Technical Report No. TR- 2011A

OBSERVATIONS OF DESIGNER BEHAVIORS IN COMPLEX SYSTEM DESIGN

Jesse Austin-Breneman Tomonori Honda
Maria C. Yang

Ideation Laboratory
Massachusetts Institute of Technology
Cambridge, MA 02139

email: jlab@mit.edu, tomonori@mit.edu, mcyang@mit.edu

Feb. 15, 2011

Abstract

The design of large-scale engineering systems requires design teams to balance a complex set of considerations. Formal approaches for optimizing complex system design assume that designers behave in a rational, consistent manner. However, observation of design practice suggests that there are limits to the rationality of designer behavior. This paper explores the gap between complex system designs generated via formal design process and those generated by teams of human designers. Results show that human design teams employed a range of strategies but arrived at suboptimal designs. Analysis of their design histories suggest three possible causes for the human design teams' performance: poorly executed global searches rather than well executed local searches, a focus on optimizing single design parameters, and sequential implementations rather than concurrent optimization strategies.

1 INTRODUCTION

The design of large-scale, complex engineering systems demand a diverse set of skills and expertise. To service this need, interdisciplinary teams are often employed, usually operating in a geographically distributed fashion. One of the continuing challenges of such teams is the balancing of conflicting considerations. Formal frameworks for complex system design such as Game Theory and Multidisciplinary Design Optimization (MDO) offer compelling strategies for arriving at design solutions in these situations. Such approaches reflect aspects of how design teams behave in practice and, in fact, hybrid approaches may be particularly effective [1]. This paper examines Game Theory more closely under the belief that improvements to it will benefit system design overall. Game Theory hinges on the assumption that subsystem designers consistently make rational choices during the design process in order to arrive at Nash Equilibrium [2]. In practice, teams that design complex systems are populated by humans who can be fallible, err in judgment, or make choices that are inconsistent with each other [3, 4, 5].

In practice, good system design is difficult to accomplish even by experienced practitioners under favorable circumstances. The broader goal of this work is a) to better understand how design teams behave during complex system design in order to create more effective, usable formal tools to support design, and b) enhance our basic understanding of how human design teams tackle complex engineering problems.

This paper presents a preliminary study assessing the role of human decision-making behavior in system design. Three human design teams were asked to design a satellite with three subsystems and their resulting designs, process and performance was analyzed. This study seeks to explore the following research questions:

1. In what ways will human decision-making differ from computer simulations?
2. How much will human-derived solutions deviate from optimal?
3. If they do deviate from optimal, what is the cause?

2 RELATED WORK

2.1 Structures for system-level design

The design of complex engineered systems is conducted by interdependent, multidisciplinary subsystems. A key challenge in system design is how to distribute limited resources among a set of subsystems. This situation is further complicated by the increasing use of distributed teams to design these systems [6] which presents communication and team cohesion problems for collaboration [7].

Large engineering systems are traditionally broken down into functional hierarchies. For example, an aircraft design can be broken down into structures and propulsion subsystems, with overlapping but not identical design parameters [8]. Furthermore, each subsystem can have thousands of input variables. In the classical approach to problems of this type, each subsystem is designed independently by discipline with system-level iterations occurring periodically throughout the process [9]. New systems-level approaches have been developed to increase the speed and effectiveness of the design process [10]. Industry has been quick to adopt systems-level approaches to interdisciplinary design [9, 10, 11, 12].

2.2 Design process models for complex systems

Metamodels are one tool used to quickly explore design spaces and converge to an optimal set of solutions. The metamodels either evaluate or approximate subsystem response to design parameter inputs. By generating system-level design outputs, the models can systematically search the design space and help guide designers towards an optimal design outcome. Limitations stem from the ability of the metamodel to accurately and quickly approximate the subsystem response to design inputs. In writing a comprehensive overview of research in the area of design and analysis of computer experiments, Simpson, et al. [13] present the wide range of problems that can be addressed through metamodels and associated algorithms. Sobieszczanski- Sobieski and Haftka's [14] survey demonstrates the range of applications in the aerospace industry.

Game Theory is an approach for modeling the multidisciplinary design process and was first proposed by Vincent [15] and further developed by Lewis and others [16, 17]. These traditional game theoretic approaches have further been combined with Decision-Based Design [18] and adopted in a broad range of design research [19, 20, 21, 22] to become a prominent framework for the study of multidisciplinary design problems [23]. Game Theoretic design attempts to identify a rational design given limits to the amount and form of information being passed between designers. The resulting designs may differ depending on the type and quantity of information exchanged. Thus, the resulting designs will be rational given limited information, but will not necessarily result in an optimal design.

2.3 Team structure and metamodels

Key components common to all of the metamodels are 1) the team structure or roles (i.e. the "direction" and "order" in which information is passed), 2) the form of the information passed between subsystems (such as point design and local sensitivities) and 3) how each subsystem makes decisions and trade-offs. We explore the last of these elements in this paper. Simulations have allowed researchers to observe the effect of changes in team structure, information passed and individual decision-making on performance metrics such as the speed and accuracy of the optimization. For example, Yi, et al. [24] compare seven MDO approaches with different hierarchical team structures. MDO models rely on the existence of a system facilitator who will make optimal trade-offs that will benefit the overall system. Honda et al. [1] also compared different team structures, comparing Game Theoretic and MDO approaches. Lewis and Mistree presented a Game Theoretic approach where each agent is involved in the optimizing task. In their model, agents made decisions using a compromise decision support problem [8]. In doing this meta-analysis, researchers have suggested best practices for design processes. Collopy outlines a strategy for reaching an optimal design based on passing of gradient information [25].

