Multidisciplinary Design Optimization of Dynamic Engineering Systems

James T. Allison,* Daniel R. Herber†

University of Illinois at Urbana-Champaign, Urbana, IL, 61801, USA

Dynamic engineering systems are playing an increasingly important role in society, especially as active and autonomous dynamic systems become more mature and prevalent across a variety of domains. Successful design of complex dynamic systems requires multidisciplinary analysis and design techniques. While multidisciplinary design optimization (MDO) has been used successfully for the development of many dynamic systems, the established MDO formulations were developed around fundamentally static system models. We still lack general MDO approaches that address the specific needs of dynamic system design. In this article we review the use of MDO for dynamic system design, identify associated challenges, discuss related efforts such as optimal control, and present a vision for fully integrated design approaches. Finally, we lay out a set of exciting new directions that provide an opportunity for fundamental work in MDO.

Nomenclature

$\mathbf{a}(\cdot)$	=	analysis function
$\mathbf{a}, \mathbf{b}$	=	example problem parameters
α, β	=	energy domain designations
$\mathbf{A}$	=	state matrix for a linear and time invariant system
$\mathbf{B}$	=	input matrix for a linear and time invariant system
c	=	suspension damping coefficient
ε	=	convergence tolerance
$f(\cdot)$	=	design objective function
$\mathbf{f}(\cdot)$	=	derivative function
$\mathbf{f_a}(\cdot)$	=	algebraic constraint
$\mathbf{g}(\cdot)$	=	design constraint functions
$\mathbf{g_p}(\cdot)$	=	physical system constraints
$\boldsymbol{\gamma}(\mathrm{t})$	=	algebraic variable vector
h_i	=	time step
i	=	time step index
j	=	Gauss-Seidel block index, multiple-shooting time segment index
k	=	iteration counter
k_s	=	suspension spring stiffness
$\mathbf{K}$	=	gain matrix
$\mathbf{K}_*$	=	optimal gain matrix

*Assistant Professor, Department of Industrial and Enterprise Systems Engineering, 104 S. Mathews Ave, Urbana IL, 61801, AIAA Member.

†Graduate Student, Department of Industrial and Enterprise Systems Engineering, 104 S. Mathews Ave, Urbana IL, 61801, AIAA Student Member.

$L(\cdot)$	=	Lagrange or running cost term
m	=	number of Gauss-Seidel coordinate blocks
n_s	=	number of states
n_t	=	number of time steps
n_T	=	number of time segments
$\phi(\cdot)$	=	cost function
$\phi_*(\cdot)$	=	optimal-value function (inner loop solution)
$\hat{\phi}(\cdot)$	=	alternative plant design objective function
$\psi(\cdot)$	=	Mayer or terminal cost term
$\pi(\cdot)$	=	augmented Lagrangian penalty function
t	=	time
t_F	=	length of the time horizon
t_i	=	time at step i
T_j	=	time at the end of time segment j
$\mathbf{u}(t)$	=	control input trajectories
$\mathbf{u}_*(t)$	=	optimal control trajectories
$\mathbf{u}_i$	=	control input at time step i
$\mathbf{U}$	=	matrix discretization of $\mathbf{u}(t)$
$\mathbf{x}$	=	optimization variable vector
$\mathbf{x}_*$	=	optimal solution
$\mathbf{x}^k$	=	solution estimate at iteration k
$\mathbf{x_c}$	=	control system design variable vector
$\mathbf{x_p}$	=	physical system design variable vector
$\mathbf{x_{p*}}$	=	optimal plant design
$\mathbb{X}$	=	Cartesian product of closed convex sets
$\boldsymbol{\xi}(t)$	=	state variable trajectories
$\boldsymbol{\xi}_*(t)$	=	optimal state trajectories
$\boldsymbol{\xi}_i$	=	state at time step i
$\hat{\boldsymbol{\xi}}(t)$	=	subset of state trajectories
$\dot{\boldsymbol{\xi}}(\cdot)$	=	time derivative of $\boldsymbol{\xi}(t)$
$\boldsymbol{\Xi}$	=	discretization of $\boldsymbol{\xi}(t)$
$\hat{\boldsymbol{\Xi}}$	=	subset of discretized state trajectories
$\mathbf{y}$	=	coupling variable
$\mathbf{Y}$	=	matrix of initial state values for multiple shooting time segments
$\boldsymbol{\zeta}(\cdot)$	=	defect constraint functions (residuals)
$\boldsymbol{\zeta}_i(\cdot)$	=	defect constraint between time segments

I. Introduction

Dynamics, or system state evolution through time, is an increasingly important aspect of systems designed by engineers. Most notably, 'smart' engineering systems that are actively controlled via electronic feedback mechanisms are becoming exceedingly prevalent, and the dynamic behavior of these systems is core to system value (e.g., renewable energy systems and vehicle electrification). In addition, many groups now recognize the importance of autonomous[1] and semi-autonomous[2] dynamic systems across several domains, including manufacturing and its impact on economic competitiveness.[3,4] Active and autonomous systems, however, pose special design challenges. Physical elements of active systems need to be designed differently than for passive dynamic systems. Physical dynamics and control systems should be designed in an integrated way to achieve the best possible system performance, and sometimes integrated design approaches are required to obtain feasible designs for especially demanding dynamic systems.

The value of integrated design approaches for mechatronic and other active systems has long been recognized,[5,6] especially for systems with strong coupling between physical and control system design (e.g., flexible robotics[7–9]). From a dynamics perspective, the design of the physical elements of a system and its control system are tightly integrated, yet compartmentalized design processes developed for the creation of passive physical systems are still in widespread use. Often a sequential design process is used, where the physical system is designed first (often relying on legacy design objectives and processes for mechanical or other physical systems), followed by control system design. Fixing physical design before moving on to control design reduces design flexibility and produces an artificially small feasible design domain, and except in rare circumstances produces suboptimal results. In extreme cases, engineers may be unable to find any design that meets system requirements using a sequential approach. Integrated approaches, however, can lead to system-optimal designs by exploiting synergy between physical and control design decisions, and in some cases enable solution of previously unsolvable problems.[10]

Adopting integrated dynamic system design approaches has clear benefit, but has proven to be challenging. Organizational and technical issues have made the transition toward integrated design methods difficult. Even with extensive integration efforts, some system interactions may be overlooked, resulting in reduced system performance. For example, modern agricultural harvesters maintain header height within a narrow window in order to harvest crops effectively (a header is the component in front of the vehicle that is designed to harvest a particular crop.) Difficulty in controlling header height is one factor that limits harvester speed. It was discovered recently that further performance improvements cannot be obtained via control design changes.[11] The interaction between physical system and control system design was not fully accounted for in the original vehicle design, and physical system redesign is necessary to achieve better performance.

Overlooking important interactions in a dynamic system model used for design may also result in unexpected results and in some cases spectacular failures. For example, at the June 2000 opening of the Millennium Bridge in the United Kingdom, pedestrians excited lateral vibrations in this passive dynamic system. These vibrations required pedestrians to walk with a synchronized teetering motion to maintain stability, amplifying lateral vibrations and rendering the bridge unusable.[12–17] While designing the bridge, engineers accounted for the effect of pedestrians walking on the bridge, but did not account for the effect of bridge motion on pedestrians. Limited redesign[18–21] was performed to attenuate lateral vibrations using active control, but more comprehensive integrated modeling and design processes early on may have prevented the need for expensive redesign. Similar unexpected behaviors that hamper dynamic system design efforts across a broad range of engineering domains may be avoided by developing models with more complete dynamic interactions, and by developing and adopting improved design methodologies created specifically for dynamic systems.

In addition to organizational issues, such as compartmentalized legacy design processes and difficulties in managing system interactions, several technical challenges exist as well. Optimization-based approaches for integrated dynamic system design require sophisticated system models. Ideally these models provide accurate representation of full system dynamics, are validated and accurate,[22,23] and are computationally efficient. Models for complete system design should also provide flexibility in both physical and control system design spaces,[24] as well as incorporate multiple disciplines and important system interactions.[25–29] Models have greater utility in design when fidelity can be adjusted to accommodate different needs at different design phases.[30] Many of these modeling objectives are competing, so tradeoffs must be made. Extensive efforts have been made to improve the value of models in dynamic system design. For example, surrogate models that approximate high-fidelity models are computationally less expensive to evaluate, but the expense of sampling required to build surrogate models may be significant.[31–34] Model reduction techniques may also be used to reduce the computational expense of dynamic system models,[34–37] but the resulting models have reduced accuracy. Scaling techniques have been investigated as a way of providing flexibility in the physical design space[38,39] without needing to develop physics-based models, but are often valid only over a fairly limited domain.

Models need to account for complete system dynamics while allowing for changes in physical system design. Often models developed for control system design provide outstanding predictions of system dynamics, but are based on a fixed physical system design. For example, engineers developing a control system for an electric motor can use a dynamic model based on measurable physical parameters, such as inductance or resistance. This is fine as long as the physical design does not change. If it does, the parameters must be identified and validated again. If another set of engineers is developing the physical design of a motor, they would need a different type of model that can predict system behavior based on independent physical design variables — variables that engineers have direct control over — such as geometric dimensions. Models that are developed for physical system design allow for physical design changes, but often are based on simplified dynamics or static analysis. Achieving full design space flexibility simultaneously with an accurate representation of system dynamics normally requires a substantial investment in model development, and can be a bottleneck in the adoption of fully-integrated dynamic system design.

While the transition to integrated design methods for dynamic systems may seem replete with obstacles, this sort of transition is not without precedent. The coupling between physical system design and control system design is analogous to the coupling between product design and manufacturing. In both cases, the conventional approach is sequential, which often proves to be restrictive and inefficient. Design for manufacture (DFM) has successfully addressed the coupling between product design and manufacturing by accounting for manufacturing needs during product development.[40] Achieving integrated design of dynamic systems will require an effort similar in magnitude to the effort that was needed to develop successful DFM methods.

Multidisciplinary design optimization (MDO) offers a solid framework for moving dynamic system design theory and practice toward a more fully-integrated state. Traditionally, MDO work has aimed to integrate previously independent analysis and design activities to improve engineering system performance and reduce development costs.[41,42] In this article we will review how MDO has been used in dynamic system design, explore opportunities for enhanced design capabilities based on MDO, and identify promising new directions for fundamental work in MDO.

While numerous dynamic systems have been designed using MDO methods, the established MDO formulations largely are based on static system analysis or black-box simulations, and do not address system dynamics explicitly. As a result, the nuances of dynamic behavior are implicitly deemphasized, and challenges related to system dynamics must be addressed on a case-by-case basis. The ever-growing scale and complexity of modern dynamic systems is taxing conventional design methodologies. Fundamentally different MDO formulations that embrace time-dependent behavior and address system dynamics directly are needed to meet these demands and realize new dynamic system capabilities.

Here we define Multidisciplinary Dynamic System Design Optimization (MDSDO) as a branch of MDO that deals with systems where the evolution of system state through time is a critical element of performance, where multiple disciplines, energy domains, models, or subsystems must be integrated, and where the unique properties of dynamic systems are exploited to improve system performance and yield efficient problem solutions. Active systems—systems that use active control to govern behavior—are playing an increasingly important role in society, and are a particularly important application of MDSDO.

We acknowledge the extensive work related to dynamic system design performed in fields such as optimal control,[43,44] robotics,[45] structural dynamics,[46,47] and cyber-physical systems.[48] Each of these areas tackles an important piece of the larger dynamic system design problem. Here we aim to bring these and other disciplines together, explore their complementary relationships, offer a high-level perspective regarding the future of dynamic system design, and develop a comprehensive vision for MDSDO. Section II discusses more deeply the elements of MDSDO, including multidisciplinary analysis, optimal physical system design, optimal control, and integrated dynamic system design methods. Section III reviews how MDO has been applied so far to dynamic system design, Section IV outlines research directions required to build more complete theory and tools for MDO-based dynamic system design, and Section V offers concluding remarks.

II. Multidisciplinary Dynamic System Design Optimization

Here we are concerned with the design optimization of engineering systems where the evolution of system state through time is central to the functionality or value of a system. These dynamic engineering systems may be passive (time variation due only to natural system dynamics) or active (controlled via electronic feedback). In most real dynamic engineering systems, multiple interacting energy domains are involved, such as mechanical, thermal, electronic, hydraulic, etc., so multidisciplinary analysis (MDA) is required for successful design. These MDA models may be used to support physical or control system design decisions, or support integrated design approaches that consider physical and control system design decisions simultaneously. In this section, the current state of each of these topics will be reviewed.

A. Multidisciplinary Analysis of Dynamic Systems

In performing MDA for dynamic systems we aim to capture the effects of each energy domain on the dynamics of the other domains. For example, when analyzing robotic systems we must account for the coupling between mechanical and electrical dynamics. Otherwise, an independent mechanical system model might predict incorrectly that electric actuator state could change instantly. Suppose we have two energy domains, α and β, with the respective state variable trajectories $\boldsymbol{\xi}_\alpha(t)$ and $\boldsymbol{\xi}_\beta(t)$. The dynamic response of each domain can be modeled using its own set of governing differential equations, but a more complete multidisciplinary model accounts for the dynamic interaction between α and β. The resulting coupled system of differential equations is:

$$\begin{aligned} \dot{\boldsymbol{\xi}}_\alpha(t) &= \mathbf{f}_\alpha\left(\boldsymbol{\xi}_\alpha(t), \boldsymbol{\xi}_\beta(t), \mathbf{u}_\alpha(t), t\right) \quad (1) \\ \dot{\boldsymbol{\xi}}_\beta(t) &= \mathbf{f}_\beta\left(\boldsymbol{\xi}_\alpha(t), \boldsymbol{\xi}_\beta(t), \mathbf{u}_\beta(t), t\right), \quad (2) \end{aligned}$$

where $\mathbf{f}_\alpha(\cdot)$ and $\mathbf{f}_\beta(\cdot)$ are the derivative functions for each domain, and $\mathbf{u}_\alpha(t)$ and $\mathbf{u}_\beta(t)$ are control inputs that are present if the system is actively controlled. Observe that $\dot{\boldsymbol{\xi}}_\alpha(t)$ depends on $\boldsymbol{\xi}_\beta(t)$, and $\dot{\boldsymbol{\xi}}_\beta(t)$ depends on $\boldsymbol{\xi}_\alpha(t)$. The fully integrated system model can be represented more compactly if we define $\boldsymbol{\xi}(t) = \left[\boldsymbol{\xi}_\alpha^{\mathrm{T}}(t), \boldsymbol{\xi}_\beta^{\mathrm{T}}(t)\right]^{\mathrm{T}}$, $\mathbf{u} = \left[\mathbf{u}_\alpha^{\mathrm{T}}(t), \mathbf{u}_\beta^{\mathrm{T}}(t)\right]^{\mathrm{T}}$, and $\mathbf{f}(\cdot) = \left[\mathbf{f}_\alpha^{\mathrm{T}}(t), \mathbf{f}_\beta^{\mathrm{T}}(t)\right]^{\mathrm{T}}$:

$$\dot{\boldsymbol{\xi}}(t) = \mathbf{f}(\boldsymbol{\xi}(t), \mathbf{u}(t), t). \quad (3)$$

The multidisciplinary dynamic system model given in Eqn. (3) may be constructed using one of several well-developed strategies, such as bond graph modeling for lumped-parameter system models[29] or high-fidelity multi-physics models for systems involving continuum mechanics.[26–28] If a commercial tool that integrates all the desired domains is unavailable, then a possible solution is to integrate separate software tools. For example, a block-diagram modeling environment appropriate for control system modeling may be incorporated with a multi-body dynamics model of a mechanical system using a co-simulation approach.[49]

In the development of dynamic engineering systems we would often like to impose constraints on state trajectories (i.e., path constraints) for design requirements or modeling expedience. Adding an equality constraint to a system of differential equations without adding a new state variable produces a system of differential algebraic equations (DAEs).[50] A DAE in semi-explicit form is:

$$\begin{aligned} \dot{\boldsymbol{\xi}}(t) &= \mathbf{f}(\boldsymbol{\xi}(t), \boldsymbol{\gamma}(t), \mathbf{u}(t), t) \quad (4) \\ \mathbf{0} &= \mathbf{f}_{\mathbf{a}}(\boldsymbol{\xi}(t), \boldsymbol{\gamma}(t), \mathbf{u}(t), t), \quad (5) \end{aligned}$$

where $\mathbf{f}_{\mathbf{a}}(\cdot)$ is an algebraic constraint and $\boldsymbol{\gamma}(t)$ is the algebraic variable (i.e., its time derivative $\dot{\boldsymbol{\gamma}}(t)$ does not appear in the equations). Most DAE algorithms require that Eqn. (5) can be solved for $\boldsymbol{\gamma}(t)$ (i.e., the Jacobian of the algebraic constraint must not be singular). A DAE that satisfies this requirement is an index-1 DAE, where the index identifies the number of differentiations required to transform a DAE into an ordinary differential equation.

Inequality path constraints are sometimes needed to model a dynamic system design problem (e.g., temperature, position, or force limits). If these bounds are reached then algebraic inequality constraints become active. Constraints may enter or exit activity multiple times during a simulation. An ODE can be transformed into a DAE when an inequality path constraint becomes active. Imposing a new algebraic relationship like this reduces a system's degrees of freedom. For every lost degree of freedom, a state variables must become an algebraic variable, i.e., a variable completely determined by state variables via the algebraic constraint. In active systems, control inputs normally become the algebraic variable since they are independent, while state variables must satisfy physics.[51] As additional inequality constraints become active, the DAE index may increase, increasing solution difficulty.

B. Optimal Control System Design

The value of MDA models extends beyond the analysis of existing systems; they are important for optimal design of new systems. Simulations of MDA models predict system behavior given its specifications, but can also be used for the inverse task (i.e., design): identifying a system specification that produces desired behavior. Physical system design and control system design both contribute to overall dynamic behavior in actively controlled systems. While physical and control system design normally are tightly coupled, they are often treated separately in conventional sequential design processes. Here we begin our exploration of dynamic system design with a brief review of optimal control, a design approach that aims to identify the control design that produces the best possible system performance.

