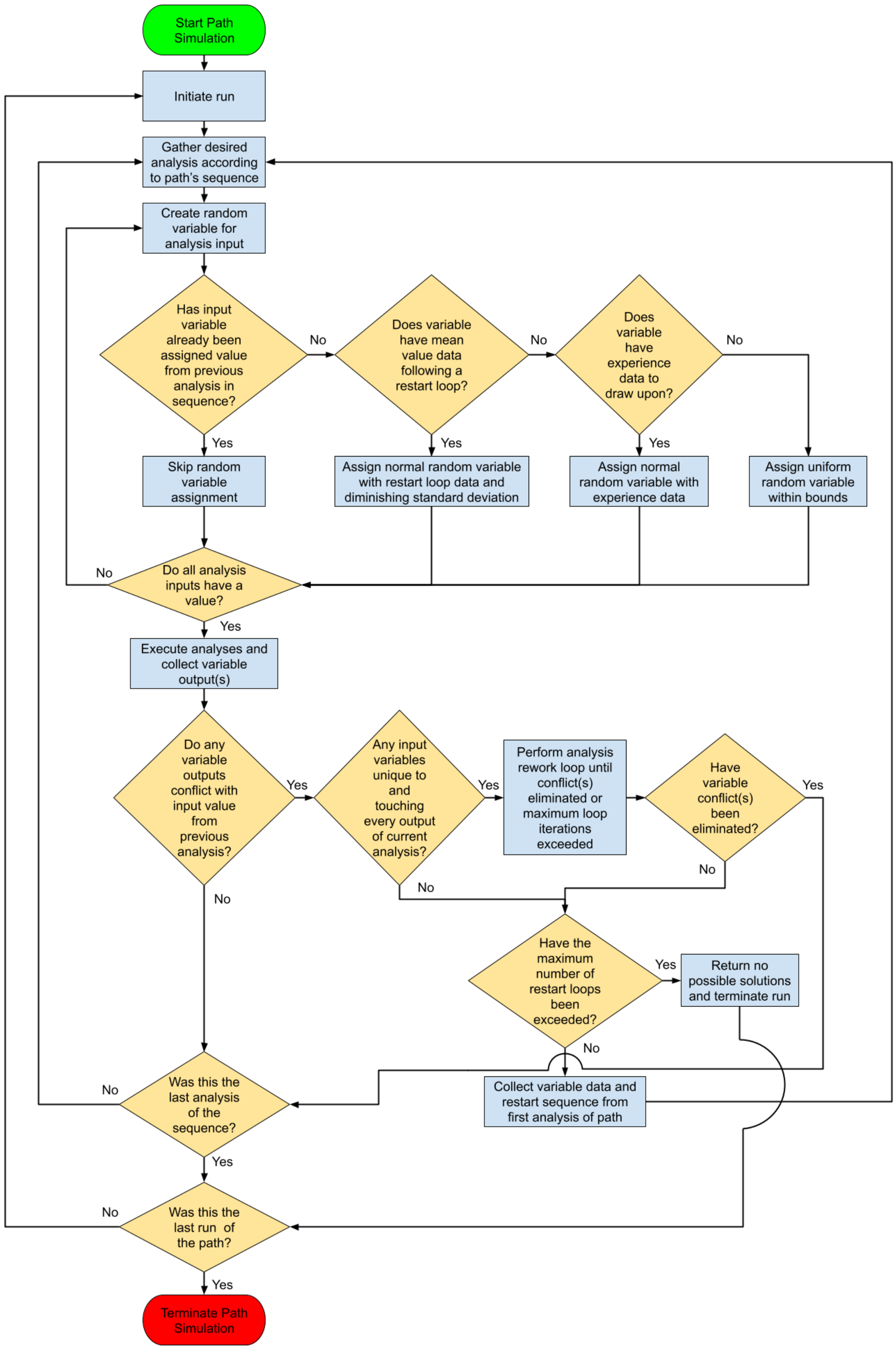

Figure 2.5: Unidirectional polynomial model algorithm incorporating analysis loops, restart loops, and experience

the first set of simulations, input variables are assigned a uniform random value within their required bounds while only including analysis loops. For the second set of simulations, a normal random value is assigned to input variables based on successful runs of the first set of simulations. Additionally, restart loops are initiated whenever analysis loops are unable to eliminate variable conflicts. Details on the specifics of the analysis and restart loops along with the normal random variable data can be found in the PRADS 2022 paper [27].

Python was used to execute a total of 100,000 Monte Carlo runs for the inexperienced path simulations and 10,000 Monte Carlo runs for the experienced path simulations. The 100,000 runs of the first simulation set were chosen to provide a sufficient amount of 10-variable success run data for the second simulation set. The results of both simulation sets adequately converged over the course of 24-36 hours. The results not only showcase variables individually and as a whole, they also present success on an analysis-by-analysis basis so that distinctions about each path can be made on the grounds of designer influence and consequential rework. Table 2.2 summarizes the comparisons between both simulation setups of the unidirectional polynomial model.

Table 2.2: Characteristics of simulation setups for the unidirectional polynomial model

Simulation Set 1	**Simulation Set 2**
100,000 runs (each path)	10,000 runs (each path)
Analysis loops	Analysis & restart loops
No experience	Experience

2.1.2. *Simulation Results*

Results were first assessed for each individual x-variable. Figure 2.6 displays the polynomial model variables on the horizontal access and tracks the percentage of runs for which each variable was successful across the four paths. One clear takeaway from these results is that *Paths 1, 2,* and *4* significantly outperform *Path 3*. The drop in success for *Path 3*'s variables is attributed to the last analysis in the sequence. Revisiting Figure 2.3, x_2 and x_3 establish their values in *Step C*, fixing two of the three input variables of *Step D*. *Step D* executes *Analysis 2*, which has two output variables (x_5 and x_6). The x_4 variable does affect both of the output variables, but alone, it is often not able to eliminate variable conflicts (within a relative tolerance) of both x_5 and x_6. When the simulations are left with unfixable variable conflicts, all variables of the run are labeled unsuccessful.

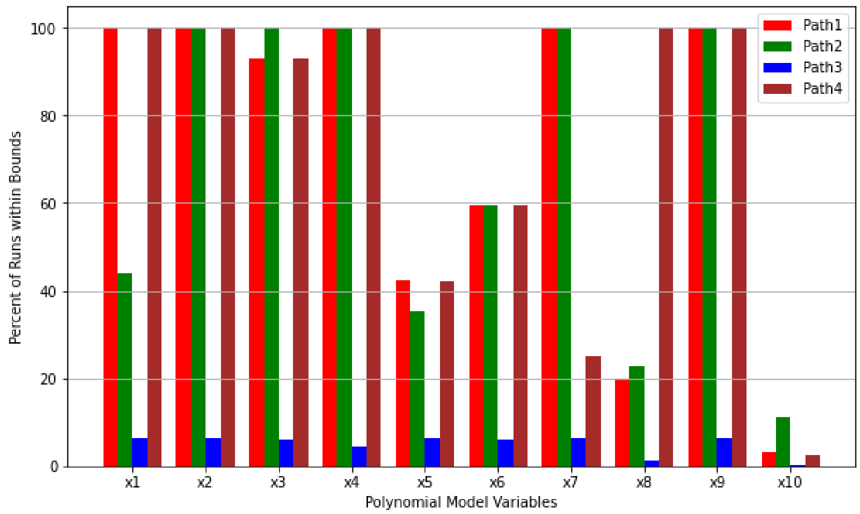

Figure 2.6: Individual variable success for unidirectional polynomial model without experience

Another observation is that certain variables see significant differences in variable success based on position in the sequence. Looking at the x_3 variable, *Path 1* sees just over 90% success, *Path 2* sees 100% success, *Path 3* sees under 10% success, and *Path 4* sees back to just over 90% success. Backtracking to Figures 2.1 - 2.4, the x_3 variable appears after *Step A* in *Path 1*, before *Step A* in *Path 2*, after *Step D* in *Path 3*, and after *Step B* in *Path 4*. For this variable, there is a trend that the earlier it appears in the sequence, the more likely it is to meet its required bounds. However, a variable appearing earlier in the sequence does not always result in more success. The x_{10} variable is calculated at the end of *Path 2* and the start of *Path 4*, yet it is more likely to be successful in *Path 2* than *Path 4*. Different sequences can lead to varying variable success, but the extent of that success is only dependent on the input variable assignments preceding the calculation of a variable.

Differing from the variable success results are the run success results shown in Figure 2.7. Rather than breaking up the results on a variable-by-variable basis, it depicts the percentage of Monte Carlo simulations (runs) having an allotted number of successful x-variables (0 - 10) across the four paths. For example, of the 100,000 simulated runs traversing *Path 2*, roughly 16% had five successful variables, 27% had six successful variables, and so on. For a design to be considered successful, all of the variables would need to meet their required bounds in the same run; however, as can be seen in Figure 2.7, not even 1% of the simulations meet this stipulation, regardless of path choice.

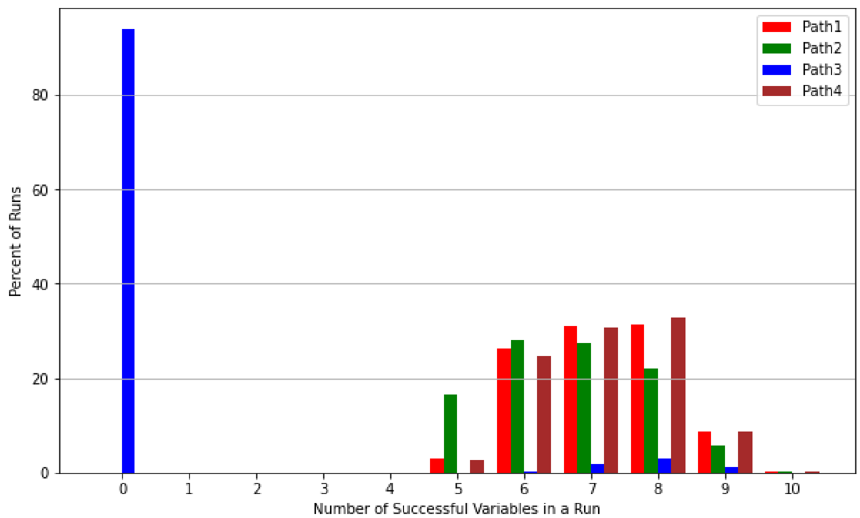

Figure 2.7: Complete run success for unidirectional polynomial model without experience

Path 1 and *Path 4* appear to be the most attractive based on their high levels of 7-, 8-, and 9-variable success. Further distinctions can be made between those two paths when observing their run success after each analysis with Table 2.3. This table shows how burdened designers later in a sequence become because of analysis order and variable assignments made earlier. Looking at *Path 1* and remembering its independent variables outlined in Table 2.1, *Step A*'s designer (lacking experience) is able to select x_1 and x_2 values that produce a successful x_3 value 92.9% of the time. Of those successful analysis runs, *Step B*'s designer is able to select an x_4 value that pairs with the already selected x_2 value and calculated x_3 value to produce successful x_5 and x_6 values 33.7% of the time. *Step C*'s and *Step D*'s results follow the same logic. These results show *Analysis 4* is very unlikely to be successful for the inexperienced designers' first pass regardless of the sequence and will usually require consequential rework. Across each path, *Analysis 4* never sees higher than a 4.3% success rate. This observation is consistent with the individual variable success results of Figure 2.6 as *Analysis 4* calculates the x_{10} variable which is largely unsuccessful across all of the paths.

Table 2.3: Probability of analysis success given preceding analyses success for inexperienced design

Path	Step A	Step B	Step C	Step D	Prob. of 10-var success
1	92.9%	33.7%	22.2%	4.3%	0.30%
2	28.6%	29.1%	39.6%	4.3%	0.14%
3	22.7%	3.4%	100%	5.4%	0.04%
4	2.4%	100%	27.5%	56.5%	0.37%

■ = *Analysis 1* | ■ = *Analysis 2* | ■ = *Analysis 3* | ■ = *Analysis 4*

The other analyses see significant fluctuations in their success rates depending on the path. Since nothing about the actual analyses changes between paths, these fluctuations must solely depend on the work completed before them. Looking specifically at *Analysis 1*, its success rate is 92.9% for *Path 1* and 100% for *Paths 3* and *4*. For *Path 2*, its success rate drops to 39.6%. Evidently, the x_2 and x_3 variable assignments resulting in successful outcomes for the first two steps of this path infrequently produce a passing x_1 value in *Step C*. Additionally, because a designer of *Path 2* does not have any influence over *Step C* (refer back to Table 2.1), the scale of the consequential rework for its failing runs must span back multiple analyses.

