On Selecting Single-Level Formulations for Complex System Design Optimization

James T. Allison
Ph.D. Candidate
e-mail: optimize@umich.edu

Michael Kokkolaras
Associate Research Scientist
Mem. ASME
e-mail: mk@umich.edu

Panos Y. Papalambros
Professor
Fellow ASME
e-mail: pyp@umich.edu

Department of Mechanical Engineering,
University of Michigan,
2250 G. G. Brown Bldg.,
Ann Arbor, MI 48109

Design of complex products with several interacting subsystems or disciplinary analyses poses substantive challenges to both analysis and optimization, necessitating specialized solution techniques. A product or system may qualify as complex due to large scale or due to strong interactions. Single-level strategies for complex system optimization centralize decision-making authority, while multilevel strategies distribute the decision-making process. This article studies important differences between two popular single-level formulations: multidisciplinary feasible (MDF) and individual disciplinary feasible (IDF). Results presented aim at aiding practitioners in selecting between formulations. Specifically, while IDF incurs some computational overhead, it may find optima hidden to MDF and is more efficient computationally for strongly coupled problems; further, MDF is sensitive to variations in coupling strength, while IDF is not. Conditions that lead to failure of MDF are described. Two new reproducible design examples are introduced to illustrate these findings and to provide test problems for other investigations.
[DOI: 10.1115/1.2747632]

Keywords: complex system design, multidisciplinary design optimization, coupling strength, multidisciplinary feasible, individual disciplinary feasible, fixed point iteration

1 Introduction

Design of complex products with several interacting subsystems or disciplinary analyses poses substantive challenges to both analysis and optimization, necessitating specialized solution techniques. A product or system may qualify as complex due to large scale (large number of members, inputs, or outputs) or due to strong interactions. These interactions complicate multidisciplinary design optimization (MDO), but provide opportunity to exploit synergy between system members. Analysis of the system as an undivided whole can be inefficient, if not intractable. An alternative is to partition the system into smaller subsystems, and coordinate the solution of the subsystems. Wagner [1] identified four paradigms of system partitioning: object-based, aspect-based, sequential, and matrix. The aspect (discipline) partitioning paradigm is used in this article.

A system consisting of N subsystems can have up to $N(N-1)$ couplings. The number and strength of these couplings influences the difficulty of addressing tasks required for system optimization. Rogers and Bloebaum quantify coupling strength with the normalized derivatives of quantities that couple subsystems [2]. Haftka, Sobieski, and Padula define coupling strength similarly, but also present an alternative metric based on the relative difference between derivatives that reflects the difficulty of analyzing a coupled system more accurately [3]. An intuitive description of this alternative approach is given in this article. Alyaqout, Papalambros, and Ulsoy define a measure of coupling strength that accounts for optimization algorithm considerations, in addition to analysis considerations [4]. Sosa, Eppinger, and Rowles present a scheme to characterize coupling strength based on design expert knowledge, related to the required frequency of communication between design groups [5,6].

In system optimization, each subsystem may need to sacrifice its own needs in deference to the overall system objective. The traditional sequential design process [7], also known as block coordinate descent [8], allows each subsystem to seek its own objective in sequence. This process begins by optimizing the first subsystem, fixing that aspect of the system design, and then proceeding with subsequent subsystem designs. The approach deals with the interactions, but cannot exploit them to synergistically improve system performance. Such improvement can be achieved through understanding the partitioning or structure of the problem, motivating decomposition-based optimization methods, such as the single-level methods discussed in this article. We consider system design problems whose underlying analysis can be partitioned into N smaller analysis blocks, or subsystems, as illustrated in Fig. 1.

Each subsystem analysis requires as input relevant design variable values, and possibly analysis outputs from other subsystems. The design vector for the entire system is $\mathbf{x}$, and the components of $\mathbf{x}$ that are input to subsystem i form the vector $\mathbf{x}_i$. The components of $\mathbf{x}_i$ fall into two categories: local variables ($\mathbf{x}_{\ell i}$) are design variables that are inputs only to subsystem i; shared variables ($\mathbf{x}_{si}$) are inputs to subsystem i and at least one other subsystem. The collection of analysis outputs generated by subsystem j and input to subsystem i is $\mathbf{y}_{ij}$. These interaction quantities are termed coupling variables. Systems generally will not possess all of the possible connections or outputs shown in Fig. 1.

In addition to computing coupling variable values, subsystems may also compute function values required for solving the system optimization problem: $\mathbf{h}_i$ and $\mathbf{g}_i$ are, respectively, the equality and inequality design constraints computed by subsystem i, and f is the system objective function, which is assumed to be computed by subsystem N. The collections of all equality and inequality constraints are $\mathbf{h}=[\mathbf{h}_1,\mathbf{h}_2,\ldots,\mathbf{h}_N]$ and $\mathbf{g}=[\mathbf{g}_1,\mathbf{g}_2,\ldots,\mathbf{g}_N]$, respectively. In summary, subsystem analysis i entails taking in values for $\mathbf{x}_i$ and $\mathbf{y}_{ij}$ $(\forall j \in \{1,2,\ldots,N\}\backslash i)$, and calculating values for $\mathbf{y}_{ji}$ $(\forall j \in \{1,2,\ldots,N\}\backslash i)$, $\mathbf{h}_i$, $\mathbf{g}_i$, and, if $i=N$, f. Subsystem analysis is commonly performed using computer-aided engineering tools, such as finite element analysis.

The coupling variables input to subsystem i form the vector

Contributed by the Design Theory and Methodology Committee of ASME for publication in the JOURNAL OF MECHANICAL DESIGN. Manuscript received April 24, 2006; final manuscript received September 27, 2006. Review conducted by P. John Clarkson. Paper presented at the ASME 2005 Design Engineering Technical Conferences and Computers and Information in Engineering Conference (DETC2005), Long Beach, CA, Sept. 24–28, 2005.

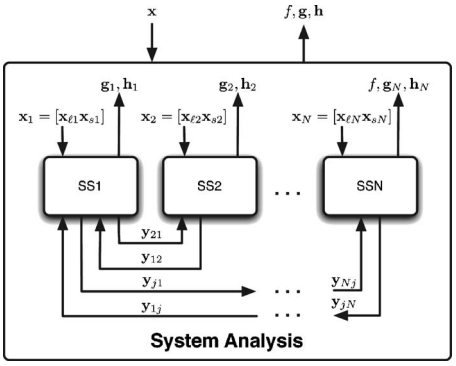

Fig. 1 The system analysis block with nonhierarchic relationships between subsystems

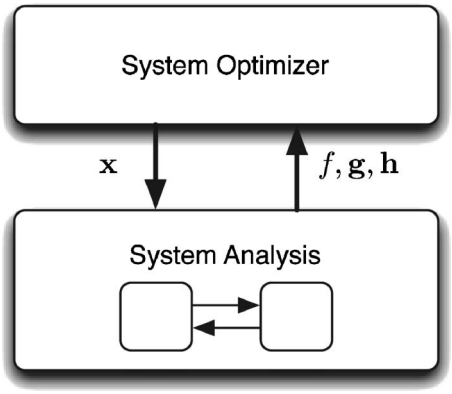

Fig. 2 MDF architecture

$\mathbf{y}_i=[\mathbf{y}_{ij_1},\mathbf{y}_{ij_2},\ldots,\mathbf{y}_{ij_{N-1}}]$, $\{j_1,j_2,\ldots,j_{N-1}\}=\{1,2,\ldots,N\}\backslash i$. The vector of all coupling variables input to all subsystems is $\mathbf{y}=[\mathbf{y}_1,\mathbf{y}_2,\ldots,\mathbf{y}_N]$. The value of each coupling variable $\mathbf{y}_{ij}$ is computed by a corresponding vector analysis function $\mathbf{a}_{ij}(\mathbf{x}_j,\mathbf{y}_j)$. More specifically, the quantities computed by subsystem j and passed to subsystem i are $\mathbf{y}_{ij}=\mathbf{a}_{ij}(\mathbf{x}_j,\mathbf{y}_j)$. The vector of all analysis functions that are used to compute coupling variable values is $\mathbf{a}(\mathbf{x},\mathbf{y})$, which is ordered such that its components correspond directly to the components of $\mathbf{y}$.

