INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED'09
24 - 27 AUGUST 2009, STANFORD UNIVERSITY, STANFORD, CA, USA

ACHIEVING PARETO OPTIMALITY IN A DECENTRALIZED DESIGN ENVIRONMENT

Tomonori Honda[1], Francesco Cucci[2] and Maria C. Yang[1]
(1) Massachusetts Institute of Technology (2) California Institute of Technology

ABSTRACT
As engineering systems grow in complexity, the teams that design them require increasingly disparate expertise and must operate in a distributed fashion. At the same time, the teams that design subsystems need to compete and compromise with each other for a limited set of resources. Thus, it becomes crucial to establish a system-level understanding of the trade-offs between subsystems. However, there is little research regarding formal design methods for determining rational designs in a decentralized environment. Lewis and others have developed an effective Game Theoretic approach based on Decision Theory to locate a Nash Equilibrium design with minimum information sharing. This paper presents a design technique to balance tradeoffs between subsystem and system performance by minimizing information sharing between subsystems, thus converging to a set of Pareto Optimal designs. In this research, designers pass a quadratic approximation of each subsystem's objective functions at each iteration. This approach is illustrated by case examples of a pressure vessel and an airplane that show how Pareto Optimal Designs may be obtained in a decentralized design environment.

Keywords: Decentralized Design, Distributed Design, Pareto Optimality, System Design

1 INTRODUCTION
In recent years, engineering systems have become steadily more complex, resulting in a rising number of disparate subsystems that must be integrated. Each subsystem is usually designed separately by experts from different fields who are often geographically dispersed, making information sharing and communication between designers of subsystems logistically more difficult. Subsystems must further compete with each other for a limited set of resources. Under such circumstances, it becomes critical to understand the trade-offs that must be made between subsystems in order to reach a final design. An example of such a system is NASA's Space Mission Design which focuses on highly specialized subsystems such as propulsion, power, and attitude control systems. One of NASA Jet Propulsion Laboratory's organizational approaches to addressing trade-offs within complex systems is the Advanced Projects Team (Team X). In Team X, the entire team is intentionally geographically co-located to facilitate rapid communication between the subsystems. This arrangement helps ensure that design trade-offs are understood at both the subsystem and system level. Balancing trade-offs is not only an issue for space missions, but are applicable to other areas where large systems are involved, such as the design and engineering of aircrafts, automobiles, ships, and buildings.

This paper investigates a method to define and effectively balance the tradeoffs between subsystem performance and overall system performance. Rather than addressing this problem from a primarily organizational point of view as in the Team X project, the way information is shared in a distributed design environment is examined, particularly when there is a lack of perfect, shared information between each designer of a subsystem. Individual designers may possess the full information necessary to design one subsystem, but only have limited information about other subsystems. In practice, it is extremely difficult to achieve perfect information sharing among even closely co-located parties; even modest geographical distribution can decrease information sharing dramatically [1]. Partly because of this lack of perfect shared information, current approaches [2, 3] model distributed group design using decision theory based on Game Theory to achieve a rational solution. If it is assumed that each designer possesses all the information necessary to design all subsystems, this becomes a group trade-off problem rather than one of design in a distributed environment. If the information passed between designers is limited, then Game Theory gives the most logical and practical approach to the problem. However, in situations in which it is possible to increase the amount of information that can be passed along, it can be shown that a designer can find a better performing design if the objective functionhas sufficient regularity.[1] This paper proposes an approach that allows designers to obtain a Pareto Optimal design in a decentralized, distributed design environment. In a Pareto Optimal design set, improvement in any design objective requires a sacrifice in at least one of the other design objectives. Thus, any design in this set can be considered an optimum assuming the priorities for the objectives are appropriate [4, 5]. The goal of this paper is to investigate a design method that converges to a set of Pareto Optimal designs while minimizing the required information flow. This will provide distributed, decentralized teams who design complex systems a process by which to arrive at optimal designs with slightly more information sharing than used in traditional approaches.

2 BACKGROUND

This work draws on several perspectives on design in a distributed environment, including decision theory and optimization.

2.1 Decision Theory

Decision theory attempts to identify the most rational design under a specific set of conditions by considering the information passed between designers. The resulting designs may differ depending on the type and quantity of information exchanged. Furthermore, the resulting designs will be rational, but not necessarily be optimal.