Furthermore, having at least some influence in a failing analysis does not guarantee that influence will even be useful. Figures 2.8 - 2.11 follow three unique Monte Carlo runs among a point cloud of other runs for *Path 1*. These three runs model three instances of design success up until the last analysis. In the last analysis of *Path 1*, the designer only has control over the x_9 variable to produce a passing x_{10} value; the x_1 and x_8 values have been established in prior analyses. The pink square represents a chosen x_9 value that produces a passing x_{10} value, the red circle represents a chosen x_9 value that does not produce a passing x_{10} value but has the potential to with consequential rework localized to the analysis, and the light blue triangle represents a chosen x_9 value that does not produce a passing x_{10} value and has no potential to pass without consequential rework cycling back multiple analyses to alter the x_1 and/or x_8 values. Most points in *Analysis 4* fall in line with the light blue triangle, meaning the designer's influence in *Analysis 4* of *Path 1* is rather useless.

In contrast to *Path 1*, *Path 4* moves *Analysis 4* all the way to the beginning of the sequence, giving the designer total influence over its input space. If the first designer of *Path 4* chooses an $\{x_1, x_8, x_9\}$-input combination that fails for x_{10}, the scale of the consequential rework will always stay localized to that analysis. Continuing to follow along with Table 2.3, the likelihood of success in every remaining step of the sequence outperforms that of *Path 1*. When *Path 4*'s design reaches *Step D*, the designer lacks influence over any of the inputs; however, based on the analyses and variable assignments made prior, that analysis will still have over a 50% chance of succeeding. When taking designer influence and consequential rework into consideration for Simulation Set 1, *Path 4* outperforms all paths.

While the setup of Simulation Set 1 is fairly reflective of complex marine design activities, it is still lacking in a couple of areas. In marine design, a designer often has access to personal experience or a database of similar designs from which to model a new design. A designer is also permitted to perform rework beyond the scope of their own analysis. With these two ideas in mind, Simulation Set 2 added experience and restart loops (larger-scale built-in rework) to the model. The variable and run success results of Simulation Set 2 are provided in Figures 2.12 and 2.13, respectively.

The variable success results see noticeable shifts in success with the additions of restart loops and experience. All variables in *Path 3* roughly have a 25-35% increase in success. For the other paths, variables that were mostly successful either maintain or see very slight drops in that success. Meanwhile, variables that were mostly unsuccessful see large increases in success. The x_{10} variable, which previously had the most difficult requirement range to satisfy, makes up significant ground on the other variables and is no longer the worst

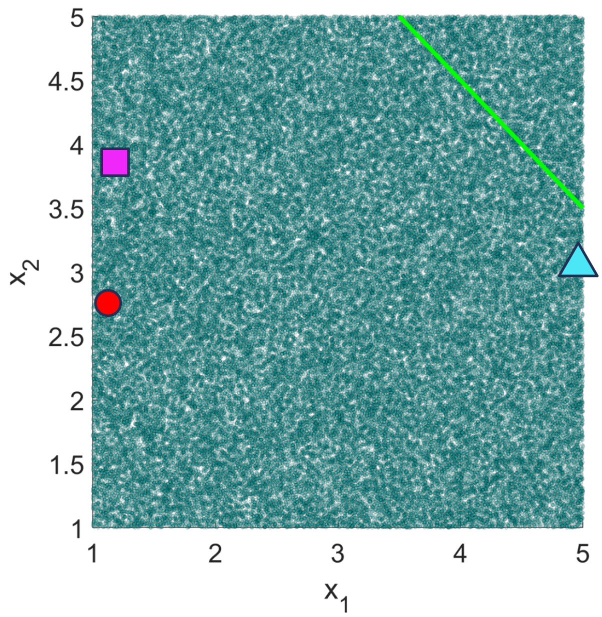

Figure 2.8: *Path 1* runs for *Analysis 1* without experience

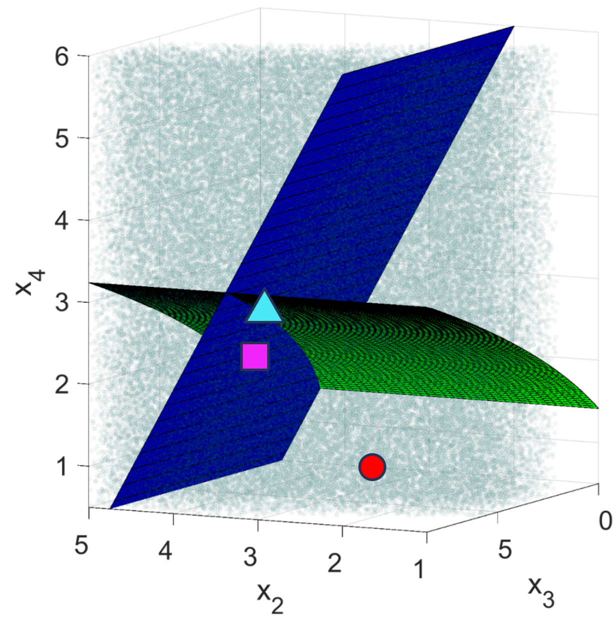

Figure 2.9: *Path 1* runs for *Analysis 2* without experience

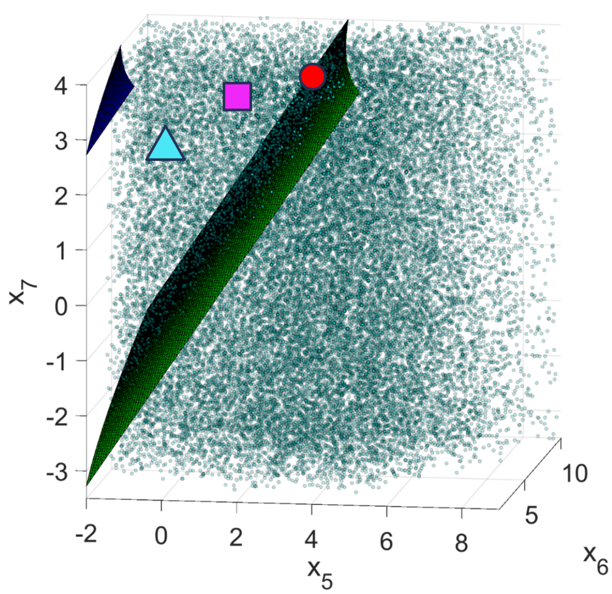

Figure 2.10: *Path 1* runs for *Analysis 3* without experience

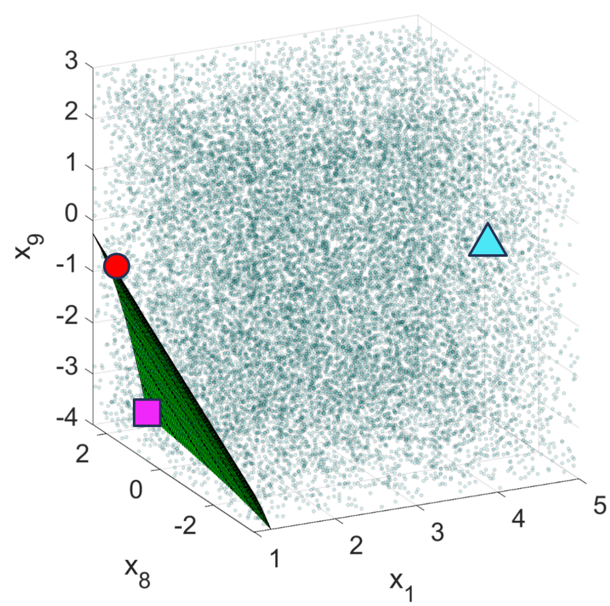

Figure 2.11: *Path 1* runs for *Analysis 4* without experience

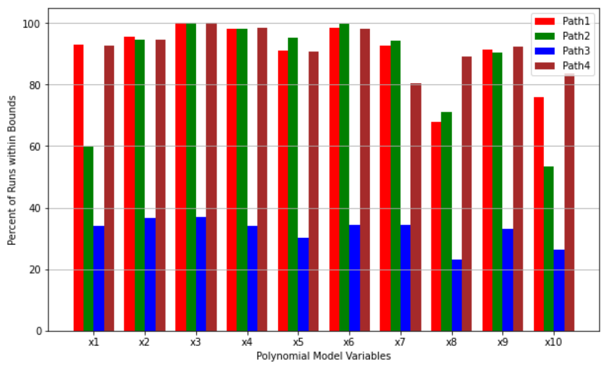

Figure 2.12: Individual variable success for unidirectional polynomial model with experience

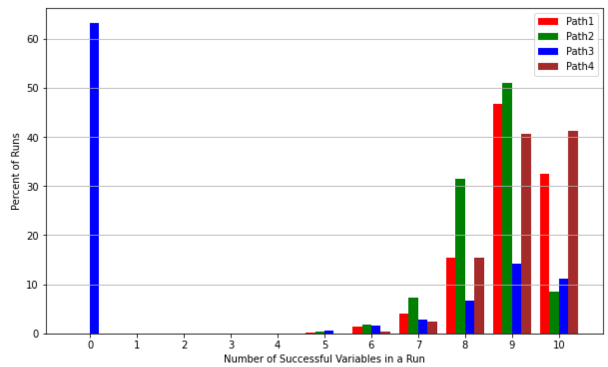

Figure 2.13: Complete run success for unidirectional polynomial model with experience

performer in *Paths 1*, *3*, and *4*.

The run success results similarly see increases in the number of successful variables in a run across every path. *Path 3*'s 0-variable success runs drop by roughly 30%, while the 7-or-less-variable success runs of *Paths 1*, *2*, and *4* all fall below 10%. Most notably, the 10-variable success runs increase dramatically from below 1% in each path to 8-11% for *Paths 2* & *3* and 32-41% for *Paths 1* & *4*. As the main design goal is still to achieve 10-variable success, *Path 1* and *Path 4* are, again, the more attractive options. The results of Table 2.4 further differentiate between these paths by allowing the impacts of designer influence and consequential rework to be considered.

Table 2.4: Probability of analysis success given preceding analyses success for experienced design

Path	Step A	Step B	Step C	Step D	Prob. of 10-var success
1	88.7%	89.2%	60.6%	67.7%	32.5%
2	88.3%	65.5%	54.8%	26.7%	8.5%
3	18.5%	64.2%	99.8%	93.4%	11.1%
4	60.3%	94.6%	88.5%	81.5%	41.1%

= *Analysis 1* | = *Analysis 2* | = *Analysis 3* | = *Analysis 4*

With experience and restart loops, the success rates of all analyses generally improve. In Simulation Set 1, *Paths 1*, *2*, and *3* are hindered by the last analysis of the sequence having a very low success rate (4.3%, 4.3%, and 5.4%) combined with having either no or very little influence over the last analysis; this combination leads to a high likelihood of requiring large-scale consequential rework. In Simulation Set 2, the success rates of the last analysis in these paths noticeably increase (67.7%, 26.7%, and 93.4%). More conscientious variable assignments made earlier in each sequence have led to lower likelihoods of requiring large-scale consequential rework.

Figures 2.14 - 2.17 show designers earlier in *Path 1*'s sequence being more thoughtful with their variable selections to benefit designers later on in the sequence. In Figure 2.14, instead of feeling like the entire design space below the green line can be filled, designers keep x_1 and x_2 assignments further away from their upper bounds knowing it will benefit designers working with these shared variables in *Analysis 2* and *Analysis 4*. The designers of *Analysis 4* still only have influence over x_9, but now that the x_1 values are concentrated much more towards the lower end of its accepted range, this influence is much more effectual. An improper x_9 variable assignment can more than likely be remedied with a consequential rework loop that does not need to revisit earlier analyses in the sequence.

With experience and restart loops added to the simulations, the best path becomes less clear. *Path 4* still outshines the rest in the percentage of 10-variable success runs, but there are other factors to consider. *Path 1*, for example, maintains designer influence in each analysis, lacks built-in rework, and benefits from thoughtful variable selections early on to reduce the chance of requiring large-scale consequential rework. The purpose behind running these simulations with the polynomial model was less intent on identifying an optimal path and more intent on understanding how impactful path selection and the ensuing order of variable decisions is on a design's success. These findings pose the question if it is worth developing new path selection strategies, or if a more holistic approach to design should be pursued altogether.