In optimal control, a control design is sought that minimizes a cost function, often of this form:

$$\phi(\boldsymbol{\xi}(t), \mathbf{u}(t), t) = \psi(\boldsymbol{\xi}(t_F), t_F) + \int_0^{t_F} L(\boldsymbol{\xi}(t), \mathbf{u}(t), t)dt, \tag{6}$$

where $\psi(\cdot)$ and $L(\cdot)$ are the Mayer (terminal cost) and Lagrange (running cost) terms and t_F is the length of the time horizon considered in the design problem. If both terms are present, the function is often termed a Bolza objective. The problem is an open-loop control problem if the control trajectory $\mathbf{u}(t)$ is the optimization variable. Observe that optimal control problems are solved with respect to an infinite-dimensional control trajectory, as opposed to a finite-dimensional optimization vector used in typical MDO formulations.

A classical approach for solving optimal control problems is to apply optimality conditions—such as Pontryagin's Maximum Principle (PMP)[43,44]—to identify the optimal control trajectory $\mathbf{u}_*(t)$ that minimizes $\phi(\cdot)$. If an analytical solution to the optimality conditions cannot be found, the resulting boundary value problem (BVP) often can be solved numerically. This approach is an 'optimize then discretize' approach, since a BVP obtained via optimality conditions is discretized and then solved.[52]

In most practical implementations we need to design a feedback controller (and often we need to design an observer to estimate states that cannot be measured directly). A simple form of feedback control is a full-state feedback regulator, where the control input is defined as $\mathbf{u}(t) = -\mathbf{K}\boldsymbol{\xi}(t)$. Assuming this control structure, the optimal control problem may be solved with respect to the gain matrix $\mathbf{K}$ instead of the control trajectory. In addition, if the system model is linear and time-invariant, i.e., the derivative function can be written as:

$$\dot{\boldsymbol{\xi}}(t) = \mathbf{f}(\boldsymbol{\xi}(t), \mathbf{u}(t)) = \mathbf{A}\boldsymbol{\xi}(t) + \mathbf{B}\mathbf{u}(t),$$

and if $\phi(\cdot)$ is quadratic, then a closed-form solution for $\mathbf{u}_*(t) = -\mathbf{K}_*\boldsymbol{\xi}_*(t)$ can be derived. The resulting optimal control law is known as a linear quadratic regulator (LQR).[53]

Optimal control approaches based on PMP are known as indirect methods. Direct methods are an alternative approach where optimal control problems are discretized first and then transcribed to a nonlinear programming (NLP) formulation. In other words, an infinite-dimensional optimal control problem is transcribed to a finite-dimensional NLP.[52,54,55] Direct Transcription (DT) is a family of 'discretize-then-optimize' methods for optimal control that use this strategy.[52,56] DT is a special case of the all-at-once (AAO) MDO formulation,[57] also known as simultaneous analysis and design (SAND).[58] In DT, an NLP algorithm simultaneously solves the system state equations and the system optimization problem, eliminating the need for forward simulation. This is accomplished by applying a numerical integration method (such as collocation[50,52]) to convert differential state equations to a system of algebraic equations, and discretizing the state and control trajectories. The resulting algebraic equations, known as defect constraints ($\boldsymbol{\zeta}(\cdot)$), are posed as equality constraints in the optimization problem, and the discretized state and control trajectories are treated as optimization variables. If $\mathbf{U}$ is a matrix where row i is the control vector at time step i (i.e., $\mathbf{u}_i$), and $\boldsymbol{\Xi}$ is a matrix where row i is the state vector at time step i (i.e., $\boldsymbol{\xi}_i$), then the following is a DT formulation for optimal control:

$$\begin{aligned} \min_{\mathbf{U},\boldsymbol{\Xi}} \quad & \sum_{i=1}^{n_t-1} L(\mathbf{u}_i, \boldsymbol{\xi}_i)h_i \\ \text{subject to:} \quad & \boldsymbol{\zeta}(\mathbf{U}, \boldsymbol{\Xi}) = \mathbf{0} \end{aligned} \tag{7}$$

Here only the Lagrange cost term is included in the objective, h_i is the time step size at step i, and n_t is the number of time steps. Note that incorporating $\mathbf{U}$ and $\boldsymbol{\Xi}$ as optimization variables increases problem dimension.

Historically DT has been applied only to open-loop optimal control (trajectory optimization in particular[56,59–62]), but recently has been extended to more general dynamic system design problems, including nonlinear feedback control design[10,63,64] and integrated physical system and control system design (co-design).[24] DT is related closely to other discretize-then-optimize techniques, such as pseudospectral methods,[60,63,65–71] adjoint state methods,[72–74] and temporal spectral element methods,[75,76] as well as model predictive control.[77] A number of commercial[78–83] and open-source[66,84,85] DT software implementations are available.

The defect constraints in DT are solved simultaneously, meaning that forward simulation is not required to obtain state trajectories. Higher-order implicit quadrature methods are normally impractical for forward simulation, but work well when the resulting defect constraints are instead solved simultaneously. When these higher-order methods are used with DT, high solution accuracy can be maintained even with large time steps, and larger step sizes reduce optimization problem dimension.[52,62,86]

Even with large time steps, DT optimization problems still have much higher dimension than other discretize-then-optimize approaches. Why then would one consider using DT? First, DT optimization problems have special structures that promote efficient computation, in some cases even exponential convergence.[65] Optimization variables appear explicitly in constraint functions, making sensitivities easier to compute, and when analytical derivatives are impractical to obtain, the Jacobian sparsity pattern enables efficient application of sparse finite differences.[56] In addition, defect constraints are independent, enabling fine-grained parallel computing. For linear dynamic systems, DT problem formulations are often either quadratic or linear programs, allowing for especially efficient problem solution.

Another reason to consider DT is the ability to impose inequality constraints on trajectories, something that generally cannot be done with indirect methods. Equality path constraints are also easily included, extending applicability to DAE systems. DT also works well on challenging singular optimal control problems, and is often successful at maintaining numerical stability when solving highly nonlinear problems.[56]

DT possesses the unique property that system dynamics are represented directly in the optimization formulation, and offers one promising direction for development of MDO formulations for dynamic system design. DT will be revisited in greater detail in Sections IV after additional context is developed, with particular emphasis on extension of DT to co-design applications.

Dynamic engineering systems are often multidisciplinary, and MDA techniques are needed to model interactions across multiple energy domains (cf. Eqn. (3)). In practice, multidisciplinary dynamic system models are used widely, but design is usually limited to a single discipline. Even if interactions across multiple domains are modeled with great sophistication, design efforts that concentrate on dynamic system performance typically address control design only. For example, many aeroservoelasticity design studies (e.g.,[87–89]) use advanced multidisciplinary models that capture complicated aerodynamic and structural interactions, but the physical system is held fixed while the control system is designed. This addresses only part of the dynamic system design problem. Physical system design has an important, if not dominant, influence on system dynamics. By some definitions of MDO, optimal control studies do not constitute MDO, even if MDA is used, since the design component involves only one discipline. A systems-oriented approach incorporates multidisciplinary design in addition to MDA.

C. Optimal Physical System Design

The role of physical system dynamics should be a core consideration in dynamic engineering system design. In other words, the onus of optimizing dynamic system performance rests also on engineers designing physical elements of the system, not just the control system engineers. When making physical system design decisions we should include comprehensive treatment of system dynamics if we want to capitalize on passive dynamics. Many physical system design optimization efforts, however, incorporate simplified system dynamics—such as steady-state or pseudo-static models—or static analysis that neglects dynamic effects altogether. Design objectives are often approximations of actual dynamic system performance metrics (e.g., mass[90] or gravity balance[91]). While these simplifications are sufficient in some cases, performance can be improved by utilizing more complete dynamic models when designing physical systems, and improved models will also enhance the ability to design more challenging dynamic systems.

While comprehensive dynamic models are in use, they are normally developed for control design, and do not allow for physical design changes. The next generation of system models need to incorporate realistic dynamics while providing flexibility in the physical design space, i.e., we need models of the form:

$$\dot{\boldsymbol{\xi}}(t) = \mathbf{f}(\boldsymbol{\xi}(t), \mathbf{x_p}, t), \tag{8}$$

where $\mathbf{x_p}$ is a vector of physical system (or plant) design variables. Models of this type require more development effort than models with a fixed physical system design (e.g., Eqn. (3)).

Consider the following passive physical dynamic system design optimization problem:

$$\begin{aligned} \min_{\mathbf{x_p}} \quad & \phi(\boldsymbol{\xi}(t), \mathbf{x_p}, t) \\ \text{s.t.} \quad & \mathbf{g_p}(\boldsymbol{\xi}(t), \mathbf{x_p}) \leq \mathbf{0} \\ & \dot{\boldsymbol{\xi}}(t) - \mathbf{f}(\boldsymbol{\xi}(t), \mathbf{x_p}, t) = \mathbf{0} \end{aligned} \tag{9}$$

Here the system objective function is minimized with respect to $\mathbf{x_p}$ only. We also introduce a new function, $\mathbf{g_p}(\cdot)$, that quantifies physical system constraints such as stress, deflection, or geometric requirements.[92] This function depends on both physical design and state variables, accounting fully for the influence of dynamic response on physical design requirements. Other more simplified design formulations neglect direct dependence of plant constraints on $\boldsymbol{\xi}(t)$. The objective function used here is the overall system objective that depends on dynamic response, as opposed to a static or simplified dynamic physical design proxy objective. Variants of the formulation given in Prob. (9) have been studied; Wang and Arora reviewed methods for solving this class of problems.[93] These formulations were based on DT where states were discretized and treated as independent variables.

D. Optimal Dynamic System Design

Approaches for optimizing the physical and control system design of dynamic systems separately were just reviewed. Independent solution of these problems, however, will not lead to the best possible system design. An integrated solution approach is required to capitalize on the synergistic relationship between physical and control system design. Conventional sequential system design approaches[6,46,94–97] only account partially for coupling between physical system (plant) and control system design decisions, producing suboptimal results.[98] In sequential design, control design is performed after plant design is complete. If optimization is employed for each task, the sequential approach consists of solving Prob. (9) to obtain the optimal plant design $\mathbf{x_{p*}}$, which is then used as the basis for solving the optimal control problem—minimizing $\phi(\cdot)$ from Eqn. (6)—to obtain the optimal control trajectory $\mathbf{u}_*(t)$. This process is illustrated in Fig. 1.

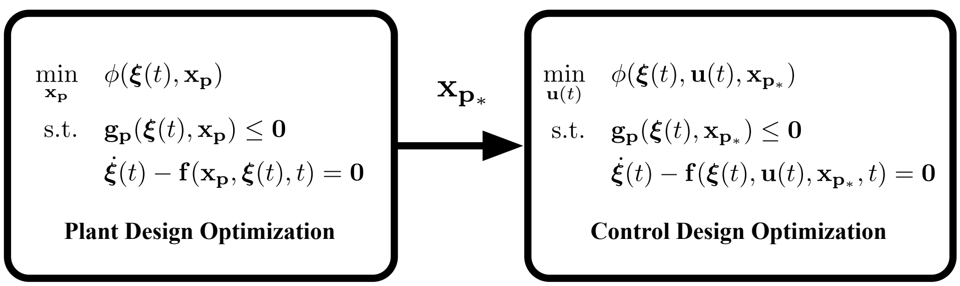

Figure 1. Sequential design process for actively-controlled dynamic systems.

While this sequential process often produces feasible system design, better methods exist. As actively controlled systems become more complex and performance requirements more stringent, sequential system design may fall short, motivating the use of more integrated design methods.

The sequential system design process illustrated in Fig. 1 represents the case where plant design is based on passive dynamics. For example, in designing a passive-active automotive suspension[99] we may start by designing the passive suspension (i.e., mechanical linkage, spring, damper, etc.) to optimize comfort and handling, but with no control actuation. The resulting plant design may then be used as a basis for designing the active control system (holding plant design fixed). Ideally we use the same system objective function $\phi(\cdot)$ in both design phases. Often, due to legacy design practices for physical systems, an alternative plant design objective function $\hat{\phi}(\cdot)$ is used instead (e.g., mass). The effects of using separate plant and control objective functions are discussed below.

We can classify plant design objectives into five types:

Case 1: Passive plant design, original system objective function: $\phi(\boldsymbol{\xi}(t), \mathbf{x_p})$.
Case 2: Passive plant design, approximate system objective function: $\hat{\phi}(\boldsymbol{\xi}(t), \mathbf{x_p})$.

Case 3: Static plant design, approximate system objective function: $\hat{\phi}(\mathbf{x_p})$.
Case 4: Active plant design, original system objective function: $\phi(\boldsymbol{\xi}(t), \mathbf{u}(t), \mathbf{x_p})$.
Case 5: Active plant design, approximate system objective function: $\hat{\phi}(\boldsymbol{\xi}(t), \mathbf{u}(t), \mathbf{x_p})$.

Case 1 corresponds to Fig. 1. Case 2 involves an approximate system objective function that still depends on the complete system dynamics, meaning that a dynamic system simulation is required to evaluate $\hat{\phi}(\cdot)$, but is still limited because it does not incorporate active control. In addition, some systems cannot be simulated without active control, so Cases 1 and 2 are not always available options. For example, an active automotive suspension[24] may be simulated in a passive mode, but a robotic manipulator requires control actuation to simulate. Bowling et al., however, did introduce 'dynamic capability' equations based on system dynamics that guide physical robot design toward improved active dynamic performance without requiring control design.[100]

Case 3 is a more significant simplification where static or frequency-based analysis eliminates the need for simulation. The objective depends only on plant design. Ravichandran et al. presented an example of a Case 3 sequential design approach for reducing energy consumption of a counterbalanced robotic manipulator.[91] For very slow pseudo-static movements, energy consumption is approximately minimized if the manipulator is designed to have perfect gravity balance (i.e., any manipulator position can be held with zero actuation torque). Using gravity balance as the plant design objective $\hat{\phi}(\cdot)$ simplifies the problem, but is inaccurate for high-speed motions.[101] Allison presented a more complete formulation that produces system-optimal results for high-speed counterbalanced manipulators.[21] Trivedi et al. also used a Case 3 approach for soft robotic manipulator design where a static model was used for physical system optimization.[97]

The commonly used Case 2 and 3 objectives arise when separate objectives for plant and control design are specified. Several researchers have asserted that active control system design problems are fundamentally multi-objective.[102–104] While co-design problems may indeed be multi-objective because of intrinsic tradeoffs in the system (e.g., cost vs. performance), a problem is not automatically multi-objective because it is a co-design problem. When separate plant and control objectives are used, the plant objective is often approximation of the real system objective (e.g., gravity balance approximating energy efficiency). Separate plant objectives may also be used because of legacy design processes. For example, when physical design is performed in isolation, using a plant objective that is not directly connected to dynamics or active control, such as mass or other static measures, is a logical choice. These legacy design paradigms, however, are firmly established. Abandoning familiar design objectives and adopting objectives that more accurately reflect overall system purpose may be challenging when working to adopt an integrated systems design approach. Part of designing with a holistic systems perspective is to develop system components that, when combined together, produce the best overall system behavior, as opposed to optimizing the components individually. Integrated system design requires consistent use of the same system objective across all system elements (or the same set of objectives if the system design problem is inherently multi-objective). Cases 1 and 4 are examples of approaches that utilize a common objective.

Cases 4 and 5 are fundamentally different from the others in that the objective depends explicitly on control design. Control input is considered during plant design, but is held fixed. Incorporating the effects of active control improves solution quality. It also opens up the possibility of an iterated sequential design process where, after completing a single pass of sequential design, we can feed $\mathbf{u}_*(t)$ back into the plant design problem and iterate. Pil and Asada demonstrated a modified form of Case 3 that allows for iteration and incorporates physical prototyping guided by control design sensitivity data,[105] and Padula et al. introduced a three-stage iterated sequential method that includes plant, control, and system-level design.[106]

The iterated sequential method based on Case 4 is a special case of the Block Coordinate Descent (BCD) optimization method.[107] To understand this connection with BCD, suppose we have an optimization problem: $\min_{\mathbf{x}} f(\mathbf{x}), \mathbf{x} \in \mathbb{X}$, where $\mathbb{X} = \mathbb{X}_1 \times \mathbb{X}_2 \times \ldots \times \mathbb{X}_m$ and each $\mathbb{X}_j$ is a closed convex set. The optimization vector may be partitioned into 'blocks' of coordinates: $\mathbf{x}_j \in \mathbb{X}_j$, $j = 1, \ldots, m$. An optimization subproblem for each coordinate block ($j = 1, \ldots, m$) may then be defined: $\min_{\mathbf{x}_j} f(\mathbf{x}), \mathbf{x}_j \in \mathbb{X}_j$. Each subproblem may either be solved simultaneously (Jacobi iteration), or in sequence using the most recently updated values for $\mathbf{x}$ (Gauss-Seidel method), and iterated. BCD converges to the solution of the original problem if each subproblem has a unique solution. When the iterated Case 4 sequential design approach is used, it is a BCD solution to the fully-integrated plant and control design (co-design) optimization problem:

$$\begin{aligned} \min_{\mathbf{x_p}, \mathbf{u}(t)} \quad & \phi(\boldsymbol{\xi}(t), \mathbf{u}(t), \mathbf{x_p}, t) \\ \text{s.t.} \quad & \mathbf{g_p}(\boldsymbol{\xi}(t), \mathbf{x_p}) \leq \mathbf{0} \\ & \dot{\boldsymbol{\xi}}(t) - \mathbf{f}(\boldsymbol{\xi}(t), \mathbf{u}(t), \mathbf{x_p}, t) = \mathbf{0} \end{aligned} \tag{10}$$

The solution to Prob. (10) is the system-optimal design; it accounts for all dynamic system interactions and plant-control design coupling, resulting in minimal $\phi(\cdot)$. This solution is our standard of comparison for all active system design methods. Note that the plant design constraints do not depend directly on $\mathbf{u}(t)$, but are influenced indirectly by control design through state trajectories. The formulation in Prob. (10) is often referred to as the simultaneous co-design method, since plant and control design decisions are made simultaneously.