Decision-Based Design grows from decision theory and was first proposed by Lewis and Mistree [3, 6, 7]. Decision-Based Design has been widely adopted in a broad range of design research [8-11]. In Decision-Based Design, it is assumed that the main activity of a designer is to make decisions and each designer's decisions are determined by optimizing individual objectives within a certain group using tools from Game Theory. Because each designer's decisions will be influenced by every other designer's decisions, the final design depends on how much information is communicated among the designers. Lewis and Mistree represent this information sharing using Game Theory [12, 13]. They formulate this design problem for the case involving two designers as the game given by:

$$
\begin{array}{cc}
\textit{Designer } 1 & \textit{Designer } 2 \\
\text{Minimize} & \text{Minimize} \\
F_1(x_1, x_{2c}) - \{F_1^1, F_1^2, \ldots, F_1^p\} & F_2(x_{1c}, x_2) - \{F_2^1, F_2^2, \ldots, F_2^q\} \\
\text{subject to} & \text{subject to} \\
g_{j_1}^1(x_1, x_{2c}) \le 0, j_1 = 1, \cdots, m_1 & g_{j_2}^2(x_{1c}, x_2) \le 0, j_2 = 1, \cdots, m_2 \\
g_{h_1}^1(x_1, x_{2c}) = 0, h_1 = 1, \cdots, l_1 & g_{h_2}^2(x_{1c}, x_2) = 0, h_2 = 1, \cdots, l_2 \\
x_{1L} \le x_1 \le x_{1U} & x_{2L} \le x_2 \le x_{2U}
\end{array}
\qquad (1)
$$

Mathematically, this states that the purpose of this design scenario is to minimize the objective functions F_i for each different subsystem under given constraints where x_i represents design variables controlled by designer i and x_{ic} describes a non-local variable that is determined by other designers (subscript $(\cdot)_c$ means constraint). In this game theory representation of distributed design, several information-sharing protocols can be used in order to reach an agreement, each resulting in a

[1]Continuity of the second derivative is needed to locally approximate an objective function using a quadratic function. Without a quadratic approximation, the information sharing scenario will be similar to a cooperative protocol rather than a noncooperative protocol.

different rational/optimal set of designs. The following are possible protocols of information sharing among designers.

2.1.1 Cooperative protocol

In a completely cooperative protocol, all information about a subsystem is shared among all designers. Each designer has sufficient information about all subsystems to work together with others to find a set of Pareto Optimal solutions. Thus, this protocol allows a design team to select among the best possible design performances of all the protocols. Lewis mentions that such a *systems thinking* approach allows members in a design team to focus on cooperation through a common vision. Even though this protocol is ideal for the organization, it is rarely realistic in a distributed environment in part because of communication and logistics challenges. It becomes even more impractical to share all information when subsystems are technologically complex or disparate.

2.1.2 Noncooperative protocol

A noncooperative protocol occurs when there is imperfect information sharing between subsystems. This means that different amounts of information are shared between subsystems. By formulating a design problem from a Game Theory perspective, the objective of the design of a system is to achieve a Nash Equilibrium. A Nash Equilibrium represents a point in design space in which every designer cannot improve his design without breaking constraints given by other subsystems. The constraints imposed by other subsystems represent the limitation in information sharing. As more information is shared between subsystems, there will be fewer constraints imposed by one subsystem on the other subsystems. A Nash Equilibrium solution may not be in a Pareto Set, but it is the most rational solution given the lack of information between subsystems.[2]

2.1.3 Leader/Follower protocol

In a leader/follower protocol, a system is designed sequentially. The first designer designs his subsystem completely, then passes the subsystem design to the next subsystem designer. The final solution of a leader/follower protocol is called a Stackelberg solution and generally differs from a Nash solution. This approach is challenging because it requires the first designer to have some reasonable assumption about the other subsystem designs which is rare in large, complex systems.

2.2 Optimization Perspective

Design in a distributed environment can also be addressed from an optimization point of view. Mathematically, engineering design may be formalized as a global multi-objective optimization problem. In this formulation, the best design found by optimization will not necessarily be one particular design, but a set of designs in a Pareto Frontier. Unlike results found using decision theory, this set should depend only on the design objectives and design constraints, not the information shared or the rationality of the decision-makers.