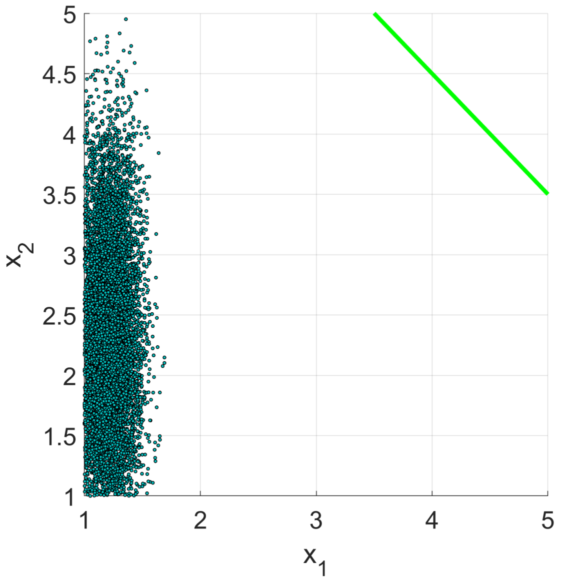

Figure 2.14: *Path 1* runs for *Analysis 1* with experience

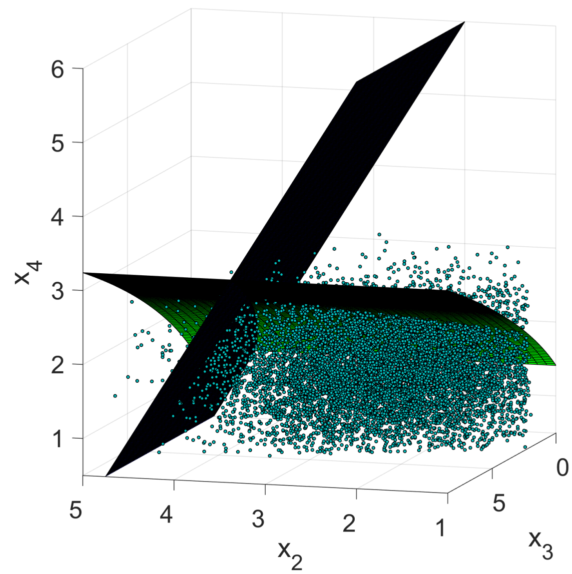

Figure 2.15: *Path 1* runs for *Analysis 2* with experience

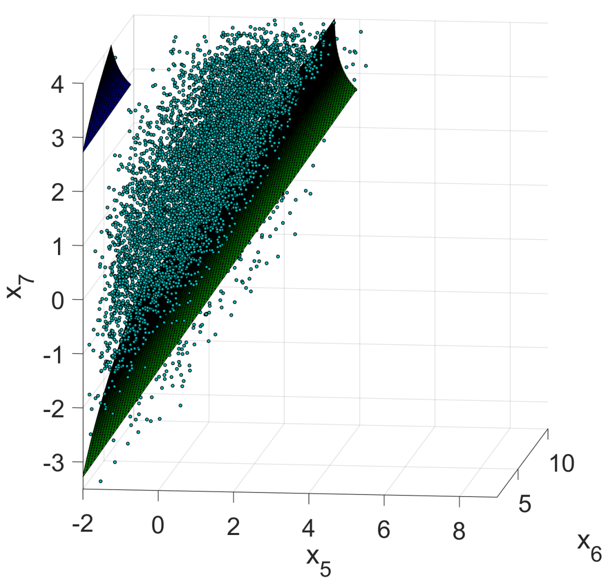

Figure 2.16: *Path 1* runs for *Analysis 3* with experience

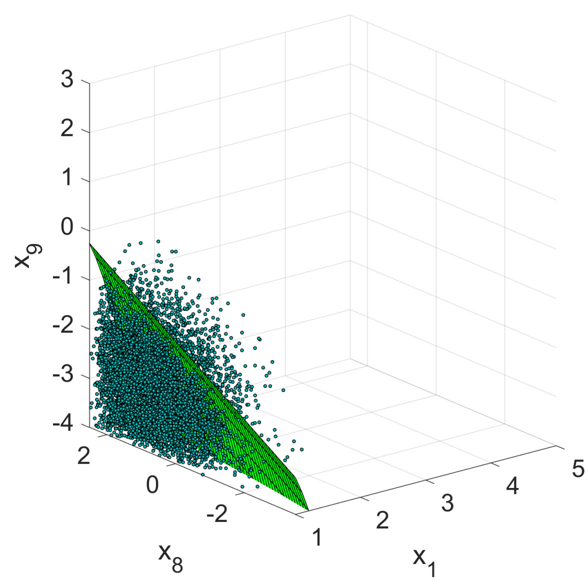

Figure 2.17: *Path 1* runs for *Analysis 4* with experience

2.2. Point-Based Design Discussion

The polynomial model has acted as a place-filler for exploring interdependent design activities. It has shown that the sequential order of design activities impacts the efficiency of design convergence. It has also helped formalize the path components of designer influence and consequential rework that ought to be considered during design path selection. The polynomial model has been helpful because its simplicity has allowed for large numbers of simulations to be performed on an arbitrary design problem in a reasonable amount of time. Unfortunately, designers working with actual complex design programs instead of generic mathematical equations cannot rely on bulk simulation to guide their path decisions. In these scenarios, the only real information designers have to work with are the known interdependent relationships between design activities. For complex design activities, these known relationships often only consist of the inputs and outputs of each analysis. Results of the polynomial model have also demonstrated the importance of incorporating experience into variable selection. In the unidirectional model, path success jumped significantly as soon as variable assignments were made with normal random variable information from previous success runs instead of uniform random values. Still, prior experience should be treated more as a commodity that is nice to have, not a necessity.

Even if a designer is able to account for designer influence and consequential rework without bulk simulation and mitigate the reliance on experience, it seems like sequential design is an inherently inefficient process. Each complex marine design problem is unique. Expectations of vessel capabilities are continuously evolving and with them, new industry and performance requirements must be met. Figure 2.18 is adopted from Wheelwright and Clark and captures the trends for design knowledge to increase and management (designer) influence to decrease over the course of a design [29]. No matter how many tweaks are made to sequential design, establishing a path order for design activities forces a designer to sacrifice influence and make decisions on vessel characteristics before the designer fully understands the repercussions of those decisions. Improper variable assignments and lack of designer influence later in a sequence lead to consequential rework that spans multiple design activities, continuously counteracting the work of others. For these reasons, this research will transition from working with a sequential design approach to a convergent one, with that new approach being SBD.

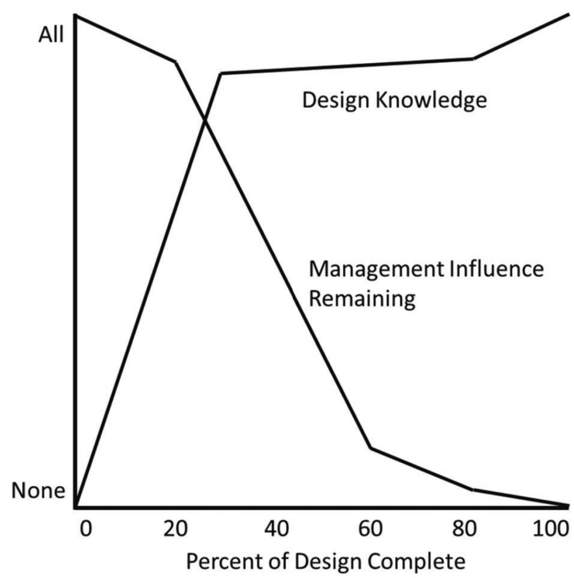

Figure 2.18: Design knowledge and management (designer) influence trends over time [29]

Convergent strategies, like SBD, delay making premature variable assignments while design knowledge is still growing. These strategies dispose of inefficient rework cycles by considering ranges of possible designs instead of a single design being pushed through a series of analyses with their own requirements. Ideally, these strategies can continue to explore each design space and delay decisions until designers are absolutely certain about variable trade-offs. In reality, convergent strategies must make decisions to reduce those ranges and keep the process moving. Reduction decisions made by design managers carry a certain amount of risk because decisions usually have to be made with information from disciplines that is incomplete or subject to change. Continued research will not study the sequence of decisions as was done with the polynomial model, but it will similarly study the consequences of decisions and investigate strategies for designers to make more informed decisions while only relying on the information at hand. More specifically, further research will propose ways that the risk of space reduction decisions can be quantified, managed, and reduced by evaluating the fragility of the design spaces that those decisions affect.

3. Set-Based Design

SBD follows a different trajectory than PBD. Instead of directing a single design through different disciplines and using their intermittent feedback to adjust that design, a set-based approach allows disciplines to operate concurrently as they communicate their preferred sets and eliminate infeasible or dominated regions. Figure 3.1 provides a visual of this process as adopted from Bernstein [1]. It begins with an initial search of all the design spaces with lower fidelity analyses, identifying preferred regions and disposing of those deemed infeasible. As these preferred regions are communicated to the group, overlapping design spaces are identified through shared variables, and disciplines work together to expand their intersections. When disciplines are satisfied with the region(s) of overlap, they revert back to shrinking the intersection(s) and subsequently increasing the fidelity of their analyses. In its idealized form, SBD continues to eliminate unfavorable regions until a single design remains as shown by last step in Figure 3.1. Although in practice, designers often lean on a SBD approach only in the early stages before transitioning back to PBD once space reduction becomes more tedious. To understand exactly how various disciplines identify preferred regions and propose to eliminate unfavorable ones, disciplines rely on unique analyses to sample their individual design spaces.

3.1. Design Spaces, Space Reduction, & Sampling

SBD is considered a convergent process because the design progresses through elimination rather than iteration. By approaching design this way, decisions on variable values are delayed, and designers perform analyses on variable *sets*. The downside of maintaining these sets is having to perform orders of magnitude more analyses to understand the design space,

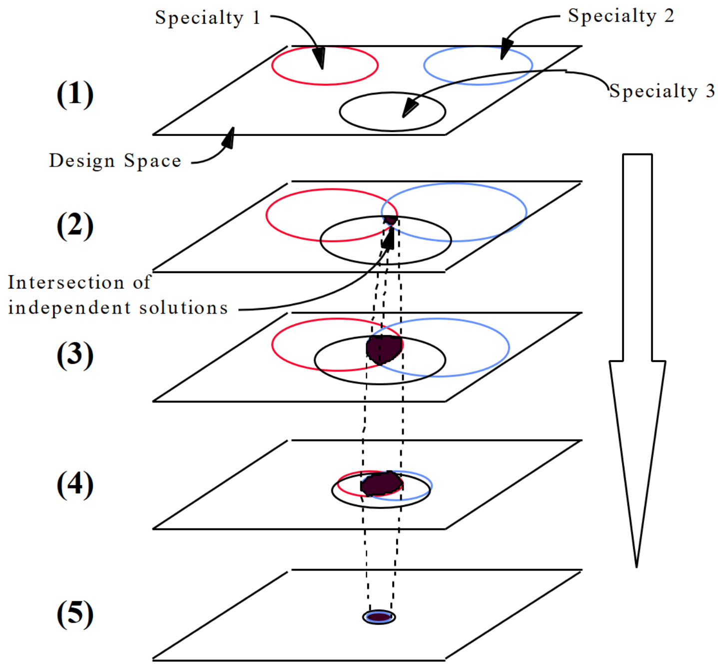

Figure 3.1: Stages of set-based design [1]

but the upside is having a much higher likelihood of ending with an optimal design than an iterative process. When reducing a design space that involves a multitude of disciplines, sound reasoning for each reduction must be provided. As it stands today, there are two primary justifications for reducing a set: infeasibility and dominance. Before defining these reduction types, one has to understand the different types of design spaces.

The two types of design spaces considered are the input space and the objective space. The input space encompasses the variables and their value ranges that the designer is free to manipulate. Principal characteristics of a ship such as length, beam, and draft are common variables found in the input space as the designer is often allowed to modify these characteristics to a certain extent. The objective space collects output values from the design activities for comparison against industry and performance requirements. Designers are not free to manipulate the output space; instead, they indirectly affect points in the objective space through the input space values they select paired with the design activities they perform.