In a BCD solution of Prob. (10) using the Gauss-Seidel method (iterated sequential method Case 4), the number of coordinate blocks is $m = 2$; $j = 1$ corresponds to the plant design coordinate block, and $j = 2$ corresponds to the control design coordinate block. If we assume for generality that a vector $\mathbf{x_c}$ is a discretization (e.g., $\mathbf{U}$ from Eqn. (7)) or parameterization (e.g., full-state feedback gain matrix $\mathbf{K}$) of the control design, the co-design problem becomes a nonlinear program. Satisfaction of the state equations is an important distinction between solution approaches, and will be discussed in the next section. The BCD co-design algorithm is:

1) set $k = 1$, initialize $\mathbf{x}^k$ and ε
2) $\mathbf{x_p}^{k+1} = \arg\min_{\mathbf{x_p}} \phi(\boldsymbol{\xi}(t), \mathbf{x_c}^k, \mathbf{x_p})$, subject to: $\mathbf{g_p}(\boldsymbol{\xi}(t), \mathbf{x_c}^k, \mathbf{x_p}) \leq \mathbf{0}$
3) $\mathbf{x_c}^{k+1} = \arg\min_{\mathbf{x_c}} \phi(\boldsymbol{\xi}(t), \mathbf{x_c}, \mathbf{x_p}^{k+1})$, subject to: $\mathbf{g_p}(\boldsymbol{\xi}(t), \mathbf{x_c}, \mathbf{x_p}^{k+1}) \leq \mathbf{0}$
4) if $\|\mathbf{x}^{k+1} - \mathbf{x}^k\| \leq \varepsilon$, terminate
5) $k = k + 1$, go to step 2,

where k is an iteration counter, ε is a convergence tolerance, and system dynamics are satisfied implicitly via simulation. Note that in step 3, the most recently updated value of $\mathbf{x_p}$ is used and control designs must also satisfy plant constraints.

While BCD is often capable of producing system-optimal solutions, it can be computationally inefficient depending on the problem at hand. This potential inefficiency motivates alternative solution approaches that will be described in the following sections. To illustrate BCD solution efficiency, consider this simple quadratic objective function:

$$\phi(\mathbf{x}) = a_1(x_1 - b_1)^2 + a_2(x_2 - b_2)^2 + a_3 x_1 x_2 \tag{11}$$

The a_i's and b_i's are constants, and x_1 and x_2 are analogous to plant and control design variables, respectively. The third term is the interaction term; larger $|a_3|$ corresponds to stronger x_1, x_2 interaction (analogous to strong plant/control design interaction). If $a_3 = 0$, there is no interaction, and the optimal solution can be obtained by solving each BCD subproblem once. Similarly, without plant–control interaction, the optimization problems could be solved independently and co-design would be unnecessary. In reality, interaction does exist between plant and control design, so integrated design approaches are required. In this simplified illustrative example, if $\mathbf{a} = [1, 5, -4]^{\mathrm{T}}$ and $\mathbf{b} = [1, 2]^{\mathrm{T}}$, then the optimal solution of $\mathbf{x}_* = [25, 12]^{\mathrm{T}}$ is obtained within a tolerance of $\varepsilon = 1 \times 10^{-5}$ in 57 BCD iterations. Increasing the magnitude of a_3 (coupling strength) increases computational expense, as illustrated in Fig. 2.

The number of iterations is minimal when the BCD starting point $\mathbf{x}^1$ is aligned with $\mathbf{x}_*$ in at least one coordinate direction (e.g., $a_3 = -3.75$), or if there is no interaction ($a_3 = 0$). To put BCD computational expense in perspective, consider the more efficient minimization of Eqn. (11) using Newton's method; only one step would be required (regardless of a_3's value) because $\phi(\cdot)$ is quadratic.[107]

We could improve upon sequential design without the computational expense of BCD by performing just a few iterations of BCD. This approach is often used in design practice in an ad hoc manner, where design iterations continue until time or budget constraints are reached.[108] This inexact BCD approach, however, produces results far from system-optimal for strongly coupled co-design problems.

Another strategy for avoiding complete iteration of BCD was introduced by Peters et al.;[104] proxy functions are incorporated into the plant design problem to account for some problem coupling without iteration. This is a Case 2 or Case 3 sequential design method, depending on whether system dynamics are considered in plant design.

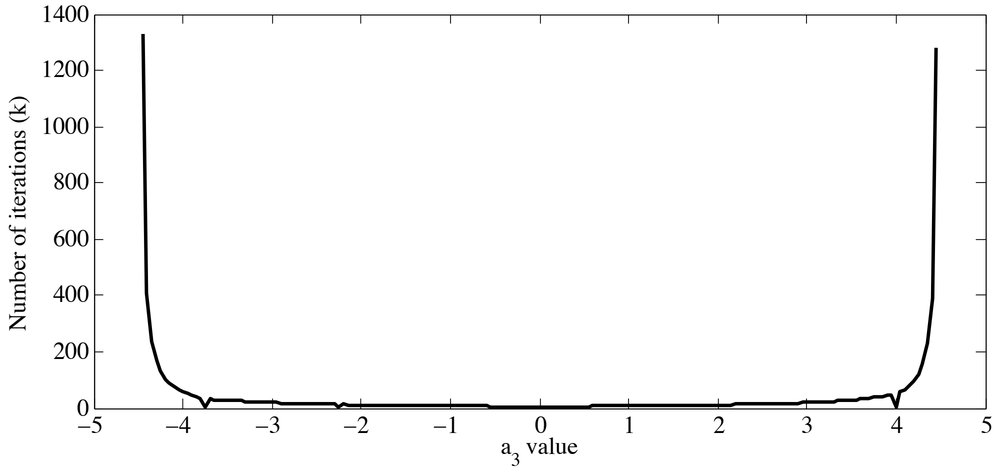

Figure 2. Influence of interaction term magnitude on BCD solution expense.

Difficulties arise when attempting to implement BCD based on Case 5. If the plant and control objectives are not equivalent, BCD is not guaranteed to converge. If BCD does converge it will not converge to the system optimum. In our numerical experiments, Case 5 BCD often cycles or diverges, and when it does converge, it is usually to a point far from the optimum of either $\hat{\phi}(\cdot)$ or $\phi(\cdot)$.

III. Existing Uses of MDO for Dynamic System Design

When applying multidisciplinary design optimization to the design of engineering systems, the aim is to account for interactions between multiple disciplines (such as structural and aerodynamic analysis) or physical subsystems (such as engine and wing). A core objective of MDO is to integrate (previously independent) disciplinary analyses and design activities to yield better system performance and reduced system development time and cost.[41,42,109–111]

Dynamic properties are of fundamental importance to the value of many multidisciplinary engineering systems, but application of MDO to dynamic system design has often been done only in a simplified or limited way. For example, rather than using simulation of nonlinear dynamics, simplified dynamic analysis is used, such as frequency domain analysis,[112–119] steady-state analysis,[120] or pseudo-static models.[91,97,121] Also, in many MDA models the interactions between disciplinary analyses are treated as static.[108] In co-design studies, a simplified dynamic or static model is often used for physical system design, while a more complete dynamic model is used for control system design. This misalignment between physical and system design formulations will prevent identification of a system-optimal solution.

One important factor that complicates the use of MDO for dynamic system design is the static nature of fundamental MDO formulations. With only a few exceptions, MDO formulations have not been developed in a way that addresses system dynamics explicitly. For example, Haftka and Sobieszczanski-Sobieksi explained that the analysis associated with MDO generally consists of nonlinear algebraic equations.[122] While dynamic system analysis may indeed be algebraic after discretization of differential equations, the 'algebraic equation analysis' mindset prevents deeper treatment of the unique needs of dynamic system design problems. In addition, while many articles do address dynamics (often in a simplified way, as discussed above), a large portion of MDO formulation development and testing has been based on example problems that are purely static or algebraic.[58,121,123–127] Some MDO test problems are abstract algebraic problems that, while effective for testing algorithms, have no direct connection to engineering design (e.g.,[41,123,124,128,129]).

Although many MDO frameworks can be challenging to use for dynamic system design, multidisciplinary optimization of dynamic systems has been investigated extensively in specific application areas, most notably in the design of dynamic structures.[47,75,76,130–139] Many studies have addressed control-structure design interaction directly,[9,46,102,106,116,120,140–147] while some focus on passive dynamic systems.[112,148–151] Sensor and actuator placement combined with control design is another extensively studied area of dynamic system design optimization.[135,152–156] Other important applications include automotive suspension and powertrain design,[24,96,99,157–165] robotic system design,[5,7,8,20,21,91,100,101,105,140,166–169] and energy systems.[114,170–173]

While substantial depth exists in select application areas for dynamic system design optimization, fundamental MDO formulations specifically designed for dynamic systems are largely not available. Many of the above studies employed either the basic multidisciplinary feasible (MDF) formulation[57,174]—where all analysis tasks are performed nested within a single optimization algorithm—or some form of the sequential design processes discussed above. Often, even when multidisciplinary analysis is performed, design is only conducted within one discipline (e.g., aeroelasticity). When multidisciplinary design is performed that incorporates physical system design, some aspect of dynamic analysis is usually simplified (e.g., co-design with static plant analysis), whereas extensive work in the area of optimal control[44] fully embraces the complexities of system dynamics, but addresses only one design discipline: control.

All the components required for fully-integrated dynamic system design—multidisciplinary analysis, physical system design methodologies, optimal control, etc.—exist, but usually in fragmented form except for specific case studies. Development of unifying MDO frameworks that can integrate these components would support the more general application of integrated dynamic system design, broadening the impact of MDO. This is an important opportunity for transformational progress in the design of dynamic engineering systems, and an exciting direction for new fundamental work in MDO.

A. Current MDO Formulations and Dynamic System Design

We will now explore approaches for using existing MDO formulations for dynamic system design, and highlight some of the difficulties that can arise. First we will look at how the MDF formulation can be applied to dynamic system design problems, and then explore distributed MDO methods.

When using MDF, all analysis tasks are performed within the optimization algorithm loop. MDF may be used to solve the fully-integrated problem (Prob. (10)),[9,160,175] or parts of a sequential design problem.[106] Starting with the simplest case, consider the (potentially multidisciplinary) plant design optimization problem introduced in Fig. 1. If we are using a Case 4 objective function that incorporates active control, the MDF formulation for the plant design portion of the sequential approach is:

$$\begin{aligned} \min_{\mathbf{x_p}} \quad & \phi(\boldsymbol{\xi}(t), \mathbf{x_c}, \mathbf{x_p}) \\ \text{s.t.} \quad & \mathbf{g_p}(\boldsymbol{\xi}(t), \mathbf{x_p}) \leq \mathbf{0} \\ \text{where:} \quad & \dot{\boldsymbol{\xi}}(t) - \mathbf{f}(\boldsymbol{\xi}(t), \mathbf{x_c}, \mathbf{x_p}, t) = \mathbf{0} \end{aligned} \tag{12}$$

Here the control design $\mathbf{x_c}$ is fixed, and the state equations are solved for the state trajectories $\boldsymbol{\xi}(t)$ using a forward simulation algorithm (such as a Runge Kutta algorithm[50,176]) for every plant design $\mathbf{x_p}$ proposed by the optimization algorithm. In other words, system analysis is nested within the optimization problem. Solution of the state equations requires time discretization ($t_1, \ldots, t_{n_t}$, where $t_1 = 0$, $t_{n_t} = t_F$, $h_i = t_{i+1} - t_i$ and n_t is the number of time steps). As noted above, the state trajectory solution may be represented in matrix form $\boldsymbol{\Xi}$, where the ith row of $\boldsymbol{\Xi}$ corresponds to $\boldsymbol{\xi}_i = \boldsymbol{\xi}(t_i)$. Also note that the state equations in Prob. 12 may span multiple engineering disciplines (cf. Eqns. (1)–(3)). MDF is a 'discretize-then-optimize' approach since discretization is performed before optimization. When applied to dynamic system design problems, MDF is also known as the single-shooting method.[52]

Once the MDF solution to the plant design problem is obtained ($\mathbf{x_{p_*}}$), the optimal control problem may be solved either via conventional optimal control methods (e.g., 'optimize-then-discretize' methods based on PMP), or 'discretize-then-optimize' methods such as direct transcription or MDF. These are good alternatives when the system Hamiltonian derivatives needed for PMP-based solutions are not obtained easily. The MDF formulation of the optimal control problem is:

$$\begin{aligned} \min_{\mathbf{x_c}} \quad & \phi(\boldsymbol{\xi}(t), \mathbf{x_c}, \mathbf{x_{p_*}}) \\ \text{s.t.} \quad & \mathbf{g_p}(\boldsymbol{\xi}(t), \mathbf{x_{p_*}}) \leq \mathbf{0} \\ \text{where:} \quad & \dot{\boldsymbol{\xi}}(t) - \mathbf{f}(\boldsymbol{\xi}(t), \mathbf{x_c}, \mathbf{x_{p_*}}, t) = \mathbf{0} \end{aligned} \tag{13}$$

Sequential design processes will not produce system-optimal solutions unless a Case 4 formulation is iterated and BCD convergence conditions are satisfied. A more efficient approach is to use MDF to solve the simultaneous problem defined in Prob. (10):

$$
\begin{aligned}
\min_{\mathbf{x_p},\mathbf{x_c}} \quad & \phi(\boldsymbol{\xi}(t), \mathbf{x_c}, \mathbf{x_p}) \\
\text{s.t.} \quad & \mathbf{g_p}(\boldsymbol{\xi}(t), \mathbf{x_p}) \leq \mathbf{0} \\
\text{where:} \quad & \dot{\boldsymbol{\xi}}(t) - \mathbf{f}(\boldsymbol{\xi}(t), \mathbf{x_c}, \mathbf{x_p}, t) = \mathbf{0}
\end{aligned} \tag{14}
$$

Here the analysis is completely integrated. Often this can be done within a single software environment, but if this is not possible, disparate analysis tools may be integrated using techniques such as co-simulation to coordinate multiple simulation environments.[177]

Nested co-design is a method that may be viewed as a special case of MDF.[98,116] An outer optimization loop optimizes the plant design, and an inner optimization loop identifies the optimal control for each plant design tested by the outer loop. Note that this inner optimization loop may have a simulation nested within it if a closed-form optimal control method (such as LQR[53]) or an AAO optimal control approach (such as direct transcription) is not employed, resulting in a double-nesting. One advantage of nested co-design is the ability to use existing optimal control algorithms (e.g., LQR, DT) to solve the inner loop problem efficiently without the complication of managing plant design variables. The outer-loop formulation is:

$$
\begin{aligned}
\min_{\mathbf{x_p}} \quad & \phi_*(\mathbf{x_p}) \\
\text{s.t.} \quad & \mathbf{g_p}(\mathbf{x_p}) \leq \mathbf{0}
\end{aligned} \tag{15}
$$

where for every objective function evaluation, the following inner-loop problem is solved to obtain $\phi_*(\cdot)$:

$$
\begin{aligned}
\min_{\mathbf{x_c}} \quad & \phi(\boldsymbol{\xi}(t), \mathbf{x_c}, \mathbf{x_p}) \\
\text{s.t.} \quad & \mathbf{g_p}(\boldsymbol{\xi}(t), \mathbf{x_p}) \leq \mathbf{0} \\
\text{where:} \quad & \dot{\boldsymbol{\xi}}(t) - \mathbf{f}(\boldsymbol{\xi}(t), \mathbf{x_c}, \mathbf{x_p}, t) = \mathbf{0}.
\end{aligned} \tag{16}
$$

The function $\phi_*(\cdot)$ in the outer-loop problem is an optimal-value function that is calculated by solving the inner loop problem. Plant design $\mathbf{x_p}$ is held fixed during the inner-loop solution. Note that the plant design constraints are retained (at least those influenced by state trajectories); $\mathbf{g_p}(\cdot)$ must be retained in any implementation where plant and control design problems are solved separately (e.g., nested methods or the sequential methods described above). Otherwise, design feasibility cannot be assured. Several have proposed using LQR to solve the inner-loop problem for linear systems.[46,98,178] If a detailed plant design formulation with substantial plant constraints is used, LQR may not be practical as it cannot manage inequality plant constraints. In this case a discretize-then-optimize approach would be more appropriate for the inner loop. Also, by the nature of the nested co-design method, the plant and control objectives are aligned since the outer loop objective is defined by the solution of the inner loop. The simultaneous and nested MDF approaches currently are some of the most widely-used solution techniques for dynamic system design problems.[179]

B. Distributed MDO Formulations

The MDF formulation is a fairly integrated approach to system design optimization. It uses a single optimization algorithm (and possibly an optimal-value function in the case of nested co-design), and system analysis is performed in a unified manner. MDF is the simplest and most prevalent MDO formulation.[180] Other formulations distribute analysis and possibly optimization tasks instead of centralizing them. These approaches allow different strategies for integrating and coordinating system design problems, and often can exploit problem sparsity for efficient computation (e.g., coarse-grained parallelism). Most of these methods, however, were motivated by the need to stitch together existing disparate analysis codes with relatively sparse interactions. This usually is not the case with systems that have abundant dynamic interactions. While some efforts to apply established distributed MDO formulations to dynamic system design have been successful, difficult challenges can arise in some cases.

To illustrate some of these potential difficulties, first let us explore the Individual Disciplinary Feasible formulation (IDF),[57] an MDO formulation with centralized optimization and distributed analysis. IDF is especially useful for systems with low-dimension analysis coupling quantities, and can be more efficient than MDF under these conditions.[174] If the quantities that couple subsystems, termed coupling variables $\mathbf{y}$, are instead high-dimension, IDF becomes inefficient. To clarify, consider a system analysis model that is comprised of several interrelated disciplinary analysis tools. Each of these analysis components may be represented by an analysis function $\mathbf{a}_i(\mathbf{x}, \mathbf{y})$, and the coupling variable $\mathbf{y}_{ij}$ is the quantity computed by $\mathbf{a}_j(\cdot)$ and required as input to $\mathbf{a}_i(\cdot)$ ($\mathbf{y}$ is the collection of all coupling variables). The combination of all analysis functions and coupling variables forms the equation $\mathbf{y} = \mathbf{a}(\mathbf{x}, \mathbf{y})$. Here we present the general IDF formulation, and will demonstrate shortly how to adapt this formulation for co-design:

$$\begin{aligned} \min_{\mathbf{x},\mathbf{y}} \quad & f(\mathbf{x}, \mathbf{y}) \\ \text{s.t.} \quad & \mathbf{g}(\mathbf{x}, \mathbf{y}) \leq \mathbf{0} \\ & \mathbf{y} - \mathbf{a}(\mathbf{x}, \mathbf{y}) = \mathbf{0}. \end{aligned} \tag{17}$$

Here $f(\cdot)$ is the design objective, $\mathbf{g}(\cdot)$ are the design constraint functions, and the equality constraint ensures analysis consistency. In MDF, analysis consistency is maintained at each optimization iteration by solving $\mathbf{y} = \mathbf{a}(\mathbf{x}, \mathbf{y})$ with an algorithm for solving nonlinear equations, such as fixed-point iteration. In IDF, this equation is solved by the optimization algorithm instead, and usually is not satisfied until convergence. In IDF, each analysis function $\mathbf{a}_i(\cdot)$ is temporarily independent, enabling coarse-grained parallel computing. One of the key points here is that in IDF, the coupling variables are optimization variables, whereas in MDF they are not. We might consider using IDF if the dimension of $\mathbf{y}$ is small, while MDF usually is more appropriate for densely-coupled problems.

