

Modeling Interactions in Multidisciplinary Design: A Game Theoretic Approach

Kemper Lewis*

State University of New York at Buffalo, Buffalo, New York 14260
and

Farrokh Mistree†

Georgia Institute of Technology, Atlanta, Georgia 30332-0405

The development, implementation, and application of approaches to modeling the interactions in multidisciplinary design is illustrated. Given that the design of complex systems involves multiple disciplinary design teams and their associated analysis and synthesis tools, the task is to model the real interactions among the designers and their tools in order to predict the resulting design. Our approach to this problem is to abstract the interactions in multidisciplinary design as a sequence of games among a set of players, which are embodied by the design teams and their computer-based tools. The developments are applied to a subsonic passenger aircraft design case study to illustrate the rich insights and results that can be generated by exercising different realistic protocols between disciplinary players in modern design processes.

Nomenclature

AR	= aspect ratio
$Ld(t, c, l)$	= lift-to-drag ratio on takeoff, cruise, and landing
R_f	= fuel balance ratio
R_{fr}	= fuel ratio required
S	= wing area
s	= state variable describing behavior of a system, functions of the design variables, and either inputs or outputs
s_L	= takeoff field length
T_i	= installed thrust
U	= useful load
V_{br}	= best range velocity
W_{to}	= takeoff weight
X_i	= control vector, under designer i 's control
x	= design variable

I. Introduction

ASSUME that a complex system such as an aircraft has been decomposed into disciplinary subsystems such as propulsion and structures. Further, assume that the propulsion designer controls X_p (design and state variables), the variables X_s are controlled by the structures designer (which the propulsion designer has no control), and f_i are respective propulsion objective functions. We assert that the following standard multiobjective formulation, representing the propulsion subsystem,

$$\underset{X_p \in f(X_s) \subset \mathbb{R}^n}{\text{minimize}} \quad f(X_p, X_s) = \{f_1(X_p, X_s), \dots, f_r(X_p, X_s)\} \quad (1)$$

is the typical starting point for much of the current research and practice in systems modeling and applied optimization. And yet in specific design instances, this assertion should be boldly challenged. For example, inasmuch as the propulsion designer only controls X_p and the structures designer controls X_s , how is X_s chosen in the propulsion design? Can the propulsion designer assume that the

structural designer will always select the vector that is most advantageous to the propulsion design? If not, how should the propulsion designer respond to this conflict? This scenario describes a two-player strategic game where one player controls X_p and the other player controls X_s , and where X_s represents all decisions that are outside the scope of the designer controlling X_p (Refs. 1–3).

We model these types of strategic relationships using game theoretical principles. A game consists of multiple decision makers who each control a specified subset of design variables and who each seek to minimize their own cost functions subject to their individual constraints.⁴ In a game, these multiple decision makers are required to select single decision strategies to optimize their set of rewards. However, each player's reward depends on the other player's strategies, i.e., it depends on decision variables that are controlled by other players. The fact that players lack control over all decision variables affecting their rewards is what makes a game and what distinguishes it from an optimization problem.

The theoretical and mathematical foundations of games are used to abstract the processes required to design a complex system as a game. The players in this game are defined as the disciplinary design teams and their associated analysis/synthesis tools. There are various game protocols depending on the level of cooperation and behavior of the players. Certain protocols lend themselves nicely to modeling interactions in design, namely, the cooperative or Pareto formulation when the players work together and communicate, the Nash or noncooperative formulation when the players act in their own self-interest, and the Stackelberg or leader/follower formulation when one player dominates another. In the next section we provide the mathematical background of each protocol.

II. Frame of Reference

In this section, the mathematics supporting the fundamental theoretical constructs used in this paper are presented. We begin with our domain-independent decision model, the compromise decision support problem (DSP). The three fundamental game theoretic protocols used in this work are also introduced. The theory and mathematics behind the protocols are integrated with the compromise DSP and implemented and applied in a design context in Secs. III and IV. In Sec. V, the resulting implications in modern design processes are explored.

Decision Model

We characterize the decisions made by each player using a compromise DSP. The mathematical form of a compromise DSP of a player i follows.

Received July 18, 1996; revision received Nov. 8, 1996; accepted for publication April 19, 1997. Copyright © 1997 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved.

*Assistant Professor, Department of Mechanical and Aerospace Engineering, Member AIAA.

†Professor, Systems Realization Laboratory, Woodruff School of Mechanical Engineering, E-mail: farrokh.mistree@me.gatech.edu. Senior Member AIAA.