Examples of the input space were shown in various figures of the polynomial model; one such example is provided in Figure 3.2. Here, the input space is characterized by three variables (x_2, x_3, and x_4) having unique upper and lower bounds. Also shown are two surfaces cutting through the input space. These surfaces were created by taking the known polynomial equations (of *Analysis 2*), substituting in the upper and lower bounds of the output variables (x_5 and x_6), and then displaying these surfaces, which are now only functions of the input variables, in the three-dimensional input space.

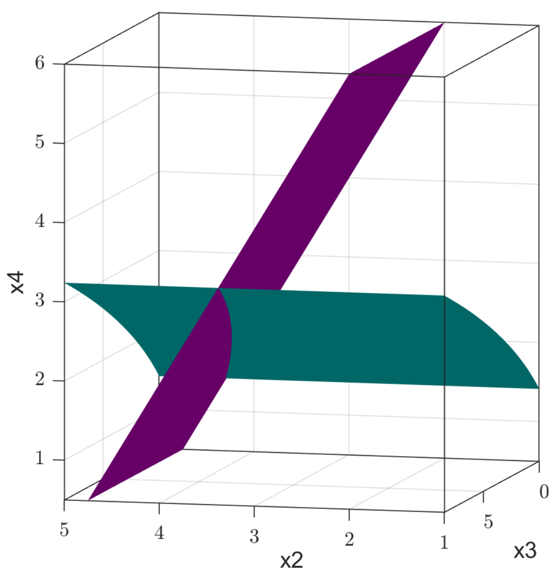

Figure 3.2: Example input space involving three variables

The surfaces of Figure 3.2 mark clear boundaries for the input variable combinations that are feasible and infeasible. Elimination via infeasibility involves reducing a design space on the basis of that space not meeting industry or performance requirements. In Figure 3.2, the bottom right quadrant is the feasible space because it encompasses input value combinations that result in calculated output values meeting their objective bounds. All other quadrants encompass input value combinations that result in at least one calculated output value not meeting the objective bounds. A designer utilizing a set-based approach could therefore eliminate the other three infeasible quadrants from design consideration.

After elimination via infeasibility, there is elimination via dominance. Design space elimination of this type involves reducing a design space on the basis of several disciplines declaring a range of variable values to be non-preferred. Looking at Figure 3.2 again, x_3 is feasible for its entire range of values from 0 to 8.5. However, if this discipline along with several other disciplines declared x_3 to be unfavorable for values less than 2, then each discipline's input space could be collectively reduced over that range.

The surfaces of Figure 3.2 are possible to visualize because as the creator of the polynomial model, I am aware of the simple equations that comprise it. A marine designer, on the other hand, will not necessarily be aware of the inner workings of the analysis program with which they are operating. The program may also be very complex with underlying theory that goes beyond mathematical formulations, making it impossible to visualize definitive feasible boundaries in the input space. In these cases, the analyses would be considered black-box functions, and the designer would be left to define their *perceived* feasible and infeasible regions through discrete tests of input space points and resulting objective space points.

Figures 3.3 and 3.4 depict the same input space as Figure 3.2, but with transparent feasible boundary surfaces - to symbolize a designer unaware of them - and different amounts of sampled points used to define the feasible regions. In the figures, the green points represent sampled input combinations with calculated outputs that meet their objective bounds, and the red points represent sampled input combinations with calculated outputs that do not. A designer reviewing these sampled input spaces would be much more confident defining the feasible spaces in Figure 3.4 than 3.3. Not only are the boundaries clearer in the case of 10,000 sampled points, there is also less unexplored space between the points of the feasible and infeasible areas. A designer assessing Figure 3.3 could formulate a general idea of feasible spaces, but their declarations would carry far more uncertainty.

The primary way to reduce uncertainty of feasible spaces for black-box programs is to

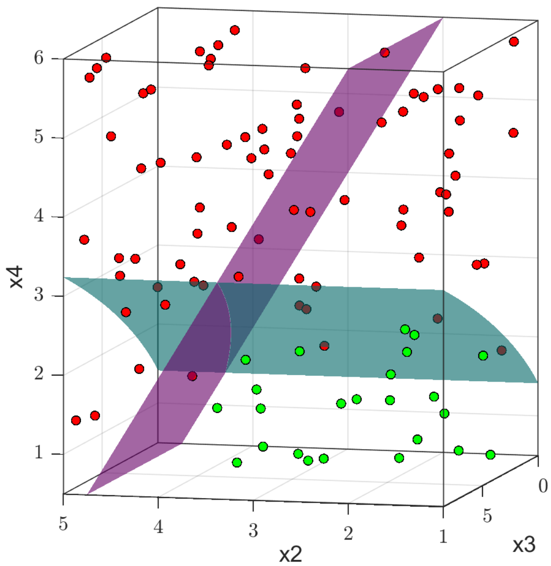

Figure 3.3: Input space having analyzed 100 sampled designs

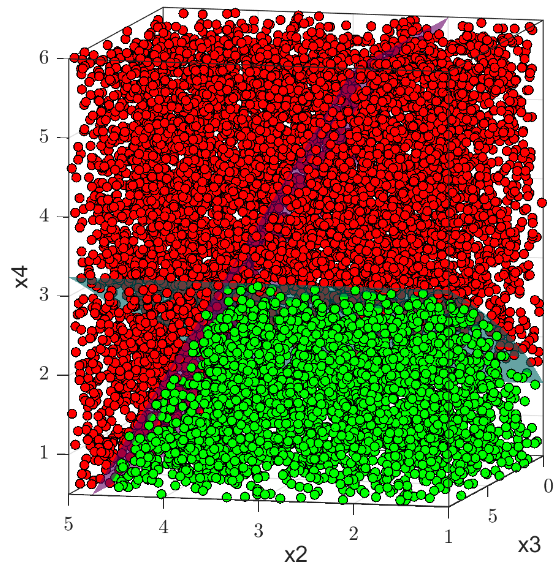

Figure 3.4: Input space having analyzed 10,000 sampled designs

sample more points. However, as a design progresses, and programs increase in fidelity, so do the time and cost commitments associated with them. If the above input space is following three variables being manipulated as part of a FEA, it is much more reasonable to gather 100 samples than 10,000 samples.

Furthermore, the marine design disciplines are extremely interdependent, and the variables impacting one discipline's design space often appear in multiple disciplines. Consequently, seemingly feasible spaces of one discipline may actually be ruled infeasible by another discipline. Take Figures 3.5 and 3.6 as an example. It was previously determined that the bottom-right quadrant of the first interdependent space is feasible, but now that quadrant contains yellow points for high values of x_3 and low values of x_4. The yellow points signify sampled designs which have outputs meeting the objective bounds of its particular discipline but not the objective bounds of another discipline. In Figure 3.5, the blue transparent surface materializes the upper bound of the x_5 objective variable into the input space, while the magenta transparent surface materializes the lower bound of the x_6 objective variable into the input space. Looking at Figure 3.6, the second interdependent space includes

x_5 and x_6 in its input space. This discipline never meets the requirements of its objective space when the x_5 variable is above a value of roughly 4. When the first interdependent space only accounts for its own results, all points in the same lower right quadrant would appear to be adequate. However, when it takes information from disciplines with interconnected variables (whether through the input spaces or objective spaces), it can further reduce its design space and avoid wasting time sampling in this infeasible region.

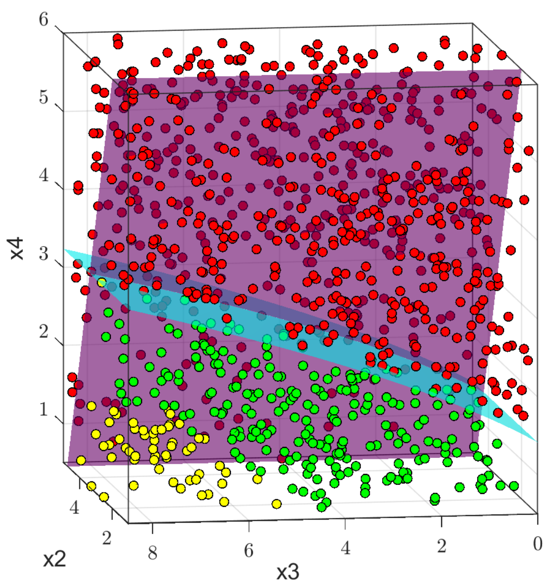

Figure 3.5: Interdependent space 1 having plotted 1,000 sample points

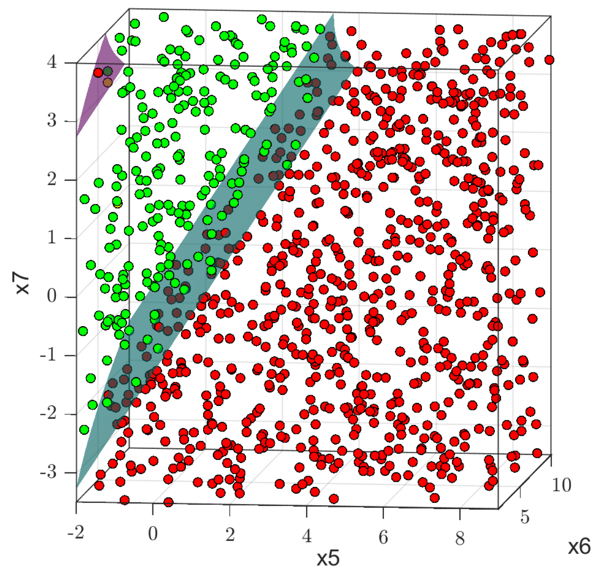

Figure 3.6: Interdependent space 2 having plotted 1,000 sample points

Sharing and receiving information between disciplines on preferred, non-preferred, and infeasible design spaces is crucial for the success of SBD. But again, as design progresses and analysis execution increases in cost and time, one cannot necessarily spend money and time indefinitely sampling points to understand the intricacies of the interdependent design spaces. Nor can one always wait on the feedback of other disciplines. The first interdependent design space would value information from the second interdependent design space on the infeasibility of the x_5 variable, but it is SBD's intent to perform analyses concurrently and continuously. By investigating previous studies that have attempted SBD, the difficulties associated with carrying it out from start to finish can be further illuminated.

3.2. Set-Based Design in Practice

A number of attempts have been made to document and impose SBD on engineering problems. Following their paper on the Second Toyota Paradox [28], Sobek II et al. made a more concerted effort to document the major principles of set-based concurrent engineering: mapping the design space, integrating by intersection, and establishing feasibility before commitment [24]. A few years later, Singer introduced a way for different disciplines to communicate and negotiate their preferences in marine design that went beyond one-on-one negotiation with fuzzy logic [22]. In 2013, McKenney completed his dissertation on a seven-step process for making structured SBD decisions [14]. The process required the user to pre-establish variable preferences and set range partitions for each input variable before using state mapping and Markov predictive models to determine the best order for space reduction. A year later, Hannapel and Vlahopoulos created an MDO algorithm that optimized the upper and lower bounds of input variables across multiple disciplines instead of singular design points [11]. This MDO improved design robustness by keeping sets of input values open until designers are ready to utilize a point-based optimizer. Most recently, Gumina documented his own SBD process involving design factor values, Latin hypercube design space exploration, and space reduction cycles [9].

While encouraging for designers to think in a set-based manner, these latter SBD implementation attempts take large, innovative leaps. McKenney's seven-step process is very involved and preselecting fixed variable sets while only making one reduction decision at a time takes away from SBD's flexibility. Hannapel and Vlahopolous showed variable sets can be merged with MDOs, but as is common for later stages of SBD, their process ultimately becomes a point-based optimization problem after the user is satisfied with the initial space reductions. Gumina demonstrates the full SBD process on an unmanned air system, but it undergoes many cycles of numerous analysis evaluations while heavily leaning on design manager intuition to continue.