2.3.1 Bounded rationality decision-making in teams

Metamodels such as those presented above often assume designers are homogeneous agents who optimize their objective functions effectively. This assumption uses a definition of objective rationality, where the decision-maker will make the "optimal" or correct choice in every decision [26]. Research in the area of bounded rationality explores the consequences of limited resources found in real-world situations [27]. Models employing bounded rationality assume that since designers may have limited information and problem-solving capabilities they cannot evaluate and therefore cannot optimize their objective functions perfectly [3]. Satisficing and fast and frugal heuristics such as take the best or take the last algorithms are among the examples of bounded rationality models [28]. Computer experiments such as Gurnani and Lewis' study of collaborative decentralized design, can use randomness to simulate this uncertainty [22]. In these situations, bounded rationality is distinct from irrationality, which is defined as making a clearly inferior or sub-optimal choice [26].

2.3.2 Communication in teams

There is a rich body of literature on factors that affect team performance from organizational behavior, psychology and sociology. Because this type of design is commonly done in teams, the most relevant research in this area tests factors which affect team success across an array of interdisciplinary problems. Supporting similar research in metamodels, communication is a key factor in many of these studies. Nardi and Whittaker [29] emphasize the need for a shared team understanding for social communication. They investigated the importance of face-to-face communication in distributed design situations. Similarly, networking in the physical space of collocated teams has been shown to be an important determinant for design quality [30]. Team communication is also addressed in the area of team cognition. Cooke and Gorman [31] demonstrate several measures using communications as a method for understanding the team decision-making process and its ability to accomplish high-level processing of information and reach an optimal decision.

2.4 Research Gap

This case study seeks to integrate the lessons from both the social science research and formal models for complex system design. Metamodels define teams using three components: communication structure, type of information passed and the subsystem decision-making process. By testing different combinations of these three parameters, metamodels offer insight into the design process for a given problem. If a particular combination of team structure, information passing method and decision-making process work well together, then that design process can be considered "optimal." This study implements the communication structure and information passing methods used in Game Theoretic approaches with human subjects representing each subsystem. In using human subjects to make decisions, the study builds upon complex system design by identifying factors which affect the sub-system decision-making process and their relative importance to the overall system optimization process. The factors identified in this case study could be used in future studies refining metamodels. In this way, the authors hope to better understand factors affecting implementation of strategies suggested by computer experiments.

3 METHODS

In this study, three three-person teams performed a design task using a Game Theoretic approach. The resulting designs were compared with a baseline Pareto optimal design.

3.1 Teams

The population used in this case study was composed of graduate students in mechanical and aeronautical engineering. This population was chosen because their skill sets closely matched those required for satellite design. Each team included one member who had completed a semester-length graduate course on MDO, but as a whole, the teams should be considered "novices" with respect to satellite design. Team members were randomly assigned to a role in charge of a subsystem. All students were offered a $10 gift certificate to Amazon.com or a local restaurant as an incentive for participating in the study. A flip video camera was awarded to each member of the team with the best performance in the study.

3.2 Procedure

Each team was given a short (10 minute) introduction to the design task. The presentation consisted of an overview of the task, communication tools to be used in the experiment, a walk-through of one iteration of the design cycle, a demonstration of the local sensitivity vector and an explanation of the performance objectives of both the satellite and of the team. The subjects then provided informed consent. A custom-built spreadsheet and other communication tools were provided to support and capture the team design activity, and are described in further detail in Section 3.4. The team was then moved to computer workstations in separate rooms and given printouts of the presentation as reference. Teams were separated because it: 1) more closely mimicked a realistic distributed team scenario and 2) allowed the electronic capture of all communication between the subsystems [32]. The subjects were given several minutes to familiarize themselves with the computational and communication software and ask any questions regarding the experimental setup. The team had up to one hour to complete the design task. A researcher was available throughout the sessions to answer questions and assist with technical difficulties. At the end of the hour, the team selected their best iteration and the message logs and design histories were archived.

3.3 Design Task

The *Firesat* satellite example from Wertz and Larson's Space Mission Analysis and Design was chosen as the design task [34] because similar problems have been studied in other complex systems optimization research [8, 35, 36]. The design problem was broken down into three subsystems: Payload & Orbital, Power and Propulsion. Figure 1 shows the linked system of input and output variables. The highly coupled nature of the system is manifested by the effect of input variables such as Mass of Payload (M_{pl}),Total Amount of Change in Velocities ($DeltaV$) and Payload Power (P_{pl}) on multiple output variables. An adapted formulation from Honda et al was used because its relatively low number of design variables made it tractable within the short time period of the controlled laboratory experiment [1]. The aim of this optimization is to minimize both Ground Resolution (GR) and Total System Mass (M_{tot}) by varying M_{pl}, ΔV, and P_{pl}. The quality of a given solution was measured by its closeness to the Pareto Optimal Frontier and its compatibility error. To convert the optimization from a sequential formulation to a concurrent formulation, "slack" variables similar to those used in Linear Programming were introduced. These "slack" variables represent the expected output from subsystems that are required by other subsystems. In this case, the "slack" variables are expected height (h_{exp}) and expected mass of power subsystems($M_{pow,exp}$). Ideally the expected input values (h_{exp} and $M_{pow,exp}$) must match the calculated values from other subsystems(h_{calc} and $M_{pow,calc}$) at the final design stages.