Given fixed values for coupling and design variables, the evaluation of $\mathbf{a}(\mathbf{x},\mathbf{y})$ returns computed values for $\mathbf{y}$. Because analysis functions are interdependent, the computed values for $\mathbf{y}$ may not match the original input values for $\mathbf{y}$. System analysis is the task of finding input values for $\mathbf{y}$ for a given $\mathbf{x}$ such that the computed coupling variable values agree with the corresponding input coupling variable values. A system in this state is said to be consistent. The system analysis problem is to find $\mathbf{y}$ such that the system analysis equations (sometimes called consistency or equilibrium equations) given in Eq. (1) are satisfied:

$$\mathbf{y}-\mathbf{a}(\mathbf{x},\mathbf{y})=\mathbf{0} \tag{1}$$

A solution to the system analysis problem is called a fixed point ($\mathbf{y}_p$), since evaluating $\mathbf{a}(\mathbf{x},\mathbf{y}_p)$ returns the original input vector $\mathbf{y}_p$. Fixed points are functions of $\mathbf{x}$, since their values vary with system design, and can be denoted $\mathbf{y}_p(\mathbf{x})$. The system analysis block, illustrated in Fig. 1, takes as input a fixed design $\mathbf{x}$, solves Eq. (1), and uses the resulting $\mathbf{y}_p(\mathbf{x})$ to calculate the system analysis outputs required by the system design optimization problem, i.e., $f(\mathbf{x},\mathbf{y}_p(\mathbf{x}))$, $\mathbf{g}(\mathbf{x},\mathbf{y}_p(\mathbf{x}))$, and $\mathbf{h}(\mathbf{x},\mathbf{y}_p(\mathbf{x}))$.

The approach to solving Eq. (1) is an important distinction between the formulations presented in this article. The system design optimization problem, given in Eq. (2), seeks $\mathbf{x}$ such that the system objective is minimized and all design constraints are satisfied, while ensuring system consistency. System consistency can be enforced by either the optimization algorithm or a separate system analysis algorithm; each approach is explored in this article:

$$\begin{aligned} \min_{\mathbf{x}=[\mathbf{x}_\ell,\mathbf{x}_s]} \quad & f(\mathbf{x},\mathbf{y}_p(\mathbf{x})) \\ \text{subject to} \quad & \mathbf{g}(\mathbf{x},\mathbf{y}_p(\mathbf{x}))\le\mathbf{0} \\ & \mathbf{h}(\mathbf{x},\mathbf{y}_p(\mathbf{x}))=\mathbf{0} \end{aligned} \tag{2}$$

This article investigates critical differences between the implementation of two single-level formulations for complex system optimization, multidisciplinary feasible (MDF), and individual disciplinary feasible (IDF) [9], to aid practitioners in problem formulation selection. Two specific phenomena are studied in detail: the effect of subsystem interdependence (coupling) on method performance, and the ability to find superior solutions that are hidden to traditional methods. Some background in single-level system optimization is provided first; two design examples are then presented, followed by conclusions.

2 Review of Single-Level MDO Formulations

The formulations and strategies for the MDF and IDF methodologies are presented in [9], and reviewed here. In these single-level formulations, all decision making is centralized and performed by a single optimization algorithm. Single-level formulations can be effective at dealing with systems possessing strong interactions, but are not well suited for problems of large dimension, where multilevel formulations may be preferred [10,11]. Balling and Sobieski provided a review of single-level formulations [12], and Balling and Wilkinson implemented these formulations in the solution of analytical test problems [13]. Hulme and Bloebaum [14] compared the implementation of single-level formulations using many test problems of varying size and coupling strength, with emphasis on solution differences that appear to be due to numerical limitations relating to increased problem dimension. None of the preceding publications, however, addresses how implementation of these formulations exhibits performance variation with respect to changes in coupling strength of a specific design problem, or address the ability of IDF to find fundamentally different solutions in practice. In addition to presenting results concerning these issues, this article clarifies cases where MDF implementations can fail.

2.1 Multidisciplinary Feasible Formulation. The most basic of MDO formulations is the MDF approach (Fig. 2), also known as "nested analysis and design" or "All-in-one". A single system-level optimizer is used, and a separate algorithm performs the system analysis task. The optimizer supplies the system analyzer with a design $\mathbf{x}$, and the system analyzer returns the function values f, $\mathbf{g}$, and $\mathbf{h}$. The MDF problem formulation is identical to the problem statement of Eq. (2); $\mathbf{y}_p(\mathbf{x})$ is computed at every step of the optimization process using a system analysis algorithm.

The fixed point iteration (FPI) algorithm is a popular system analysis method for MDF. Section 3 will elucidate some challenges in using FPI in conjunction with system optimization. Other system analysis methods exist, but also exhibit their own difficulties, and will not be discussed in this article. A design optimization strategy is classified as MDF if a complete system analysis is performed for every optimization iteration. The analysis is "nested" within the design. The optimizer is responsible to find the optimal design $\mathbf{x}^*$ (the design solution), while the system analyzer is responsible to find $\mathbf{y}_p(\mathbf{x})$ (the system analysis solution).

This approach may be desirable if the subsystems are weakly coupled (fast system analysis convergence), and if the subsystem analyses are not computationally expensive. In addition, MDF

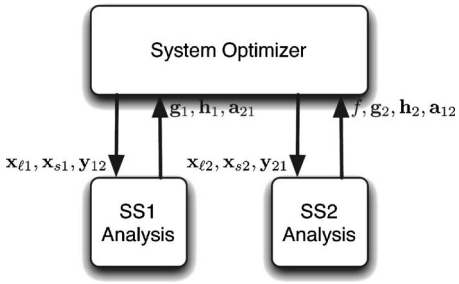

Fig. 3 IDF architecture

eases the incorporation of legacy analysis tools. If a design organization already performs a complete analysis before making a design decision, MDF is a natural fit.

Although the merits of MDF are notable, its shortcomings must be clearly understood. MDF is dependent upon the effectiveness of the system analyzer. If the analyzer does not converge at any point in the process, the optimizer may fail. The nested analysis and optimization process required by MDF can be computationally inefficient, and motivates approaches that eliminate the need for repeated system analysis [15]. In addition, typical MDF implementations cannot exploit the potential coarse-grained parallelism of distinct subsystem analyses. MDF has been aptly termed a "brute force" approach [16].

2.2 Individual Disciplinary Feasible Formulation. In the IDF formulation, an analyzer for each subsystem is employed and a single system-level optimizer is used, but the optimizer, rather than a system analysis algorithm, coordinates the interactions between the subsystem analyses. The IDF architecture is illustrated in Fig. 3 using a two-element system. The optimizer chooses values for both design and coupling variables: system analysis and design are performed simultaneously. Since the system optimizer provides all inputs required for all subsystems concurrently, subsystem analyses may be executed in parallel.

The IDF formulation is given in Eq. (3). It differs from the MDF formulation in that the decision variable vector includes both design variables $\mathbf{x}$ and coupling variables $\mathbf{y}$, while auxiliary constraints $\mathbf{h}_{\text{aux}}$ are added to ensure system consistency eliminating the need to solve for $\mathbf{y}_p(\mathbf{x})$ at each optimization iteration.

$$\begin{aligned} \min_{\mathbf{x}=[\mathbf{x}_\ell,\mathbf{x}_s],\mathbf{y}} \quad & f(\mathbf{x},\mathbf{y}) \\ \text{subject to} \quad & \mathbf{g}(\mathbf{x},\mathbf{y}) \leq \mathbf{0} \\ & \mathbf{h}(\mathbf{x},\mathbf{y}) = \mathbf{0} \\ & \mathbf{h}_{\text{aux}}(\mathbf{x},\mathbf{y}) = \mathbf{y} - \mathbf{a}(\mathbf{x},\mathbf{y}) = \mathbf{0} \end{aligned} \tag{3}$$

IDF facilitates coarse-grained parallelism, improves convergence properties, and drives the design toward better solutions if multiple analysis solutions exist. If the solution process is interrupted, the intermediate design may not be consistent or feasible. In contrast, an interrupted MDF solution will yield a consistent, but potentially infeasible, design. Since IDF does not require the frequently expensive task of achieving system consistency when far from the solution, the optimization algorithm can trace a more efficient path toward the solution and computational expense is reduced through the elimination of repeated system analysis steps [17].