Given an alternative to be improved and assumptions used to model the domain of interest, where

- n = number of local design variables
- \mathbf{X} = vector of local control variables of player i , $\{x_i, s_i\}$
- q = number of inequality constraints
- $p + q$ = number of system constraints
- m = number of system goals
- $g_i(\mathbf{X})$ = system constraint functions
- $f_k(\mathbf{X})$ = function of deviation variables to be minimized at priority level k for the preemptive case

find the system design variables x_i , $i = 1, \dots, n$, and deviation variables d_i^- and d_i^+ , $i = 1, \dots, m$, where x is an independent variable over which a designer has control.

Satisfy the system constraints (linear, nonlinear)

$$g_i(\mathbf{X}) = 0, \quad i = 1, \dots, p$$

$$g_i(\mathbf{X}) \geq 0, \quad i = p + 1, \dots, p + q$$

The system goals (linear, nonlinear) are

$$A_i(\mathbf{X}) + d_i^- - d_i^+ = G_i, \quad i = 1, \dots, m$$

where the bounds are

$$x_i^{\min} \leq x_i \leq x_i^{\max}, \quad i = 1, \dots, n$$

$$d_i^-, d_i^+ \geq 0, \quad d_i^- \cdot d_i^+ = 0, \quad i = 1, \dots, m$$

Minimize the deviation function

$$f = [f_1(d_i^-, d_i^+), \dots, f_k(d_i^-, d_i^+)], \quad i = 1, \dots, m$$

The compromise DSP is a multiobjective decision model, which is a hybrid formulation based on mathematical programming and goal programming.⁵ It is used to determine the values of design variables that satisfy a set of constraints while achieving a set of conflicting goals as well as possible. The system descriptors, namely, design and deviation variables, system constraints, system goals, bounds, and the deviation function are described in detail in Ref. 5. We use deviation variables in the same manner as in goal programming where they are used to measure the difference between the aspiration level for a goal and the current attainment of the goal. The deviation function is a function of the deviation variables. In the compromise DSP, the aim is to minimize the difference between that which is desired and that which can be achieved. Therefore, the aim is to minimize the deviation function. The closer you are to attaining the system goals, the better off you are and the smaller the deviation function becomes. The strategy of a player, embodied by the compromise DSP, is to minimize the deviation function. The compromise DSP is solved using the adaptive linear programming algorithm⁵ if only continuous variables are used. If discrete and continuous variables are used, the compromise DSP is solved using the foraging-directed adaptive linear programming algorithm.⁶ Both algorithms are part of the decision support software called decision support in designing engineering systems (DSIDES®) (Ref. 7).

Game Theoretical Protocols

For this discussion, assume there are two players who each control X_1 and X_2 , respectively, and who try to minimize their own deviation functions f_1 and f_2 , respectively. When $\min f$ is used in the discussion, it succinctly represents the solution of a compromise DSP. The term $f_i(X_1, X_2)$ represents the deviation function of player i for values of the control variables X_1 and X_2 of each player. Three protocols are used in this paper, namely, the Pareto, Nash, and Stackelberg protocols. A brief description of each follows.

Pareto Cooperation

The mathematics of this protocol follows. A pair (x_1^p, x_2^p) is Pareto optimal if no other pair (x_1, x_2) exists such that

$$f_1(x_1, x_2) < f_1(x_1^p, x_2^p) \quad \text{and} \quad f_2(x_1, x_2) < f_2(x_1^p, x_2^p) \quad (2)$$

This is interpreted as follows: no player can do better without hurting another player. Both players cannot simultaneously improve their standing.

The practical implication of this is full cooperation implies that each player has exact representations of the necessary nonlocal state variables. Approximate cooperation implies that each player only has an approximation of the necessary nonlocal state variables.

Nash Noncooperation

The mathematics of this protocol follows. Player 1 constructs the solution set as a function of required (but unknown) nonlocal design and state variables, e.g.,

$$\mathbf{X}_1 = \{C^1, C^2, f(\mathbf{X}_2), C^3, f(\mathbf{X}_2)\} \quad (3)$$

based on the strategy to minimize their own deviation function, $\min f_1(\mathbf{X}_1, \mathbf{X}_2)$. In the example of Eq. (3), the rational reaction set (RRS) of player 1 is given as

$$\mathbf{X}_1^{\text{RRS}}(\mathbf{X}_2) = \{f(\mathbf{X}_2), f(\mathbf{X}_2)\} \quad (4)$$

where the C^i are numerical values for variables that do not depend on nonlocal information. For the variables that do depend on nonlocal variables (the third and fifth ones in this case), a function of the form $f(\mathbf{X}_2)$ is found. The RRS consists of the subset of variables that depend on nonlocal information.