Moreover, none of these papers consider how reduction decisions affect the fragility of each discipline's design space. A design space's fragility will refer to its susceptibility for new information to expose and invalidate the feasibility of remaining areas of the input space that have not been reduced. Put more simply, a fragile design space is the opposite of a robust design space. The reason for using the term fragility instead of robustness is that it more naturally coincides with the scope of this research. The goal is to quantify and determine the maximum risk a design manager should tolerate for space reduction decisions, or the maximum amount of design space fragility to tolerate. This phrasing fits better than saying the minimum amount of design space robustness to ensure. New information can come from unexplored areas of the input space, changes to requirements, changes to the underlying execution or assumptions of current analyses, and changes of present analyses to other analyses varying in fidelity. These last three types of new information can more broadly be defined as "design changes". As set ranges are collectively reduced across interdependent design spaces, they become more rigid and vulnerable to new information. To develop strategies that use the fragility of interconnected spaces to quantify the risk of space reduction decisions, one must first understand exactly how the specific combination of previous space reductions and future information can affect the fragility of remaining design spaces differently.

3.3. Reduction Decisions and Design Space Fragility

A design space is least fragile (or most robust) before any space reductions have been made. At this point in time, disciplines continue to gather information from discrete points tested in their input space to grow their understanding of the problem. No designs have been eliminated yet, meaning that each discipline should be able to handle any sort of new information they gather without any issue. Once reduction decisions are made, the fragility of each discipline's design space starts to increase. The extent to which the fragility increases depends on previous reductions and the potential of future information to impact those reductions. To better understand, Figure 3.7 acts as a copy of the third layer in

Bernstein's explanation of SBD involving three disciplines identifying feasible regions of overlap in their design spaces [1]. In the figure, the perceived feasible regions of each discipline are outlined by the red, blue, and black circles. If each discipline is absolutely confident in their perceptions of their feasible regions and there is no possibility of new information changing those perceptions, then design space fragility would not exist. Disciplines could feel free to eliminate the white, green, and yellow design spaces without issue before picking the best design in the orange space. However, perceptions of feasible regions are based on *discretely tested points* using a *wide range of analyses* that test designs against *arbitrarily or industry set requirements*, which are all subject to change in marine design. For those reasons, new information has the potential to impact present perceptions of feasible and dominated regions, meaning design space fragility does exist. Moreover, the interaction between feasible spaces of different disciplines is never as straightforward as outlined by Figure 3.7. Actual marine design problems involve a multitude of disciplines with some shared and some unique variables that the two-dimensional circles do not capture. A design manager needs to be careful with their space reduction decisions such that they do not exacerbate the fragility of any one discipline.

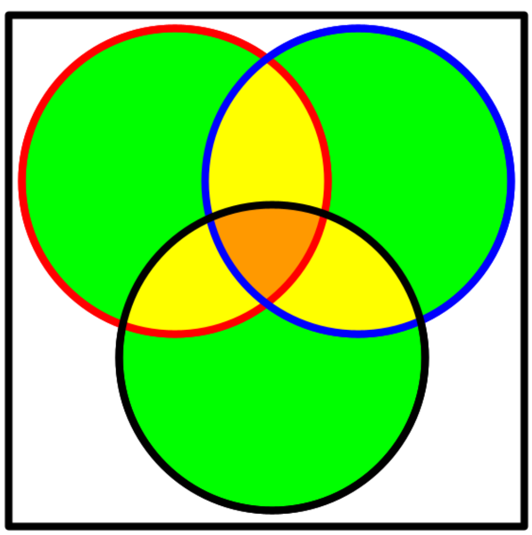

Figure 3.7: Overlapping regions of perceived feasible spaces for three disciplines of a design problem

Before showing how specific space reductions can worsen a design space's fragility, a couple of assumptions will be made for this work. First, it will be assumed that each discipline has thoroughly explored their design space before making any reduction request. This assumption will eliminate the possibility of new information in unexplored areas of a discipline's design space from contributing to its fragility and instead make fragility solely affected by past reductions and future design changes. Secondly, it will be assumed that at each reduction cycle, the design manager has received and merged all requests and supporting information so that the proposed reductions are consistent across shared variables of interdependent disciplines. With those assumptions, design space fragility as affected by previous reduction decisions and future design changes can be discussed from the perspective of a particular discipline.

The fragility of the red discipline's design space can be impacted in a multitude of ways. First and foremost, there must be some sort of initial space removal in any area (feasible or infeasible) of the design space shown in Figure 3.7. If design decisions were primarily influenced by the red discipline, the reduction hierarchy would more than likely begin by removing all white, green, and yellow space outside of the red discipline's perceived feasible bounds, followed by green then yellow space removal within the red discipline's perceived feasible bounds, before being left with the orange space. However, space reduction decisions are a collective effort that often compromise with contradictory preferential feedback of shared design variables from many disciplines, so the red discipline will have to manage fragility from *undesired* space reductions of its own perceived feasible region along with *desired* space reductions outside of its perceived feasible region. With these reductions, the red discipline becomes vulnerable to new information. One source of fragility arises from prematurely removing areas of the design space before learning of a design change of the red discipline, as depicted by Figure 3.8. In this figure, the dashed red circle signifies the newly perceived feasible space after gathering new information from a design change. If reduction decisions were made in the pink region before receiving this information, then there would be higher levels of fragility because the discipline would not have any of this newly perceived feasible space with which to work. Fortunately for this discipline, if any of the grey region was removed before receiving information of the design change, its fragility would slightly recover because that area of the design space is no longer perceived as feasible. Additionally, there is another source of fragility that arises from prematurely removing areas of the design space before learning of a design change of an interdependent discipline, such as the blue discipline depicted by Figure 3.9. In this figure, the dashed blue circle signifies the newly perceived feasible space of the blue discipline after gathering information from a design change. If reduction decisions were made in the pink region before receiving this information (which used to be a perceived infeasible area of the design space for the overlapping blue discipline), then there would be higher levels of fragility because these disciplines would no longer have this shared feasible area of the design space with which to work. Fortunately for the red discipline, if any of the grey region was removed before receiving new information from the blue discipline's design change, its fragility would slightly recover because that area of the design space is no longer perceived as a shared feasible space.

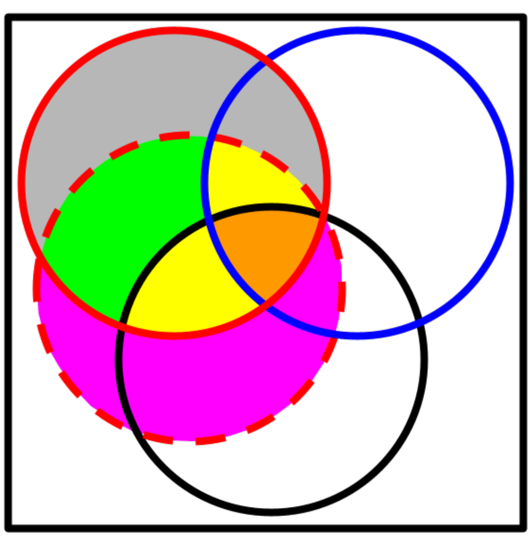

Figure 3.8: Fragility attributed to premature space reductions and design change of main discipline

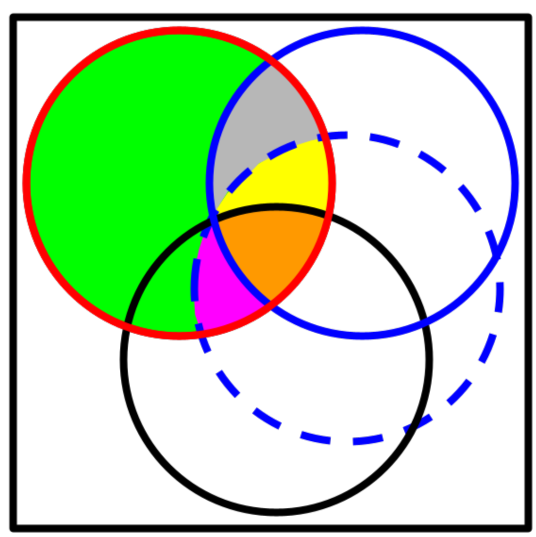

Figure 3.9: Fragility attributed to premature space reductions and design change of interdependent discipline

Design managers want to avoid space reduction decisions that lead to exceedingly fragile design spaces before all available information is known, yet they have to make decisions to keep the design process moving. At each space reduction cycle, every design space is susceptible to increases in design space fragility that can be further exacerbated by previous reductions and future design changes. And while the aforementioned figures have portrayed instances of increasing fragility in the context of premature space reductions and future design changes that lead to diminishing feasible spaces, the size of a design space alone does not correlate to its fragility. For example, if a multidimensional design space is very large, but one of its input variables only has two feasible values, that design space is vulnerable to new information from its own discipline and/or other disciplines declaring those two values infeasible. By effect, there are varying levels of risk for new space reduction decisions due to the varying levels of fragility that result from specific combinations of prior reductions and future changes. To truly be able to evaluate a design space's fragility and effectively quantify the risk of space reduction decisions, a design manager needs to observe how well each proposed reduction to a discipline's design space handles new information compared to the present, non-reduced design space. Later, risk levels worth accepting can be determined in context of remaining project time and budget. The key to accomplishing these feats may be in developing a fragility assessment framework that adopts various metrics from Information Theory.

4. Information Theory

In the broadest sense, Information Theory develops strategies and metrics to quantify how informative data is. One such metric that is instrumental to Information Theory is called entropy, which is a measure of the amount of uncertainty, or "surprise", in data. Information Theory and entropy have many different applications, with one application using entropy for exploration in a data set to identify *perceived* feasible boundaries. As entropy is a quantifier of uncertainty, there is much more uncertainty for data along feasible boundaries than there is in clearly feasible or infeasible regions. With that being said, this work plans on using entropy less as an exploration tool for a data set, and more as a quantifier of risk before making space reduction decisions. The following sections will first provide more background on information and entropy with basic examples and then detail the potential for integrating Information Theory with space reduction decision-making in SBD.

4.1. Background

When describing "informational value" of an event, learning a piece of information for an event that is unlikely to occur carries far more weight than an event that is likely to occur. For example, choosing numbers of a winning lottery ticket is very unlikely. While scrutinizing over what numbers to select, learning a specific number *will be* drawn would be much more informative than learning a specific number *will not be* drawn [5]. There is often a discrepancy in the utility of information an event conveys, or the extent to which the event is "illuminating" to the observer. To that end, Delgado-Bonal and Marshak define information as a decrease in the ambiguity of knowledge when observing an event [3].

Entropy builds on information as it is a measure of the *amount* of information an event is *expected* to reveal, or the expected informational value of an event [3]. Often times, the words "uncertainty" and "entropy" are used interchangeably; when one is uncertain about the outcome of an event, entropy is high because there is a lot of information to be learned. With this definition, entropy is only a function of the probability distribution of various events rather than the actual values of those events [3]. A common example used to more clearly explain entropy involves a coin flip [5]. When flipping a fair coin, there is a 50-50 chance the coin will land on heads or tails. With this perfect split in probability, a person has *no idea* whether the coin will land on heads or tails, leading to maximum uncertainty, or maximum entropy. For a double-sided coin that has two heads, a person is *certain* the coin will land on heads, leading to minimum uncertainty, or minimum entropy. For a coin that has heads and tails but is weighted such that there is a higher likelihood to land on one side than the other, a person has a *general idea* of how the coin will land, leading to moderate uncertainty, or moderate entropy. The specific equation used to calculate this expected value of information (H) is provided in Equation 4.1, where b is the base of the logarithm (commonly 2 for a binomial random variable) and p refers to the probability of the event, E [3]. A graph depicting binomial entropy levels (as was the case for the coin having two possible outcomes) is provided in Figure 4.1. As expected, the graph has the highest entropy levels for binomial events having a 50% chance of happening ($H = 1$) and the lowest entropy levels for binomial events having a 0% and 100% chance of happening ($H = 0$).

$$H(E) = -\sum_{i=1}^{\infty} p(E) \log_b p(E) \tag{4.1}$$

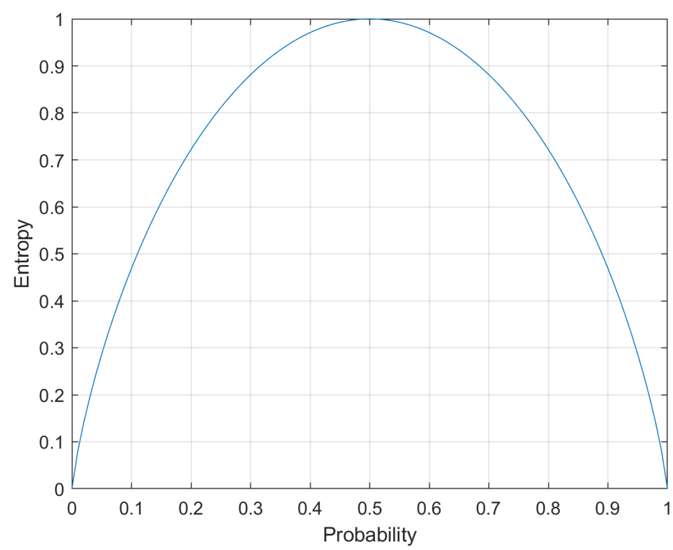

Figure 4.1: Entropy versus probability of first event occurring for binomial events

Using entropy to quantify known or uncertain information originated from Shannon's 1948 paper involving noise interference of transmitted messages [19]. In his work, Shannon quantifies the rate of information transfer of telegraphed messages, where in the case of telegraphy, information transfer refers to the reduction of uncertainty as to what the sequence of alphabetic symbols are describing. For a transmitted telegraphic message that is being revealed one symbol at a time, the message could mean absolutely anything before any symbols are revealed, corresponding to a high level of entropy. As symbols are revealed, known probabilities involving English language rules paired with any other sources of knowledge can be used to draw conclusions as to what the completed message will mean. If a message is eight letters long, there is more uncertainty as to what the message is describing before any letters are revealed compared to after a "TH" is revealed. Thought of in another way, the auto-complete function of an iPhone is able to offer better predictions for the word being written when five letters have already been typed compared to only two.

