

DETC2011-48521

**CO-DESIGN OF AN ACTIVE SUSPENSION USING SIMULTANEOUS DYNAMIC
OPTIMIZATION**

James T. Allison

Tufts University
Department of Mechanical Engineering
Medford, MA 02155

Email: james.allison@tufts.edu

Zhi Han

MathWorks, Inc.
Natick, MA 01760

ABSTRACT

Design of physical systems and associated control systems are coupled tasks; design methods that manage this interaction explicitly can produce system-optimal designs, whereas conventional sequential processes may not. Here we explore a new technique for combined physical system and control design (co-design) based on a simultaneous dynamic optimization approach known as direct transcription, which transforms infinite-dimensional control design problems into finite dimensional non-linear programming problems. While direct transcription problem dimension is often large, sparse problem structures and fine-grained parallelism (among other advantageous properties) can be exploited to yield computationally efficient implementations. Extension of direct transcription to co-design gives rise to a new problem structures and new challenges. Here we illustrate direct transcription for co-design using a new automotive active suspension design example developed specifically for testing co-design methods. This example builds on prior active suspension problems by incorporating a more realistic physical design component that includes independent design variables and a broad set of physical design constraints, while maintaining linearity of the associated differential equations.

1 Introduction

Often dynamic systems developed by engineers employ electronic control. Successful design of controlled systems is becoming increasingly important as their number and complex-

ity rises. In conventional design processes the physical system is designed first, followed by control system design [1]. This sequential approach does not fully account for coupling between physical artifact and control system design, and produces sub-optimal results [2, 3]. More effective multidisciplinary design methods that manage the artifact-control coupling explicitly are being developed, and are often termed *co-design* methods.

Engineers often employ simulation to predict dynamic system performance and support design decisions. Simulation can be linked to numerical optimization algorithms to aid effective exploration of design alternatives. Typically a nested approach is used: the optimization algorithm proposes a candidate design, which is then tested via simulation, and the results are then used in finding a new candidate design. Alternative approaches that perform simulation and optimization simultaneously are explored here, specifically a family of methods known as direct transcription (DT).

DT has been employed successfully for control design and parameter estimation across wide a range of applications, from chemical process design [4, 5], to aerospace trajectory optimization [6–8], to walking dynamics [9], to epidemiology [10], but has not yet been applied to co-design. In this article we introduce an extension of DT for co-design, explore the associated advantages and challenges, and demonstrate this approach using a newly-developed active suspension system example.

1.1 Direct Transcription

Here we assume the system to be designed can be modeled using a system of continuous differential equations:

$$\dot{\boldsymbol{\xi}} = \mathbf{f}_d(\boldsymbol{\xi}(t), \mathbf{u}(t), t) \quad (1)$$

$$\mathbf{0} = \mathbf{f}_a(\boldsymbol{\xi}(t), \mathbf{u}(t), t) \quad (2)$$

$$\mathbf{0} = \boldsymbol{\Psi}(\boldsymbol{\xi}(t_F), \mathbf{u}(t_F), t_F) \quad (3)$$

Equation (1) defines the time derivatives of a system model, where $\boldsymbol{\xi}(t)$ (length n_s) is the vector of state variables and $\mathbf{u}(t)$ is the control input. Equation (2) is the algebraic or path constraint, and if present, the model is a system of differential-algebraic equations (DAEs) [11]. The system may also have a boundary condition enforced at the simulation end time t_F (Eqn. (3)). If the model consists only of Eqn. (1) with initial conditions $\boldsymbol{\xi}(t_0) = \boldsymbol{\xi}_0$, it can be solved with forward simulation using a numerical ODE solver, such as the first-order Runge-Kutta method (Euler's method):

$$\boldsymbol{\xi}_i = \boldsymbol{\xi}_{i-1} + h_i \mathbf{f}_d(\boldsymbol{\xi}_{i-1}, \mathbf{u}(t_{i-1}), t_{i-1}) \quad (4)$$

Here h_i is the integration step size at t_i . Higher-order Runge-Kutta methods improve accuracy, as do implicit methods that employ an inner iterative method to solve for $\boldsymbol{\xi}_i$.

The open-loop control design problem can be written as:

$$\min_{\boldsymbol{\xi}(t), \mathbf{u}(t), t_F} J = \phi(\boldsymbol{\xi}(t), \mathbf{u}(t), t_F) \quad (5)$$

subject to Eqns. (1)–(3), where J is the response of a cost function. The final simulation time t_F often is an important quantity, particularly if the cost function depends on t_F or $\boldsymbol{\xi}(t_F)$. Final time might also be treated as a fixed parameter. When t_F varies, the number of time steps is fixed, but the time each step represents is scaled by the value of t_F . Feedback control can be accommodated by replacing $\mathbf{u}(t)$ with control design variables \mathbf{x}_c . Time-independent parameters \mathbf{p} can also be added to the set of optimization variables for parameter estimation problems:

$$\min_{\boldsymbol{\xi}(t), \mathbf{x}_c, \mathbf{p}} J = \phi(\boldsymbol{\xi}(t), \mathbf{x}_c, \mathbf{p}) \quad (6)$$