Succinctly, the RRS of player 1 is defined as

$$\mathbf{X}_1^{\text{RRS}}(\mathbf{X}_2) := \{\mathbf{X}_1^{\text{RRS}} \in \mathbf{X}_1 \text{ such that } f_1(\mathbf{X}_1^{\text{RRS}}, \mathbf{X}_2) = \min f_1(\mathbf{X}_1, \mathbf{X}_2)\} \quad (5)$$

A strategy pair (X_1^N, X_2^N) is a Nash solution for the two-player example if

$$(X_1^N, X_2^N) \in \mathbf{X}_1^{\text{RRS}} \cap \mathbf{X}_2^{\text{RRS}} \quad (6)$$

This is interpreted as follows: each player poses their solution set based on unknown information from the other players. If these solution sets intersect, then there exists a solution. The RRS of a player embodies how a player would react (what solution they would choose) for any given set of variables from the other players.

The practical implication of this is players must make decisions in isolation due to organizational barriers, time schedules, or geographical constraints.

Stackelberg Leader/Follower

Here the mathematics are

$$\text{minimize } f_1(\mathbf{X}_1, \mathbf{X}_2) \quad (7)$$

$$\text{satisfying } \mathbf{X}_2 \in \mathbf{X}_2^{\text{RRS}}(\mathbf{X}_1)$$

This is interpreted as follows: the leader makes the decision first, based on the assumption that the follower behaves rationally. Thus, the leader knows, in effect, the follower's RRS. The follower then makes the decision without having to assume anything about the leader.

The practical implication of this is sequential decision making, still common and unavoidable in multidisciplinary design.

These protocols are implemented within and between multiple compromise DSPs to model multiple decisions, which may or may not be coupled. Development of the practical implementations for each protocol are presented in the next section.

III. Implementation

Case Study

The case study in this paper is the design of a 727-200 aircraft. This study is derived from Refs. 8 and 9. Two distinct players are identified, each with their own analysis and synthesis routines: the aerodynamics player responsible for the wing and fuselage lift characteristics and the weights player responsible for setting the thrust and takeoff weight through a fuel balance. In an earlier single-level version of the 727-200 template,^{8,9} the existing 727-200 design was reproduced. The model presented here is an updated version. The

primary difference is the inclusion of two coupled disciplinary problems, as opposed to the previously studied single problem. Simplified forms of each players' compromise DSP are given subsequently. The full form of the players' compromise DSPs is given in Ref. 10.

In the analytical model of the aerodynamics designer, there are three control variables required by the aerodynamics designer from the weights designer. These are the design variables x_W , that is, W_{to} and T_i , and the state variable s_W , that is, R_{fr} . In the analytical model of the weights player, there are five control variables required by the weights designer from the aerodynamics designer. These are the design variable x_A , that is, S , and the state variables s_A , that is, Ld_i , Ld_c , Ld_j , and V_{br} . What are the resulting designs when each discipline has the information they need, when they do not have the information, and when they are solved sequentially? It is the resolution of these coupling and coordination challenges that are of interest. The different game protocols studied that are used to model these scenarios are discussed.

Aerodynamics Player's Compromise DSP

Given the relevant constants, the important aerodynamic relationships and equations, and the expressions for local state variables s_A (outputs from aerodynamics discipline), find the aerodynamics design variables x_A , that is, S (in square feet), wing span b (in feet), and fuselage length l (in feet), and the deviation variables associated with the aerodynamic goals.

Satisfy the following aerodynamic constraints (nonlinear): aspect ratio ≤ 10.5 ; achievable climb gradient, landing ≥ 2.4 deg; achievable climb gradient, takeoff ≥ 2.7 deg; landing field ≤ 4500 ft; takeoff field ≤ 6500 ft; drag coefficient, landing, and takeoff ≤ 0.02 ; and drag coefficient in cruise ≤ 0.02 . Satisfy the following aerodynamic goals (nonlinear): achievable climb gradient, landing = 3.0 deg; achievable climb gradient, takeoff = 3.0 deg; landing field = 4500 ft; takeoff field = 4500 ft; and $AR = 10.5$ with the following bounds: $1200 \leq S \leq 2500$ ft, $105 \leq l \leq 150$ ft, and $85 \leq d \leq 140$ ft.

Minimize the deviation function

$$Z_A = \sum (d_{Ai}^- + d_{Ai}^+), \quad i = 1, \dots, 5 \quad (\text{five goals})$$

Weights Player's Compromise DSP

Given the relevant constants, the important fuel/weight relationships and equations, and the expressions for local state variables s_W (outputs from weights discipline), find the weight design variables x_W , that is, W_{to} (in pounds) and T_i (in pounds) and the deviation variables associated with the weight goals.