The compatibility error at a given iteration was defined as the percentage error between either h_{exp} and h_{calc} or $M_{pow,exp}$ and $M_{pow,calc}$, whichever is higher. Compatibility error was calculated using the following equation:

$$\%err = \max\left(\frac{||h_{exp} - h_{calc}||}{(h_{exp} + h_{calc})/2}, \frac{||M_{pow,exp} - M_{pow,calc}||}{(M_{pow,exp} + M_{pow,calc})/2}\right) * 100\% \quad (1)$$

Ideally the "slack" variables would be equal at the final design state and the compatibility error would be zero. However an allowable discrepancy of 10% was set for the final iteration to avoid forcing teams to "polish" their result during the short time frame of the experiment.

3.4 Communication Tools

Figure 2 shows the team structure and communication links between team members. In the Game Theoretic approach, the subsystems can communicate freely directly with

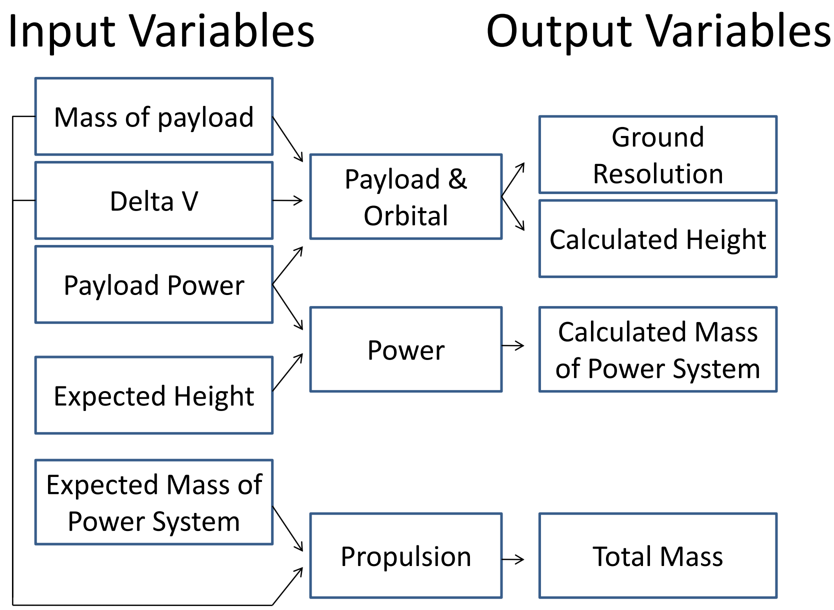

Figure 1: Input and Output Variables for Satellite Design Task

each other and try to improve system design rationally by fully utilizing shared information.

An Excel spreadsheet inspired by NASA's Jet Propulsion Laboratory ICEMaker tool [37] was customized to facilitate the exploration of the design space. The spreadsheet included an associated Visual Basic macro for each subsystem. This spreadsheet allowed the subsystem designer to calculate the output parameters for any given input vector.

The macro also calculated the local sensitivity vectors (gradient) to provide a design indicator to help the designer optimize the objective. A fundamental challenge of system design is a lack of visibility on how one design decision affects the overall system. The gradient gives the designer information on the local effect of the input variables on each output variable. The dot product of the change in the input variables and the gradient vector should be negative in order to minimize the output variable. In this way, the gradient indicates both the desired magnitude and direction of change in the input parameters for minimizing a given output. However, because there are multiple objective outputs, the designer must balance the information provided by the gradient for each objective and decide on a final direction and magnitude.

Table 1 contains gradient information as it might appear to a team during one iteration. In this case, the designer has to compromise on a direction (whether to increase or decrease) with respect to M_{pl}, P_{pl}, and ΔV. Note that a good choice might be to

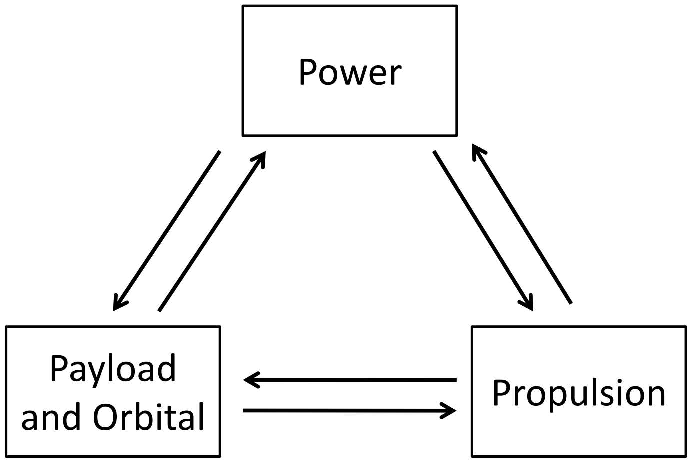

Figure 2: Game Theoretic team structure for the design team and task

Table 1: Sample gradient information

$\frac{\partial GR}{\partial M_{pl}}$	$\frac{\partial GR}{\partial P_{pl}}$	$\frac{\partial GR}{\partial \Delta V}$	$\frac{\partial M_{tot}}{\partial M_{pl}}$	$\frac{\partial M_{tot}}{\partial P_{pl}}$	$\frac{\partial M_{tot}}{\partial \Delta V}$
-0.15	-0.12	-0.04	3.47	0.08	0.04