IDF might be applied to dynamic system design problems in one of several ways, distinguished by how the separate analysis functions are defined. To illustrate the first of three IDF approaches discussed in this article, suppose the system analysis is in the form of a Simulink® model. We could group model components into clusters; a simulation of each cluster would comprise an analysis function, and the signals connecting the clusters would become the coupling variables. Each cluster corresponds to a portion of the derivative function, similar to the partitioned state equations in Eqns. (1-2). For example, the model in Fig. 3, based on the vane airflow (VAF) sensor problem in reference,[174] can be partitioned into blocks used to compute torque due to air resistance, and blocks used to quantify the dynamic response of the sensor vane. This partition cuts across torque (τ) and position (θ) signals (time histories are shown in Fig. 3).

The challenge with this decomposition approach is that each signal corresponds to a time history—a function-valued quantity—not just a scalar or small vector. Signals that cut across partitions are coupling variables, adding significantly to the number of optimization variables in the IDF formulation. In the VAF example, IDF requires the optimization algorithm to specify the complete torque and position trajectories so that the two subsystems may be simulated independently for each optimization function call.

If we assume that partitioned signals in this first IDF approach correspond to states, we can define $\hat{\boldsymbol{\xi}}(t)$ as the subset of state trajectories that cut across system partitions. Solving this IDF problem using nonlinear programming requires that we discretize $\hat{\boldsymbol{\xi}}(t)$. The matrix $\hat{\boldsymbol{\Xi}}$ is a subset of discretized state trajectories where the ith row corresponds to the value of states that cross partitions at time t_i. $\hat{\boldsymbol{\Xi}}$ are coupling variables, so they are included in the set of optimization variables in this IDF formulation:

$$\begin{aligned} \min_{\mathbf{x_p},\mathbf{x_c},\hat{\boldsymbol{\Xi}}} \quad & \phi\left(\boldsymbol{\Xi}, \mathbf{x_p}, \mathbf{x_c}\right) \\ \text{s.t.} \quad & \mathbf{g_p}\left(\boldsymbol{\Xi}, \mathbf{x_p}\right) \leq \mathbf{0} \\ & \hat{\boldsymbol{\Xi}} - \mathbf{a}\left(\hat{\boldsymbol{\Xi}}, \mathbf{x_p}, \mathbf{x_c}\right) = \mathbf{0}. \end{aligned} \tag{18}$$

Here $\boldsymbol{\Xi}$ is determined by simulating each of the subsystems. Each simulation requires a priori specification of the state trajectories that are inputs to the corresponding subsystem (i.e., the corresponding elements of $\hat{\boldsymbol{\Xi}}$). The components of $\hat{\boldsymbol{\Xi}}$ are local copies of state trajectories that correspond to trajectories computed in other subsystem simulations. Having the optimization algorithm specify these local copies enables independent simulation of each subsystem. The relationship between these quantities is made more clear by the last constraint in Prob. (18). The analysis functions $\mathbf{a}(\cdot)$ here output the state trajectories that cross subsystem boundaries as computed by the simulations. The analysis function outputs must match the local state trajectory copies $\hat{\boldsymbol{\Xi}}$. In other words, if we specify $\hat{\boldsymbol{\Xi}}$ and use it in evaluating $\mathbf{a}(\cdot)$, the resulting state trajectories must match the input $\hat{\boldsymbol{\Xi}}$. If the last constraint is satisfied, $\hat{\boldsymbol{\Xi}}$ is a fixed-point, and the solution to this decomposed problem will match the solution to the undecomposed MDF formulation.

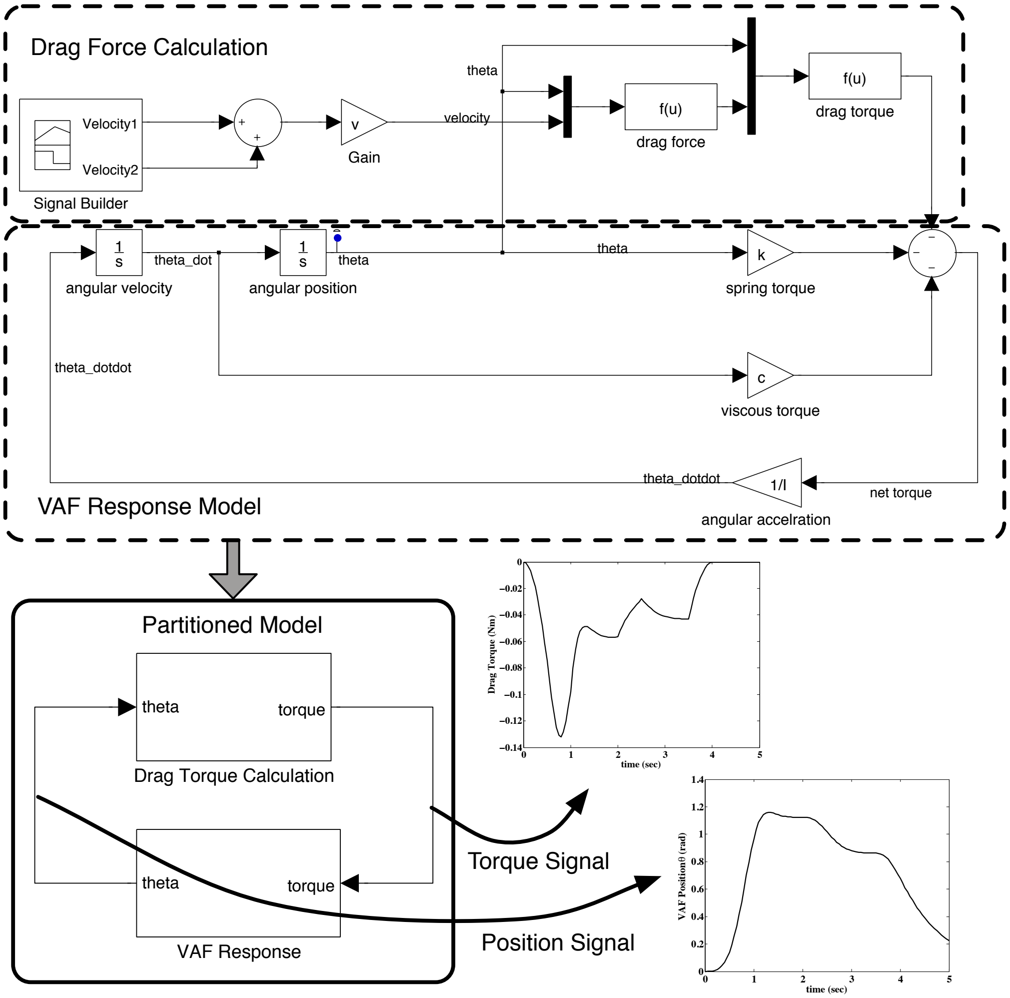

Figure 3. Partitioning of a block diagram model of a vane airflow sensor design problem.

Accurate solution using the above IDF approach requires fine discretization of state trajectories ($\hat{\Xi}$). This fine discretization, however, increases IDF problem dimension since $\hat{\Xi}$ is an optimization variable, often resulting in computationally expensive solutions. In other high-dimension formulations (such as direct transcription), problem structure and easily obtained analytical derivatives can be exploited for efficient solution. Analytical derivatives are difficult to obtain for Prob. (18) due to the subsystem simulations. While IDF partitioning enables coarse-grained parallelism, its problem structure does not readily allow for further efficiency improvements.

High-dimension coupling variables (such as $\hat{\Xi}$), also known as vector-valued coupling variables (VVCVs), can be approximated with low-dimension representations to aid efficient computation.[181–183] We can extend the usefulness of IDF and other MDO formulations for the design of dynamic systems using this and other workarounds, but 'force-fit' approaches like this are fundamentally limited. There is a bound on how far trajectory representation dimension can be reduced before solution accuracy suffers. We need to explore fundamentally different solution methods that fit the properties of dynamic system design problems more naturally.

Cutting across signals is not the only decomposition available for dynamic systems. Problems may be partitioned temporally by splitting the simulation into n_T time segments instead of partitioning system model elements. The state trajectories across each one of these time segments is obtained via simulation. We can consider each of these independent simulations to be an analysis function, and the state of the system between time segments makes up the set of coupling variables. The coupling variables in this case are normally lower dimension than the coupling variables in the first IDF variant given in Prob. (18). This second IDF decomposition approach, known as multiple-shooting,[24,52] has practical motivations. It helps ameliorate numerical instabilities for highly nonlinear systems, and also enables coarse-grained parallel computing. The IDF formulation for this approach is:

$$\begin{aligned} \min_{\mathbf{x_p},\mathbf{x_c},\mathbf{Y}} \quad & \phi\left(\boldsymbol{\Xi}, \mathbf{x_p}, \mathbf{x_c}\right) \\ \text{s.t.} \quad & \mathbf{g_p}\left(\boldsymbol{\Xi}, \mathbf{x_p}\right) \leq \mathbf{0} \\ & \boldsymbol{\zeta}_i\left(\boldsymbol{\Xi}, \mathbf{Y}\right) = \mathbf{0}, \quad i = 1, 2, \ldots, n_T. \end{aligned} \tag{19}$$

Here $\mathbf{Y}$ is the matrix of coupling variables; each row corresponds to the state value at the beginning of a time segment. $\boldsymbol{\zeta}_i(\cdot)$ are defect constraints that ensure the initial state values for each time segment (corresponding to rows of $\mathbf{Y}$) match the final state value from the previous time segment simulation (corresponding to the appropriate rows of $\boldsymbol{\Xi}$). These quantities are illustrated in Fig. 4. The optimization algorithm chooses initial state values for the simulations in time segments 1 and 2. These initial values correspond to rows in $\mathbf{Y}$. In time segment 1 the state trajectory is obtained by simulating through t_7. The defect constraint quantifies the difference between $\boldsymbol{\xi}_7$ (the state value at t_7 obtained via simulation in time segment 1) and the initial state value used in time segment 2. At IDF convergence these two quantities should match. While not widely used for co-design at present, this multiple-shooting approach is a well-known optimal control technique.

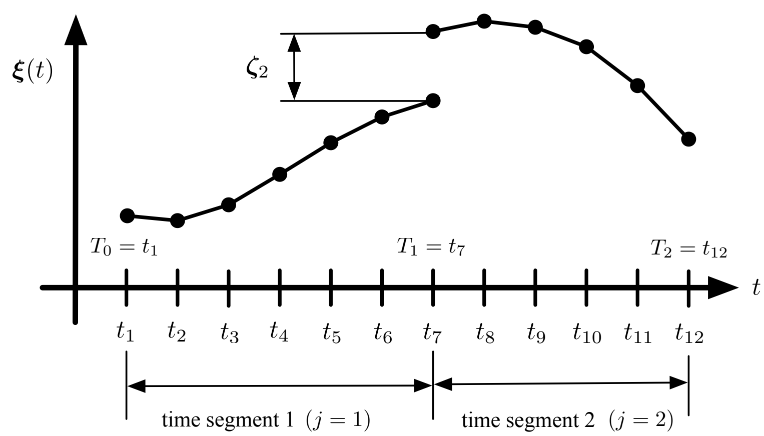

Figure 4. Illustration of time discretization and defect constraint between multiple shooting time segments.

The third IDF approach involves a popular model for co-design problems that addresses the link between plant analysis and control system analysis. Suppose the objective function takes the form $\phi(\boldsymbol{\xi}(t), \mathbf{x_c}, \mathbf{x_p})$, but the dependence of $\phi(\cdot)$ on $\mathbf{x_p}$ is through intermediate variables $\mathbf{y}$. For example, Allison and Han modeled a passive-active automotive suspension where the system dynamics model depends on spring stiffness and damping coefficients, k_s and c, respectively.[24] These coefficients, however, are not independent design variables. They are intermediate (coupling) variables that depend on other quantities that designers have direct control over, such as geometric dimensions. If $\mathbf{y} = [k_s, c]$, we can represent this dependence using the analysis function notation, $\mathbf{y} = \mathbf{a}(\mathbf{x_p})$, and the IDF formulation becomes:

$$\begin{aligned} \min_{\mathbf{x_p},\mathbf{x_c},\mathbf{y}} \quad & \phi\left(\mathbf{a}(\boldsymbol{\Xi}, \mathbf{x_p}), \mathbf{x_c}\right) \\ \text{s.t.} \quad & \mathbf{g_p}\left(\boldsymbol{\Xi}, \mathbf{x_p}\right) \leq \mathbf{0} \\ & \mathbf{y} - \mathbf{a}(\mathbf{x_p}) = \mathbf{0}. \end{aligned} \tag{20}$$

As with the other IDF approaches, $\boldsymbol{\Xi}$ is obtained via simulation, but in this case a single undecomposed simulation is used. The coupling variables here are low-dimension, and offer yet another opportunity for problem decomposition. In this way we can look at co-design as a two-discipline—plant design and control design—MDO problem.[106,160,184]

Other decompositions are possible. Engineers may define subsystems that correspond to observers, distributed control systems, or multiple disciplines associated with plant design. Hybrids of the three IDF variants discussed above may also be implemented. For example, the second and third variants could be combined to create a formulation with separate plant and control analyses, and multiple shooting is used to perform the simulation in a distributed manner for each subsystem.

The IDF formulations involve a single optimization problem. Alternative ‘multi-level’ MDO formulations employ multiple distributed optimization problems that each solve a piece of the system design problem.[185,186] Multi-level formulations are especially useful when desirable sparsity patterns exist in both analysis couplings and design variable dependence structures.[126] One important multi-level formulation is augmented Lagrangian coordination (ALC),[123,187] which is a non-hierarchical generalization of analytical target cascading (ATC).[124]

The three dynamic system decompositions described above for IDF also apply to ALC. For example, the third IDF variation has been demonstrated in several ALC studies where analysis functions compute parameters that are used in a dynamic simulation. Alexander et al. demonstrated how to use this approach to solve an electric vehicle design problem that involves function-valued coupling variables.[181–183,188] An ALC subproblem is defined for electric motor design, and a second subproblem is defined for powertrain or vehicle system design. The vehicle design subproblem computes the system objective function based on dynamic simulation. This vehicle-level simulation depends on motor properties, such as the torque-speed curve, that are computed by the motor subproblem. These properties are coupling variables that link motor design to the system objective function. While these coupling variables are not time histories, they are function-valued and need to be represented using VVCVs. Reduced-dimension representations for this specific problem have been investigated, some of which render solution via ALC practical.

Another way to extend the third IDF decomposition approach to ALC involves defining ALC subproblems for plant and control design, but then capitalizing on existing optimal control theory to develop an ‘optimize-then-discretize’ solution for the controls subproblem. Allison and Nazari demonstrated this approach using an electric circuit design problem.[184] As with the electric vehicle studies, the objective function in this case is linked to the plant design variables via plant analysis functions and coupling variables. The suspension co-design problem discussed above is another example of this decomposition approach, since the objective function is linked to plant design variables via coupling variables: $\mathbf{y} = [k_s, c] = \mathbf{a}(\mathbf{x_p})$.

The details of the two ALC approaches discussed above are available in the literature.[181–184,188] Another possible ALC formulation based on the second IDF decomposition (multiple shooting) is introduced here. Suppose the objective function is of the form presented in Eqn. (6), but in discretized form:

$$\phi(\Xi, \mathbf{x_c}) = \psi(\boldsymbol{\xi}_{n_t}, t_F) + \int_0^{t_F} L(\Xi, \mathbf{x_p}, \mathbf{x_c})dt. \tag{21}$$

Here we assume that numerical integration is performed using discretized state and control trajectories to compute the Lagrange cost. The final state value at t_F is $\boldsymbol{\xi}_{n_t}$. Observe that this function is additively separable if the problem is partitioned temporally into n_T smaller time segments. If T_j is the time at the end of time segment j ($T_0 = 0$ and $T_{n_T} = t_F = t_{n_t}$), and if $\boldsymbol{\Xi^{(j)}}$ and $\mathbf{x_c}^{(j)}$ are the discretized state and control trajectories over time segment j, the objective function may be rewritten as:

$$\phi(\Xi, \mathbf{x_c}) = \psi(\boldsymbol{\xi}_{n_t}, t_F) + \sum_{j=1}^{n_T} \int_{T_{j-1}}^{T_j} L(\Xi^{(j)}, \mathbf{x_p}, \mathbf{x_c}^{(j)})dt. \tag{22}$$

The optimization problem now can be divided into n_T subsystems, where the jth objective function is the jth term of the sum in Eqn. (22), and the Mayer term is included in the objective function for subsystem n_T. If coordinated using ALC, the decomposed problem is equivalent to the original undecomposed problem. In this ALC approach the states at time segment interfaces are coupling variables, and defect constraints ensure that the states at these interfaces are consistent. More specifically, if $\mathbf{y}^{(j)}$ is the state at the beginning of time segment j (the coupling variable), it must match $\boldsymbol{\xi}_F^{(j-1)}$, which is the state at the end of time segment $j-1$ obtained via simulation. The coupling variable $\mathbf{y}^{(j)}$ is an independent optimization variable in the jth ALC optimization subproblem, whereas $\boldsymbol{\xi}_F^{(j-1)}$ is computed in subproblem $j-1$ and held fixed in subproblem

j. The formulation for ALC subproblem j ($j \neq n_T$) is:

$$\begin{aligned}
\min_{\mathbf{x_p}^{(j)},\mathbf{x_c}^{(j)},\mathbf{y}^{(j)}} \quad & \int_{T_{j-1}}^{T_j} L(\boldsymbol{\Xi}^{(j)}, \mathbf{x_p}^{(j)}, \mathbf{x_c}^{(j)})dt \\
& + \pi\left(\mathbf{y}^{(j)} - \boldsymbol{\xi}_F^{(j-1)},\, \mathbf{x_p}^{(j)} - \mathbf{x_p}^{(j-1)}\right) \\
\text{s.t.} \quad & \mathbf{g_p}\left(\boldsymbol{\Xi}^{(j)}, \mathbf{x_p}^{(j)}\right) \leq \mathbf{0}
\end{aligned} \tag{23}$$

Instead of posing the defect equations as equality constraints as in the multiple shooting formulation of IDF given in Prob. (19), the defect constraints are enforced using an augmented Lagrangian penalty function $\pi(\cdot)$. A coordination algorithm guides all of the subproblems toward agreement so that at ALC convergence the defect constraints are satisfied within a given tolerance. Another primary difference between this ALC formulation and the corresponding IDF formulation is that optimization tasks are distributed in addition to analysis tasks. Also, $\mathbf{x_p}^{(j)}$ is a local copy of the plant design vector, and at ALC convergence the copies from all subproblems must match (enforced with $\pi(\cdot)$). Here plant and control variables are solved for simultaneously in the optimization subproblem. A variant of this ALC formulation is to adopt a nested approach similar to Probs. (15) and (16). The outer loop of the ALC subproblem would solve for $\mathbf{x_p}^{(j)}$ and $\mathbf{y}^{(j)}$, whereas $\mathbf{x_c}^{(j)}$ would be obtained in an inner optimization loop.