Satisfy the following weight constraints (nonlinear): useful load ≥ 0.3 ; fuel balance ≥ 1.0 ; achievable climb gradient, landing ≥ 2.4 deg; achievable climb gradient, takeoff ≥ 2.7 deg; landing field ≤ 4500 ft; and takeoff field ≤ 6500 ft. Satisfy the following weight goals (nonlinear): productivity index (PRI) = 270 (Ref. 11); useful load fraction = 0.5; fuel balance = 1.0; achievable climb gradient, landing = 3.0 deg; achievable climb gradient, takeoff = 3.0 deg; landing field = 4500 ft; and takeoff field = 4500 ft with the following bounds: $27,750 \leq T_i \leq 55,000$ lb, and $140,000 \leq W_{to} \leq 250,000$ lbs.

Minimize the deviation function

$$Z_W = \sum (d_{Wi}^- + d_{Wi}^+), \quad i = 1, \dots, 7 \quad (\text{seven goals})$$

Protocol Implementation

Two forms of cooperation are studied, namely, full and approximate cooperation.

Full Cooperation

Full cooperation is the classical game theory cooperative formulation (Pareto optimality). In our implementation of full cooperation both players solve their problems within an integrated computer environment and have access to full representations (equations) of any nonlocal information they need.

Approximate Cooperation

Full cooperation, where every player has complete information from the other players, is rarely achievable in modern design processes. Therefore, an approximate form of cooperation is asserted as being a more realistic representation of cooperation. In the approximate cooperative formulation, the players may not be working in an integrated environment but have access to first-order approximations of the nonlocal state variables they need:

$$s(x) \approx s^0 + \nabla s(x^0)(x - x^0) \quad (8)$$

This information is provided by a processor that constructs the first-order approximations at each solution iteration for each player and feeds them to the appropriate player.

In the noncooperative and leader/follower formulations, players must make assumptions about how the other players are going to make their decisions. The formal realization of these assumptions in game theory is the RRS. A player's RRS succinctly captures how the player will react (make their decisions) based on the (unknown) decisions another player may make.

Construction of the RRS

Each player must construct the RRS based on unknown information about the other players. In simple problems, exact RRSs can be found (Ref. 12). However, in complex, nonlinear problems, finding the exact RRSs is a difficult task because it involves finding a symbolic solution of a model as a function of the necessary but unknown variables of the other players. Therefore, to approximate the RRSs, we use the design of experiments (DOE) and response surface methodology (RSM). Specifically, the RRSs are approximated by a second-order response surface and are constructed by linking a statistical package NORMAN^{®13} with DSIDES.⁷ In effect, NORMAN acts as the experimental meta-controller, intelligently sampling a nonlocal design space, and then DSIDES solves the local model (compromise DSP) at each nonlocal sample point. This is illustrated in Fig. 1. At the left of the figure the nonlocal design space of player j is shown. By using NORMAN as the automated DOE package, certain points (as determined by the choice of experimental design) are sampled in this space and fed into the local compromise DSP of player i . At each point, the compromise DSP is solved using DSIDES.

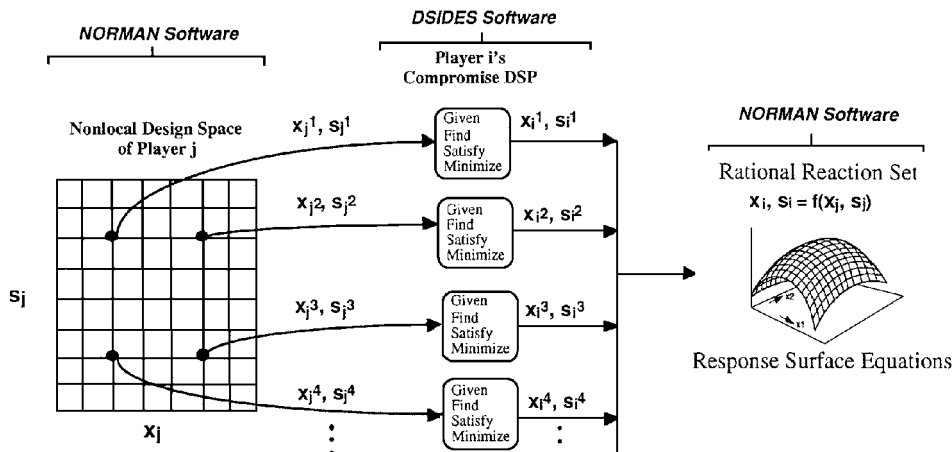


Fig. 1 Construction of player i 's RRS.