keep ΔV the same since the directions are opposite and the magnitudes are the same. The magnitude indicates that decreasing M_{pl} has a large effect on M_{tot} when compared to P_{pl} and the largest effect overall. Therefore it make sense to decrease M_{pl}. However, the direction of the gradient with respect to GR shows that decreasing M_{pl} increases GR. To compensate for this increase, P_{pl} must be increased by factor of at least 1.25 times more than the decrease in M_{pl} to reduce GR simultaneously. This can be seen by comparing the relative magnitudes of $\frac{\partial GR}{\partial M_{pl}}$ and $\frac{\partial GR}{\partial P_{pl}}$. One solution is for the designer to decrease M_{pl} and increase P_{pl} with a ratio of 1 : 1.5 respectively. Thus, the gradient provides a way for the designer to understand the ideal direction for each input parameter in order to minimize both M_{tot} and GR.

A shared Google documents spreadsheet was also created to allow for communication of these vectors between team members. The Google document also combined the gradient information from each subsystem into an overall sensitivity vector for output parameters GR and M_{tot} with respect to system input variables. The Google document was accessible to multiple team members in near-real-time. The Skype messaging system was also used to allow for real-time communication between team members. The team structure was reflected in both the Google document and Skype programs with each subsystem able to see and edit all of the group documents. To accommodate the different spreadsheets and messaging windows used in this study, each workstation was equipped with two monitors.

4 RESULTS

The following results and observations were drawn from the archived design histories and message logs. The three teams tested were numbered and will be consistently referred to as Teams 1 through 3.

4.1 Optimization Results

The history of design choices of the three teams were analyzed to ascertain the optimality of the final solutions and compared to a baseline of the Pareto Frontier. The Pareto Frontier was generated via simulated annealing and provides a set of global optima. All teams opted to use the full hour to generate designs. At the end of the hour, they were asked to select the design they felt was their "best" design. These self-selected "best" designs are plotted in Figure 3 along with the Pareto Frontier. Beside each "best" design is a percentage value that indicates the error as calculated by the compatibility constraint (Eq. 1). Overall, none of the design teams generated a feasible solution that was close to the Pareto Frontier. Only Team 1 was able to keep the compatibility constraint to within 10% (Figure 3). Team 2 appears to achieve Pareto Optimality, but the compatibility error is unacceptable (*125%*) causing it to be an infeasible solution.

Figure 4 shows the history of the designs that each team explored over the hour. Teams 1 and 2 generated 8 designs each, and Team 3 generated 7 designs in total. None of teams managed to improve both GR and M_{tot} simultaneously in any iteration.

Figure 5 shows the high variability of compatibility error among the teams at each design iteration. Team 1 hand consistently low compatibility error. Team 2's initial design had low compatibility error but this increased as they generated new designs. Compatibility dropped back down after they returned to their initial designs. Finally, Team 3 had high compatibility error throughout the hour.

4.2 Types of decision-making strategies

An analysis of the design histories and instant messenger logs showed that all three groups arrived at sub-optimal solutions when compared to computer simulations. The sub-optimal choices can be classified into three types of decision-making errors: 1) performing a global search poorly rather than focusing on executing a local search efficiently, 2) optimizing a single input parameter at a time rather than exploiting coupling information between input parameters represented by the gradient and 3) optimizing the subsystems sequentially instead of concurrently. Table 2 shows the number of

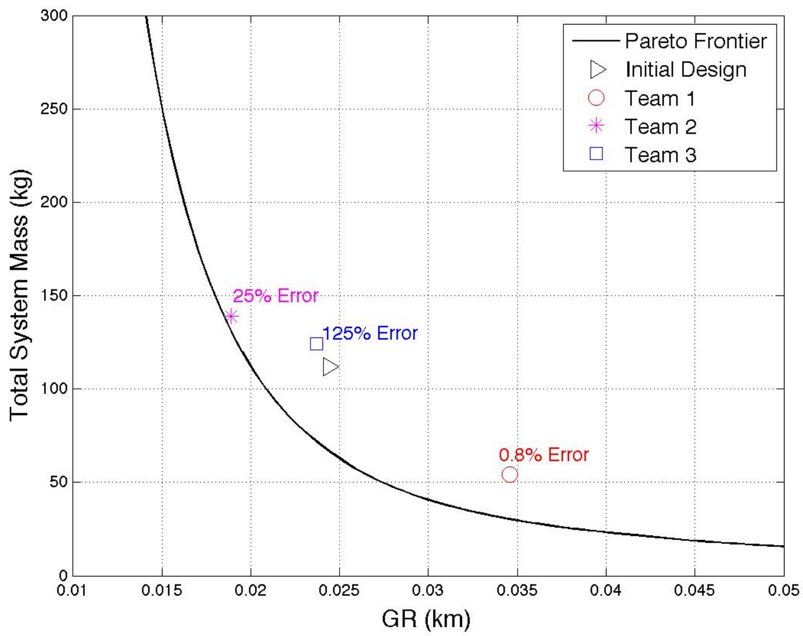

Figure 3: Comparison of 3 "best" design results selected by the teams. The Pareto frontier serves as a baseline. The percentage next to each point is the compatibility error of that solution.