IDF is more centralized than MDF, and the dimension of the optimization problem is increased since coupling variables are made decision variables. This increase in dimension can reduce numerical solution accuracy when the problem size is large, as evident in the results presented by Hulme and Bloebaum [14]. MDF may be preferable when the dimension of $\mathbf{y}$ is much larger than the dimension of $\mathbf{x}$ [3,17]. Furthermore, auxiliary equality constraints can introduce numerical solution difficulties [17,18].

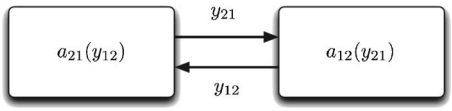

Fig. 4 Two-element coupled system

Section 3 will show that using the IDF formulation can help find superior solutions that are hidden to MDF implementations, and Secs. 4 and 5 will demonstrate that IDF results in improved computational efficiency for strongly coupled problems (as predicted in [9]). If a high level of centralization is acceptable, IDF may be an ideal design strategy.

3 System Analysis

The system analysis equations of Eq. (1) may be solved with iterative methods such as Newton-Raphson or FPI [19]. FPI is regularly employed as the analysis tool for the MDF formulation. Due to its intuitive implementation, MDF is the most frequently utilized MDO strategy [16]. However, it should not be applied without recognition of its shortcomings. As an alternative to nesting FPI within an optimization algorithm, solution of some or all system analysis equations may be performed by the optimization algorithm, as is the case with IDF, which can alleviate difficulties encountered with MDF implementations. This section reviews the nature of FPI, explores some issues with its use in MDF, and presents convergence conditions for FPI to aid intuition. These convergence conditions are foundational to understanding coupling strength. Haftka et al. [3] presented two definitions for coupling strength—the first accounting for the magnitude of interanalysis derivatives, and the second for the relationship between these derivatives. This section, along with subsequent design examples, strengthens the position of the second definition.

3.1 Fixed Point Iteration. A two-element coupled system is depicted in Fig. 4, which possesses feedback coupling, since a_{21} depends on the output of a_{12} and vice versa. Since $\mathbf{x}$ is fixed during system analysis, it is omitted from the current discussion.

To employ FPI for system analysis, an initial guess is made for the input to the subsystem that is executed first, and the analyses are iteratively performed with updated coupling variable values until consistency is achieved, i.e., coupling variables match analysis outputs, satisfying Eq. (1). If the system meets certain criteria, this process will converge to a fixed point. The FPI algorithm for the two-dimensional example problem is [19]:

Step 0: choose initial guess y_{12}^0, set $k=0$
Step 1: $k=k+1$
Step 2: $y_{21}^k=a_1(y_{12}^{k-1})$
Step 3: $y_{12}^k=a_2(y_{21}^k)$
Step 4: if $\|\mathbf{y}^k-\mathbf{y}^{k-1}\|<\varepsilon$, then stop, otherwise go to Step 1.

When the stopping criterion in Step 4 is met, the system is epsilon-consistent, approximately satisfying Eq. (1). The norm in Step 4 is typically the Euclidian or infinity norm. Figure 5 illustrates the analysis space of a sample two-element system, which possesses two fixed points at the intersections of the analysis functions. Following the algorithm above, FPI will converge to the fixed point $\mathbf{y}_{p_A}$ if y_{12}^0 is near this point, but never to $\mathbf{y}_{p_B}$ for any y_{12}^0.

Studying this result, it can be seen graphically that if the line traced by a_{12} is steeper than the line traced by a_{21} in the neighborhood of a fixed point, then FPI will converge to that fixed point. This observation agrees with the well-known necessary and sufficient conditions for FPI convergence [20]:

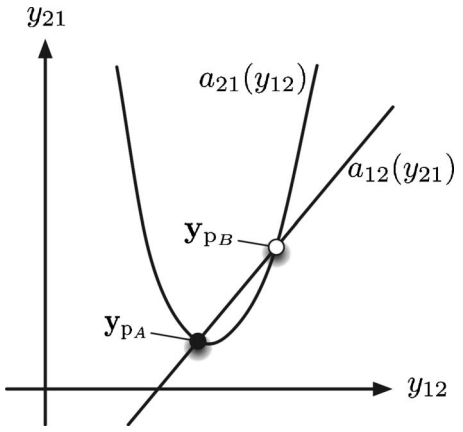

Fig. 5 System with multiple fixed points

$$\left(\frac{\partial a_{21}(y_{12})}{\partial y_{12}}\right)^{-1} < \frac{\partial a_{12}(y_{21})}{\partial y_{21}} \tag{4}$$

The derivatives in Eq. (4), in normalized form, are used by Rogers and Bloebaum [2] to quantify coupling strength between subsystems. Intuitively, higher sensitivity between subsystems will require more iterations during analysis. When the specific solution algorithm is FPI, however, computational effort depends instead on the relationship between these sensitivities. For example, if the relation in Eq. (4) is satisfied but is near equality, convergence will require numerous iterations, and will cycle without convergence if the inequality becomes equality. If we define coupling strength as the effort required to bring a coupled system into a consistent state, rather than just the influence that the subsystems have on each other, then coupling strength is more aptly quantified through a comparison of derivative values than through absolute derivative magnitudes.

3.2 Example: Hidden Optima. Consider the IDF formulation of the two-element system optimization problem given in Eq. (5).

$$\begin{aligned}
\min_{\mathbf{x},\mathbf{y}} \quad & f(\mathbf{x},\mathbf{y}) = y_{12}^2 - 100y_{21} + 0.1\mathbf{x}'\mathbf{x} \\
\text{subject to} \quad & \mathbf{h}_{\text{aux}}(\mathbf{x},\mathbf{y}) = \mathbf{y} - \mathbf{a}(\mathbf{x},\mathbf{y}) = \mathbf{0} \\
\text{where} \quad & a_{21}(y_{12},x_1) = \phi_1(x_1)(y_{12} - \alpha_1)^2 \\
& a_{12}(y_{21},x_2) = \phi_2(x_2)y_{21} + \alpha_2 \\
& \phi_1(x_1) = \frac{0.25}{1 + e^{x_1}} + 0.5 \\
& \phi_2(x_2) = -\left(\frac{1}{1 + e^{x_2}} + 0.5\right) \\
& \alpha_1 = 3, \quad \alpha_2 = 3.5
\end{aligned} \tag{5}$$

For any $\mathbf{x} \in \mathbb{R}^2$, two fixed points exist, similar to the system in Fig. 5. FPI is capable of finding only a point with small y_{21} and large y_{12}, which is a local optimum. The second fixed point has the reverse properties, and leads to the global optimum. Even when started at the global optimum, the MDF implementation moves toward the inferior local optimum $f(\mathbf{x}^*_{\text{MDF}}) = -0.244$ at $\mathbf{x}^*_{\text{MDF}} = [-1.902, 2.273]$. The IDF implementation finds the global optimum $f(\mathbf{x}^*_{\text{IDF}}) = -975.692$ at $\mathbf{x}^*_{\text{IDF}} = [5.824, 7.754]$. It can be shown using Eq. (4) that MDF implemented using FPI is incapable of finding the global optimum. Physically meaningful models can also exhibit such behavior. Note that the MDF and IDF implementations solve identical design problems; the different solutions follow from limitations of FPI. The MDF formulation is indirectly limited by its dependence on available system analysis tools.