Using the resulting input and output pairs, e.g., $\{(x_j^1, s_j^1)\}$ and $\{(x_j^2, s_j^2)\}$, response surfaces are constructed in NORMAN to approximate the behavior of a local player as a function of the unknown nonlocal variables of another player. Because we are using second-order surfaces, the generic representation is

$$X_i^{\text{RRS}} = \{x_i, s_i\} = f(x_j, s_j) = A x_j + B s_j + C (x_j \times x_j) + D (s_j \times s_j) + E (x_j \times s_j) \quad (9)$$

where A, B, C, D , and E are numerical constants as determined in the RSM. We use the central composite design (CCD) as the experimental design, which uses a three-level design along with center and star points to sample the design space.¹⁴

In the aircraft study, the weights player needs five variables from the aerodynamics player. The variables needed by the weights player in this study are the design variable S and the state variables Ld_c, Ld_i, Ld_l , and V_{br} .

The design of experiments is set up based on the minimum and maximum values of these variables. The CCD design uses the following upper and lower bounds to generate the three-level design plus the additional star and center points: $1200 \leq S \leq 2500$, $5 \leq Ld_i \leq 17$, $12 \leq Ld_c \leq 20$, $8 \leq Ld_l \leq 18$, and $500 \leq V_{br} \leq 1000$. With five input variables and using a CCD, 43 experiments are run for various values of S, Ld_i, Ld_c, Ld_l , and V_{br} to approximate the weight player's RRS. An experiment consists of solving the weight player's compromise DSP at one combination of values of S, Ld_i, Ld_c, Ld_l , and V_{br} .

Conversely, the unknown parameters that the aerodynamics player needs from the weights player are the design variables W_{to} and T_i and the state variable R_{fr} . The CCD is set up based on the minimum and maximum values of the following parameters: $140,000 \leq W_{to} \leq 250,000$, $27,750 \leq T_i \leq 55,000$, and $0.2 \leq R_{fr} \leq 0.6$. Only three variables are needed by the aerodynamics player, as opposed to five variables that are needed by the weights player. With the decreased number of input variables for the aerodynamics player, the number of experiments needed to construct the RRS of the weights player decreases from 43 to 15 using the same order of response surface in a CCD design. An experiment in this case consists of solving the weight player's compromise DSP at one combination of values of W_{to}, T_i , and R_{fr} .

Noncooperation

Once the players have constructed their RRSs (or had them constructed), the noncooperative solution, if it exists, occurs at the intersection of the RRSs:

$$(X_A^N, X_W^N) = X_A^{\text{RRS}} \cap X_W^{\text{RRS}} \quad (10)$$

In this aircraft study, the RRSs of the two players consists of eight nonlinear equations with eight unknowns (eight coupled control

variables, five from aerodynamics and three from weights). Solving this system of equations results in multiple noncooperative solutions. In Sec. IV, only the best noncooperative solution is used for comparison purposes.

Leader/Follower

The leader/follower protocol models a sequential decision-making process among decision makers. The advantages and disadvantages of being the leader or follower are given in Table 1. The implementation of the two leader/follower scenarios in our study is discussed.

Aerodynamics as leader/weight as follower. This formulation corresponds to a design process in which the aerodynamics design team takes the lead in a design process and makes the decision first about the aircraft profiles based on an assumption that the other disciplines will make decisions rationally [in the form of the follower's RRS, see Eq. (7)]. Then the weights player, the follower, makes decisions based on the leader's solution. The leader, aerodynamics, has access to the follower's RRS ($W_{to}, T_i, R_{fr} \in X_W^{\text{RRS}}$), and the follower, the weight discipline, knows (but has to wait for) the leader's solution ($\{Ld_i, Ld_c, Ld_l, V_{br}, S\}$), and then uses that information in the model.

Weight as leader/aerodynamics as follower. This formulation corresponds to a design process in which designers first choose an engine configuration, based on an assumption that the other disciplines will behave rationally. Then, the other disciplines have to design according to the engine specification. This is fundamentally different from the formulation with aerodynamics as the leader. In this case, the leader, weight, has access to the follower's RRS ($\{Ld_i, Ld_c, Ld_l, V_{br}, S \in X_A^{\text{RRS}}\}$), and the follower, aerodynamics, knows (but has to wait for) the leader's solution ($\{W_{to}, T_i, R_{fr}\}$), and then uses that information in the model. The solution to each implementation of the leader/follower protocols as well as the cooperative and noncooperative protocols are discussed in Sec. IV.