Skype messages that each team sent in each category. Since the total number of messages was different for all groups and this tally is only for messages concerning each type of decision, absolute numbers are not significant. Rather, the prevalence of the messages indicates what type of error each group was committing.

In analyzing each team individually, the errors can be broadly labeled as optimizing from a local instead of a system perspective. In essence, the teams preferred a trial-and-error strategy instead of other common optimization techniques used in the computer simulations such as sequential linear programming [38] and sequential conjugate gradient-restoration method [39]. For example, Team 1 optimized a single parameter at a time. Since the subsystems are highly coupled, this method converges to an artificial local optima. In other words, fixing design parameters will provide addi-

Table 2: Errors mentioned in real-time messaging system

Problem Type	Team 1	Team 2	Team 3
Non-local search	2	16	0
Optimizing single input	22	9	12
Optimizing sequentially	3	4	1

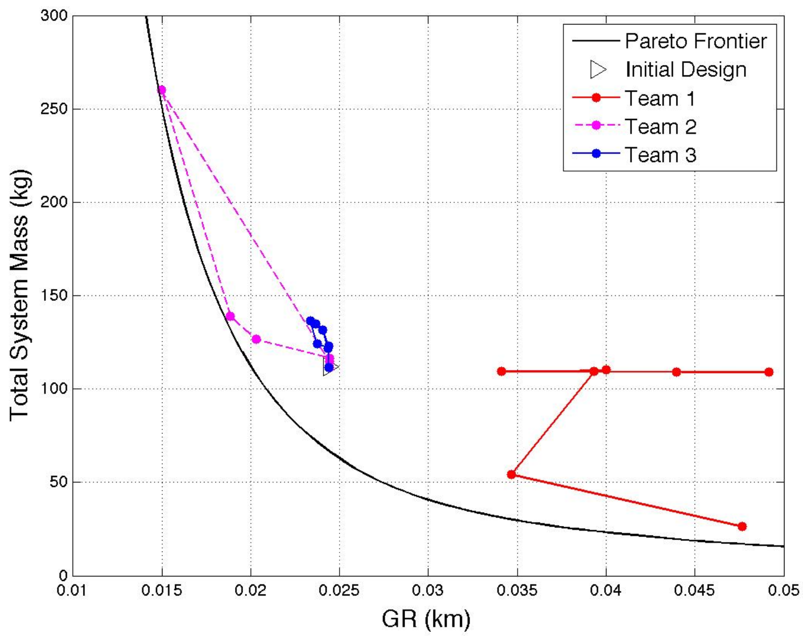

Figure 4: Design History for each team. Each path shows the design points explored by individual teams.

tional constraints that create local optima that do not exist without these constraints. Thus, optimizing input values independently tends to converge to suboptimal solutions in coupled systems. Team 1 also performed subsystem iterations sequentially instead of concurrently. This choice increased the iteration time, slowing the down the overall process. Given the short time frame of the experiment, concurrent iterations by each subsystem would have allowed for more iterations and possibly a more optimal solution. This issue is less critical for this particular case study because computational time for the Payload & Orbital subsystem is substantially slower than the other subsystems. Thus, average time per iteration for a sequential approach is about 1.5 times (rather than 3 times) slower than a concurrent approach. However, the sequential iterations avoided compatibility issues as the outputs h_{calc} and $Mpow_{calc}$ were used as the inputs for the next subsystem. This choice of a sequential strategy can be thought of as an example of bounded rationality. Although the sequential strategy is slower than the concurrent approach and therefore objectively inferior, it could be considered the ”best” decision for this team given a limited understanding of how to enforce compatibility between the subsystems. Overall, Team 1 performed the best of the 3 teams in terms of optimality and compatibility error. It must be noted that they chose as their ”best” solution an iteration which favored minimizing GR over their final iteration which was actually closer to Pareto Optimality. This may be due to the team’s limited information regard-

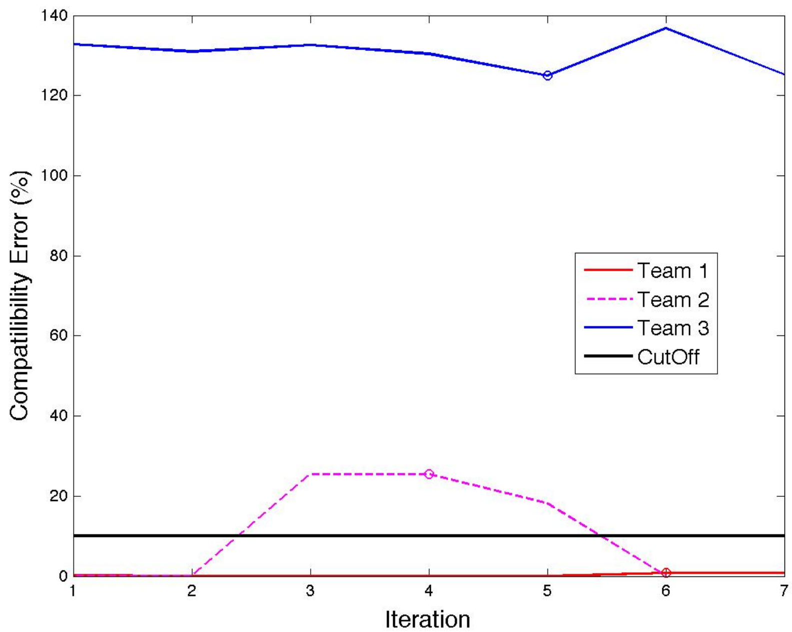

Figure 5: Compatibility Error between Subsystems as Function of Design Iteration

ing the location of the Pareto Optimal Frontier. Their decision may indicate that the group was not using gradient information to evaluate how close the solutions were to the Pareto Frontier.