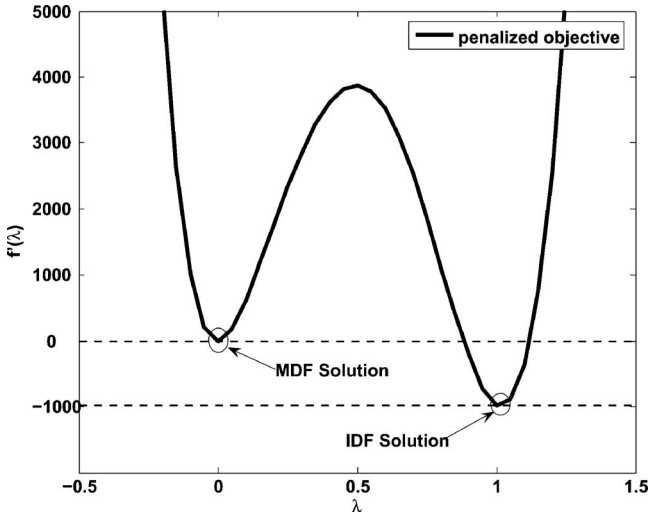

Fig. 6 IDF optimization space visualization

Although the optimization space for IDF is more complex, the problem design space can be explored more effectively. When the $f(\mathbf{x})$ response surface (as computed with FPI) is visualized graphically, only a single optimum is seen at $\mathbf{x}^*_{\text{MDF}}$. Since the IDF optimization space is in $\mathbb{R}^4$, the objective function cannot be visualized easily. One approach is to plot the objective function along the line that connects the MDF and IDF solutions, i.e., plot $f(\lambda) = f(\lambda[\mathbf{x}^*_{\text{IDF}}\mathbf{y}^*_{\text{IDF}}] + (1-\lambda)[\mathbf{x}^*_{\text{MDF}}\mathbf{y}_p(\mathbf{x}^*_{\text{MDF}})])$, where $\mathbf{y}^*_{\text{IDF}}$ is the coupling variable vector at the IDF solution, and $\mathbf{y}_p(\mathbf{x}^*_{\text{MDF}})$ is the coupling variable fixed point computed by FPI at $\mathbf{x}^*_{\text{MDF}}$. The auxiliary constraints can be included in the visualization by adding a penalty for constraint violation to the objective function: $f'(\lambda) = f(\lambda) + 500\|\mathbf{h}_{\text{aux}}(\mathbf{x},\mathbf{y})\|_2^2$. Figure 6 illustrates how using IDF can reveal optima that are hidden to MDF. For the points represented in this plot, the auxiliary constraint violation is zero only at the MDF and IDF solutions. It is hypothesized that other methods that employ simultaneous analysis and design, such as analytical target cascading[21], share this desirable behavior with IDF.

Thus, although FPI implementation is straightforward, it presents several difficulties: FPI may not converge to an analysis solution; if multiple solutions exist, FPI may not find them all; the sequential nature of FPI prevents the parallel execution of analyses. When FPI is used as the system analysis tool for MDF, all of these same issues arise. The optimization problem may not converge, and when it converges, the globally optimal solution may not be found. In addition, the resulting nested optimization and analysis process can be inefficient. These algorithmic considerations are critical factors in making problem formulation decisions.

This section explored issues associated with FPI, established the ability of IDF to find "hidden" optima, and laid a foundation for the understanding of coupling strength. The following two sections use reproducible engineering design examples to illustrate how coupling strength influences the computational performance of MDF and IDF implementations. Each of the examples enable variation of coupling strength without changing the dimension or nature of the problem. This facilitates numerical experiments that reveal the distinct response of MDF and IDF to increased coupling strength.

4 Air Flow Sensor Design

An air flow sensor design problem that incorporates structural and aerodynamic analysis is considered. Vane airflow (VAF) sensors are used in automotive applications to monitor the rate at

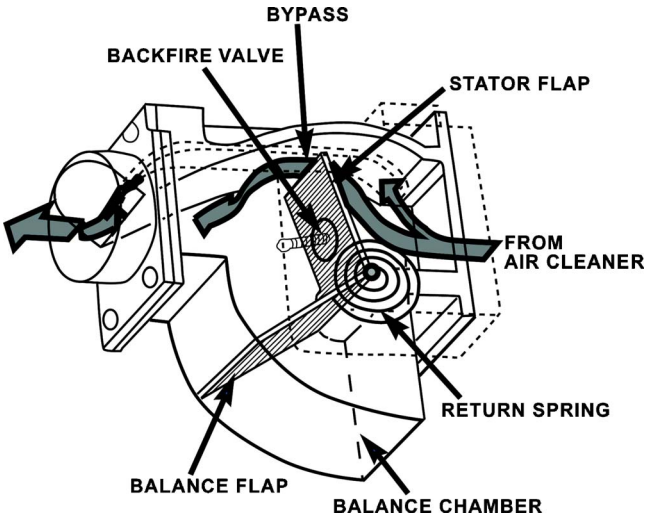

Fig. 7 Vane airflow sensor schematic (after [22])

which air enters the engine for use in fuel injection control. A VAF sensor is illustrated in Fig. 7 [22]. Incoming air flows past the stator flap, which deflects in proportion to air flow velocity. A bypass channel reduces the sensor's impedance on airflow. A potentiometer measures this deflection angle and provides a signal to the engine control unit.

A simplified model of a VAF sensor is used in this design example (Fig. 8). The stator flap has length ℓ and width w, is attached to its base with a revolute joint, and is biased to the vertical position with a torsional spring of stiffness k. The plate is subject to horizontal air flow of speed v that results in a drag force F. The design objective is to choose ℓ and w such that the plate deflects an amount θ (for a fixed air speed) that closely matches a target deflection value $\hat{\theta}$. The plate area $A=\ell w$ is constrained to a fixed value, and the drag force on the plate must not exceed $F_{\max}$. This task, summarized in Eq. (6), is in essence a sensor calibration problem.

$$\begin{aligned} \min_{\ell,w} \quad & (\theta-\hat{\theta})^2 \\ \text{subject to} \quad & F-F_{\max}\le 0 \\ & \ell w - A = 0 \end{aligned} \tag{6}$$

The structural analysis computes the plate deflection θ for a given sensor design and drag force. Note that the governing equation is transcendental, requiring iterative solution for θ:

$$k\theta = \frac{1}{2}F\ell\ \cos\theta \tag{7}$$

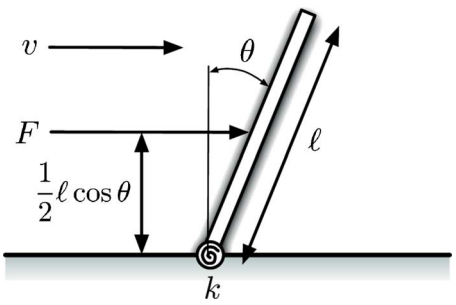

Fig. 8 Simplified representation of a vane airflow sensor

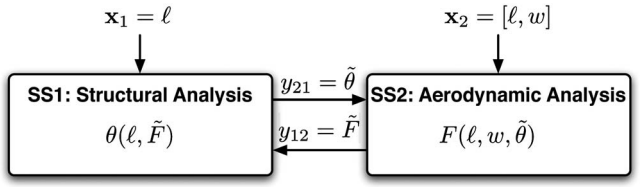

Fig. 9 Coupling relationship in airflow sensor analysis

The aerodynamic analysis computes the drag force on the plate F for a given sensor design and plate deflection; C is a constant that incorporates air density and the drag coefficient, $C=\frac{1}{2}\rho C_D$, and A_f is the plate frontal area, $A_f=\ell w\cos\theta$:

$$F = CA_f v^2 = C\,\ell\,w\cos\theta v^2 \tag{8}$$

The analyses depend on each other—Fig. 9 illustrates this relationship. The coupling variables are $\tilde{\theta}$ and $\tilde{F}$. This notation is used to distinguish coupling variables from the corresponding analysis functions, $\theta(\ell,\tilde{F})$ and $F(\ell,w,\tilde{\theta})$. This notational approach is used for this and the following example in order to preserve straightforward interpretation of formulations. The shared variable is ℓ ($\mathbf{x}_{s1}=\mathbf{x}_{s2}=\ell$), and w is a local variable ($\mathbf{x}_{\ell 1}=w$). Fixed point iteration can be used to find consistent values of $\tilde{F}$ and $\tilde{\theta}$ for a given design ($\mathbf{x}=[\ell,w]$).