IV. Results

In this section we compare the results of applying the game theoretical implementations of the preceding section to the aircraft study and to gain some insight into the role of each protocol in design. The configurations corresponding to the solutions of each protocol, along with the configuration of an existing 727-200 aircraft, are shown in Fig. 2. Again, only the best noncooperative solution is used in this section.

In Fig. 3 the deviation functions corresponding to the protocols are shown. Some interesting observations can be made from the results.

1) The best overall results occur, as expected, when cooperation exists among the players. The term overall is meant to imply that both players cumulatively do well. Whether full or approximate cooperation is exercised does not affect the result significantly. By using approximate representations, favorable results are obtained with less computational demands transfer than full cooperation, as the disciplinary analysis and synthesis are kept isolated.

2) Player aerodynamics does very well (same as it does in the cooperative formulations) when it is the leader in the leader/follower formulation, but at the expense of the weight player. Player weight as the leader in the leader/follower formulation actually does better than it does in the cooperative formulations, but at the expense of the aerodynamics player.

3) Even the best noncooperative solution is significantly worse for both players. Therefore, the best the players can hope for, if they have

Table 1 Advantages and disadvantages of being leader or follower

	Advantage	Disadvantage
Being the leader	Get to make your decision first	Have to make an assumption about how the follower will react
Being the follower	Do not have to make any assumptions	Are constrained by the leader's decision

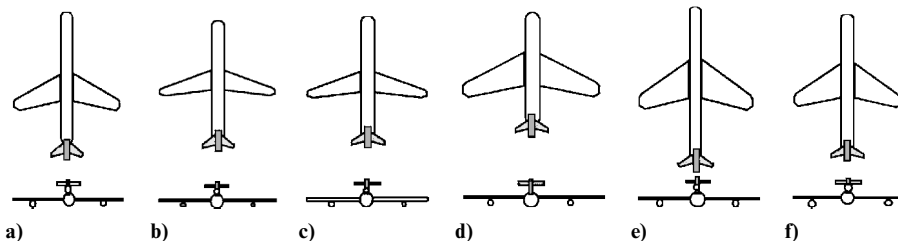
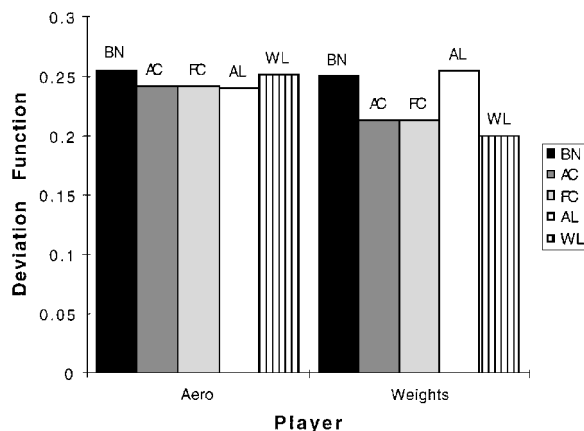


Fig. 2 Aircraft configurations corresponding to the protocols, approximately 1:1500 scale: a) best noncooperative solution (BN), b) approximate cooperative (AC), c) full cooperative (FC), d) aerodynamics as leader (AL), e) weight as leader (WL), and f) 727-200.

Table 2 State variables of various solutions

State variable	BN ^a	FC ^a	AC ^a	AL ^a	WL ^a
U	0.48	0.49	0.49	0.46	0.48
PRI	158	177	177	155	174
Ld_i	13.1	15.1	15.0	16.0	12.9
Ld_t	10.0	11.8	11.8	12.6	9.9
Ld_c	18.0	19.9	19.9	20.7	18.3
AR	7.2	9.66	9.65	9.89	7.91
s_L , ft	3940	4490	4500	4310	4470

^aSee Fig. 2.**Fig. 3** Protocol results as compared to an existing design; see Fig. 2.

to make their decisions in isolation, is an inferior solution. Although many better solutions exist, the players are not able to find them because of the lack of collaboration and the limiting assumptions that must be made.

4) The existing 727-200 values do not match with any one protocol exercised in this work. This is not unexpected, as the model used in this work is a simplified model of a complete aircraft model. The simplified model is used to illustrate the rich insights and benefits that could be generated when the behavior of the disciplines is modeled as a strategic interaction using game theory. Note that, if one aircraft had to be chosen as being closest to the 727-200 aircraft, it would be the aircraft from the leader/follower formulation with weights as the leader (Figs. 2e and 2f). The significant differences are the values of the wing span and fuselage length, which are slightly larger in the weight as leader aircraft, and the installed thrust, which is slightly larger in the 727-200.