One common approach to accomplishing this is Multidisciplinary Design Optimizaion (MDO) [14-16]. MDO was first formulated by Sobieszczanski-Sobieski as Concurrent Subspace Optimization (CSSO) [17]. Other classes of MDO developed later include Collaborative Optimization (CO) [18], Bi-Level integrated system synthesis (BLISS) [19], Multiple-Discipline-Feasible (MDF), Individual-Discipline-Feasible (IDF), All-at-once (AAO), and Multidisciplinary Optimization based on Independent Subspaces (MDOIS) [14, 20]. These can be broadly classified into hierchical and non-hierchical methods depending on the role of the system and subsystems. The main difference between MDO and the approach described in this paper is that this strategy does not require a human system designer as a facilitator. It is often the case that subsystems can interact with each other without the guidance of a system designer, especially in the case of two highly coupled systems.

The main question explored in this paper is: can a decentralized design team share less information than required in a cooperative protocol and still obtain a Pareto optimal set of design results? If so, this

[2] This is a rational solution rather than an optimal solution because there could be a design such that it is better than a rational solution for every design objective. However, if there is too little information shared between one subsystem and another, there will be no deterministic method to find these optimal solutions.

will allow designers of complex systems to find optimal designs with affordable and feasible information sharing.

3 MATHEMATICAL FORMULATION OF A DISTRIBUTED DESIGN PROBLEM

This paper focuses on an adapted version of a well known distributed design problem first posed by Lewis [13]. This problem was chosen because the original formulation is well scoped and illustrative. The analysis of the design problem will be limited to two players who are optimizing two different objective functions F_1 and F_2, depending upon the variables $\mathbf{x}$. These objectives represent different design goals given by performance, utility, or preference variables. Furthermore, each player is constrained to be in some set U_i. Because each player passes sets of constraints, this approach is also similar to A. C. Ward's Set Based Concurrent Engineering [21, 22] in which sets of intervals for feasible designs are passed iteratively between design groups to converge on a "good" design. In this case, quadratic constraints are exchanged rather than feasible intervals.

The problem is formally given by:

$$\begin{array}{ll} \textit{Designer } 1 & \textit{Designer } 2 \\ \min\limits_{x_1,x_2} F_1(x_1,x_2) & \min\limits_{x_1,x_2} F_2(x_1,x_2) \\ x \in \mathsf{U}_1 & x \in \mathsf{U}_2 \end{array} \tag{2}$$

The procedure necessary to solve the above optimization problem will be determined by modifying the algorithm that the following equation is based on the ε-constraint method [23, 24] and the method of equality constraints [25].

Designer 1 starts from a point in $(x_1^{(0,0)}, x_2^{(0,0)})$ in design space and minimizes his objective function $F_1(x_1,x_2)$ is constrained to the intersection of the sets U_1 and $\mathsf{A}_1^{(1)}$ where $\mathsf{A}_1^{(1)} = \{x \mid F_2(x_1,x_2) = F_2(x_1^{(0,0)}, x_2^{(0,0)})\}$. He will chose a minimization point $(x_1^{(1,0)}, x_2^{(1,0)})$ and pass it to the Designer 2 along with the $\mathsf{A}_2^{(1)} = \{x \mid F_1(x_1,x_2) = F_1(x_1^{(1,0)}, x_2^{(1,0)})\}$. Designer 2 will then minimize the objective function $F_2(x_1,x_2)$ constrained to the intersection of the sets U_2 and $\mathsf{A}_2^{(1)}$. Designer 2 will chose a minimization point $(x_1^{(1,1)}, x_2^{(1,1)})$ and pass it to the Designer 2 along with the $\mathsf{A}_1^{(2)}$. Designer 1 and 2 will then iterate until they reach a convergent design.