While Shannon uses telegraphy as an analogy to explain the mathematical theory behind entropy, it has countless other applications. Another application centers around Goodrum's work tracking information transfer within his "Knowledge-Information" (K-I) frameworks [8]. Goodrum's K-I framework has local and global layers that track information flow within and between disciplines to help a designer determine the difficulty in generating a design solution as well as the robustness of a design path. Coupled with the framework, he develops different entropy metrics such as topological entropy to track information contained in a knowledge structure, data status entropy to track a network's calculability over time, and target value entropy to track uncertainty of calculated values from different analysis methods. Goodrum demonstrates the usefulness of pairing his knowledge structures with these entropy metrics through an analysis of alternatives study for the design of three differently sized Landing Helicopter Dock hullforms. With this framework, a design manager is able to observe the knowledge generation of different disciplines that persuade actions such as reallocating resources to disciplines experiencing significant design churn or projecting the likelihood of meeting deadlines subject to new requirements.

Other types of entropy metrics, along with Goodrum's and Shannon's, could be useful for assessing the fragility of information across interdependent disciplines. Delgado-Bonal and Marshak break down the advantages and disadvantages of using approximate entropy versus sample entropy to track uncertainty based on the existence of patterns in a data set [3]. Oladyshkin and Nowak show how linking Bayesian inference with various metrics from Information Theory, such as relative entropy and cross entropy, benefit off each other to simplify computations for model selection and experimental design [16]. Chen et al. introduce a maximum entropy distribution function with an improved method of moments for parameter derivation and prove their method outperforms fitting extreme wave height data compared to other non-entropic methods [2]. With that being said, a much deeper literature search on these entropy types and Information Theory in general is still needed to be of use for this new SBD application. Before solidifying the exact entropy types to investigate for evaluating design space fragility and space reduction risk, the framework that would end up using or developing these metrics should be established.

4.2. Connecting Entropy with Fragility and Space Reduction Risk

The goal of creating a fragility assessment framework that tracks informational value and robustness is to be able to quantify the risk of various space reduction decisions subject to future design changes. With this information, design managers would be able to approve of a reduction, reduce the magnitude of a reduction, or hold off on a reduction until later when more information is gathered and/or time and budget constraints necessitate the decision. The key to creating such a tool is likely in understanding the value of present information and the fragility of that information when affected by reasonable perturbations that either replicate or imitate design changes. Entropy seems like it would make for a useful pairing with this framework as it tracks the expected value of information while fragility is an assessment of the extent to which new information affects the feasibility of a design space. Determining the types of entropy to pursue and exactly how to configure them into a fragility assessment framework becomes the main challenge.

To create such a framework, it helps to first visualize exactly how design changes impact perceptions of a design space. The two types of design changes to consider are requirement changes and model changes. These design changes and their effects on the input and objective spaces can be visualized with example design spaces involving three variables as shown in Figures 4.2 and 4.3. In these input spaces, *Variable 1* is shared between both disciplines, and designers run analyses for various locations in their input space to gain an understanding of the space as a whole. Green points in the input space represent tested points that meet design requirements, while red points represent tested points that do not. With the tested points, perceptions of feasible boundaries in the input space are represented by the dashed curves. Using this discrete information, suppose a design manager decides to reduce the design spaces of both disciplines as represented by the black boxes. In this case, the reduction decision is largely swayed by *Discipline 1* and its perceived infeasibility for small values of *Variable 1*. *Discipline 2* is also largely infeasible for small values of *Variable 1* except for a small perceived feasible region for when *Variable 1* is paired with larger values of *Variable 3*.

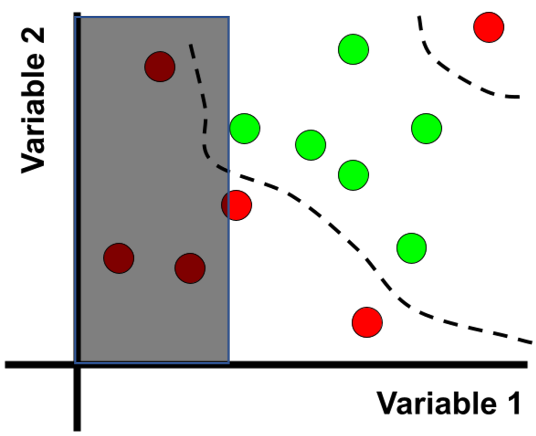

Figure 4.2: Input space of *Discipline 1* with example space reduction

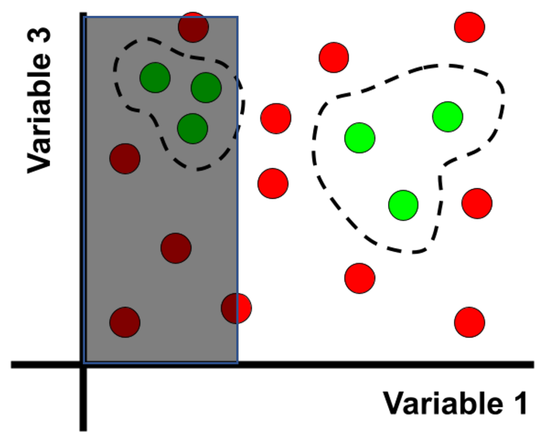

Figure 4.3: Input space of *Discipline 2* with example space reduction

For a design change, the one item that always remains the same is the precise locations that have already been tested in the input space. For a requirements change, the precise locations of the calculated output points also remain the same, but the feasible borders separating passing points from failing points in the output space shift. For a model change, the points in the output space shift, but the feasible borders in the output space stay the same. Figures 4.4 and 4.5 establish a potential objective space corresponding with *Discipline 1*'s input space where the solid lines represent known feasible boundaries and the green shaded region encompasses outputs meeting all requirements. Figures 4.6 and 4.7 exemplify a shift in a requirement in the output space and the corresponding impact it has on the previously tested points in the input space (two points now passing below the lower perceived feasible boundary). Figures 4.8 and 4.9 exemplify a model shift which *would cause* shifts in the location of calculated outputs and the corresponding impact it has on previously tested points in the input space (two points now passing below the lower perceived feasible boundary and three now failing within it). The words "would cause" have to be used for model changes because exactly how the points shift cannot be exactly known without actually testing the same points again with the new model.

With the new information following the requirement and model changes, the design manager's proposed space reduction is less than ideal. New information from a model change would have shed light on the original reduction decision had the design manager not committed to it, or that large of a space reduction, too soon. By effect, the fragility of each discipline's design space, bearing the weight of that early or that large of a space reduction, becomes increasingly exposed. Had the design manager planned for future design changes and paired them with the current state of the design space, they would avoid this dilemma.

The basic idea for the framework is to develop entropy metrics that compare the expected value of information of remaining design spaces before and after a proposed reduction. In Figure 4.4, there will be different entropy levels associated with different areas of the design space. These varying levels of entropy can be higher in areas where there is not a lot of information because of sparse data points, as well as in areas close to perceived feasible borders because of the abrupt transition of passing and failing designs. When perceptions of feasible borders are altered by a requirements change (as shown in Figure 4.6), the entropy

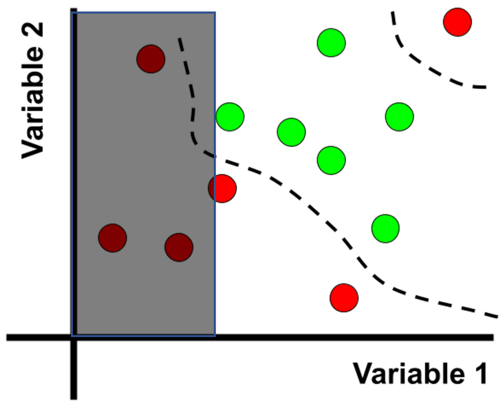

Figure 4.4: Input space of *Discipline 1* with example space reduction

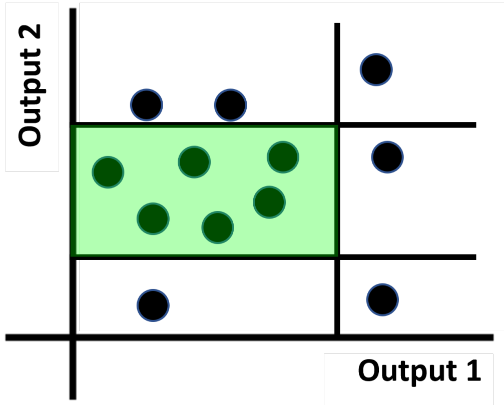

Figure 4.5: Objective space of *Discipline 1* before design change

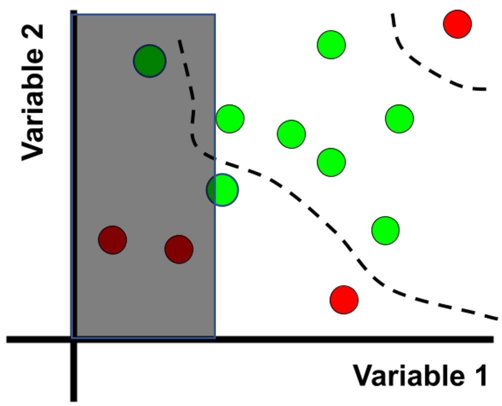

Figure 4.6: Input space of *Discipline 1* with example space reduction and impact of requirement change

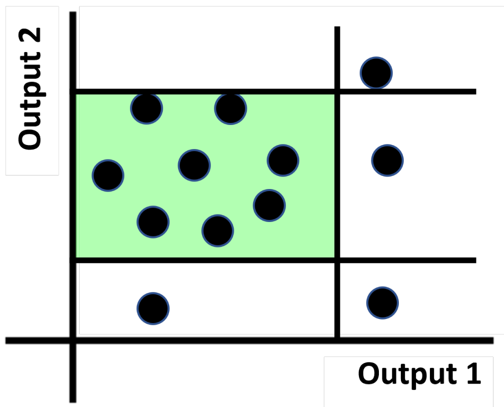

Figure 4.7: Objective space of *Discipline 1* after requirement change

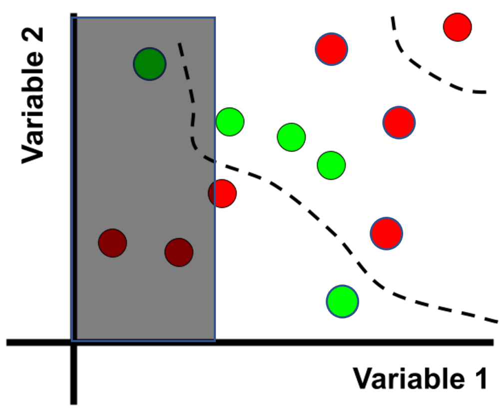

Figure 4.8: Input space of *Discipline 1* with example space reduction and impact of model change

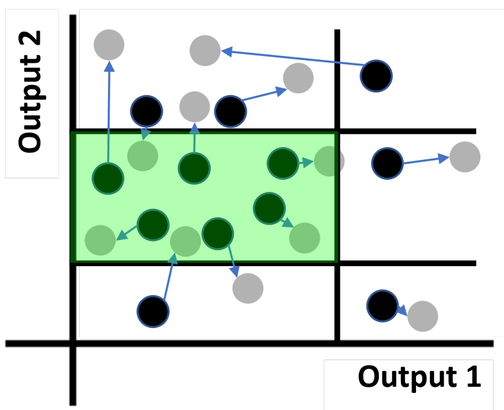

Figure 4.9: Objective space of *Discipline 1* after model change

levels will change in the areas of the design space that remain. These entropy changes will fluctuate depending on what areas of the design space the design manager is proposing to reduce and the design changes that are considered. If the remaining design space after the proposed reduction has very high entropy levels compared to before the reduction, designers would benefit from gathering more information. There would be a high risk for that specific reduction, so the designer manager should reconsider if the proposed reduction is suitable for that point in time. To create and test the framework, it will be helpful to develop a new design problem similar in complexity to the polynomial model.