To summarize, MDO has been utilized extensively for solving dynamic system design problems. In most cases, however, a simple MDF formulation has been used to solve the simultaneous or nested co-design problems, parts of a sequential design process, or the problem has been limited to MDA with single-discipline design (e.g., optimal control). Solving dynamic system design problems with MDO formulations that are more sophisticated than MDF has proven to be challenging. Some success has been realized via distributed MDO methods, particularly when specialized optimization algorithms are employed or the unique structure of dynamic systems is exploited (e.g., IDF variant two – multiple shooting). However, some approaches for decomposing tightly integrated dynamic system models produce VVCVs (e.g., IDF variant one) or are otherwise poorly-suited for solving dynamic system design problems. The solution approach in these cases is not a good fit. Instead of attempting to 'force-fit' a particular solution method onto a given problem, new MDO strategies should be developed and explored that are compatible with the unique demands of dynamic system design. Multidisciplinary design problems that involve complex system dynamics are fundamentally different from the static, pseudo-static, simplified dynamic problems that much of MDO development has been based on. Returning to the foundations of MDO and developing formulations specifically for dynamic systems will advance both MDO research and efforts to design increasingly complex dynamic systems. The remainder of this article outlines promising directions for building up a more general theory for multidisciplinary dynamic system design optimization (MDSDO).

IV. Intrinsically Dynamic MDO Formulations

The need is clear for MDO methodologies that are deeply compatible with the nature of dynamic system design problems, but how do we move forward? Optimization has been used very successfully for a number specific dynamic system design applications, and in some cases MDO has been applied, but how do we move toward a more general theory for MDSDO? MDSDO must extend to a wide array of dynamic engineering systems, address dynamic issues directly, and be used more comprehensively throughout the product development process. We envision MDSDO as a vital branch of MDO and believe that MDSDO should embody the following characteristics:

- **Intrinsically Dynamic:** Most real engineering systems are dynamic. Many are nonlinear. System dynamics must be a core component of MDSDO formulations, comparable to its importance in optimal control.
- **Multidisciplinary and Integrated:** MDSDO should incorporate both multidisciplinary analysis and multidisciplinary design. Integration should be central to MDSDO, spanning analysis domains, time scales, and length scales. Decomposition should be strategic and congruent with dynamic system characteristics.
- **Systems-oriented:** Legacy design mindsets should be replaced with a balanced approach to dynamic system design, including avoiding unnecessary multi-objective co-design formulations and bias toward control-design. Additionally, larger systems-of-systems views should be incorporated into MDSDO.

- **Utilize Passive Dynamics:** A deeper treatment of physical system dynamics in an integrated MDSDO approach enables greater use of passive dynamics, reducing control system demands and advancing system capabilities.
- **Parallel:** Computational resources increasingly rely on more processors to enhance performance rather than processor speed. It is imperative that algorithms used with MDSDO are parallel in nature to exploit this trend.

In the remainder of this article we discuss four important fronts for advancing the state of MDSDO and satisfying the above requirements.

A. Balanced Co-Design

Most co-design studies has been performed with a strong emphasis on control design, and tend to deemphasize physical system design.[88, 99] For example, while controls engineers often recognize the importance of integrating control design with plant design, they often construct co-design implementations with fairly simplified plant design formulations. Dependent quantities often are treated as independent plant design variables. For example, Fathy et al. treat spring and damper coefficients as design variables,[99, 157] when in reality these quantities depend on detailed geometric design variables, such as spring wire and helix diameters. In other words, spring and damper coefficients are coupling variables $\mathbf{y}$ that link plant and control design analyses. Using coupling variables in place of design variables results in an incomplete problem formulation; its solution produces plant requirements instead of a plant design, and usually neglects plant design constraints $\mathbf{g_p}(\cdot)$ (e.g., suspension packaging, fatigue, damper temperature, etc.[24]).

Plant design simplification is particularly problematic when working with nested co-design formulations. The plant constraints, at least those that depend on state trajectories, should be included in both the inner and outer loops (Eqns. (15-16)), but simplified plant design obscures this requirement. Fathy et al. presented the nested co-design formulation sans $\mathbf{g_p}(\cdot)$ where the inner loop is solved using LQR (for linear systems).[98] LQR, however, cannot incorporate plant constraints, so a more general inner-loop solution method, such as MDF or direct transcription,[24, 52] must be used when more realistic plant design models are used in co-design.

Generations of engineers have developed mechanical systems without active control. Design paradigms appropriate for passive systems have evolved, matured, and now permeated the collective engineering consciousness. These mindsets often are taken as given, and it is hard to imagine any other design perspective, even for engineers seeking multidisciplinary design solutions. These legacy design approaches are evident in many existing co-design formulations in two ways. First, many studies posit that co-design is by nature multi-objective,[102–104] i.e., the plant design objective is distinct from the control design objective (Case 2, 3, or 5 plant design). Often the objective used for the plant is passive or static, a clear artifact of legacy design paradigms. These co-design formulations overlook to some degree the integrated nature of actively controlled systems. Active systems are single, unified systems, not two systems each with a distinct design objective. Plant and control systems work together toward a system-wide goal.[189] Co-design formulations therefore should include a single system-wide objective that reflects the primary purpose of the overall system (Case 4 plant design).[142] As noted earlier, co-design problems may indeed be multi-objective due to inherent system tradeoffs (e.g., cost vs. performance), but a problem is not automatically multi-objective if it is a co-design problem. If fundamental tradeoffs exist, the set of multiple system objectives should be used consistently across both plant and control design, qualifying as Case 4 plant design.

Second, co-design studies often assume unidirectional coupling between plant and control design, i.e., control performance depends on plant design: $\phi(\mathbf{a}(\mathbf{x_p}), \mathbf{x_c}, \boldsymbol{\xi}(t))$, but not vice versa. This premise is understandable if a control system is viewed as an 'add-on' to the physical system due to legacy design mindsets. The properties of active dynamic systems, however, typically depend simultaneously on plant and control design. If this is the case, unidirectional formulations are incomplete. Many properly-modeled plant design constraints depend on dynamic response $\boldsymbol{\xi}(t)$, which depends on both plant and control design. For example, material fatigue constraints depend on stress oscillation properties, which are a function of state trajectories (see the active suspension example in[24]). While bidirectional coupling is challenging to model, it is required for co-design formulations to accurately represent the system design problem. Any of the co-design formulations discussed above that use a Case 4 objective and include plant design constraint dependence on $\boldsymbol{\xi}(t)$ are bidirectional.

A balanced approach to co-design, such as the approach demonstrated in reference,[24] enables engineers to construct a formulation based on what is best for the overall system, rather than retaining elements of legacy design formulations from plant or control design. These formulations appropriately balance plant and control design depth, have single system objective functions (or consistently applied sets of multiple objectives), and account for bidirectional coupling.

B. Passive Dynamics

Passive dynamics refers to the dynamic behavior of a system without active control. Many systems are designed to operate passively (e.g., most automotive suspension systems,[162, 190] vibration absorbers,[151] passive walkers[191]), and often this design approach is desirable to reduce system complexity and improve stability and reliability. Passive dynamics, however, play a critical role in active systems as well. The physical elements of a system should be designed so that their passive dynamic properties combine synergistically with active control to enhance performance.[102, 146, 192, 193] Doing so can have profound impact on dynamic performance and energy consumption. For example, extensive research in building design has resulted in numerous passive technologies for reducing energy consumption,[194] such as night ventilation,[195, 196] passive cooling,[197] solar walls,[198] and passive systems combined with advanced control systems.[199]

Most passive dynamics studies to date have employed a sequential design approach; the plant is designed first as a passive system (Case 1 or 2 plant design), followed by control system design. While this strategy may be effective, it cannot fully exploit the synergy between physical and control systems, and cannot achieve system-optimality. Co-design can be an effective approach for tailoring passive dynamics to enhance active system performance, but only if plant design is treated with sufficient depth, and if the plant design objective matches the system objective (Case 4 plant design). Allison successfully demonstrated the co-design of a robotic manipulator where the control effort and energy required to perform a task was reduced dramatically.[20, 21] Future MDSDO development should enhance our ability to leverage passive dynamics. In some cases, this may even eliminate the need for active control, or if not, significantly reduce control system complexity and energy requirements.

C. Direct Transcription

Direct Transcription (DT) is a class of 'discretize-then-optimize' optimal control methods that was introduced in Section II. Here we explore additional details, discuss its extension to co-design, and examine its role in emerging MDSDO developments.

Exploring the relationship between DT (an AAO method) and IDF for dynamic system design provides some useful insights. When using IDF, some of the system analysis burden is shifted to the optimization algorithm via consistency constraints, whereas in AAO the optimization algorithm performs all system analysis directly. In the case of IDF for dynamic systems (variant 2, or multiple shooting), simulations of subdivided time segments comprise the analysis functions, and consistency (defect) constraints ensure continuity between time segments. Now consider what happens if the time segment size is reduced to that of a single time step. The number of coupling variables would increase to $n_s \times n_t$ (the number states times the number of time steps), and a consistency constraint would be required for each time step. The coupling variables would then be the complete set of discretized states $\mathbf{\Xi}$, and the consistency constraints would be the discretized state equations.

The optimal control formulation for DT (without inequality constraints) was presented in Prob. (7). Allison and Han demonstrated an extension of DT for the solution of co-design problems,[24] and Tava and Suzuki employed a similar technique for launch vehicle co-design.[200] Using DT for co-design produces a problem that requires satisfaction of optimality conditions for both plant and control design, in addition to satisfying defect constraints. The following is a simultaneous DT co-design formulation:

$$\begin{aligned} \min_{\mathbf{x_p},\mathbf{U},\mathbf{\Xi}} \quad & \sum_{i=1}^{n_t-1} L(\mathbf{x_p},\mathbf{u}_i,\boldsymbol{\xi}_i)h_i \\ \text{subject to:} \quad & \boldsymbol{\zeta}(\mathbf{x_p},\mathbf{U},\mathbf{\Xi}) = \mathbf{0} \\ & \mathbf{g_p}(\mathbf{x_p},\mathbf{\Xi}) \leq \mathbf{0}, \end{aligned} \tag{24}$$

When DT is applied to optimal control, the defect constraint Jacobian typically has a sparse diagonal structure that supports efficient problem solution. In the DT co-design extension, defect constraints are dependent on $\mathbf{x_p}$, increasing constraint Jacobian density. In most practical co-design problems, the coupling between plant and co-design is bidirectional, i.e., plant constraints depend on state trajectories, which in turn depend on control design. The dependence of plant constraints on both state and plant design variables increases constraint Jacobian density further. While initial studies have addressed these challenges for specific co-design problems, many open questions remain regarding the extension of DT for co-design. Note that other DT co-design formulations are possible, such as nesting a DT optimal control implementation within a plant optimization outer loop, or using DT to solve an ALC controls subproblem as described in.[184]

Computational efficiency, parallelism, and numerical stability are desirable properties of DT, but other qualities motivate on a more fundamental level the investigation of formulations like DT for dynamic system design. For example, Prob. (24) imposes no assumptions on control structure, which is especially helpful during early-stage design when control architecture is undefined. DT solutions provide insights into upper system performance limits without the restrictions imposed by specific control system designs. Open-loop solutions also often provide insights into complex system dynamics and possible directions for physical system design,[173] and can also serve as a basis for developing implementable feedback control systems. Most importantly, DT addresses system dynamics directly; dynamics are an integral part of the MDO formulation in Prob. (24). The DT co-design extension is a fully-integrated approach for dynamic system design that manages control, plant design, and state variables simultaneously.

While DT is promising for co-design, it can be challenging to implement at present. Commercial and open-source software is available for using DT to solve optimal control problems, but using DT for co-design changes the underlying problem structure and currently requires custom software development. Also, as a fully integrated AAO approach, DT usually does not mesh well with popular modeling environments. Sophisticated multidisciplinary analysis is difficult to incorporate, requiring integration at the equation level, a decidedly ‘advanced maneuver’. Progress must be made in theoretical and algorithmic development, the development of design tools, and awareness among design engineers before DT can become a practical solution for co-design.

D. Dynamic System Topology Optimization

Design optimization with respect to continuous variables, such as geometric dimensions or control parameters, is fairly mature and often can be performed with efficient gradient-based algorithms. In continuous optimization, however, system configuration is defined a priori. The design space is restricted, and in essence we are ‘tuning’ existing designs rather than generating completely new designs. Fundamentally new designs require configuration or topology modifications, i.e., changes in the existence of or interaction between system elements.[201] Topology design is traditionally the domain of engineering creativity and intuition, but often we lack the intuition required to make decisions regarding large-scale complex system that deviate very far from established design configurations (particularly if dynamics are important). The development of efficient methodologies for topology design is particularly important because new configurations can precipitate significant improvements in system performance, often much more so than continuous optimization alone. Success in topology optimization will help MDO break free of its ‘gilded cage’[202] and transition from design improvement to design synthesis.

Topology optimization methods for continuum systems, such as homogenization methods for structural topology optimization,[166,203,204] are well-established. These methods, however, do not apply to dynamic systems with discrete components with unique properties or functions (as opposed to a homogeneous continuum). For example, hybrid electric vehicle (HEV) powertrains combine multiple power sources to provide forward motion, and we have numerous options in how to specify the number, type, and connectivity of these power sources. Traditionally, engineers have explored design configurations via engineering intuition, or by enumerating possible configurations[205] (sometimes aided by automated modeling[25]) and comparing the optimal designs of each. In either case, we are limited to investigating systems of only very small size, or limited to only partial enumerations of larger systems. For example, design of genetic regulatory circuits, a critical element of synthetic biology, has plateaued at a maximum circuit size of six nodes using scientific intuition or exhaustive enumeration.[206,207] Improved methods are required to develop circuits of practical size.

Practical topology design problems are too large to use exhaustive enumeration as a solution approach. We can solve larger non-continuum problems using heuristic methods, although these approaches lead to improved instead of optimal solutions. For example, heuristic filters based on engineering knowledge can help reduce the number of design configurations that we need to compare. Liu demonstrated the use of heuristic filters in HEV powertrain design.[208,209] Rule-based techniques can also be used to generate feasible system topologies and reduce the number of designs that must be compared.[210] Genetic algorithms and other heuristic algorithms have also been applied successfully for topology design,[211] but largely are limited to continuum design problems or small heterogeneous systems.

Many current topology optimization approaches use centralized decision-making, i.e., all configuration decisions are micromanaged by an optimization algorithm, and the available decisions are usually determined a priori. Centralization limits the scale of systems that can be considered. We need methods that can scale up to larger systems and that can address the specific needs of dynamic system configuration design problems. Theory and methods from complex systems research offer some promising directions for efficient large-scale system topology design. In complex systems, global behavior emerges from distributed local decisions (e.g., market systems, ant colonies, immune systems). Keeping decisions local makes scaling up to very large systems possible. If engineers adopt design methods that involve automated local decision-making, they can concentrate their efforts on the rules that guide local decisions instead of on managing a large number of low-level decisions. In this way a complex systems approach can help us abstract topology design problems and operate at a higher level to 'transcend the overwhelming details of individual systems'.[212] Novel topology design approaches based on complex systems will allow us to explore new design configurations without requiring decisions at the lowest-level. Initial work in applying cellular automata to structural topology optimization,[213] and in applying cellular division algorithms to dynamic system topology design problems has produced promising results,[214–216] and is an example of the type of complex systems strategy that may enable topology optimization of large-scale dynamic systems.

Individual mechatronic systems often are part of a larger 'System-of-Systems' (SoS), where many mechatronic and other systems are coordinated to perform a larger task.[189,217] Transportation systems,[218] space construction,[219] military operations,[189] and farming[1] are all examples of SoSs. The interface between physical and control systems is important to investigate, but the additional interfaces between individual mechatronic systems and issues surrounding distributed control design[220] lead to a particularly challenging topology design problem. SoS design should address simultaneously the interfaces at the individual system and SoS level.

SoS design is sometimes referred to as 'site-level design', where we are not only interested in the design of individual mechatronic systems, but also in how they communicate and interact with each other, the environment, and humans. SoS design is too involved to perform from scratch whenever new needs evolve,[218] often due to investment in existing infrastructure, complexity of associated social or economic systems, or due to sheer complexity.[19] When complete system design is impractical, strategic redesign of limited portions of the larger system is performed instead;[18,20,21] design methods that can accommodate uncertain future changes in system requirements are vital in SoS design.