The optimal solution to this problem may be found using monotonicity analysis (MA) [23], and can be used to benchmark computational results. The analytical solution is given in Eqs. (9):

$$\ell^* = \frac{2k\cos^{-1}\left(\dfrac{F_{\max}}{CAv^2}\right)CAv^2}{F_{\max}^2}, \quad w^* = A/\ell^* \tag{9}$$

For parameter values $k=0.050$ N/rad, $v=40.0$ m/s, $C=1.00$ kg/m^3, $F_{\max}=7.00$ N, A$=0.01$ m^2, and $\hat{\theta}=0.250$ rad, the optimal design is $[\ell^*,w^*]=[0.0365,0.274]$. The drag coefficient of a finite flat plate is approximately 2.0, resulting in a value of $C=1.00$ if we assume air density to be 1.00 kg/m^3.

The MDF solution entails solving Eq. (6), where each optimization iteration requires a fixed point iteration solution to Eqs. (7) and (8) to obtain consistent values for $\tilde{\theta}$ and $\tilde{F}$. The MDF result matches the MA solution. The IDF solution requires the addition of $\tilde{\theta}$ and $\tilde{F}$ to the decision variable set, as well as auxiliary constraints on these values, as shown in Eq. (10). The IDF solution also matches the MA solution:

$$\begin{aligned} \min_{\ell,w,\tilde{\theta},\tilde{F}} \quad & (\tilde{\theta}-\hat{\theta})^2 \\ \text{subject to} \quad & \tilde{F}-F_{\max}\le 0 \\ & \ell w - A = 0 \\ & \tilde{\theta}-\theta(\ell,\tilde{F}) = 0 \\ & \tilde{F}-F(\ell,w,\tilde{\theta}) = 0 \end{aligned} \tag{10}$$

If the spring constant k is large, the plate deflection will be small, resulting in only minor changes to the frontal area and drag force. Quantitatively, increasing k will reduce $\partial F(\ell,w,\tilde{\theta})/\partial\tilde{\theta}$, but not affect $\partial\theta(\ell,\tilde{F})/\partial\tilde{F}$, resulting in reduced coupling strength between analyses. Conversely, small k results in high coupling strength; as verified experimentally, the number of iterations required for FPI convergence increases with decreasing k. Consequently, the computational expense of the MDF implementation is expected to increase with decreasing k. IDF eliminates the need to converge to consistent analysis results at points far from the optimal solution, but incurs its own computational overhead due to

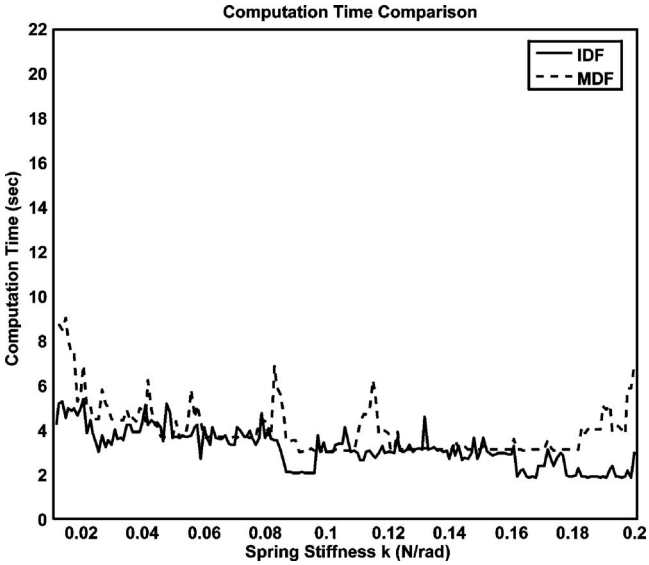

Fig. 10 Comparison of MDF and IDF solution time as a function of coupling strength

increased problem dimension. The value of k was varied from 0.01 to 0.20 N/rad, and the MDF and IDF computation times (on a 3.4 GHz Pentium® 4 PC) were recorded. The result, displayed in Fig. 10, reveals that MDF does incur more computational expense with small values of k, as expected, while IDF is only slightly sensitive to changes in coupling strength. The MDF and IDF solutions agreed within 0.01% over the specified stiffness range.

Figure 11 compares the number of function evaluations required for the MDF and IDF implementations. A function evaluation is defined as the calculation of both structural and aerodynamic outputs, including calculations required for finite differencing. Since the analysis expense for this example is low, the additional computational overhead required for the IDF implementation is a significant factor in solution time. With respect to function evaluations, IDF solution expense truly is insensitive to coupling strength. It is also clear from this plot that the noise displayed in Fig. 10 is purely computational. The example in the following section requires more analysis time, which is large compared to computational noise, resulting in a smoother plot of solution time.

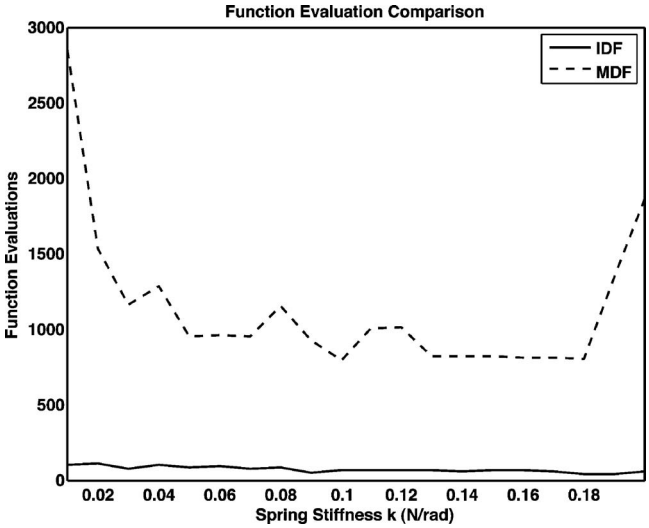

Fig. 11 Comparison of MDF and IDF function evaluations as a function of coupling strength

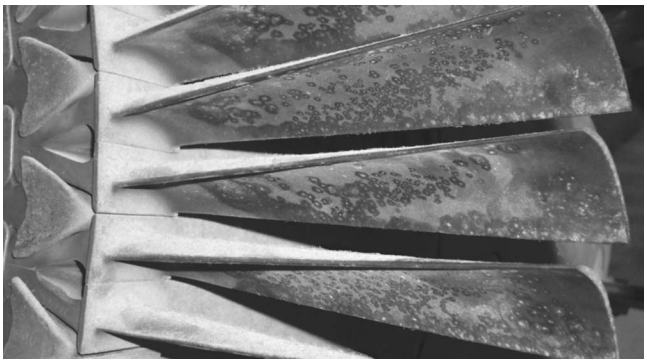

Fig. 12 GE J-79 turbojet engine turbine blades [24]

At stiffness values much larger than the range displayed in Fig. 11, the design problem becomes infeasible since the equilibrium plate deflection is low enough that the resulting large frontal area incurs drag force values that exceed $F_{\max}$. It is interesting to note that in this infeasible domain, MDF satisfies the drag force constraint and violates the area constraint, while IDF exhibits the converse. This phenomenon comes about because MDF finds consistent values for $\tilde{\theta}$ and $\tilde{F}$ at each optimization iteration, while IDF does not. IDF is free to choose an infeasible $\tilde{F}$ in order to satisfy the area constraint, but MDF does not have this flexibility. As stiffness is increased and the design problem approaches infeasibility, the MDF computation time increases as observed in the plot. At stiffness values below the displayed range, MDF fails due to excessive coupling strength. In this design example, MDF time increases with coupling strength due to excessive analysis effort, and increases with decreasing coupling strength due to excessive optimization effort for narrowly feasible design problems.