Further insight into the different aircraft can be gained by exploring the values of the state variables that describe the behavior of the aircraft. In Table 2, the significant state variables for the aircraft are given for each protocol solution. From the system requirements and compromise DSPs presented in Sec. III, the desired values of the state variables for each player are as follows: weight player wants $U = 0.5$ and PRI to be maximized; aerodynamics player wants Ld_i , Ld_t , Ld_c to be maximized and $AR = 10.5$; and both players want $s_L = 4500$ ft. Each state variable is investigated for the various protocol cases.

Useful Load

The useful load fraction for each player in the FC, AC, and WL cases (see Fig. 2) are close to 0.50 but in the AL solution is only 0.46. This is intuitive because, in the AL case, the aerodynamics player is not concerned with U and, subsequently, does not leave the weight player enough freedom to improve U . In the noncooperative protocol, the assumptions made by the players result in an acceptable useful load fraction value, but this is rare.

PRI

In the FC, AC, and WL cases, the productivity index is the maximum, whereas in the others it is significantly less. When weights is the leader (WL), the PRI is high because the weight player controls PRI and has the freedom to maximize it. In both cooperative for-

mulations, the players cooperate and achieve the highest PRI of the scenarios. In the AL case, the aerodynamics player is not concerned with PRI and, subsequently, does not leave the weight player enough freedom to improve PRI. In the BN case, the PRI is significantly worse than the best cases.

Lift-to-Drag Ratios

The lift-to-drag ratios are maximum when aerodynamics is leader (AL). This is interesting, as when the players cooperate (FC and AC), the aerodynamics player sacrifices some of the lift-to-drag to benefit the weight player in other state variables, such as U and PRI. When aerodynamics is only concerned with its own requirements, the lift-to-drag ratios are maximum, but this adversely affects the weight player and, in turn, the goodness of the overall aircraft.

Aspect Ratio

Similar to the lift-to-drag ratios, AR is closest to 10.5 in the AL case. However, when cooperation is exercised, player aerodynamics realizes that the AR can be sacrificed to benefit both players. In the BN case, the AR is significantly worse than the best cases.

Landing Field Length

Unlike the previous cases, in this case both players are striving to bring s_L to 4500 ft. Of course, when cooperation exists, this goal is achieved, as shown in Table 2. However, in the leader/follower cases, because the leader is unable to meet this goal alone and these decisions restrict the follower, there is not enough collaboration to meet the goal even though, through cooperation, it is possible. Also, the effects of isolation/noncooperation (BN) are illustrated in Table 2 by a significantly inferior s_L . Although both players have a common goal, because of the isolation and assumptions that have to be made, they are unable to reach their goal.

V. Observations and Implications for Design

The developments and results in Secs. III and IV have computational and theoretical implications in modern design processes.

1) The leader/follower protocol embodies a philosophy that is not consistent with principles such as concurrent engineering (CE) and integrated product and process development. However, in complex systems where design teams are located throughout the world and are governed by different management with different objectives and priorities, true concurrency is very difficult. Therefore, tools and methods that accept and engage in some form of sequential processes where the different decision makers have their own objectives and requirements have important roles in complex systems design.

2) The noncooperative case (BN) used in Sec. IV is the best noncooperative case but is still inferior to the other solutions. Noncooperation should be avoided at all costs. Even largely sequential processes, as modeled in the leader/follower protocol, are shown to be more advantageous to the final design than the noncooperative case.

3) The computational requirements of constructing a player's RRS is a direct function of the number of variables needed from another player. Research in reducing the number of feedbacks and coupling between disciplines can facilitate the efficient construction of players' RRS¹⁵ and can be used to help provide effective decision support in the design of complex, multidisciplinary systems.

The results and observations documented in this paper have largely been driven by descriptive motivations, as opposed to prescriptive motivations. In other words, this work, in general, describes the resulting designs when various design process structures are used, or when different strategies are used by different design teams. We do not intend to prescribe managerial remedies to a noncooperative or leader/follower relationship, but describe the results if these relationships exist. Because relationships such as these certainly exist and will continue to exist in modern design of complex systems, the descriptive power of this work is beneficial to explore certain scenarios and the inherent tradeoffs between them.

VI. Closure

The design of multidisciplinary systems requires multiple decision makers, design teams, or organizations to make decisions that

may affect each other. CE principles have been used to facilitate this decision-making process at a personal interaction level. In this paper, game theory is being used to make similar strides but at the level of the interactions of mathematical models, analysis packages, and/or synthesis and optimization routines. The use of game theory to model decision-making processes where cooperation may not exist among decision makers in engineering design is of relatively recent origin; its usefulness in many other decision-making sectors such as economics, politics, and strategic warfare is well established. We strive to establish the usefulness of game theory in design by abstracting complex design processes as a series of games and analyzing the resulting insights into design problem and process structure.