V. Conclusion

The design of multidisciplinary dynamic systems presents unique challenges to engineers, and is becoming an increasingly important technical issue as the number and complexity of smart and autonomous systems rises, and as their role in society becomes more crucial (e.g., energy and transportation systems). A phenomenal amount of work has been performed in the area of dynamic system design, but has addressed control system design primarily. Physical system design is integral to the dynamic system design problem and must be addressed as well. Unfortunately, dynamic properties are often simplified or neglected when performing plant design. While control design efforts more fully embrace system dynamics, if the plant design problem is addressed in conjunction with control design, the plant design problem is usually simplified (e.g., treating dependent variables as design variables). Co-design methods have been developed to design dynamic systems in a more integrated way, but often exhibit a strong control systems emphasis, and sometimes retain many elements of siloed design methodologies (i.e., separate plant and control design). For example, distinct objectives for plant and control design often are kept instead of adopting a system-wide objective. A more balanced system design approach is needed. While many have recognized the need for a more balanced and integrated approach to engineering system design, we lack many of the methods and tools required to put these concepts into practice.

We need a fresh systems perspective to advance MDSDO. It must be more than merging plant and control design and constructing interface mechanisms between existing design frameworks. The underlying design philosophies for plant and control systems need fundamental changes; each needs to move from a disciplinary design approach toward a completely integrated approach focused on the system design problem. MDO is the right framework for this transition.

The problem of dynamic system design optimization with balanced consideration of physical and control system design falls squarely into the domain of MDO. While MDO has been applied to the design of active dynamic systems using basic formulations such as MDF, or within the limited scope of specific applications, the established MDO formulations largely do not address explicitly the unique characteristics of dynamic systems. We advocate for a concerted effort in the MDO research community to develop more general theory and methodologies for MDSDO. These efforts should result in MDO approaches with intrinsically dynamic formulations that provide a balanced approach to co-design, more fully utilize passive dynamics, embrace more sophisticated dynamic plant design models, aid early-stage design efforts, and ultimately go beyond individual mechatronic systems to support the design of dynamic systems-of-systems.

References

[1]Eaton, R., Katupitiya, J., Siew, K. W., and Howarth, B., "Autonomous Farming: Modeling and Control of Agricultural Machinery in a Unified Framework," *In the Proceedings of the 15th International Conference on Mechatronics and Machine Vision in Practice*, IEEE, Dec. 2008, pp. 499–504.

[2]Anderson, S. J., Peters, S. C., Pilutti, T. E., and Iagnemma, K., "An Optimal-Control-Based Framework for Trajectory Planning, Threat Assessment, and Semi-Autonomous Control of Passenger Vehicles in Hazard Avoidance Scenarios," *International Journal of Vehicle Autonomous Systems*, Vol. 8, No. 2-4, 2010, pp. 190–216.

[3]"Report to the President on Ensuring American Leadership in Advanced Manufacturing," June 2011, United States President's Council of Advisors on Science and Technology.

[4]"A National Strategic Plan for Advanced Manufacturing," Feb. 2012, United States National Science and Technology Council.

[5]Youcef-Toumi, K., "Modeling, Design, and Control Integration: A Necessary Step in Mechatronics," *IEEE/ASME Transactions on Mechatronics*, Vol. 1, No. 1, March 1996, pp. 29–38.

[6]Roos, F., *Towards a Methodology for Integrated Design of Mechatronic Servo Systems*, Ph.D. Dissertation, Royal Institute of Technology, 2007.

[7]Asada, H., Park, J. H., and Rai, S., "A Control-Configured Flexible Arm: Integrated Structure/Control Design," *In the Proceedings of the 1991 IEEE International Conference on Robotics and Automation*, IEEE, Sacramento, CA, USA, April 1991, pp. 2356–2362.

[8]Nishigaki, H. and Kawashima, K., "Motion Control and Shape Optimization of a Suitlike Flexible Arm," *Structural Optimization*, Vol. 15, No. 3-4, June 1998, pp. 163–171.

[9]Cardoso, J. B. and Moita, P. P., "Design and Control of Nonlinear Mechanical Systems for Minimum Time," *Shock and Vibration*, Vol. 15, No. 3-4, 2008, pp. 315–323.

[10]Bollino, K. P., Lewis, L. R., and Sekhavat, P., "Pseudospectral Optimal Control: A Clear Road for Autonomous Intelligent Path Planning," *In the Proceedings of the AIAA-Infotech@Aerospace Conference and Exhibit*, No. AIAA-2007-2831, AIAA, Rohnert Park, CA, USA, May 2007.

[11]Xie, Y., Alleyne, A. G., Greer, A., and Deneault, D., "Fundamental Limits in Combine Harvester Header Height Control," *In the Proceedings of the 2011 American Control Conference*, San Francisco, CA, USA, June 2011.

[12]Dallard, P., Fitzpatrick, T., Flint, A., Low, A., Smith, R. R., Willford, M., and Roche, M., "London Millennium Bridge: Pedestrian-Induced Lateral Vibration," *Journal of Bridge Engineering*, Vol. 6, No. 6, 2001, pp. 412–417.

[13]Fitzpatrick, T., Dallard, P., Le Bourva, S., Low, A., Roger, R. S., and Willford, M., "Linking London: The Millennium Bridge," *Royal Academy of Engineering*, June 2001.

[14]Newland, D. E., "Pedestrian Excitation of Bridges," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, Vol. 218, No. 5, May 2004, pp. 477–492.

[15]Reiterer, M., "Control of Pedestrian-Induced Bridge Vibrations by Tuned Liquid Column Dampers," *the Proceedings of the 3rd European Conference on Structural Control*, Vienna, Austria, July 2004.

[16]Strogatz, S. H., Abrams, D. M., McRobie, A., Eckhardt, B., and Ott, E., "Theoretical Mechanics: Crowd Synchrony on the Millennium Bridge," *Nature*, Vol. 438, No. 7064, Nov. 2005, pp. 43–44.

[17]Nakamura, S., Kawasaki, T., Katsuura, H., and Yokoyama, K., "Experimental Studies on Lateral Forces Induced by Pedestrians," *Journal of Constructional Steel Research*, Vol. 64, No. 2, Feb. 2008, pp. 247–252.

[18]Harper, S. R. and Thurston, D. L., "Incorporating Environmental Impacts in Strategic Redesign of an Engineered System," *Journal of Mechanical Design*, Vol. 130, March 2008, pp. 031101.

[19]de Weck, O. L., Roos, D., and Magee, C L andVest, C. M., *Engineering Systems: Meeting Human Needs in a Complex Technological World*, The MIT Press, Oct. 2011.

[20]Allison, J. T., "Engineering System Co-Design with Limited Plant Redesign," *Engineering Optimization*, 2012, to appear.

[21]Allison, J. T., "Plant-Limited Co-Design of an Energy-Efficient Counterbalanced Robotic Manipulator," *In the Proceedings of the ASME 2012 Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, No. DETC2012-71108, ASME, Chicago, IL, USA, Aug. 2012.

[22]Li, J., Mourelatos, Z. P., Kokkolaras, M., and Papalambros, P. Y., "Validating Designs through Sequential Simulation-Based Optimization," *In the Proceedings of the ASME 2010 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, No. DETC2010-28431, Montreal, Quebec, Canada, Aug. 2010, pp. 1023–1031.

[23]Sarin, H., Kokkolaras, M., Hulbert, G., Papalambros, P., Barbat, S., and Yang, R.-J., "Comparing Time Histories for Validation of Simulation Models: Error Measures and Metrics," *Journal of Dynamic Systems, Measurement, and Control*, Vol. 132, No. 6, Nov. 2010, pp. 061401.

[24]Allison, J. T. and Han, Z., "Co-Design of an Active Suspension Using Simultaneous Dynamic Optimization," *In the Proceedings of the ASME 2011 Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, No. DETC2011-48521, ASME, Washington D.C., USA, Aug. 2011.

[25]Wu, Z., Campbell, M. I., and Fernández, B. R., "Bond Graph Based Automated Modeling for Computer-Aided Design of Dynamic Systems," *Journal of Mechanical Design*, Vol. 130, No. 4, 2008, pp. 041102.

[26]Mola, A., *Multi-physics and Multilevel Fidelity Modeling and Analysis of Olympic Rowing Boat Dynamics*, Ph.D. Dissertation, Virginia Polytechnic Institute and State University, June 2012.

[27]Chen, Y.-Y., *A Multi-Physics Software Framework on Hybrid Parallel Computing for High-Fidelity Solutions of Conservation Laws*, Ph.D. Dissertation, The Ohio State University, 2011.

[28]Keyes, D. E. et al., "Multiphysics Simulations: Challenges and Opportunities," Tech. Rep. ANL/MCS-TM-321, Argonne National Laboratory, Park City, UT, USA, July 2011.

[29]Karnopp, D. C., Margolis, D. L., and Rosenberg, R. C., *System Dynamics: Modeling and Simulation of Mechatronic Systems*, Wiley, 5th ed., Jan. 2012.

[30]Robinson, T. D., Eldred, M. S., Willcox, K. E., and Haimes, R., "Surrogate-Based Optimization Using Multifidelity Models with Variable Parameterization and Corrected Space Mapping," *AIAA Journal*, Vol. 46, No. 11, Nov. 2008, pp. 2814–2822.

[31]Wang, G. G. and Shan, S., "Review of Metamodeling Techniques in Support of Engineering Design Optimization," *Journal of Mechanical Design*, Vol. 129, No. 2, April 2007, pp. 370–380.

[32]Shan, S. and Wang, G. G., "Survey of Modeling and Optimization Strategies to Solve High-Dimensional Design Problems with Computationally-Expensive Black-Box Functions," *Structural and Multidisciplinary Optimization*, Vol. 41, No. 2, 2010, pp. 219–241.

[33]Allaire, D. and Willcox, K., "Surrogate Modeling for Uncertainty Assessment with Application to Aviation Environmental System Models," *AIAA Journal*, Vol. 48, No. 8, Aug. 2010, pp. 1791–1803.

[34]Frangos, M., Marzouk, Y., Willcox, K., and van Bloemen Waanders, B., "Surrogate and Reduced-Order Modeling: A Comparison of Approaches for Large-Scale Statistical Inverse Problems," *Large-Scale Inverse Problems and Quantification of Uncertainty*, edited by L. Biegler et al., Vol. 7, Wiley, 2010, pp. 123–149.

[35]Ersal, T., Kittirungsi, B., Fathy, H. K., and Stein, J. L., "Model Reduction in Vehicle Dynamics Using Importance Analysis," *In the Proceedings of the ASME 2008 Dynamic Systems and Control Conference, Parts A and B*, No. DSCC2008-2125, ASME, Ann Arbor, MI, USA, Oct. 2008, pp. 179–186.

[36]D. Galbally, D., Fidkowski, K., Willcox, K., and Ghattas, O., "Non-Linear Model Reduction for Uncertainty Quantification in Large-Scale Inverse Problems," *International Journal for Numerical Methods in Engineering*, Vol. 81, No. 12, March 2010, pp. 1581–1608.

[37]Lieberman, C., Willcox, K., and Ghattas, O., "Parameter and State Model Reduction for Large-Scale Statistical Inverse Problems," *SIAM Journal on Scientific Computing*, Vol. 32, No. 5, 2010, pp. 2523–2542.

[38]Kittirungsi, B., *A Scaling Methodology for Dynamic Systems: Quantification of Approximate Similitude and Use in Multiobjective Design*, Ph.D. Dissertation, University of Michigan, 2008.

[39]Petersheim, M. D. and Brennan, S. N., "Scaling of Hybrid Electric Vehicle Powertrain Components for Hardware-In-The-Loop Simulation," *In the Proceedings of the IEEE 2008 International Conference on Control Applications*, IEEE, San Antonio, TX, USA, Sept. 2008, pp. 720–726.

[40]O'Driscoll, M., "Design for Manufacture," *Journal of Materials Processing Technology*, Vol. 122, No. 2-3, March 2002, pp. 318–321.

[41]Hulme, K. F. and Bloebaum, C. L., "Development of a Multidisciplinary Design Optimization Test Simulator," *Structural Optimization*, Vol. 14, No. 2-3, 1997, pp. 129–137.

[42]Sobieszczanski-Sobieski, J. and Haftka, R. T., "Multidisciplinary Aerospace Design Optimization: Survey of Recent Developments," *Structural Optimization*, Vol. 14, No. 1, Aug. 1997, pp. 1–23.

[43]Pontryagin, L. S., *The Mathematical Theory of Optimal Processes*, Interscience, 1962.

[44]Bryson, A. E. and Ho, Y. C., *Applied Optimal Control: Optimization, Estimation and Control*, Taylor & Francis, 1975.

[45]Spong, M. W., Hutchinson, S., and Vidyasagar, M., *Robot Modeling and Control*, Wiley, 1st ed., Nov. 2005.

[46]Rao, S. S., "Combined Structural and Control Optimization of Flexible Structures," *Engineering Optimization*, Vol. 13, No. 1, Jan. 1988, pp. 1–16.

[47]Kang, B. S., Park, G. J., and Arora, J. S., "A Review of Optimization of Structures Subjected to Transient Loads," *Structural and Multidisciplinary Optimization*, Vol. 31, No. 2, 2006, pp. 81–95.

[48]Sha, L., Gopalakrishnan, S., Liu, X., and Wand, Q., "Cyber-Physical Systems: A New Frontier," *Machine Learning in Cyber Trust*, Vol. 1, 2009, pp. 3–13.

[49]Vaculín, O., Krüger, W. R., and Valášek, M., "Overview of Coupling of Multibody and Control Engineering Tools," *Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility*, Vol. 41, No. 5, 2004, pp. 415–429.

[50]Ascher, U. M. and Petzold, L. R., *Computer Methods for Ordinary Differential Equations and Differential-Algebraic Equations*, SIAM, 1998.

[51]Feehery, W. F. and Barton, P. I., "Dynamic Optimization with State Variable Path Constraints," *Computers & Chemical Engineering*, Vol. 22, No. 9, Aug. 1998, pp. 1241–1256.

[52]Biegler, L. T., *Nonlinear Programming: Concepts, Algorithms, and Applications to Chemical Processes*, SIAM, 2010.

[53]Friedland, B., *Control System Design: An Introduction to State-Space Methods*, Dover, 1986.

[54]Biegler, L. T., "An Overview of Simultaneous Strategies for Dynamic Optimization," *Chemical Engineering and Processing: Process Intensification*, Vol. 46, No. 11, Nov. 2007, pp. 1043–1053.

[55]Biegler, L. T. and Zavala, V. M., "Large-Scale Nonlinear Programming using IPOPT: An Integrating Framework for Enterprise-Wide Dynamic Optimization," *Computers & Chemical Engineering*, Vol. 33, No. 3, March 2009, pp. 575–582.

[56]Betts, J. T., *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*, SIAM, 2nd ed., Nov. 2009.

[57]Cramer, E. J., Dennis, Jr., J. E., Frank, P. D., Lewis, R. M., and Shubin, G. R., "Problem Formulation for Multidisciplinary Optimization," *SIAM Journal on Optimization*, Vol. 4, No. 4, 1994, pp. 754–776.

[58]Haftka, R. T., "Simultaneous Analysis and Design," *AIAA Journal*, Vol. 23, No. 7, July 1985, pp. 1099–1103.

[59]Hargraves, C. R. and Paris, S. W., "Direct Trajectory Optimization using Nonlinear Programming and Collocation," *Journal of Guidance, Control, and Dynamics*, Vol. 10, No. 4, July 1987, pp. 338–342.

[60]Bollino, K. P. and Ross, I. M., "A Pseudospectral Feedback Method for Real-Time Optimal Guidance of Reentry Vehicles," *In the Proceedings of the 2007 American Control Conference*, IEEE, New York, NY, USA, July 2007, pp. 3861–3867.

[61]Jain, S. and Tsiotras, P., "Trajectory Optimization Using Multiresolution Techniques," *Journal of Guidance, Control, and Dynamics*, Vol. 31, No. 5, Sept. 2008, pp. 1424–1436.

[62]Williams, P., "Hermite-Legendre-Gauss-Lobatto Direct Transcription in Trajectory Optimization," *Journal of Guidance, Control, and Dynamics*, Vol. 32, No. 4, July 2009, pp. 1392–1395.

[63]Gong, Q., Kang, W., and Ross, I. M., "A Pseudospectral Method for the Optimal Control of Constrained Feedback Linearizable Systems," *IEEE Transactions on Automatic Control*, Vol. 51, No. 7, July 2006, pp. 1115–1129.

[64]Ross, I. M., Sekhavat, P., Fleming, A., and Gong, Q., "Optimal Feedback Control: Foundations, Examples, and Experimental Results for a New Approach," *Journal of Guidance, Control, and Dynamics*, Vol. 31, No. 2, March 2008, pp. 307–321.

[65]Benson, D., *A Gauss Pseudospectral Transcription for Optimal Control*, Ph.D. Dissertation, Massachusetts Institute of Technology, 2005.

[66]Rao, A. V., Benson, D. A., Darby, C., Patterson, M. A., Francolin, C., Sanders, I., and Huntington, G. T., "Algorithm 902: GPOPS, A MATLAB Software for Solving Multiple-Phase Optimal Control Problems Using the Gauss Pseudospectral Method," *ACM Transactions on Mathematical Software*, Vol. 37, No. 2, April 2010, pp. 1–39.

[67]Ross, I. M. and Fahroo, F., "Legendre Pseudospectral Approximations of Optimal Control Problems," *New Trends in Nonlinear Dynamics and Control and their Applications*, edited by W. Kang, C. Borges, and M. Xiao, Vol. 295, Springer, Berlin, Heidelberg, 2003, pp. 327–342.

[68]Garg, D., Patterson, M., Hager, W. W., Rao, A. V., Benson, D. A., and Huntington, G. T., "A Unified Framework for the Numerical Solution of Optimal Control Problems using Pseudospectral Methods," *Automatica*, Vol. 46, No. 11, Nov. 2010, pp. 1843–1851.

[69]Kang, W., Gong, Q., and Ross, I. M., "Convergence of Pseudospectral Methods for a Class of Discontinuous Optimal Control," *In the Proceedings of the 44th IEEE Conference on Decision and Control, and the European Control Conference*, IEEE, Seville, Spain, Dec. 2005, pp. 2799–2804.

[70]Kang, W., Ross, I. M., and Gong, Q., "Pseudospectral Optimal Control and Its Convergence Theorems," *Analysis and Design of Nonlinear Control Systems*, edited by A. Astolfi and L. Marconi, Springer, Berlin, Heidelberg, 2008, pp. 109–124.