3.1 Approximation of the Isosurface

The goal of this paper is to formulate a design method that minimizes information flow between subsystems, but still converges to a set of Pareto Optimal designs. Thus, an important issue is the *form* of the information that is passed. In principle, passing a manifold[3] requires less information sharing between subsystems than in a cooperative protocol, but it may still require significant time and effort to exchange a complete manifold atlas between subsystems due to design environment and/or complexity of the manifold. To simplify the algorithm, the manifold $\mathsf{A} = \{\mathbf{x} \mid F(\mathbf{x}) = F(x_0)\}$ is approximated quadratically as:

$$\mathsf{A} \approx \{\mathbf{x} \mid \nabla F(\mathbf{x}_0) \times (\mathbf{x} - \mathbf{x}_0) + (\mathbf{x} - \mathbf{x}_0) \times \mathrm{He}_F(\mathbf{x}_0)(\mathbf{x} - \mathbf{x}_0) = 0\} \tag{3}$$

Both the gradient and Hessian can be numerically computed and passed between subsystems much more efficiently than a complete manifold for most cases. Because the algorithm is iterative, the error

[3]In this case, a manifold represents a contour of the desired objective value in design variable space.

due to approximation diminishes as a design converges. Therefore, the manifold constraint will be satisfied in a convergent design.

Because a quadratic approximation is used for the manifold, this approach is similar to sequential quadratic constrained quadratic programming (SQCQP) [26] for each optimization step. In a mathematically rigorous way, SQCQP shows that quadratic approximation of the nonlinear constraint can still achieve global optimality as long as the constraints and objectives have enough regularity. Here, unlike SQCQP, the objective function does not need to be approximated as a quadratic function. Each designer considers the objective functions of other designers as quadratic constraints.

One characteristic of Lewis and Mistree's formulation is that the convergence and stability of the underlying algorithm is not guaranteed [8]. In this paper's approach, convergence criteria are expressed more rigorously. A bounded, monotonic sequence (a contraction) always has a unique point of convergence, thus when designers pass a complete manifold to one another as an epsilon constraint, each subsystem's set of objectives will always be contracted, guaranteeing the convergence in the full manifold case. If the information each designer passes is approximated by a constraining manifold, an approximation error will be introduced. It is clear that if the erroneous increase in the objective value caused by this approximation error is smaller compared to each objective's improvement for every iteration, then the sequence will also be monotonically decreasing. Thus, the key to guaranteeing convergence lies in the accuracy of the approximation. This also implies that if oscillation occurs, the designer can reduce the "step" size by bounding the amount that a design variable can change.

4 CASE EXAMPLES

4.1 Case Study 1: Pressure Vessel

This first example examines the distributed design of the pressured vessel from Lewis and et al. [12]. It assumes that one designer is responsible for deciding the weight of the vessel and another designer is responsible for selecting its volume. Each designer's design objective and constraints are given as follows

Design Problem for Volume

Maximize: $V(R,L)=\frac{4}{3}\pi R^3+\pi R^2 L$

Design Variables: R and L

Stress Constraints: $\sigma_{circ}=\frac{PR}{T}\le S_t$

Geometric Constraints:

$$5T-R\le 0$$
$$R+T-40\le 0$$
$$L+2R+2T-150\le 0$$

Size Constraints:

$$0.1\le R\le 36$$
$$0.1\le L\le 140$$

Design Problem for Weight

Minimize: $W(R,L,T)=\rho\left[\frac{4}{3}\pi(R+T)^3+\pi(R+T)^2L-\frac{4}{3}\pi R^3-\pi R^2L\right]$

Design Variable: R, L, T [4]

[4]For a Game Theoretic Approach, T is the only design variable for Weight

Stress Constraints: $\sigma_{circ} = \frac{PR}{T} \leq S_t$

Geometric Constraints:

$$5T - R \leq 0$$

$$R + T - 40 \leq 0$$

$$L + 2R + 2T - 150 \leq 0$$

Size Constraints: [5]