Team 2's message logs show that they also preferred a trial and error strategy. The team searched the design space by doubling or halving input parameters and evaluating the effect on the objective variables. It is possible that this team aimed to look for global minima, rather than local minima. However, this strategy also led the team to arrive at a suboptimal solution. Given a highly-coupled complex system, small local searches are important in order to take advantage of information gained from the current design state. The nonlinear response to input vectors means that a general "downhill" direction can not be established from global searches. The large changes in input parameters also led the team to several infeasible solutions during their exploration of the design space. This strategy also caused large compatibility errors. At two points in their search, the team was close to a Pareto optimal solution, though with large discrepancies between h_{exp} and h_{calc}. At these two times, the team should have used the gradient information to correct the compatibility error. They instead moved the input parameters again and arrived at a final solution very close to the original starting point.

Like Team 1, Team 3 also optimized input variables independently on some iterations, mentioning this a total of 12 times in their message logs. Their searches were more local in nature and they did not explore the breadth of the design space well.

Although the group did reference local sensitivity vectors when discussing design decisions, they did not record the gradient information in their design history. They also had a largest compatibility error of over 100%. The group did not mention this large discrepancy or compatibility error in their message logs, even though they had been instructed to keep the compatibility error of at least the final solution to less than 10%. It is not possible from the given message logs and design histories to state whether the team simply focused on other objectives and ignored the compatibility error or did not correctly compute the compatibility error.

Notably, none of the three teams recorded the gradient information in their design history for several of the iterations. In the message logs, Teams 1 and 3 mentioned the gradient 11 and 8 times respectively, Team 2 only mentioned local sensitivities once. This coupled with the teams' failure to minimize both objective variables simultaneously in one iteration indicates that teams were using the gradient information sparingly in their decisions. Since only the systems-level design histories were archived, it cannot be determined if individual subsystems were using the gradient. However, given the coupled nature of the problem, a systems-level use of the gradient information is more critical.

5 DISCUSSION

In this preliminary study, the three main components of metamodels were implemented in the context of a human design team. The study used a Game Theoretic team structure with each subsystem being represented by one designer. Gradient information in the form of a local sensitivity vector was available and freely passed between the sub-systems. The individual subsystem decision process was controlled by the human designers.

It was expected that teams would look to the local sensitivities vector for guidance in generating their designs. In fact, teams used the gradient information very little. Because of this, the influence of the type of decision-making strategy became much more important. However, these results show that teams had a difficult time choosing an effective strategy.

Two major components of the decision-making process are the choice of optimization strategy and the convergence criteria. Common optimization techniques utilized by computer simulations are gradient-based strategies such as conjugate gradient techniques and constrained linear programming. These techniques are also widely used by in industry due to their step-by-step procedure and ease of implementation. Furthermore, the objective variables can be optimized either simultaneously or sequentially as in the case of constrained linear programming. These techniques contrast with the trial-and-error strategy chosen by the designers in this study. Convergence criteria are not applicable to the results in this study as all of the groups used the full amount of time without converging to a Pareto Optimal solution.

In this study, a lack of systems perspective in the decision-making process dominated team performance. The results in this case study may be due to a variety of factors including novice strategy choices, limited human problem-solving capability or bounded rationality, irrationality and team dynamics. The novice strategy choice may be due to a lack of training or knowledge about system level optimization. This would considered bounded rationality as the students could have been making rational choices given their limited human resources. However, each team did have a member who had taken a semester-length graduate course on MDO and so was at least familiar with the basic strategies and principles of formal design optimization. Irrationality, such as the decision by Team 3 which resulted in an increase in both objective variables, may have also played a role in the sub-optimal results. It is difficult in this study to differentiate this from bounded rationality. The results could also be explained by a combination of both irrationality and bounded rationality.

Based on the Skype instant messages exchanged within the teams, team dynamics also played a role in the strategy choice. In accordance with the rational model of group decision-making [40], all of the groups discussed what they should do before they began. However, suboptimal strategic choices were made during this initial stages for all three groups. For example, Team 2 decided to double and halve input parameters to explore the design space in a basic trail-and-error strategy. In the particular case of Team 2, one member suggested the doubling strategy and the other team members may have accepted it because of pluralistic ignorance. This is likely example of Abilene Paradox [41], in which one team member's suggestion is not refuted because the others perceive that the particular team member has expertise and/or information that they do not possess. The distributed nature of the teams in this study meant individual members did not have information on the relative expertise of other members.