[71]Pietz, J. A., *Pseudospectral Collocation Methods for the Direct Transcription of Optimal Control Problems*, Master's Thesis, Rice University, 2003.

[72]Plessix, R.-E., "A Review of the Adjoint-State Method for Computing the Gradient of a Functional with Geophysical Applications," *Geophysical Journal International*, Vol. 167, No. 2, Nov. 2006, pp. 495–503.

[73]Choi, S., Potsdam, M., Lee, K., Iaccarino, G., and Alonso, J. J., "Helicopter Rotor Design Using a Time-Spectral and Adjoint-Based Method," *In the Proceedings of the 12th AIAA/ISSMO Multidisciplinary Analysis and Optimization Conference*, No. AIAA-2008-5810, AIAA, Victoria, British Columbia, Canada, Sept. 2008.

[74]Freund, J. B., "Adjoint-Based Optimization for Understanding and Suppressing Jet Noise," *Procedia Engineering*, Vol. 6, 2010, pp. 54–63.

[75]Kurdi, M. and Beran, P., "Optimization of Dynamic Response using Temporal Spectral Element Method," *In the Proceedings of the 46th AIAA Aerospace Sciences Meeting and Exhibit*, No. AIAA-2008-0903, AIAA, Reno, NV, USA, Jan. 2008.

[76]Kurdi, M. H. and Beran, P. S., "Optimization of Dynamic Response using a Monolithic-Time Formulation," *Structural and Multidisciplinary Optimization*, Vol. 39, No. 1, 2009, pp. 83–104.

[77]Zheng, T., editor, *Model Predictive Control*, Sciyo, Rijeka, Croatia, Aug. 2010.

[78]Vlases, W. G., Paris, S. W., Lajoie, R. M., Martens, M. J., and Hargraves, C. R., "Optimal Trajectories by Implicit Simulation," Tech. Rep. WRDC-TR-90-3056, Boeing Aerospace and Electronics, Wright-Patterson Air Force Base, Ohio, 1990.

[79]Betts, J. T. and Huffman, W. P., "Sparse Optimal Control Software SOCS. Mathematics and Engineering Analysis Technical Document MEA-LR-085," Tech. rep., Boeing Information and Support Services, Seattle, WA, USA, July 1997.

[80]Institute of Flight Mechanics and Control of Stuttgart University, *GESOP & ASTOS - The New Generation of Optimization Software*, 2003.

[81]Ross, I. M. and Fahroo, F., "User's Manual for DIDO 2001 α: A MATLAB Application for Solving Optimal Control Problems," Tech. Rep. AAS-01-03, Department of Aeronautics and Astronautics, Naval Postgraduate School, Monterey, CA, USA, 2001.

[82]Williams, P., *User's Guide for DIRECT 2.0*, Royal Melbourne Institute of Technology, Melbourne, Australia, 2008.

[83]Rutquist, P. E. and Edvall, M. M., "PROPT - Matlab Optimal Control Software - One of a Kind, Lightning Fast Solutions to Your Optimal Control Problems!" 2010, Tomlab Optimization, Inc.

[84]Milam, M. B., *Real-Time Optimal Trajectory Generation for Constrained Dynamical Systems*, Ph.D. Dissertation, California Institute of Technology, Pasadena, CA, USA, May 2003.

[85]Becerra, V. M., "Solving Complex Optimal Control Problems at No Cost with $\mathcal{PSOPT}$," *In the Proceedings of the 2010 IEEE International Symposium on Computer-Aided Control System Design*, Yokohama, Japan, Sept. 2010, pp. 1391–1396.

[86]Herman, A. L. and Conway, B. A., "Direct Optimization using Collocation Based on High-Order Gauss-Lobatto Quadrature Rules," *Journal of Guidance, Control, and Dynamics*, Vol. 19, No. 3, 1996, pp. 592–599.

[87]Karpel, M., "Procedures and Models for Aeroservoelastic Analysis and Design," *ZAMM*, Vol. 81, No. 9, Sept. 2001, pp. 579–592.

[88]Zimmermann, H., "Aeroservoelasticity," *Computer Methods in Applied Mechanics and Engineering*, Vol. 90, No. 1-3, Sept. 1991, pp. 719–735.

[89]Pratt, R. W., "Aeroservoelasticity: Key Issues Affecting the Design of Flight Control Systems," *In the Proceedings of the 1994 International Conference on Control*, No. 389, IEEE, Coventry, United Kingdom, March 1994, pp. 1522–1527.

[90]Hadj, N. B., Tounsi, S., Neji, R., and Sellami, F., "Real Coded Genetic Algorithm for Permanent Magnet Motor Mass Minimization for Electric Vehicle Application," *In the Proceedings of the 2007 International Symposium on Computational Intelligence and Intelligent Informatics*, IEEE, Agadir, Morocco, March 2007, pp. 153–158.

[91]Ravichandran, T., Wang, D., and Heppler, G., "Simultaneous Plant-Controller Design Optimization of a Two-Link Planar Manipulator," *Mechatronics*, Vol. 16, No. 3-4, April 2006, pp. 233–242.

[92]Papalambros, P. and Wilde, D., *Principles of Optimal Design: Modeling and Computation*, Cambridge University Press, Cambridge, UK, 2nd ed., 2000.

[93]Wang, Q. and Arora, J. S., "Several Simultaneous Formulations for Transient Dynamic Response Optimization: An Evaluation," *International Journal for Numerical Methods in Engineering*, Vol. 80, No. 5, 2009, pp. 631–650.

[94]Friedland, B., *Advanced Control System Design*, Prentice-Hall, 1st ed., 1996.

[95]Li, Q., Zhang, W. J., and Chen, L., "Design for Control—A Concurrent Engineering Approach for Mechatronic Systems Design," *IEEE/ASME Transactions on Mechatronics*, Vol. 6, No. 2, June 2001, pp. 161–169.

[96]Butsuen, T., *The Design of Semi-Active Suspensions for Automotive Vehicles*, Ph.D. Dissertation, Massachusetts Institute of Technology, 1989.

[97]Trivedi, D., Dienno, D., and Rahn, C. D., "Optimal, Model-Based Design of Soft Robotic Manipulators," *Journal of Mechanical Design*, Vol. 130, No. 9, 2008, pp. 091402.

[98]Fathy, H. K., Reyer, J. A., Papalambros, P. Y., and Ulsov, A. G., "On the Coupling between the Plant and Controller Optimization Problems," *In the Proceedings of the 2001 American Control Conference*, Vol. 3, IEEE, Arlington, VA, USA, June 2001, pp. 1864–1869.

[99]Fathy, H. K., Papalambros, P. Y., and Ulsoy, A. G., "Integrated Plant, Observer, and Controller Optimization With Application to Combined Passive/Active Automotive Suspensions," *In the Proceedings of the 2003 ASME International Mechanical Engineering Congress and Exposition*, No. IMECE2003-42014, ASME, Washington D.C., USA, Nov. 2003, pp. 225–232.

[100]Bowling, A. P., Renaud, J. E., Newkirk, J. T., Patel, N. M., and Agarwal, H., "Reliability-Based Design Optimization of Robotic System Dynamic Performance," *Journal of Mechanical Design*, Vol. 129, No. 4, April 2007, pp. 449.

[101]Park, J. H. and Asada, H., "Concurrent Design Optimization of Mechanical Structure and Control for High Speed Robots," *In the Proceedings of the 1993 American Control Conference*, IEEE, San Francisco, CA, USA, June 1993, pp. 2673–2679.

[102]Sunar, M. and Rao, S. S., "Simultaneous Passive and Active Control Design of Structures using Multiobjective Optimization Strategies," *Computers & Structures*, Vol. 48, No. 5, Sept. 1993, pp. 913–924.

[103]Frischknecht, B. D., Peters, D. L., and Papalambros, P. Y., "Pareto Set Analysis: Local Measures of Objective Coupling in Multiobjective Design Optimization," *Structural and Multidisciplinary Optimization*, Vol. 43, No. 5, 2011, pp. 617–630.

[104]Peters, D. L., Papalambros, P. Y., and Ulsoy, A. G., "Control Proxy Functions for Sequential Design and Control Optimization," *Journal of Mechanical Design*, Vol. 133, No. 9, 2011, pp. 091007.

[105]Pil, A. C. and Asada, H. H., "Integrated Structure/Control Design of Mechatronic Systems using a Recursive Experimental Optimization Method," *IEEE/ASME Transactions on Mechatronics*, Vol. 1, No. 3, 1996, pp. 191–203.

[106]Padula, S. L., Sandridge, C. A., Walsh, J. L., and Haftka, R. T., "Integrated Controls-Structures Optimization of a Large Space Structure," *Computers & Structures*, Vol. 42, No. 5, March 1992, pp. 725–732.

[107]Bertsekas, D. P., *Nonlinear Programming*, Athena Scientific, 2nd ed., Sept. 1999.

[108]Alonso, J. J., "Some Thoughts on Applicability of Aerospace Analysis and Design Techniques to Wind Energy," Presented at the 2010 Wind Energy Systems Engineering Workshop in Louisville, CO, USA.

[109]Agte, J., Weck, O., Sobieszczanski-Sobieski, J., Arendsen, P., Morris, A., and Spieck, M., "MDO: Assessment and Direction for Advancement—An Opinion of One International Group," *Structural and Multidisciplinary Optimization*, Vol. 40, No. 1-6, 2010, pp. 17–33.

[110]Giesing, J P abd Barthelemy, J. F. M., "A Summary of Industry MDO Applications and Needs," *In the Proceedings of the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, No. AIAA-1998-4737, AAIA, St. Louis, MO, USA, Sept. 1998.

[111]Bartholomew, P., "The Role of MDO within Aerospace Design and Progress towards an MDO Capability," *In the Proceedings of the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, No. AIAA-1998-4705, AIAA, St. Louis, MO, USA, Sept. 1998.

[112]Brackett, D. J., Ashcroft, I. A., and Hague, R. J. M., "Multi-Physics Optimisation of 'Brass' Instruments—A New Method to Include Structural and Acoustical Interactions," *Structural and Multidisciplinary Optimization*, Vol. 40, No. 1-6, 2010, pp. 611–624.

[113]Falnes, J. and Hals, J., "Heaving Buoys, Point Absorbers and Arrays," *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, Vol. 370, No. 1959, Jan. 2012, pp. 246–277.

[114]Trimble, A. Z., Lang, J. H., Pabon, J., and Slocum, A., "A Device for Harvesting Energy From Rotational Vibrations," *Journal of Mechanical Design*, Vol. 132, No. 9, 2010, pp. 091001.

[115]Chen, T.-Y., "Design Optimization with Static and Dynamic Displacement Constraints," *Structural Optimization*, Vol. 4, No. 3-4, Sept. 1992, pp. 179–185.

[116]Fonseca, I. M., "Integrated Structural and Control Optimization," *Journal of Vibration and Control*, Vol. 10, No. 10, Oct. 2004, pp. 1377–1391.

[117]Lee, J. and Hajela, P., "Parallel Genetic Algorithm Implementation in Multidisciplinary Rotor Blade Design," *Journal of Aircraft*, Vol. 33, No. 5, Sept. 1996, pp. 962–969.

[118]Peters, D. L., Kurabayashi, K., Papalambros, P. Y., and Ulsoy, A. G., "Co-Design of a MEMS Actuator and its Controller Using Frequency Constraints," *In the Proceedings of the ASME 2008 Dynamic Systems and Control Conference, Parts A and B*, No. 43352, ASME, Ann Arbor, MI, USA, Oct. 2008, pp. 801–807.

[119]Tosserams, S., Etman, L. F. P., and Rooda, J. E., "A Micro-accelerometer MDO Benchmark Problem," *Structural and Multidisciplinary Optimization*, Vol. 41, No. 2, 2010, pp. 255–275.

[120]Ou, J. S. and Kikuchi, N., "Integrated Optimal Structural and Vibration Control Design," *Structural Optimization*, Vol. 12, No. 4, Dec. 1996, pp. 209–216.

[121]Krishnamachari, R. S. and Papalambros, P. Y., "Optimal Design of a Hybrid Electric Powertrain System," *Mechanics of Structures and Machines*, Vol. 25, No. 3, Jan. 1997, pp. 267–286.

[122]Haftka, R. T., Sobieszczanski-Sobieski, J., and Padula, S. L., "On Options for Interdisciplinary Analysis and Design Optimization," *Structural Optimization*, Vol. 4, No. 2, June 1992, pp. 65–74.

[123]Tosserams, S., Etman, L. F. P., and Rooda, J. E., "An Augmented Lagrangian Decomposition Method for Quasi-Separable Problems in MDO," *Structural and Multidisciplinary Optimization*, Vol. 34, No. 3, 2007, pp. 211–227.

[124]Kim, H. M., Michelena, N. F., Papalambros, P. Y., and Jiang, T., "Target Cascading in Optimal System Design," *Journal of Mechanical Design*, Vol. 125, No. 3, Sept. 2003, pp. 474–480.

[125]Allison, J. T., Walsh, D., Kokkolaras, M., Papalambros, P. Y., and Cartmell, M., "Analytical Target Cascading in Aircraft Design," *In the Proceedings of the 44th AIAA Aerospace Sciences Meeting and Exhibit*, No. AIAA-2006-1325, AIAA, Reno, NV, USA, Jan. 2006.

[126]Allison, J. T., Kokkolaras, M., and Papalambros, P. Y., "Optimal Partitioning and Coordination Decisions in Decomposition-Based Design Optimization," *Journal of Mechanical Design*, Vol. 131, No. 8, Aug. 2009, pp. 081008.

[127]Allison, J. T. and Papalambros, P. Y., "Consistency Constraint Allocation in Augmented Lagrangian Coordination," *Journal of Mechanical Design*, Vol. 132, No. 7, July 2010, pp. 071007.

[128]Yi, S. I., Shin, J. K., and Park, G. J., "Comparison of MDO Methods with Mathematical Examples," *Structural and Multidisciplinary Optimization*, Vol. 35, No. 5, 2008, pp. 391–402.

[129]Schittkowski, K., *More Test Examples for Nonlinear Programming*, Lecture Notes in Economics and Mathematical Systems, Springer, 1987.

[130]Sousa, L. G., Cardoso, J. B., and Valido, A. J., "Optimal Cross-Section and Configuration Design of Elastic-Plastic Structures Subject to Dynamic Cyclic Loading," *Structural Optimization*, Vol. 13, No. 2-3, April 1997, pp. 112–118.

[131]Eldred, M. S., Giunta, A. A., and van Bloemen Waanders, B. G., "Multilevel Parallel Optimization using Massively Parallel Structural Dynamics," *Structural and Multidisciplinary Optimization*, Vol. 27, No. 1-2, 2004, pp. 97–109.

[132]Araújo, A. L., Martins, P., Mota Soares, C. M., Mota Soares, C. A., and Herskovits, J., "Damping Optimization of Viscoelastic Laminated Sandwich Composite Structures," *Structural and Multidisciplinary Optimization*, Vol. 39, No. 6, 2009, pp. 569–579.

[133]Ohno, T., Kramer, G. J. E., and Grierson, D. E., "Least-Weight Design of Frameworks under Multiple Dynamic Loads," *Structural Optimization*, Vol. 1, No. 3, Sept. 1989, pp. 181–191.

[134]Meisami-Azad, M., Mohammadpour, J., and Grigoriadis, K. M., "An H_2 Upper Bound Approach for Control of Collocated Structural Systems," *In the Proceedings of the 2007 American Control Conference*, IEEE, New York, NY, USA, July 2007, pp. 4631–4636.

[135]Trease, B. P., *Topology Synthesis of Compliant Systems with Embedded Actuators and Sensors*, Ph.D. Dissertation, University of Michigan, 2008.

[136]Nieto, F., Hernández, S., and Jurado, J. A., "Optimum Design of Long-Span Suspension Bridges Considering Aeroelastic and Kinematic Constraints," *Structural and Multidisciplinary Optimization*, Vol. 39, No. 2, 2009, pp. 133–151.

[137]Arora, J. S. and Wang, Q., "Review of Formulations for Structural and Mechanical System Optimization," *Structural and Multidisciplinary Optimization*, Vol. 30, No. 4, 2005, pp. 251–272.

[138]Carmichael, D. G., "Structural Optimization and System Dynamics," *Structural Optimization*, Vol. 2, No. 2, 1990, pp. 105–108.

[139]Chahande, A. I. and Arora, J. S., "Development of a Multiplier Method for Dynamic Response Optimization Problems," *Structural Optimization*, Vol. 6, No. 2, June 1993, pp. 69–78.

[140]Albers, A. and Ottnad, J., "Integrated Structural and Controller Optimization in Dynamic Mechatronic Systems," *Journal of Mechanical Design*, Vol. 132, No. 4, 2010, pp. 041008.

[141]Ling, J. and Kabamba, P., "Combined Design of Structures and Controllers for Optimal Maneuverability," *Structural and Multidisciplinary Optimization*, Vol. 230, No. 4, 1991, pp. 214–230.

[142]McLaren, M. D. and Slater, G. L., "A Disturbance Model for Control/Structure Optimization with Output Feedback Control," *Structural Optimization*, Vol. 6, No. 2, June 1993, pp. 123–133.

[143]Andreaus, U., Colloca, M., Iacoviello, D., and Pignataro, M., "Optimal-Tuning PID Control of Adaptive Materials for Structural Efficiency," *Structural and Multidisciplinary Optimization*, Vol. 43, No. 1, 2011, pp. 43–59.

[144]Fonseca, I. M. and Bainum, P. M., "Large Space Structure Integrated Structural and Control Optimization, Using Analytical Sensitivity Analysis Introduction," *Journal of Guidance, Control, and Dynamics*, Vol. 24, No. 5, 2001, pp. 978–982.

[145]Fonseca, I. M., Bainum, P. M., and Santos, M. C., "CPU Time Consideration for LSS Structural/Control Optimization Models with Different Degrees of Freedom," *Acta Astronautica*, Vol. 54, No. 4, Feb. 2004, pp. 259–266.

[146]Smith, M. J., Grigoriadis, K. M., and Skelton, R. E., "The Optimal Mix of Passive and Active Control in Structures," *In the Proceedings of the 1991 American Control Conference*, IEEE, Boston, MA, USA, June 1991, pp. 1459–1464.