$$0.1 \leq R \leq 36$$

$$0.1 \leq L \leq 140$$

$$0.1 \leq T \leq 6$$

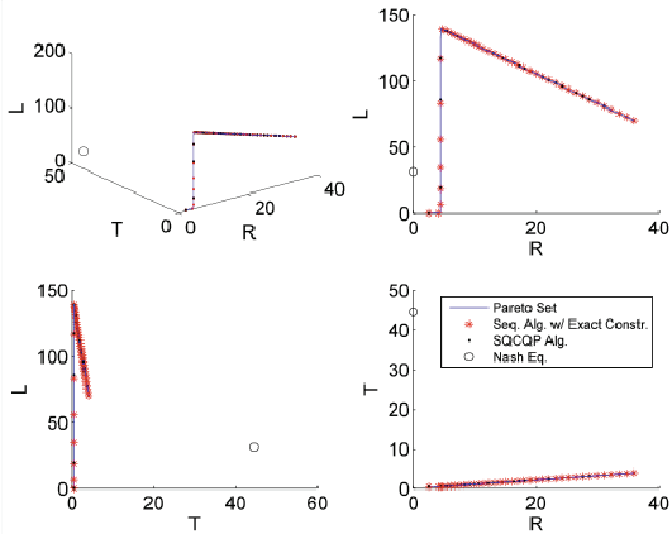

Figure 1: Pareto Frontier in Design Space for Pressure Vessel

The results of the new algorithm applied to this example are shown in Figures 1 and 2. The circle-shaped points on the figures represent the Nash solution using a Game Theoretic approach. There is only one Nash solution for this example, which means that the Nash solution is independent of the starting point as long as it starts from the attractor set. In this algorithm, if the Pareto Set contains more than one point, each starting point could end up being any point in the Pareto Set. In order to determine most of the points in the Pareto Set, the two extrema are located by optimizing each objective separately. Then the process starts from the affine combination points of these two extrema in design variable space. The result shows that these points converge along the Pareto Set ranging from one extrema to the other. Thus, the whole the Pareto Set can be estimated from this approach. Finally, this result shows that passing the quadratic approximation for each designer's objective function is sufficient to obtain Pareto Optimal design solutions. Note that the accuracy of a Pareto Frontier must be balanced with information transfer burden.

[5] For a Game Theoretic Approach, a constraint for R and L won't be needed.

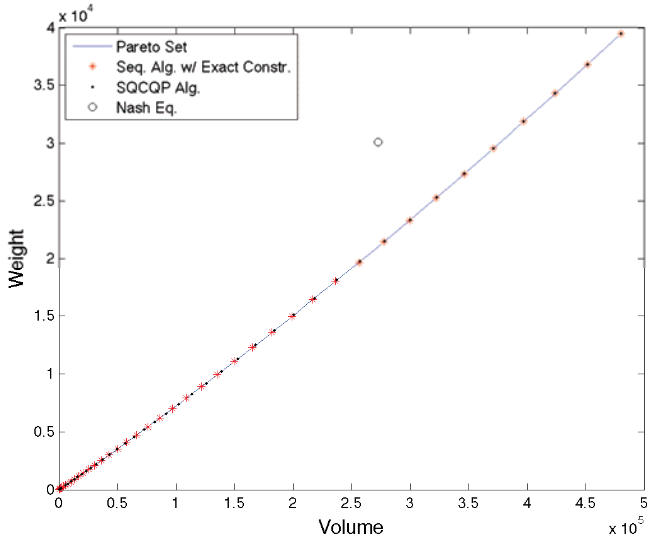

Figure 2: Pareto Frontier in Performance Space for Pressure Vessel

4.2 Case Study 2: Aircraft Design

In this second example, the method is applied to the aircraft example given in Lewis, et al.'s work [3] and Mistree, et al.'s NASA report [27]. As before, there are two designers. One designer optimizes the weight of the aircraft and the other optimizes its aerodynamics. The mathematical details of the design scenarios are not included here due to space considerations, and may be downloaded from [28]. The key equations are the same as described in Lewis' previous work, but one constant b_t and weighting for objectives are different from their choices.

The results of new algorithm applied to this example are shown in Figure 3. The figure includes a set of Nash Equilibria, Pareto Optimal points, and convergence points from this new algorithm. In this example, there is set of Nash Equilibrium points rather than a single equilibrium point. Note that sections of the Nash solution are close to the Pareto Optimal solution, but other sections of it are not. The Pareto Optimal designs are obtained using Simulated Annealing. Using this new algorithm, the Pareto Optimal Set can be obtained. Thus, even for this complex example, if each designer can pass a quadratic approximation of his or her objective function, it is sufficient for a system to reach Pareto Optimality.