Optimization skills and systems-level perspectives may be more apparent in design teams with more experience in complex system design. An expert design team may also be more likely to use gradient information. This case study suggests that metamodels could incorporate more information about the human aspects of the decision-making process. In this study, the skill level of the designers with respect to optimization and their inability to think from a systems-level perspective dominated the overall optimization and led to sub-optimal solutions. A metamodel of these groups would have to include bounded rationality with a high level of uncertainty. Also, the teams preferred to not use the gradient information. Since gradient-based optimization approaches are often more efficient, this suggests the need for either alternative methods of presenting the gradient information for effective use or a design protocol which is robust to novice mistakes.

Limitations to this preliminary study include the size and makeup of the population, usability of the software and the distributed nature of the team. First, the small sample size and student status of the teams means conclusions drawn from this study are not generalizable to all designer populations, though it serves as a useful starting point for future studies. Second, Team trust and cohesion has been shown to be important to team success [7]. Subjects were assigned to teams randomly, but teams who have worked together before or have a stake in working together in the future may have performed better in this study. Third, the communication tool was unfamiliar to the subjects and the Excel spreadsheet computation time for each subsystem varied. Slower than real-time communication certainly influenced the number of iterations possible and may have also confused the designers. Fourth, the choice of team structure may have also affected the results. In MDO structures, teams have a dedicated systems facilitator charged with thinking from a systems perspective. In Game Theoretic approaches, however, there is not centralized facilitator, and decision strategies rely on the expertise of individual subsystem designers. Finally, although the team members were separated to mimic the work environments of real-life distributed teams, a body of research suggests that co-located teams often perform better than virtual teams [10].

6 CONCLUSIONS AND FUTUREWORK

Results showed that a number of possible human factors, such as bounded rationality, irrationality and simple errors, dictated the outcome of the decision making process. Each designer preferred utilizing a trial-and- error strategy or drawing on design history rather than using more accurate gradient information that indicated how to best change a design parameter. When individual designers attempted to optimize their subsystems via trial and error, each assumed that his or her subsystem functions were separable with respect to input variables and so optimized each input independently. In reality, the subsystem functions were highly coupled, and this strategy led to suboptimal solutions. It was also found that designers focused on their individual subsystems rather than on the overall system perspective. This case study demonstrates the necessity of a design protocol that is robust to these types of mistakes.

1. In what ways will human decision-making differ from computer simulations?

   Human designers differed from computer simulations in their choice of design strategy and in the rationality of their behavior. It was expected that the designers would utilize the gradient information provided to guide their choices, but they did not. Without the aid of gradient information, designers relied on various decision-making strategies to generate designs.

2. How much will human-derived solutions deviate from optimal?

   The solutions that resulted from the above strategies deviated substantially from optimal with several teams searching infeasible design spaces.

3. If they do deviate from optimal, what is the cause?

   This study identified several possible causes such as a lack of system-level optimization knowledge or training, irrational or bounded rational behavior by the designers and team dynamics.

Future work should involve studying teams with more experience in designing engineering systems to assess their behavior in this type of system design scenario. In particular, it would be useful to understand what strategies such designers employ. Future work should also include testing team structures such as MDO on human decision-making. The work presented in this paper also has ramifications for how we train and educate engineering students. Most engineering systems, whether simple or complex, require some understanding of how decisions for one subsystem affect those for another subsystem. The results of this study suggest that students could benefit from more training in system level thinking.

ACKNOWLEDGMENTS

The work described in this paper was supported in part by the National Science Foundation under Award CMMI-0830134. The work was also supported in part by a Ford Foundation Predoctoral Fellowship from the National Research Council of the National Academies. The opinions, findings, conclusions and recommendations expressed are those of the authors and do not necessarily reflect the views of the sponsors.

References

[1] Honda, T., F. Ciucci, et al. (2010). "A Comparison of Information Passing Strategies in System Level Modeling." International Design Engineering Technical Conferences. Montreal, Canada.

[2] J. F. Nash (1951). "Non-Cooperative Games. " Annals of Mathematics, 54, pp. 286-295.

[3] H. A. Simon (1997). Models of Bounded Rationality, MIT Press, Cambridge, MA.

[4] Devendorf, E., and K. Lewis. (2008). "Planning on Mistakes: An Approach to Incorporate Error Checking into the Design Process," ASME Design Engineering Technical Conferences. New York, NY.

[5] Gurnani, A. P., and K. Lewis. (2008). "Using Bounded Rationality to Improve Decentralized Design," AIAA Journal, 46(12): 3049-3059.

[6] G. Mark (2002). "Extreme collaboration." Communication of the ACM 45(6): 89-93.

[7] Yang, M. C. and Y. Jin (2008). "An Examination of Team Effectiveness in Distributed and Co-located Engineering Teams." International Journal of Engineering Education 24(2): 400-408.

[8] Lewis, K. and F. Mistree (1997). "Modeling Interactions in Interdisciplinary Design: A Game Theoretic Approach." AIAA Journal 35(8): 1387-1392.

[9] Bandecchi M, B. Melton, et al. (2000). "The ESA/ESTEC Concurrent Design Facility". 2nd European Systems Engineering Conference. Munich, Germany.

[10] Karpati, G., Martin, J., Steiner, M., and Reinhardt, K. (2003). "The Integrated Mission Design Center (IMDC) at NASA Goddard Space Flight Center." Proceedings of the IEEE Aerospace Conference. IEEE, Big Sky, MT, Vol. 8, pp. 8_3657- 8_3667.