[147]Khot, N. S., "Structure/Control Optimization to Improve the Dynamic Response of Space Structures," *Computational Mechanics*, Vol. 3, No. 3, 1988, pp. 179–186.

[148]Craig, K. J. and Kingsley, T. C., "Design Optimization of Containers for Sloshing and Impact," *Structural and Multidisciplinary Optimization*, Vol. 33, No. 1, 2007, pp. 71–87.

[149]Krack, M., Secanell, M., and Mertiny, P., "Cost Optimization of a Hybrid Composite Flywheel Rotor with a Split-Type Hub using Combined Analytical/Numerical Models," *Structural and Multidisciplinary Optimization*, Vol. 44, No. 1, 2011, pp. 57–73.

[150]Sun, T.-C., "Design Sensitivity Analysis and Optimization of Nonlinear Dynamic Response for a Motorcycle Driving on a Half-Sine Bump Road," *Structural Optimization*, Vol. 11, No. 2, April 1996, pp. 113–119.

[151]Dimitrovová, Z. and Rodrigues, H. C., "Optimization of Passive Vibration Isolators Mechanical Characteristics," *Structural and Multidisciplinary Optimization*, Vol. 42, No. 3, 2010, pp. 325–340.

[152]Colburn, C., Zhang, D., and Bewley, T., "Gradient-Based Optimization Methods for Sensor & Actuator Placement in LTI Systems," *Working Paper*, 2011, pp. 1–21.

[153]Maghami, P. G. and Joshi, S. M., "Sensor/Actuator Placement for Flexible Space Structures," *IEEE Transactions on Aerospace and Electronic Systems*, Vol. 29, No. 2, April 1993, pp. 345–351.

[154]Hiramoto, K., Mohammadpour, J., and Grigoriadis, K. M., "Integrated Design of System Parameters, Control and Sensor/Actuator Placement for Symmetric Mechanical Systems," *In the Proceedings of the Joint 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference*, IEEE, Shanghai, P.R. China, Dec. 2009, pp. 2855–2860.

[155]Li, F., de Oliveira, M., and Skelton, R. E., "Integrating Control Design and Sensor/Actuator Selection," *In the Proceedings of the 45th IEEE Conference on Decision and Control*, IEEE, San Diego, CA, USA, Dec. 2006, pp. 6612–6617.

[156]Li, F. and Skelton, R. E., "Sensor/Actuator Selection for Tensegrity Structures," *In the Proceedings of the 45th IEEE Conference on Decision and Control*, IEEE, San Diego, CA, USA, Dec. 2006, pp. 2332–2337.

[157]Fathy, H. K., Papalambros, P. Y., Ulsoy, A. G., and Hrovat, D., "Nested Plant/Controller Optimization with Application to Combined Passive/Active Automotive Suspensions," *In the Proceedings of the 2003 American Control Conference*, Vol. 4, IEEE, Denver, CO, USA, June 2003, pp. 3375–3380.

[158]Filipi, Z., Louca, L., Daran, B., Lin, C.-C., Yildir, U., Wu, B., Kokkolaras, M., Assanis, D., Peng, H., Papalambros, P., Stein, J., Szkubiel, D., and Chapp, R., "Combined Optimization of Design and Power Management of the Hydraulic Hybrid Propulsion System for the 6x6 Medium Truck," *International Journal of Heavy Vehicle Systems*, Vol. 11, No. 3/4, 2004, pp. 371–401.

[159]Alyaqout, S. F., Papalambros, P. Y., and Ulsoy, A. G., "Combined Design and Robust Control of a Vehicle Passive/Active Suspension," *In the Proceedings of the 2007 European Control Conference*, European Control Association, Kos, Greece, July 2007.

[160]He, Y. and McPhee, J., "Multidisciplinary Design Optimization of Mechatronic Vehicles with Active Suspensions," *Journal of Sound and Vibration*, Vol. 283, No. 1-2, May 2005, pp. 217–241.

[161]Bourmistrova, A. and Storey, I., "Multiobjective Optimisation of Active and Semi-Active Suspension Systems with Application of Evolutionary Algorithm," *In the Proceedings of the 2005 International Conference on Modeling and Simulation*, Melbourne, Australia, Dec. 2005, pp. 1217–1223.

[162]Deo, H. V., *Axiomatic Design of Customizable Automotive Suspension Systems*, Ph.D. Dissertation, Massachusetts Institute of Technology, 2007.

[163]Smith, M. C. and Walker, G. W., "Interconnected Vehicle Suspension," *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, Vol. 219, No. 3, Jan. 2005, pp. 295–307.

[164]Storey, I. and Bourmistrova, A., "Control Over Limited-Stroke Suspensions using Jerk," *In the Proceedings of the 18th World IMACS / MODSIM Congress*, Cairns, Australia, July 2009, pp. 838–844.

[165]Zhang, Y., Lin, H., Zhang, B., and Mi, C., "Performance Modeling and Optimization of a Novel Multimode Hybrid Powertrain," *Journal of Mechanical Design*, Vol. 128, No. 1, 2006, pp. 79–89.

[166]Bendsøe, M. P. and Kikuchi, N., "Generating Optimal Topologies in Structural Design Using a Homogenization Method," *Computer Methods in Applied Mechanics and Engineering*, Vol. 71, No. 2, Nov. 1988, pp. 197–224.

[167]Kurtoglu, T. and Campbell, M. I., "Automated Synthesis of Electromechanical Design Configurations from Empirical Analysis of Function to Form Mapping," *Journal of Engineering Design*, Vol. 20, No. 1, Feb. 2009, pp. 83–104.

[168]Tekeş, A., Sönmez, U., and Güvenç, B. A., "Trajectory Control of Compliant Parallel-Arm Mechanisms," *Journal of Mechanical Design*, Vol. 132, No. 1, 2010, pp. 011006.

[169]Sars, V. D., Haliyo, S., and Szewczyk, J., "A Practical Approach to the Design and Control of Active Endoscopes," *Mechatronics*, Vol. 20, No. 2, March 2010, pp. 251–264.

[170]Guevara, J. R. M., *Optimization Strategies for the Synthesis / Design of Highly Coupled, Highly Dynamic Energy Systems*, Ph.D. Dissertation, Virginia Polytechnic Institute and State University, 2000.

[171]Muñoz, J. R., Costiner, S., and Ghosh, S., "A Multi-Disciplinary Optimization Approach for Process and Energy Systems," *In the Proceedings of the 9th AIAA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, No. AIAA-2002-5467, AIAA, Atlanta, GA, USA, Sept. 2002.

[172]Benini, L., Bogliolo, A., and De Micheli, G., "A Survey of Design Techniques for System-Level Dynamic Power Management," *IEEE Transactions on VLSI Systems*, Vol. 8, No. 3, June 2000, pp. 299–316.

[173]Allison, J. T., Kaitharath, A., and Herber, D. R., "Wave Energy Extraction Maximization using Direct Transcription," *In the Proceedings of the ASME 2012 International Mechanical Engineering Congress & Exposition*, No. IMECE2012-86619, ASME, Houston, TX, USA, Nov. 2012.

[174]Allison, J. T., Kokkolaras, M., and Papalambros, P. Y., "On Selecting Single-Level Formulations for Complex System Design Optimization," *Journal of Mechanical Design*, Vol. 129, No. 9, 2007, pp. 898–906.

[175]Ahn, K., Whitefoot, J., Atluri, V. P., Tate, E., and Papalambros, P. Y., “Comparison of Early-Stage Design Methods for a Two-Mode Hybrid Electric Vehicle,” *In the Proceedings of the 2011 Vehicle Power and Propulsion Conference*, IEEE, Sept. 2011, pp. 1–6.

[176]Shampine, L. F., *Numerical Solution of Ordinary Differential Equations*, Springer, Aug. 1994.

[177]Trčka, M., *Co-simulation for Performance Prediction of Innovative Integrated Mechanical Energy Systems in Buildings*, Ph.D. Dissertation, Eindhoven University of Technology, 2008.

[178]Belvin, W. K. and Park, K. C., “Structural Tailoring and Feedback Control Synthesis - An Interdisciplinary Approach,” *Journal of Guidance, Control, and Dynamics*, Vol. 13, No. 3, May 1990, pp. 424–429.

[179]Balesdent, M., Bérend, N., Dépincé, P., and Chriette, A., “A Survey of Multidisciplinary Design Optimization Methods in Launch Vehicle Design,” *Structural and Multidisciplinary Optimization*, Vol. 45, No. 5, 2012, pp. 619–642.

[180]Braun, R. D., *Collaborative Optimization: An Architecture for Large-Scale Distributed Design*, Ph.D. Dissertation, Stanford University, 1996.

[181]Alexander, M. J., Allison, J. T., and Papalambros, P. Y., “Reduced Representations of Vector-Valued Coupling Variables in Decomposition-Based Design Optimization,” *Structural and Multidisciplinary Optimization*, Vol. 44, No. 3, 2011, pp. 379–391.

[182]Alexander, M. J. and Allison, J. T., “Decomposition-Based Design Optimisation of Electric Vehicle Powertrains using Proper Orthogonal Decomposition,” *International Journal of Powertrains*, Vol. 1, No. 1, 2011, pp. 72–92.

[183]Alexander, M. J., Allison, J. T., Papalambros, P. Y., and Gorsich, D. J., “Constraint Management of Reduced Representation Variables in Decomposition-Based Design Optimization,” *Journal of Mechanical Design*, Vol. 133, No. 10, 2011, pp. 101014.

[184]Allison, J. T. and Nazari, S., “Combined Plant and Controller Design Using Decomposition-Based Design Optimization and the Minimum Principle,” *In the Proceedings of the ASME 2010 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, No. DETC2010-28887, ASME, Montreal, Quebec, Canada, Aug. 2010, pp. 765–774.

[185]Tosserams, S., Etman, L. F. P., and Rooda, J. E., “A Classification of Methods for Distributed System Optimization Based on Formulation Structure,” *Structural and Multidisciplinary Optimization*, Vol. 39, No. 5, 2009, pp. 503–517.

[186]Martins, J. R. R. A. and Lambe, A. B., “Multidisciplinary Design Optimization: A Survey of Architectures,” *AIAA Journal*, to appear.

[187]Tosserams, S., Etman, L. F. P., and Rooda, J. E., “Augmented Lagrangian Coordination for Distributed Optimal Design in MDO,” *International Journal for Numerical Methods in Engineering*, Vol. 73, No. 13, March 2008, pp. 1885–1910.

[188]Allison, J. T., *Optimal Partitioning and Coordination Decisions in Decomposition-Based Design Optimization*, Ph.D. Dissertation, University of Michigan, 2008.

[189]Wolf, R. A., *Multiobjective Collaborative Optimization of Systems of Systems*, Ph.D. Dissertation, Massachusetts Institute of Technology, 2005.

[190]Lu, X.-P., Li, H.-L., and Papalambros, P., “Design Procedure for the Optimization of Vehicle Suspensions,” *International Journal of Vehicle Design*, Vol. 5, No. 1-2, 1984, pp. 129–142.

[191]McGeer, T., “Passive Dynamic Walking,” *The International Journal of Robotics Research*, Vol. 9, No. 2, April 1990, pp. 62–82.

[192]Pitti, A., Lungarella, M., and Kuniyoshi, Y., “Exploration of Natural Dynamics through Resonance and Chaos,” *In the Proceedings of the 9th International Conference on Intelligent Autonomous Systems*, Tokyo, Japan, March 2006, pp. 558–565.

[193]Williamson, M. M., “Oscillators and Crank Turning: Exploiting Natural Dynamics with a Humanoid Robot Arm,” *Philosophical Transactions. Series A, Mathematical, Physical, and Engineering Sciences*, Vol. 361, No. 1811, Oct. 2003, pp. 2207–23.

[194]Ries, C. and Jenkins, J., “Improving the Energy Performance of Buildings: Learning from the European Union and Australia,” Tech. rep., RAND Corporation, Santa Monica, CA, USA, 2009.

[195]Pfafferott, J., Herkel, S., and Jäschke, M., “Design of Passive Cooling By Night Ventilation: Evaluation of a Parametric Model and Building Simulation with Measurements,” *Energy and Buildings*, Vol. 35, No. 11, Dec. 2003, pp. 1129–1143.

[196]Pfafferott, J., Herkel, S., and Wambsganß, M., “Design, Monitoring and Evaluation of a Low Energy Office Building with Passive Cooling by Night Ventilation,” *Energy and Buildings*, Vol. 36, No. 5, May 2004, pp. 455–465.

[197]Voss, K., Herkel, S., Pfafferott, J., Löhnert, G., and Wagner, A., “Energy Efficient Office Buildings with Passive Cooling - Results and Experiences from a Research and Demonstration Programme,” *Solar Energy*, Vol. 81, No. 3, March 2007, pp. 424–434.

[198]Hirunlabh, J., Kongduang, W., Namprakai, P., and Khedari, J., “Study of Natural Ventilation of Houses by a Metallic Solar Wall under Tropical Climate,” *Renewable Energy*, Vol. 18, No. 1, Sept. 1999, pp. 109–119.

[199]Oldewurtel, F., Parisio, A., Jones, C. N., Morari, M Gyalistras, D., Gwerder, M., Stauch, V., Lehmann, B., and Wirth, K., “Energy Efficient Building Climate Control using Stochastic Model Predictive Control and Weather Predictions,” *In the Proceedings of the 2010 American Control Conference*, IEEE, Baltimore, MD, USA, July 2010, pp. 5100–5105.

[200]Tava, M. and Suzuki, S., “Multidisciplinary Design Optimization of the Shape and Trajectory of a Reentry Vehicle,” *Transactions of the Japan Society for Aeronautical and Space Sciences*, Vol. 45, No. 147, 2002, pp. 10–19.

[201]Chakrabarti, A., Shea, K., Stone, R., Cagan, J., Campbell, M., Hernandez, N. V., and Wood, K. L., “Computer-Based Design Synthesis Research: An Overview,” *Journal of Computing and Information Science in Engineering*, Vol. 11, No. 2, 2011, pp. 021003.

[202]Simpson, T. W. and Martins, J. R. R. A., “Multidisciplinary Design Optimization for Complex Engineered Systems: Report From a National Science Foundation Workshop,” *Journal of Mechanical Design*, Vol. 133, No. 10, Oct. 2011, pp. 101002.

[203]Bendsoe, M. P., *Topology Optimization*, Springer, 2nd ed., 2004.

[204]Rozvany, G. I. N., “A Critical Review of Established Methods of Structural Topology Optimization,” *Structural and Multidisciplinary Optimization*, Vol. 37, No. 3, 2009, pp. 217–237.

[205]Karbowski, D., Pagerit, S., Kwon, J., Rousseau, A., and von Pechmann, K. F. F., “”Fair” Comparison of Powertrain Configurations for Plug-In Hybrid Operation using Global Optimization,” *SAE Technical Paper*, Vol. 4970, 2009.

[206]Purnick, P. E. M. and Weiss, R., "The Second Wave of Synthetic Biology: From Modules to Systems," *Nature Reviews Molecular Cell Biology*, Vol. 10, No. 6, June 2009, pp. 410–422.

[207]Ma, W., Trusina, A., El-Samad, H., Lim, W. A., and Tang, C., "Defining Network Topologies that Can Achieve Biochemical Adaptation," *Cell*, Vol. 138, No. 4, Aug. 2009, pp. 760–773.

[208]Liu, J., *Modeling, Configuration and Control Optimization of Power-Split Hybrid Vehicles*, Ph.D. Dissertation, University of Michigan, 2007.

[209]Liu, J. and Peng, H., "A Systematic Design Approach for Two Planetary Gear Split Hybrid Vehicles," *Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility*, Vol. 48, No. 11, Nov. 2010, pp. 1395–1412.

[210]Swantner, A. and Campbell, M. I., "Topological and Parametric Optimization of Gear Trains," *Engineering Optimization*, Vol. 44, No. 11, Nov. 2012, pp. 1351–1368.

[211]Chapman, C. D., Saitou, K., and Jakiela, M. J., "Genetic Algorithms as an Approach to Configuration and Topology Design," *Journal of Mechanical Design*, Vol. 116, No. 4, Dec. 1994, pp. 1005–1012.

[212]Mitchell, M., *Complexity: A Guided Tour*, Oxford University Press, USA, April 2009.

[213]Setoodeh, S., Abdalla, M. M., and Gürdal, Z., "Combined Topology and Fiber Path Design of Composite Layers using Cellular Automata," *Structural and Multidisciplinary Optimization*, Vol. 30, No. 6, 2005, pp. 413–421.

[214]Hornby, G., Lipson, H., and Pollack, J., "Generative Representations for the Automated Design of Modular Physical Robots," *IEEE Transactions on Robotics and Automation*, Vol. 19, No. 4, 2003, pp. 703–719.

[215]Pedro, H. T. C. and Kobayashi, M. H., "On a Cellular Division Method for Topology Optimization," *International Journal for Numerical Methods in Engineering*, Vol. 88, No. 11, Dec. 2011, pp. 1175–1197.

[216]Stanford, B., Beran, P., and Kobayashi, M., "Simultaneous Topology Optimization of Membrane Wings and Their Compliant Flapping Mechanisms," *In the Proceedings of the 53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, No. AIAA-2012-1357, AIAA, Honolulu, HI, USA, April 2012.

[217]Maier, M. W., "Architecting Principles for Systems-of-Systems," *Systems Engineering*, Vol. 1, No. 4, Feb. 1998, pp. 267–284.

[218]Delaurentis, D. A., "Understanding Transportation as a System-of-Systems Design Problem," *In the Proceedings of the 43rd AIAA Aerospace Sciences Meeting and Exhibit*, No. AIAA-2005-0123, AIAA, Reno, NV, USA, Jan. 2005.

[219]Huntsberger, T., Stroupe, A., and Kennedy, B., "System of Systems for Space Construction," *In the Proceedings of the 2005 IEEE International Conference on Systems, Man and Cybernetics*, Vol. 4, IEEE, Waikoloa, HI, USA, Oct. 2005, pp. 3173–3178.

[220]Scattolini, R., "Architectures for Distributed and Hierarchical Model Predictive Control — A Review," *Journal of Process Control*, Vol. 19, No. 5, 2009, pp. 723–731.