5. Application to New Set-Based Problem

A new problem has been created to provide a simple, simulated design environment for building and testing a fragility assessment framework and entropy metrics. The new problem is similar to the polynomial model in that it consists of arbitrary mathematical equations used in place of actual marine design activities. However, the manner in which the new variables and equations interact with each other differs. As a reminder, the new design problem should lay the groundwork for the following research questions to be answered:

1. How can design managers track uncertainty associated with the behavior of unexplored areas of interdependent design spaces when making a space reduction?
2. To what extent can design managers quantify reduction risk across multiple disciplines by comparing design space informational content before and after a proposed reduction?
3. How might design managers leverage *present* information to suggest where continued exploration would elicit the most beneficial *new* information to alleviate a high-risk space reduction?
4. What is an acceptable level of space reduction risk to tolerate relative to project-based

time and budget constraints?

The following sections will break down the specifics of the new problem, detail the process for addressing each research question, and explain how the utility of the framework and entropy metrics will be evaluated against a SBD procedure that does not include them.

5.1. New Problem Definition

The new set-based problem considers a design involving three different disciplines as depicted in Figure 5.1. Each discipline has a series of inputs interconnected with the two other disciplines and output(s) unique to their discipline. The disciplines are limited to three dimensions for the input sets so that the design spaces can be easily visualized, but solution strategies for this problem should also be scalable for higher dimensionality. The possible range for each input variable goes from 0 to 1 as it is assumed that they are all normalized variables.

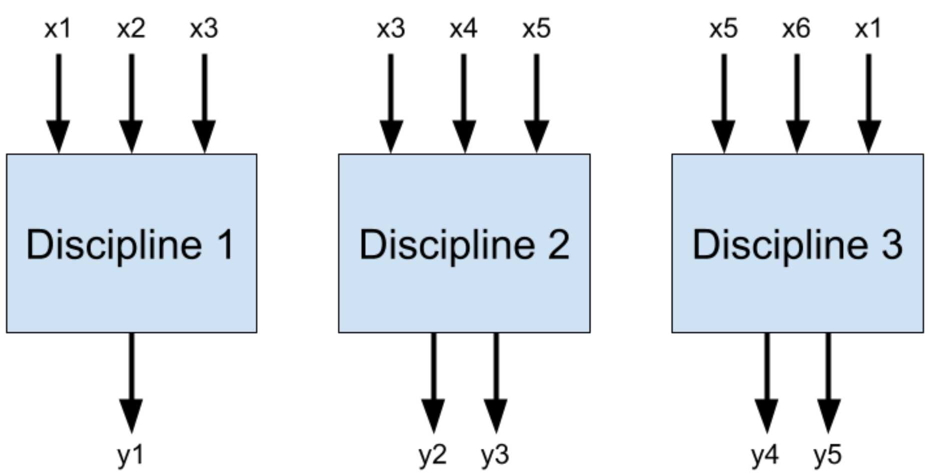

Figure 5.1: Input and output variables of three SBD disciplines

With the disciplines and their design spaces established, arbitrary mathematical equations transferring points from the input space to the objective space are created and provided by Equations 5.1 - 5.5. These equations are stand-ins for black-box programs and will not be known by the designers or design manager overseeing sampling and space reduction decisions in each discipline.

Discipline 1:

$$y_1 = 0.8x_1^2 + 2x_2^2 - x_3 \tag{5.1}$$

Discipline 2:

$$y_2 = 1.25x_5 - 12.5x_3^3 + 6.25x_3^2 \tag{5.2}$$

$$y_3 = \left(x_4^3 + x_5\right)^2 \tag{5.3}$$

Discipline 3:

$$y_4 = 2x_5 + 0.2\sin\left(25x_6\right) - x_1^{\frac{1}{5}} \tag{5.4}$$

$$y_5 = x_1^{\frac{1}{3}} - \cos\left(3x_5\right) \tag{5.5}$$

The required objective variable bounds that each discipline must meet are as follows: $y_1 = [0.00, 0.40]$ or $[1.20, 1.60]$, $y_2 = [0.50, 0.70]$, $y_3 = [0.20, 0.50]$, $y_4 = [0.00, 0.50]$, $y_5 = [0.80, 1.60]$. These bounds were chosen primarily because they help introduce a lot of surface curvature and intricate overlapping behavior when transferred over to the input space. The upper and lower bounds of each output variable are substituted into their respective equations so that the feasible spaces of each discipline can be observed in Figures 5.2 - 5.4. Again, the surfaces are not known to anyone performing or overseeing design activities because the equations of each discipline are considered black-box equations. Instead, the sampled points (green for passing and red for not passing) are supposed to help the designers and design manager visualize the feasible spaces. Because a limited amount of these sampled points will be used to understand a large design space, there will always be a certain level of uncertainty in their perceptions. Aside from the equations and variable bounds, the only other part of the problem that will be initially established are finite time and budget constraints for the project as a whole.

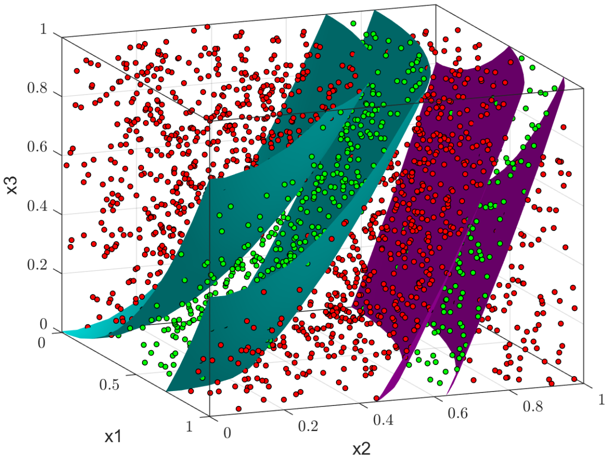

Figure 5.2: *Discipline 1*'s input space sampled with 2,000 points

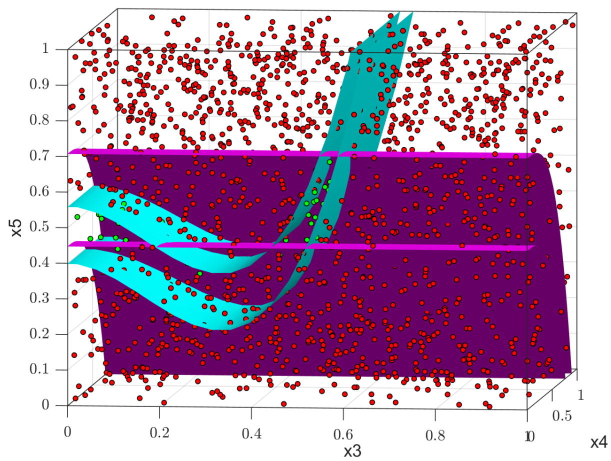

Figure 5.3: *Discipline 2*'s input space sampled with 2,000 points

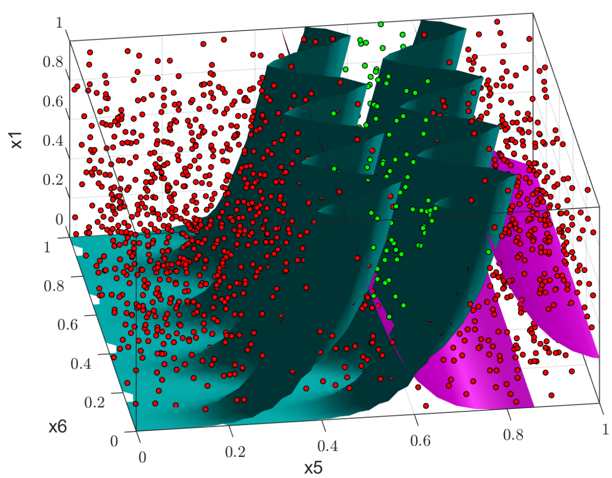

Figure 5.4: *Discipline 3*'s input space sampled with 2,000 points

5.2. Tentative Fragility Assessment Process

These steps will analyze the fragility of remaining design spaces before and after a proposed reduction. They will allow a design manager to assess how vulnerable each space becomes in comparison to the original spaces that do not include the proposed reduction.

1. Calculate various entropy metrics within the remaining design space before a proposed reduction using sampled feedback across each discipline

2. Introduce design changes by either perturbing sampled data or bounds to reflect po-

tential design changes or comparing distances of points from other points as well as points from feasible bounds in the input and objective spaces

3. Calculate the same entropy metrics across the same design space after the perturbations
4. Merge entropy metrics across each discipline and potentially prioritize feedback from the most vulnerable discipline
5. Repeat *Steps 1-4* for the design space that remains after the proposed reductions with the same perturbation strategy

5.3. Tentative Risk Management Process

These steps will use the fragility of each discipline's design space to quantify the risk of reduction for comparison to remaining project time and budget allowances so that the appropriate reduction decisions or precautions can be taken.

1. Quantify the risk of a reduction by comparing the change in the entropy results following perturbations of the entire design space against the reduced design space
2. Establish the amount of risk a design manager should be willing to take on based on time and cost to run each analysis and time and budget remaining for the project
3. For high-risk reductions, provide guided feedback to necessary disciplines on areas of the design space to explore that will likely return the most useful information for further space reduction proposals and assessment

5.4. Evaluating and Growing the Fragility Framework

The effectiveness of the fragility framework will be evaluated by comparing a SBD process that uses the framework versus one that does not. The downfall of the regular SBD process may be in that it converges on a design too quickly and is unable to handle any sort of design changes that arise, especially later on in the design process. The final design of the regular SBD process may also be less optimal than the design determined with the fragility framework, if a final design is converged on at all. To ensure any differences in results can be attributed to the fragility framework, the same steps that propose and merge space reduction requests will need to be used for the SBD process with and without the framework.

The framework will also need to be evaluated for more complicated SBD problems. The current SBD problem can first be made more challenging by also intertwining various outputs of each discipline with inputs of another discipline as is often the case for interconnected marine design activities. Later, the framework can be tested for an actual marine design problem consisting of many interdependent disciplines of multiple dimensions with analyses that are actual black-box programs and not mathematical substitutes. However, before pursuing fragility framework applications with new problems, disciplines would benefit from guidance on where to quickly and effectively explore their design spaces when the design manager decides to delay a high-risk reduction.

5.5. Guiding Exploration for High-Risk Reductions

SBD exploration is usually performed within each individual design discipline without outsider influence from a design manager or other disciplines. But for high-risk reductions, designers may need feedback from the design manager on areas of their design spaces where they should focus on sampling to efficiently mitigate that risk. Furthermore, it may be worthwhile having these interdependent disciplines work collectively to quickly generate new, high-value information for the design manager. In these particular instances, adaptive sampling (or active learning) strategies may be worth pursuing.

Adaptive sampling takes probabilistic and entropic-related feedback from previous samples to guide future samples towards particular areas of a design space. These strategies provide more direction than various one-shot sampling strategies that determine the sample size and points to test in a single stage [13]. They also provide more direction than various space-filling sampling strategies, such as Latin Hypercube Sampling (LHS), that ensure each level of each variable of a design space are equally represented [10]. Liu et al. document different types of single-objective, adaptive sampling methods which include those that are variance-based, query-by-committee-based, cross-validation-based, and gradient-based [13]. There are also existing studies that apply adaptive sampling strategies to complex engineering problems. Shahan and Seepersad opt for a generative, kernel-based Bayesian network classifier in place of a discriminant one to explore and exploit a two-dimensional design space of an unmanned aerial vehicle wing [18]. Jang et al. pursue active learning with a support vector machine when defining the feasible density and core thickness boundaries of an aluminum, foam-core sandwich panel meeting blast resistance specifications [12]. Shintani et al. approach a multi-disciplinary design problem involving the suspension of a car with a Bayesian active learning model which uses Gaussian probability and entropy for space exploration [21]. Each of these sampling strategies help a designer understand a design space with fewer points than their one-shot and space-filling counterparts, but they still have areas that can be improved for marine design application.