[11] Gough, K., B. D. Allen, et al. (2005). "Collaborative Mission Design at NASA Langley Research Center." Space Systems Engineering Conference. Atlanta, GA.

[12] Ancona, D., H. Bresman, et al. (2002). "The Comparative Advantage of X-Teams." MIT Sloan Management Review 43(3): 33-39.

[13] Simpson, T. W., J. D. Poplinski, et al. (2001). "Metamodels for Computer-based Engineering Design: Survey and recommendations." Engineering with Computers 17(2): 129-150.

[14] Sobieszczanski-Sobieski, J. and R. T. Haftka (1997). "Multidisciplinary Aerospace Design Optimization: Survey of Recent Developments." Structural and Multidisciplinary Optimization 14(1): 1-23.

[15] T. L. Vincent (1983). “Game Theory as a Design Tool.” Journal of Mechanism, Transmissions, and Automation in Design 105: 165-170.

[16] K. Lewis (1996). An Algorithm for Integrated Subsystem Embodiment and System Synthesis. PhD thesis, Georgia Institute of Technology.

[17] Whitfield,R. I. , Duffy, R. I., et al. (2002). “Distributed design coordination.” Research in Engineering Design 13: 243-252.

[18] G. A. Hazelrigg (1998). “A framework for decision-based engineering design.” Journal of Mechanical Design 120: 653-658.

[19] Chanron, V. and K. Lewis (2005). “A study of convergence in decentralized design processes.” Research in Engineering Design 16(3):133-145.

[20] Chanron, V., Singh, T. and K. Lewis (2005). “Equilibrium stability in decentralized design systems.” International Journal of Systems Science 36(10):651-662.

[21] Xiao, A., Zheng, S., and et al. (2005). “Collaborative Multidisciplinary Decision Making using Game Theory and Design Capability Indices.” Research in Engineering Design 16(1-2):57-72.

[22] Gurnani, A. P., and K. Lewis (2008) “Using Bounded Rationality to Improve Decentralized Design.” AIAA Journal 46(12): 3049-3059.

[23] Lewis, K. E., Chen, W. and L. C. Schmidt (2006). Decision Making in Engineering Design. American Society of Mechanical Engineers.

[24] Yi, S., J. Shin, et al. (2008). “Comparison of MDO methods with mathematical examples.” Structural and Multidisciplinary Optimization 35(5): 391-402.

[25] P. Collopy (2001). “Economic-based Distributed Optimal Design.” AIAA SPACE 2001 Conference and Exposition. Albuquerque, NM

[26] Gilboa, I., F. Maccheroni, et al. (2010). “Objective and Subjective Rationality in A Multiple Prior Model” Econometrica 78(2): 755-770.

[27] J. Hermann (2010). “Progressive Design Processes and Bounded Rational Designers” Journal of Mechanical Design 132: 0810051-0810058.

[28] Gigerenzer, G. and D. D. Goldstein (1996). “Reasoning the Fast and Frugal Way: Models of Bounded Rationality.” Psychological Review 103(4): 650-669.

[29] Nardi, B. and S. Whittaker (2002). The Place of Face-to-face Communication in Distributed Work. Distributed Work. P. Hinds and S. Keisler. Cambridge, MA, MIT Press: 83-109.

[30] A. Kendon (1990). Conducting Interaction: Patterns of Behavior in Focused Encounters. New York, NY, Cambridge University Press.

[31] Cooke, N. J. and J. C. Gorman (2006). Assessment of Team Cognition. International Encyclopedia of Ergonomics and Human Factors. P. Karwowski. UK, Taylor & Francis Ltd.: 270-275.

[32] Jin, Y. and M. Geslin (2010). “A study of argumentation-based negotiation in collaborative design.” Artificial Intelligence for Engineering Design, Analysis and Manufacturing 24: 35-48.

[33] Jet Propulsion Laboratory, (2010). “Team X,” http://jplteamx.jpl.nasa.gov/.

[34] Larson, W. and J. Wertz (1999). Space Mission Analysis and Design. El Segundo, CA, Microcosm Press and Kluwer Academic Publishers.

[35] Freiheit, T. and S. Rao (1991). “A modified Game Theory Approach to Multiobjective Optimization.” Journal of Mechanical Design 113: 286-291.

[36] Shiau, C. and J. Michalek (2008). “Should Designers Worry About Market Systems?” Journal of Mechanical Design 131: 1-9.

[37] Parkin, K. L. G., J. C. Sercel, et al. (2003). “Icemaker: an excel-based environment for collaborative design.” 2003 IEEE Aerospace Conference. Big Sky, MT

[38] R. Bhowmik (2008). “Building Design Optimization Using Sequential Linear Programming” Journal of Computers 3(4): 58-64.

[39] Miele, A., Huang. H. Y., and J. C. Heideman. (1969). “Sequential Gradient-Restoration Algorithm for the Minimization of Constrained Functions- Ordinary and Conjugate Gradient Versions” Journal of Optimization Theory and Applications: 4(4): 213-243.

[40] D. Forsyth (1990). Group Dynamics. Pacific Grove, CA, Brooks/Cole.

[41] J. Harvey (1974). “The Abilene Paradox: The Management of Agreement.” Organizational Dynamics, 3(1): 63-80.