5 Turbine Blade Design

The analysis and design of a turbine blade for a gas turbine engine is presented, followed by a parametric study on coupling strength. The results substantiate the findings discussed in the previous sections, and provide further insights.

5.1 Design Problem Description. A turbine blade in a gas turbine engine is exposed to high temperature combustion gases moving at high velocity, and is subject to high forces due to aerodynamic drag force and centripetal acceleration. Figure 12 illustrates turbine blades from a GE J-79 turbojet engine [24]. Each blade is attached to the rotor at the left of the figure, and combustion gases moving from the left cause the turbine to rotate.

Several phenomena were modeled in order to capture the design tradeoffs and coupling behavior, specifically: thermal expansion of the turbine blade in the axial direction, stress and elongation due to centripetal acceleration, aerodynamic drag force and the resulting bending stresses, and the temperature dependence of thermal conductivity, elastic modulus, and rupture stress.

The blade temperature profile depends upon its dilated length. Elongation due to thermal expansion or centripetal forces exposes more surface area to hot combustion gases, affecting the heat transfer through the blade and the associated temperature profile. The model also captures the dependence of elastic modulus and thermal conductivity on temperature. Higher temperatures (caused by changes in length) result in lower stiffness, causing greater elongation. In summary, temperature depends on length, and length depends on temperature. Thus, turbine blade analysis consists of two coupled disciplinary analyses, similar to the previous

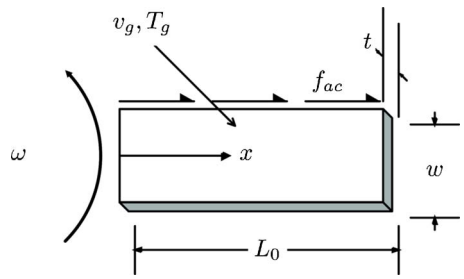

Fig. 13 Turbine blade model schematic

example. The design task is to minimize the blade mass m and the heat transfer through the blade q. Both of these metrics influence turbine thermal efficiency.

5.2 Analysis Model. The turbine blade is modeled as a simple rectangular fin (Fig. 13). The design variables are the blade width w and thickness t. The blade has an initial undeformed length of L_0, and is subjected to combustion gas at temperature T_g and velocity v_g. The blade is affixed to a rotor with angular velocity ω, resulting in an inertial force f_{ac}. The axial position x is measured from the blade base. Four failure modes are considered: melting, interference between the blade and the turbine housing due to elongation, and structural failure due to bending stress σ_b or axial stress σ_a. Several simplifying assumptions were made: constant coefficient of thermal expansion α, no internal blade cooling, constant inertial force f_{ac} over the blade, and no lateral contraction. The dependence of thermal conductivity (k), elastic modulus (E), and rupture stress (σ_r) on temperature is modeled with curve fits based on empirical data.

The turbine blade optimization problem is presented in Eq. (11), which has been formulated as a single objective problem by creating a mass constraint. The coupling variables $\tilde{T}(x)$ and $\tilde{L}$ are required to be consistent with the corresponding analysis functions $T(w,t,\tilde{L},x)$ and $L(\tilde{T}(x))$ at the solution:

$$\begin{aligned} \min_{w,t} \quad & q(w,t,\tilde{L}) \\ \text{subject to} \quad & T(w,t,\tilde{L},x) - T_{\text{melt}} \le 0 \\ & \delta_{\text{total}}(\tilde{T}(x)) - \delta_{\text{allow}} \le 0 \\ & \sigma_a(\tilde{L},x) - \sigma_r(\tilde{T}(x),x) \le 0 \\ & \sigma_b(t,\tilde{L},x) - \sigma_r(\tilde{T}(x),x) \le 0 \\ & m(w,t) - m_{\max} \le 0 \\ & \text{and} \quad 0 \le x \le L_0 + \delta_{\text{total}}(\tilde{T}(x)) \end{aligned} \tag{11}$$

T_{melt} is the melting temperature, $\delta_{\text{total}}(\tilde{T}(x))$ is the blade elongation, δ_{allow} is the initial clearance between the blade and housing, and $\sigma_a(\tilde{L},x)$, $\sigma_b(t,\tilde{L},x)$, and $\sigma_r(\tilde{T}(x),x)$ are the axial, bending, and rupture stress distributions along the blade, respectively. The analysis for each discipline (structural and thermal) follows.

Structural Analysis. The structural analysis calculates blade mass ($m=wtL_0\rho$), where ρ is the blade density, the total blade elongation (δ_{total}), which is the sum of the thermal expansion δ_{th} and elongation due to axial acceleration δ_{ax}, and bending and axial stress distributions ($\sigma_b(x)$, $\sigma_a(x)$). We begin with the elongation calculation. The first elongation term is calculated as follows:

$$d\delta_{\text{th}} = \alpha(T(x) - T_0)dx$$

$$\delta_{\text{th}} = \int_0^{L_0} T(x)dx - \int_0^{L_0} \alpha T_0 dx \tag{12}$$

$$\delta_{\text{th}} = \int_0^{L_0} T(x)dx - \alpha T_0 L_0$$

T_0 is the initial blade temperature, and α, the coefficient of thermal expansion, is assumed constant. The temperature profile, calculated by the thermal analysis, is required to evaluate δ_{th}. To calculate δ_{ax}, the axial load as a function of axial position is determined. The portion of the blade outboard of a position x pulls with load $P_a(x)$. The tangential velocity of the blade $v=\omega r$ is assumed to be constant over the blade length, and is valid if $L_0 \ll r$:

$$\begin{aligned} P_a(x) &= \int_x^{L_0+\delta_{\text{total}}} \frac{v^2}{r}\rho A_c dx = \frac{v^2}{r}\rho wt(L_0 + \delta_{\text{total}} - x) \\ &= \omega^2 r\rho wt(L_0 + \delta_{\text{total}} - x) \end{aligned} \tag{13}$$

$$\delta_{ax} = \int_0^{L_0+\delta_{\text{total}}} \frac{P_a(x)dx}{A_c E(T(x))} = \omega^2 r\rho \int_0^{L_0+\delta_{\text{total}}} \frac{(L_0 + \delta_{\text{total}} - x)}{E(T(x))}dx \tag{14}$$

$$\delta_{\text{total}} = \int_0^{L_0} T(x)dx - \alpha T_0 L_0 + \omega^2 r\rho \int_0^{L_0+\delta_{\text{total}}} \frac{(L_0 + \delta_{\text{total}} - x)}{E(T(x))}dx \tag{15}$$

Since Eq. (15) is transcendental, an iterative solution procedure is required to solve for δ_{total} given $T(x)$.