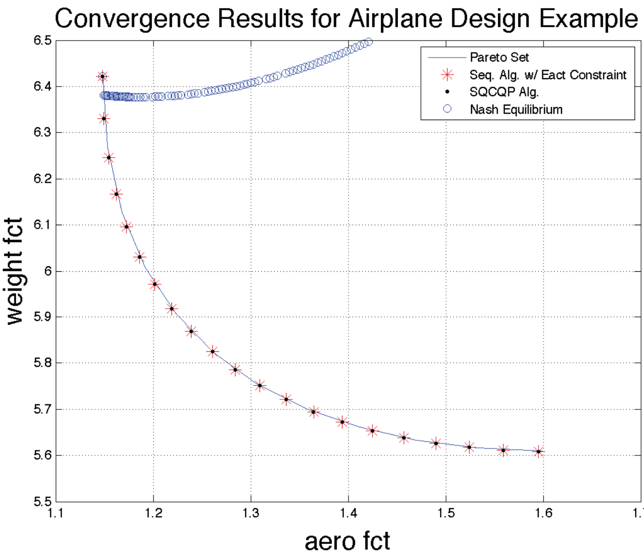

Figure 3: Pareto Frontier in Performance Space for Aircraft Design

5 CONCLUSIONS

In a decentralized environment, there are few formal design techniques for obtaining rational solutions. Lewis and others have developed an effective, widely used Game Theoretic approach based on Decision Theory. When there is limited information sharing between subsystems, this approach is very reliable as it will find a Nash Equilibrium design using minimum information sharing. The main question explored in this paper is: if a subsystem is able to share more information than in a non-cooperative environment, but less than in a fully cooperative environment, is there an approach that can obtain a Pareto Optimal set in a decentralized environment? In this paper, designers were permitted to share a quadratic approximation to the each subsystem's objective function at each iteration. In two examples, one for a Pressure Vessel design and the other for Airplane Design, Pareto sets were obtained by passing along quadratic approximations. Thus, this paper demonstrates two cases in which the Pareto Optimal Set of design points can be found in decentralized design environment by sharing a quadratic approximation to the objective functions as long as each subsystem's objective function is well behaved.

6 FUTURE WORK

As engineering systems become increasingly complex, the design process will likely become more decentralized as more varied and specialized expertise will be needed to develop these subsystems. Thus, it is important to explore and understand different design techniques that may be applied in such environments. This paper demonstrates a method for obtaining Pareto Sets in a distributed design scenario with limited information sharing when an objective function is smooth enough. Some possible extensions of this work include:

- Multiple subsystems.The cases presented in this paper assume only two designers must interact. Increasing the number of subsystems can lead to a drastic increase in the amount of information sharing required. How does this technique scale with an increasing number of subsystems?
- Sequences of designs. Future work should consider the whole sequence of designs rather than only the design from the previous iteration. In real world design, teams usually consider all previous designs they explored.
- Subsystem specific constraints. For the sake of simplicity, examples shown in this paper have the same constraints for all subsystems. Future work will examine more realistic ways to represent constraints as they are passed between subsystems.

ACKNOWLEDGEMENT

The work described in this paper was supported in part by the National Science Foundation under Award DMI-0547629. The opinions, findings, conclusions and recommendations expressed are those of the authors and do not necessarily reflect the views of the sponsors.