Acknowledgments

We gratefully acknowledge the following grants: NASA NGT 51102 and National Science Foundation DMI-9420405. The cost of computer time was underwritten by the Systems Realization Laboratory of the Georgia Institute of Technology. We thank Rudi Cartuyvels (E-mail: cartuyvl@imec.be) for providing us with a copy of NORMAN to automate the process of creating response surface models.

References

- ¹Aubin, J. P., *Mathematical Methods of Game and Economic Theory*, North-Holland, Amsterdam, 1979.
- ²Dresher, M., *Games of Strategy*, Dover, New York, 1981.
- ³Von Neumann, J., and Morgenstern, O., *Theory of Games and Economic Behavior*, Princeton Univ. Press, Princeton, NJ, 1944.
- ⁴Myerson, R. B., *Game Theory: Analysis of Conflict*, Harvard Univ. Press, Cambridge, MA, 1991.
- ⁵Mistree, F., Hughes, O. F., and Bras, B. A., "The Compromise Decision Support Problem and the Adaptive Linear Programming Algorithm," *Structural Optimization: Status and Promise*, edited by M. P. Kamat, Vol. 150, Progress in Astronautics and Aeronautics, AIAA, Washington, DC, 1993, pp. 247–286.
- ⁶Lewis, K., and Mistree, F., "Foraging-Directed Adaptive Linear Programming: An Algorithm for Solving Nonlinear Mixed Discrete/Continuous Design Problems," *ASME Design Automation Conference* (Irvine, CA), 96-DETC/DAC-1601, American Society of Mechanical Engineers, New York, 1996.
- ⁷Reddy, R., Smith, W. F., Mistree, F., Bras, B. A., Chen, W., Malhotra, A., Badrinath, K., Lautenschlager, U., Pakala, R., Vadde, S., Patel, P., and Lewis, K., "DSIDES User Manual," Systems Realization Lab., George Woodruff School of Mechanical Engineering, Georgia Inst. of Technology, Atlanta, GA, July 1996.
- ⁸Lewis, K., Lucas, T., and Mistree, F., "A Decision Based Approach to Developing Ranged Top-Level Aircraft Specifications: A Conceptual Exposition," *AIAA/USAF/NASA/ISSMO 5th Symposium on Recent Advances in Multidisciplinary Analysis and Optimization* (Panama City, FL), AIAA, Washington, DC, 1994, pp. 465–481.
- ⁹Mistree, F., Marinopoulos, S., Jackson, D., and Shupe, J. A., "The Design of Aircraft Using the Decision Support Problem Technique," NASA CR 4134, April 1988.
- ¹⁰Lewis, K., "An Algorithm for Integrated Subsystem Embodiment and System Synthesis," Ph.D. Dissertation, Georgia Inst. of Technology, Atlanta, GA, 1996.
- ¹¹Rohl, P. J., Schrage, D. P., and Mavris, D. N., "Combined Aerodynamic and Structural Optimization of a High-Speed Civil Transport Wing," *Proceedings of the AIAA/ASME/ASCE/AHS/ASC 36th Structures, Structural Dynamics, and Materials Conference* (New Orleans, LA), AIAA, Washington, DC, 1995, pp. 548–557 (AIAA Paper 95-1222).
- ¹²Rao, J. R. J., Badhrinath, K., Pakala, R., and Mistree, F., "A Study of Optimal Design Under Conflict Using Models of Multi-Player Games," *Engineering Optimization*, Vol. 28, 1997, pp. 63–94.
- ¹³Cartuyvels, R., and Dupas, L. H., "NORMAN/DEBORA: A Powerful CAD-Integrating Automatic Sequencing System Including Advanced DOE/RSM Techniques for Various Engineering Optimization Problems," *Design of Information Infrastructure Systems for Manufacturing (DIISM'93)*, JSPE-IFIP WG 5.3, Tokyo, Japan, Nov. 1993.
- ¹⁴Montgomery, D., *Design and Analysis of Experiments*, 3rd ed., Wiley, New York, 1991.
- ¹⁵McCulley, C., and Bloebaum, C. L., "Complex System Design Task Sequencing for Cost and Time Considerations," *AIAA/USAF/NASA/ISSMO 6th Symposium on Multidisciplinary Analysis and Optimization* (Bellevue, WA), AIAA, Washington, DC, 1996, pp. 1488–1496.

A. D. Belegundu
Associate Editor