The axial stress is a function of axial position, and is calculated with the relation $\sigma_a = P_a/A_c$, where P is the axial load, $A_c = wt$ is the cross-sectional area as before, and $L = L_0 + \delta_{\text{total}}$ is the elongated length.

$$\sigma_a(L,x) = \omega^2 r\rho(L - x) \tag{16}$$

The aerodynamic load is calculated using $P_{\text{aero}} = \frac{1}{2}A_f C_D \rho v^2$, where $A_f = wL$ is the frontal area, C_D is the drag coefficient, ρ is the combustion gas density, and v is the combustion gas velocity (assumed perpendicular to the blade). For convenience, the constant $K = \frac{1}{2}C_D\rho v^2$ is defined, giving $P_{\text{aero}} = KwL$. The total drag force acting on the blade outboard of a position x is $P_{\text{aero}}(x) = Kw(L - x)$, and the bending moment at point x is $M(x) = Kw(L-x)^2/2$, resulting in a bending stress of:

$$\sigma_b(w,L,x) = \frac{3K(L-x)^2}{4t^2} \tag{17}$$

Thermal Analysis. The thermal model, which calculates the temperature profile and heat transfer, was derived from the steady-state heat equation using constant base temperature and an adiabatic tip boundary condition [25]. The average convection coefficient $\bar{h}$ was approximated using empirical correlations involving the average Nusselt number $\overline{\text{Nu}}$ and the Prandtl number Pr: $\overline{\text{Nu}} = \bar{h}w/k_g = C\,\text{Re}_D^z\,\text{Pr}^{1/3}$. The combustion gas conduction coefficient is k_g, $\text{Re}_D = vw/\nu$ is the appropriate Reynold's number, z is an empirical exponent of 0.731, and C is the heat capacity of the combustion gas. Solving for $\bar{h}$, and substituting values for the other parameters with SI units (at $T_\infty = 900\,^\circ\text{C}$), we find: $\bar{h}(v,w) = 9.196v^{0.731}w^{-0.269}$. The temperature profile and the heat transfer through the blade into the rotor at the point of attachment are found through solution of the heat equation with the appropriate boundary conditions:

$$T(w,t,L,x) = \frac{\cosh(s(L-x))}{\cosh(sL)}(T_b - T_\infty) + T_\infty \tag{18}$$

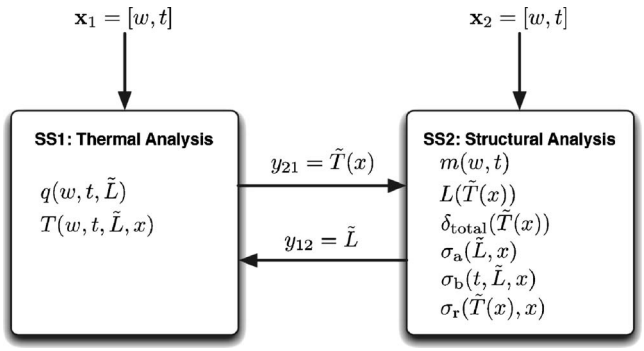

Fig. 14 Turbine blade coupling and functional relationships

$$q(w,t,L) = wt(T_b - T_\infty)\tanh(sL)\sqrt{2\bar{h}(w+t)wtk} \tag{19}$$

where $s = \sqrt{2\bar{h}(t+w)/ktw}$.

Curve Fits. Surrogate models based on empirical data [26] were employed in order to capture temperature dependence. The rupture stress σ_r for Inconel X-750 was approximated using a modified sigmoid function:

$$\sigma_r(T) = \frac{1300}{1 + e^{0.011(T-675)}} \tag{20}$$

The conductivity of the blade k was modeled using a linear fit. The dependence on average temperature $\bar{T}$ was captured from empirical data

$$k(\bar{T}) = 6.8024 + 0.0172\bar{T} \tag{21}$$

A fourth-order polynomial was fit to the modulus of elasticity for the blade material:

$$E(T) = 209.8 - 0.0487T - 0.0002T^2 + 6 \times 10^{-7}T^3 - 6 \times 10^{-10}T^4 \tag{22}$$

5.3 System Analysis. Figure 14 illustrates the analysis problem structure. The system has two shared design variables, and no local design variables.

The analysis functions evaluated by the thermal analysis are the heat loss $q(w,t,\tilde{L})$ and the temperature distribution $T(w,t,\tilde{L},x)$. The structural analysis evaluates several analysis functions, including the mass $m(w,t)$, dilated length $L(\tilde{T}(x))$, total deflection, $\delta_{\text{total}}[\tilde{T}(x)]$, and the bending, axial, and rupture stress distributions $\sigma_{\text{b}}(t,\tilde{L},x)$, $\sigma_{\text{a}}(\tilde{L},x)$, and $\sigma_{\text{r}}(\tilde{T}(x),x)$. Both design variables are shared, i.e., $\mathbf{x}_{s1} = \mathbf{x}_{s2} = [w,t]$. The function-valued quantities (temperature and stress distributions) are discretized along the length of the blade to facilitate numerical calculations. Using the parameter values from Table 1 and a sample design of $[w,t] = [0.08, 0.005]$ (meters), the analysis outputs (using FPI) are $q = 0.2046$ W, $m = 0.1702$ kg, and $L = 0.057$ m.

5.4 System Design Optimization. The turbine blade design problem was solved using both MDF and IDF. A parametric study on coupling strength was then performed, demonstrating the sensitivity of MDF and IDF to this factor. In the MDF implementation, consistent coupling variable values were obtained using FPI at every optimization iteration. Note that in the MDF formulation, given in Eq. (11), the third and fourth constraints are function-valued; these were discretized and implemented as vector-valued constraints. The mass was constrained not to exceed 0.04 kg. The parameter values from Table 1 were used, and the optimal design was found to be $[w^*, t^*] = [0.0131, 0.0075]$ (both in meters). The IDF formulation is:

$$\begin{aligned}
\min_{w,t,\tilde{T}(x),\tilde{L}} \quad & q(w,t,\tilde{L}) \\
\text{subject to} \quad & T(w,t,\tilde{L},x) - T_{\text{melt}} \le 0 \\
& \delta_{\text{total}}(\tilde{T}(x)) - \delta_{\text{allow}} \le 0 \\
& \sigma_a(\tilde{L},x) - \sigma_r(\tilde{T}(x),x) \le 0 \\
& \sigma_b(t,\tilde{L},x) - \sigma_r(\tilde{T}(x),x) \le 0 \\
& m(w,t) - m_{\max} \le 0 \\
& \tilde{T}(x) - T(w,t,\tilde{L},x) = 0 \\
& \tilde{L} - L(\tilde{T}(x)) = 0 \\
& \text{and} \quad 0 \le x \le L_0 + \delta_{\text{total}}(\tilde{T}(x))
\end{aligned}$$

The function-valued coupling variable $\tilde{T}(x)$ was discretized and implemented as a vector-valued coupling variable, substantially increasing the optimization problem dimension. The sixth and seventh constraints were added to ensure system consistency at algorithmic convergence, since the requirement that the system is consistent at every step is relaxed in the IDF formulation. Using parameter values from Table 1, the optimal design found using IDF was almost identical to the MDF results: $[w^*, t^*] = [0.0128, 0.0074]$ (both in meters). The small numerical discrepancy is not unexpected due to the increase in problem dimension.

The computation time required for both MDF and IDF solutions was recorded over a range of coupling strength levels, varied by adjusting the modulus of elasticity $E(T)$. A more compliant blade results in increased blade elongation and exposed surface area, increasing the impact that the structural analysis results have on the thermal analysis. The $E(T)$ curve from Eq. (22) was multiplied by a scaling factor to produce changes in coupling strength. Figure 15 illustrates the dependence of MDF and IDF computa-

Table 1 Turbine blade design parameters

ρ	8510 kg/m^3	ρ_g	3.522 kg/m^3
L_0	0.05 m	C_d	2.0
α	12.6×10^{-6} m/K	v	100 m/s
r_b	0.5 m	T_b	300°C
ω	2100 rad/s	T_g	900°C
$\delta_{\max}$	0.05 m	ε	1.0×10^{-8}

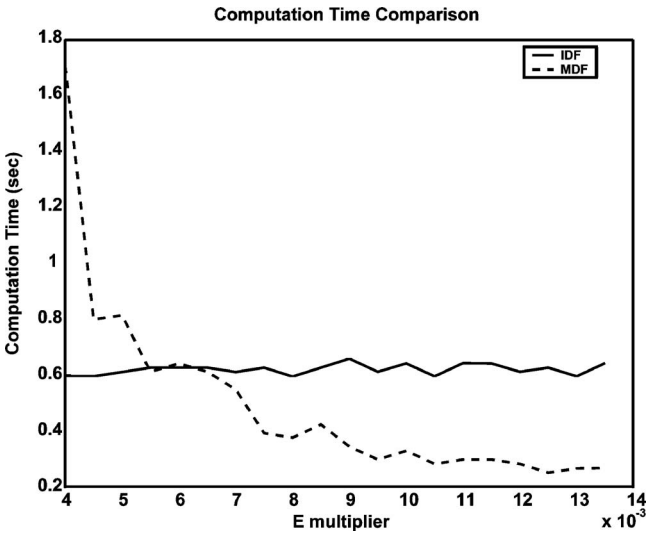

Fig. 15 Comparison of MDF and IDF solution time as a function of coupling strength

tion time on this modulus multiplier, and hence the dependence on coupling strength.