One shortcoming of relevance to interdependent marine design activities is the inability to seamlessly transfer information from one discipline to another. Shintani et al. claim their sampling strategy is geared towards multi-disciplinary design problems, but in their actual application to a car's suspension, they fix all but three design variables that end up being shared between each discipline's objective variables [21]. In a more realistic SBD setting, different disciplines have certain variables that are shared and certain variables that are unique, and a design manager is not immediately aware of which variables, if any, are worth fixing to simplify the problem. Liu et al. make an effort to document the limited multi-objective sampling methods available today [13]. There are a few general methods which use alternating or ranked objective variable feedback, symmetric methods which treat feedback from objective variables equally, and asymmetric methods which enhance a primary objective variable while also accounting for secondary objective variables. A marine SBD problem sharing feedback from interdependent disciplines and having multi-fidelity analyses is likely to benefit the most from an asymmetric method. Whether an ideal, asymmetric, multi-objective sampling method currently exists or requires refinement for marine design problems is a lingering question as Liu et al. admit, "Defining adaptive sampling strategies in multi-response modeling framework is of interest and is still an open problem that needs more research efforts" [13].

An adaptive sampling approach spanning multiple disciplines is not recommended for the entire duration of the SBD process. Rather, it could be very useful when the design manager has to help disciplines reduce fragility for a high-risk reduction while a project's timeline and budget continue to dwindle. Determining exactly how an adaptive sampling strategy would tie into the end of the fragility framework is still a long way off. In any case, it is the last piece in not only helping a design manager identify high-risk space reductions, but helping them work with disciplines to efficiently mitigate the risk of those reductions.

6. Recap & Future Work

In the first year of my studies, I completed extensive background research on decisions made in marine design and the consequences those decisions have on the design activities that follow. This research first led to the creation of a polynomial design model used to study the effect of different PBD paths on design convergence while introducing the concepts of designer influence and consequential rework. Work from the path-based research was submitted and accepted by two international conferences where I gave a virtual and in-person presentation on my work. The results of the path-based research show that PBD is an inherently inefficient process that is deeply rooted in prior experience to make decisions, so I transitioned to working with a SBD approach that I believe can be more efficient than PBD if design managers are given the proper guidance. Specifically, I advocate for creating a fragility assessment framework that uses metrics rooted in entropy from Information Theory to quantify the risk of proposed space reductions. With this framework, set-based design managers would be able to better understand the implications of their reduction decisions before actually committing to them. Upcoming work will focus on performing a much more extensive background search on Information Theory and entropy. Then work will concentrate on the new SBD problem described in the previous section. I have also completed all of my course credit requirements along with the four required workshops for new PhD students. Following the acceptance of this prospectus, I will progress from PhD student to PhD candidate status. The following section will chronicle the major contributions I hope to make during my remaining research years and lay out a tentative timeline for accomplishing these goals.

6.1. Major Contributions

The anticipated contributions over the course of my PhD are meant to help a designer gauge the risk of their proposed SBD reduction decisions in regards to the remaining design spaces' vulnerability to design changes. Traditional SBD has disciplines act concurrently and independently with their own unique design models and collective design requirements before reporting back on their preferred and unpreferred design spaces to a design manager. Design managers then pair their preferences with the information that was used to form them to make universal space reduction decisions before instructing disciplines to continue to independently explore their remaining design spaces. Unfortunately for the design manager, they have no way of understanding how fragile their reduction decisions make each design space until new information exposes their vulnerability. To help design managers stay wary of the affect that their reduction decisions have on the fragility of interdependent design spaces, the following contributions are planned:

1. Develop entropy-based metrics that quantify spatial and feasible boundary uncertainties of a sampled design space (*Research Question 1*)

2. Merge feedback from entropy-based metrics across multiple disciplines and develop a fragility framework to quantify the risk of space reduction decisions (*Research Question 2*)

3. Introduce adaptive sampling strategies that leverage informational feedback from the entropy-based metrics to suggest where further sampling should take place (*Research Question 3*)

4. Incorporate project-based time and budget constraints into the fragility framework to gauge acceptable risk levels worth tolerating at different stages of the design process (*Research Question 4*)

In pursuit of these contributions, immediate work will continue to replace actual marine design problems with mathematical equations analogous to these problems. Continuing research in this way keeps the focus on the *interactions* between design disciplines rather than the actual analyses of each discipline. Once each contribution has been adequately studied, the final fragility framework can be applied to an actual marine design problem.

6.2. Tentative Timeline

The timeline provided in Table 6.1 has been laid out for the remainder of my PhD. The dates and tasks provided are projections based on my current research standing that are subject to change.

Acknowledgements

This work is supported through the DoD National Defense Science and Engineering Graduate (NDSEG) Fellowship Program. The program's continued support is greatly appreciated.

Table 6.1: Tentative timeline over remainder of PhD

Date(s)	Task/Event
January 19, 2023	Prospectus presentation
February 2023	Gather and read through entropy related research
February 2023 - March 2023	Develop framework for fragility assessment
March 2023 - May 2023	Test framework & entropy metrics on new problem
June 2023 - July 2023	Compile findings involving the new problem
July 31 - August 4, 2023	NDSEG fellows conference in San Antonio, TX
August 2023	Isolate & gather code for different marine design programs to be used for marine design application
September 1, 2023	(Hopefully) join Dr. Kana and his lab at TU Delft
September 2023 - November 2023	Integrate entropy metrics and fragility framework with the different marine design programs
December 2023 - April 2024	Test framework on new marine design problem
May 2024 - July 2024	Compile findings involving marine design problem
July 2024 - August 2024	Prepare for defense
August 30, 2024	PhD defense & dissertation in Ann Arbor, MI

References

[1] Joshua I Bernstein. "Design Methods in the Aerospace Industry: Looking for Evidence of Set-Based Practices". In: (1998), pp. 1–226.

[2] Bai-yu Chen et al. "Maximum Entropy Distribution Function and Uncertainty Evaluation Criteria". In: *China Ocean Engineering* 35.2 (Apr. 2021), pp. 238–249.

[3] Alfonso Delgado-Bonal and Alexander Marshak. "Approximate Entropy and Sample Entropy: A Comprehensive Tutorial". In: *Entropy* 21.6 (May 28, 2019), pp. 541–578.

[4] Norbert Doerry. *Measuring Diversity in Set-Based Design*. Feb. 11, 2015, p. 14.

[5] *Entropy (information theory)*. In: *Wikipedia*. Nov. 5, 2022. URL: `https://en.wikipedia.org/wiki/Entropy_(information_theory)` (visited on 12/05/2022).

[6] Steven D Eppinger et al. "Organizing the Tasks in Complex Design Projects". In: *Massachusetts Institute of Technology* (Oct. 1989), pp. 1–19.

[7] J. Harvey Evans. "Basic Design Concepts". In: *Journal of the American Society of Naval Engineers* 71 (1959), pp. 671–678.

[8] Conner J Goodrum. "Conceptually Robust Knowledge Generation in Early Stage Complex Design". PhD thesis. Ann Arbor, MI, USA: University of Michigan, 2020. 181 pp.

[9] Jamie M Gumina. "A Set-Based Approach to Systems Design". PhD thesis. Monterey, CA: Naval Postgraduate School, Mar. 2019. 227 pp.

[10] Jamie M. Gumina, Clifford Whitcomb, and Alejandro S. Hernandez. “Latin Hypercube Sampling Strategies Applied to Set-Based Design”. In: SNAME Maritime Convention. Tacoma, Washington, Oct. 30, 2019.

[11] Shari Hannapel and Nickolas Vlahopoulos. “Implementation of set-based design in multidisciplinary design optimization”. In: *Structural and Multidisciplinary Optimization* 50.1 (July 2014), pp. 101–112.

[12] Sungwoo Jang et al. “Inductive Design Exploration Method with Active Learning for Complex Design Problems”. In: *Applied Sciences* 8.12 (Nov. 28, 2018).

[13] Haitao Liu, Yew-Soon Ong, and Jianfei Cai. “A survey of adaptive sampling for global metamodeling in support of simulation-based complex engineering design”. In: *Structural and Multidisciplinary Optimization* 57.1 (Jan. 2018), pp. 393–416.

[14] Thomas A McKenney. “An Early-Stage Set-Based Design Reduction Decision Support Framework Utilizing Design Space Mapping and a Graph Theoretic Markov Decision Process Formulation”. PhD thesis. Ann Arbor, MI, USA: University of Michigan, 2013. 317 pp.

[15] Farrokh Mistree et al. “Decision-Based Design: A Contemporary Paradigm for Ship Design”. In: *Society of Naval Architects and Marine Engineers* 98 (Jan. 1990), pp. 565–597.

[16] Sergey Oladyshkin and Wolfgang Nowak. “The Connection between Bayesian Inference and Information Theory for Model Selection, Information Gain and Experimental Design”. In: *Entropy* 21.11 (Nov. 4, 2019), pp. 1081–1100.

[17] Pedro Parraguez, Steven D. Eppinger, and Anja M. Maier. “Information Flow Through Stages of Complex Engineering Design Projects: A Dynamic Network Analysis Approach”. In: *IEEE Transactions on Engineering Management* 62.4 (Nov. 2015), pp. 604–617.

[18] David Shahan and Carolyn C. Seepersad. “Sequential Sampling With Kernel-Based Bayesian Network Classifiers”. In: *37th Design Automation Conference, Parts A and B*. ASME 2011 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference. Vol. 5. Washington, DC, USA: ASMEDC, Jan. 1, 2011, pp. 877–890.

[19] C E Shannon. “A Mathematical Theory of Communication”. In: *The Bell System Technical Journal* 27.3 (Oct. 1948), pp. 379–423.

[20] Colin P. F. Shields and David J. Singer. “Naval Design, Knowledge-Based Complexity, and Emergent Design Failures”. In: *Naval Engineers Journal* 129.4 (Dec. 2017), pp. 75–86.

[21] Kohei Shintani, Atsuji Abe, and Minoru Tsuchiyama. *Data-Driven Set Based Concurrent Engineering Method for Multidisciplinary Design Optimization*. Technical Paper 2022-01-0793. Toyota Motor Corporation, Mar. 2022, pp. 1–13.

[22] David J. Singer. “A Hybrid Agent Approach for Set-Based Conceptual Ship Design through the Use of a Fuzzy Logic Agent to Facilitate Communications and Negotiation”. PhD thesis. Ann Arbor, MI, USA: University of Michigan, 2003. 139 pp.

[23] David J. Singer, Norbert Doerry, and Michael E. Buckley. “What Is Set-Based Design?” In: *Naval Engineers Journal* 121.4 (Oct. 2009), pp. 31–43.

[24] Durward K. Sobek II, Allen C. Ward, and Jeffrey K. Liker. “Toyota’s Principles of Set-Based Concurrent Engineering”. In: *Sloan Management Review* 40.2 (1999), pp. 67–83.

[25] Donald V. Steward. “The design structure system: A method for managing the design of complex systems”. In: *IEEE Transactions on Engineering Management* EM-28.3 (Aug. 1981), pp. 71–74.

[26] J B Van Houten and M D Collette. “Exploration of Solution Path Dependencies with a Polynomial Model”. In: *Trends in Maritime Technology and Engineering* 1 (2022), pp. 629–633.

[27] Joseph Van Houten, David Singer, and Matthew Collette. “Balancing Designer Influence with Rework for Design Paths of a Simple Polynomial Model”. In: *Practical Design of Ships and Other Floating Structures* 1 (2022), pp. 508–527.

[28] Allen Ward et al. “The second Toyota paradox: how delaying decisions can make better cars faster”. In: *Sloan Management Review* 36.3 (1995), pp. 43–61.

[29] S. C. Wheelwright and K. B. Clark. *Revolutionizing Product Development: Quantum Leaps in Speed, Efficiency, and Quality*. New York: The Free Press, 1992.