REFERENCES

[1] Allen, T. J., 1984, Managing the Flow of Technology: Technology Transfer and the Dissemination of Technological Information, The MIT Press.
[2] Lewis, K., 1996, "An Algorithm for Integrated Subsystem Embodiment and System Synthesis," Georiga Institute of Technology.
[3] Lewis, K., and Mistree, F., 1997, "Modeling the Interactions in Multidisciplinary Design: A Game Theoretic Approach," AIAA Journal of Aircraft, 35(pp. 1387-1397.
[4] Scott, M. J., 1999, "Formalizing Negotiation in Engineering Design," California Institute of Technology, Pasadena, CA.
[5] Scott, M. J., and Antonsson, E. K., 2000, "Arrow's Theorem and Engineering Design Decision Making," 11(4), pp. 218-228.
[6] Vincent, T., 1983, "Game Theory as a Design Tool," ASME Journal of Mechanisms, Transmissions, and Automation in Design, 105(pp. 165-170.
[7] Whitfield, R. I., Duffy, A. H. B., Coates, G., and Hills, B., 2002, "Distributed Design Coordination," Research in Engineering Design, 13(pp. 243-252.
[8] Chanron, V., and Lewis, K., 2004, "Convergence and Stability in Distributed Design of Large Systems," Salt Lake City, Utah.
[9] Chanron, V., Lewis, K., Murase, Y., Izui, K., Nishiwaki, S., and Yoshimura, M., 2005, "Handling Multiple Objectives in Decentralized Design," Long Beach, CA.
[10] Xiao, A., Zheng, S., Allen, J. K., Rosen, D. W., and Mistree, F., 2005, "Collaborative Multidisciplinary Decision Making Using Game Theory and Design Capability Indices.," Research in Engineering Design, 16(1-2), pp. 57-72.
[11] Fernandez, M. G., Panchal, J. H., Allen, J. K., and Mistree, F., 2005, "An Interactions Protocol for Collaborative Decision Making - Concise Interactions and Effective Management of Shared Design Spaces," Long Beach, CA USA.
[12] Lewis, K., and Mistree, F., 1999, "Collaborative, Sequential, and Isolated Decisions in Design," ASME Journal of Mechanical Design, 120(pp. 643-652.
[13] Lewis, K., and Mistree, F., 2001, "Modeling Subsystem Interactions: A Game Theoretic Approach," Journal of Design, Manufacturing, and Automation, 1(pp. 17-36.
[14] Park, G. J., 2007, Analytic Methods in Design Practice, Springer, Germany.
[15] Yi, S. I., Shin, J. K., and Park, G. J., 2008, "Comparison of Mdo Methods with Mathematical Examples," Structural Multidisciplinary Optimization, 35(pp. 391-402.
[16] Cramer, E. J., Dennis, J. E., Frank, P. D., Lewis, R. M., and Shubin, G. R., 1993, "Problem Formulation for Multidisciplinary Optimization," Center for Research on Parallel Computation, Rice University.
[17] Sobieszczanski-Sobieski, J., 1988, "Optimization by Decomposition: A Step from Hierarchic to Non-Hierarchic Systems," NASA
[18] Tappeta, R. V., and Renaud, J. E., 1997, "Multiobjective Collaborative Optimization," Journal of Mechanical Design, 119(Sept.), pp. 403-411.
[19] Sobieszczanski-Sobieski, J., Agte, J. S., and Sandusky, R. R., 1998, "Bi-Level Integrated System Synthesis (Bliss)," Proc. AIAA/USAF/NASA/ISSMO symposium on multidisciplinary analysis and optimization, AIAA-98-4916, pp. 1543-1557.
[20] Shin, M. K., and Park, G. J., 2005, "Multidisciplinary Design Optimization Based on Independent Subspaces," INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN ENGINEERING, 64(pp. 599–617.
[21] Ward, A. C., 1989, "A Theory of Quantitative Inference for Artifact Sets, Applied to a Mechanical Design Compiler," MIT.
[22] Ward, A. C., Lozano-Pérez, T., and Seering, W. P., 1990, "Extending the Constraint Propagation of Intervals," Artificial Intelligence in Engineering Design and Manufacturing, 4(1), pp. 47-54.

[23] Marglin, S., 1967, Public Investment Criteria, MIT Press, Cambridge, MA.
[24] Haimes, Y., 1973, "Integrated System Identification and Optimization," Control and Dynamic Systems: Advances in Theory and Applications, 10(pp. 435-518.
[25] Lin, J. I., 1976, "Multiple-Objective Problems: Pareto-Optimal Solutions by Method of Proper Equality Constrains," IEEE Transactions on Automatic Control, AC-21(5), pp. 641-650.
[26] Jian, J. B., Ju, Q. J., Tang, C. M., and Zheng, H. Y., 2007, "A Sequential Quadratically Constrained Quadratic Programming Method of Feasible Directions," Applied Mathematics and Optimization, 56(56), pp. 343-363.
[27] Mistree, F., Marinopoulos, S., Jackson, D. M., and Shupe, J. A., 1988, "The Design of Aircraft Using the Decision Support Problem Technique," NASA.
[28] Honda, T., Ciucci, F., and Yang, M. C., http://web.mit.edu/ideation/aero_example.pdf