As with the previous example, IDF computation time is insensitive to coupling strength, while MDF computation time increases with coupling strength. At modulus multiplier values larger than the range illustrated, very little change in computation time was observed. In contrast to the previous example, increased stiffness does not induce infeasibility, but does result in very weak coupling as expected. Since design infeasibility is not a confounding factor as in the VAF example, MDF time monotonically increases with coupling strength. A very stiff blade results in effectively independent analyses—only one or two FPI iterations are required for system analysis. At modulus multiplier values smaller than the range presented, MDF failed due to strong coupling.

In summary, weakly coupled systems are efficiently solved with MDF, while strongly coupled systems require excessive iterations for the inner analysis loops of MDF. The computation time required for the IDF approach is virtually constant for all levels of coupling strength investigated here. In addition to verifying the predictions of IDF efficiency [9] for the case of a strongly coupled system, Figs. 10 and 15 show a very clear relationship between coupling strength and computational performance.

6 Conclusion

The implementation of the MDF and IDF formulations was studied with respect to coupling strength, solution quality, and computational implications, as well as illustrated with new and reproducible design problems. Theoretical predictions that IDF is more efficient than MDF for strongly coupled problems were verified empirically for the examples presented. An intuitive understanding of convergence conditions for FPI helped demonstrate that the relationship between inter-analysis derivative magnitudes, rather than absolute magnitudes, determines coupling strength. When considering the use of MDF and IDF formulations, clear tradeoffs exist. MDF requires less setup effort and results in lower problem dimensionality, but can incur expensive analysis iterations during solution and is not amenable to coarse-grained parallelism. If FPI is used for solution, the MDF approach may fail if convergence conditions are not met, and may be unable to find some solutions. In contrast, IDF requires more extensive setup effort and increases optimization problem dimension, but is worth the additional overhead when coupling strength is high and when there are few coupling variables in comparison to design variables. Further computational advantage for IDF implementations may be realized through concurrent processing. This enhanced understanding of the strengths and weaknesses of these two single-level formulations facilitates effective selection for the problem at hand, and points toward the possibility that other methods using simultaneous analysis and design share the desirable properties of IDF.

Acknowledgment

This work was partially supported by a National Science Foundation Graduate Research Fellowship and by the Automotive Research Center, a US Army Center of Excellence at the University of Michigan. The authors would also like to thank Ross Morrow for reviewing the manuscript and providing helpful suggestions.

References

[1] Wagner, T. C., 1993, "General Decomposition Methodology For Optimal System Design," Ph.D. thesis, University of Michigan, Ann Arbor, MI.
[2] Rogers, J. L., and Bloebaum, C. L., 1994, "Ordering Design Tasks Based on Coupling Strengths," *5th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, Panama City Beach, FL*, Sept. 7–9, Paper No. AIAA-1994-4326.
[3] Haftka, R., Sobieszczanski-Sobieski, J., and Padula, S. L., 1992, "On Options for Interdisciplinary Analysis and Design Optimization," Struct. Optim., **4**(2), pp. 65–74.
[4] Alyaqout, S. F., Papalambros, P. Y., and Ulsoy, A. G., 2005, "Quantification and Use of System Coupling in Decomposed Design Optimization Problems," *Proceedings of International Mechanical Engineering Congress and Exposition*, Nov. 5–11, Paper No. IMECE2005-81364.
[5] Sosa, M. E., Eppinger, S. D., and Rowles, C. M., 2003, "Identifying Modular and Integrative Systems and Their Impact on Design Team Interactions," ASME J. Mech. Des., **125**, pp. 240–252.
[6] Sosa, M. E., Eppinger, S. D., and Rowles, C. M., 2004, "The Misalignment of Product Architecture and Organizational Structure in Complex Product Development," Manage. Sci., **50**(12), pp. 1674–1689.
[7] Chanron, V., and Lewis, K., 2004, "Convergence and Stability in Distributed Design of Large Systems," *ASME Design Engineering Technical Conference*, Sept. 28–Oct. 2, Paper No. DETC2004-57344.
[8] Bertsekas, D. P., 1999, *Nonlinear Programming*, 2nd ed., Athena Scientific, Nashua, NH.
[9] Cramer, E. J., Dennis, J. E. Jr., Frank, P. D., Lewis, R. M., and Shubin, G. R., 1994, "Problem Formulation for Multidisciplinary Optimization," SIAM J. Optim., **4**(4), pp. 754–776.
[10] Allison, J. T., 2004, "Complex System Optimization: A Review of Analytical Target Cascading, Collaborative Optimization, and Other Formulations," MS's thesis, Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI.
[11] Allison, J. T., Kokkolaras, M., Zawislak, M., and Papalambros, P. Y., 2005, "On the Use of Analytical Target Cascading and Collaborative Optimization for Complex System Design," *6th World Conference on Structural and Multidisciplinary Optimization*, May 30–June 3.
[12] Balling, R. J., and Sobieszczanski-Sobieski, J., 1996, "Optimization of Coupled Systems: A Critical Overview of Approaches," AIAA J., **34**(1), pp. 6–17.
[13] Balling, R. J., and Wilkinson, C. A., 1997, "Execution of Multidisciplinary Design Optimization Approaches on Common Test Problems," AIAA J., **35**, pp. 178–186.
[14] Hulme, K. F., and Bloebaum, C. L., 2000, "Simulation-Based Comparison of Multidisciplinary Design Optimization Solution Strategies Using Cascade," Struct. Multidiscip. Optim., **19**(1), pp. 17–35.
[15] Sobieszczanski-Sobieski, J., and Haftka, R. T., 1997, "Multidisciplinary Aerospace Design Optimization: Survey of Recent Developments," Struct. Optim., **14**(1), 1–23.
[16] Braun, R. D., 1996, "Collaborative Optimization: An Architecture for Large-Scale Distributed Design," Ph.D. thesis, Stanford University, Stanford, CA.
[17] Alexandrov, N. M., and Lewis, R. M., 2000, "Algorithmic Perspective on Problem Formulations in MDO," *8th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, Sept. 6–8.
[18] Thareja, R., and Haftka, R., 1986, "Numerical Difficulties Associated With Using Equality Constraints to Achieve Multilevel Decomposition in Structural Optimization," *AIAA/ASME/ASCE/AHS 27th Structures, Structural Dynamics and Materials Conference. Part 1: Structures and Materials*.
[19] Chapra, S. C., and Canale, R. P., 1998, *Numerical Methods for Engineers*, 3rd ed., McGraw-Hill, New York.
[20] Hildebrand, F. B., 1974, *Introduction to Numerical Analysis*, 2nd ed., McGraw-Hill, New York.
[21] Kim, H. M., 2001, "Target Cascading in Optimal System Design," Ph.D. thesis, University of Michigan, Ann Arbor, MI.
[22] Wells Manufacturing Corp., 1999, "Making Sense of Engine Airflow," *Counterpoint: The Electronic Diagnostic and Driveability Resource*, Vol. 3(3), pp. 1–3.
[23] Papalambros, P. Y., and Wilde, D. J., 2000, *Principles of Optimal Design: Modeling and Computation*, 2nd ed., Cambridge University Press, New York.
[24] University of Michigan, Aerospace Engineering display, François-Xavier Bagnoud Building, April, 2006.
[25] Incropera, F. P., and DeWitt, D. P., 2002, *Introduction to Heat Transfer*, John Wiley and Sons, Inc., New York.
[26] MATWEB Material Property Data, http://www.matweb.com/, accessed April 9, 2004.