Optimal control problems may be solved using techniques based on the calculus of variations, including classical optimal control methods such as LQR [12]. This approach, however, requires derivation of derivatives of the system's Hamiltonian, so is limited in applicability. A common alternative eliminates $\boldsymbol{\xi}(t)$ from the optimization variable set by solving for state variable values using a nested simulation within the optimization problem:

$$\min_{\boldsymbol{\Xi}} J = \phi(\boldsymbol{\Xi}, \mathbf{x}_c) \quad (7)$$

Here $\boldsymbol{\Xi}$ is a discretized representation of the state variables, where the i th row of $\boldsymbol{\Xi}$ is $\boldsymbol{\xi}_i = \boldsymbol{\xi}(t_i)$, $i = 1, 2, \dots, n_t$. This nested approach enables use of finite-dimensional optimization methods, such as sequential quadratic programming [13], and is implemented commercial software [14], but has limitations. Discretization and variable step sizes results in a cost function that

is not smooth or arithmetically consistent [8], i.e., arithmetic operations used to compute cost vary with \mathbf{x}_c . Non-smoothness can be managed by increasing optimization finite difference step sizes or by using gradient-free methods, but this approach may still suffer from numerical sensitivity, particularly for highly nonlinear or stiff systems. Numerical sensitivity can be reduced by partitioning the simulation into n_T time segments using a technique know as multiple shooting. The state at the interfaces between time segments (\mathbf{Y}) is controlled directly by the optimization algorithm, and state continuity is enforced at convergence by defect constraints $\boldsymbol{\zeta}(\boldsymbol{\Xi}, \mathbf{Y})$:

$$\begin{aligned} \min_{\mathbf{x}_c, \mathbf{Y}} \quad & J = \phi(\boldsymbol{\Xi}, \mathbf{x}_c) \\ \text{s.t.} \quad & \boldsymbol{\zeta}_i(\boldsymbol{\Xi}, \mathbf{Y}) = \mathbf{0}, \quad i = 1, 2, \dots, n_T - 1 \end{aligned} \quad (8)$$

Row j of \mathbf{Y} is the state at the beginning of time segment $j + 1$. Within each time segment the system is simulated, and defect constraints require the state at the end of time segment j to match the j th row of \mathbf{Y} . For example, in Fig. 1 the value for $\boldsymbol{\xi}_7$ calculated using simulation over time segment 1 does not match the value for $\boldsymbol{\xi}_7$ specified by \mathbf{Y} that serves as the initial conditions for the simulation over time segment 2. This discontinuity in state at the interface is quantified using the defect constraint $\boldsymbol{\zeta}_1$.

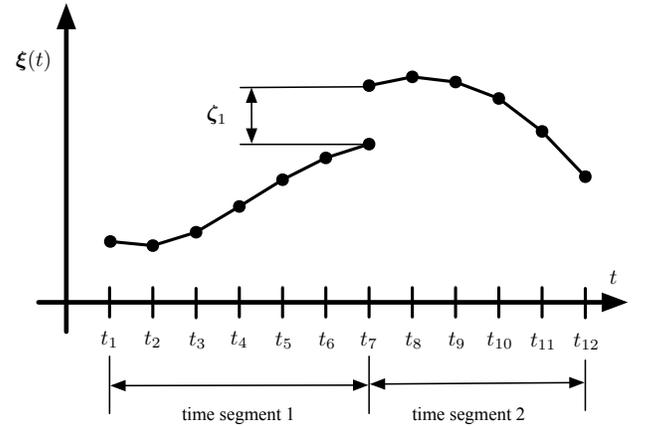


Figure 1: Illustration of continuity defect $\boldsymbol{\zeta}_1$ between time segments 1 and 2 using the multiple shooting method.

Because \mathbf{Y} is specified by the optimization algorithm, the n_T simulations are independent and can be calculated using coarse-grained parallel computing. The shorter simulation segments reduce numerical sensitivity.

In the limit, as $n_T \rightarrow n_t$, the simulation for each time segment collapses to a single difference equation. The set of defect constraints replace simulation completely, and $\boldsymbol{\Xi}$ becomes an optimization variable. This is Direct Transcription (DT); the infinite-dimensional optimal control problem is transcribed di-

rectly to a finite-dimension nonlinear programming problem:

$$\begin{aligned}
 \min_{\mathbf{\Xi}, \mathbf{x}_c, t_F} \quad & J = \phi(\mathbf{\Xi}, \mathbf{x}_c, t_F) \\
 \text{s.t.} \quad & \boldsymbol{\zeta}_i(\mathbf{\Xi}, \mathbf{x}_c, t_F) = \mathbf{0} \\
 & \hat{\mathbf{f}}_{ai}(\mathbf{\Xi}, \mathbf{x}_c, t_F) = \mathbf{0} \\
 \text{where} \quad & i = 2, \dots, n_t - 1, n_t
 \end{aligned} \tag{9}$$

A defect constraint is defined for each time step, and path constraints are discretized ($\hat{\mathbf{f}}_{ai}$). This formulation allows for variable t_F , and \mathbf{x}_c can be replaced easily with discretized $\mathbf{u}(t)$. As $\max(h_i) \rightarrow 0$, the optimality conditions for the DT problem converge to the optimality conditions for Eqn. (5).

While the dimension of the DT problem is large ($\mathbf{\Xi}$ has $n_t \cdot n_s$ elements), its problem structure is advantageous. At every optimization iteration $\mathbf{\Xi}$ is specified completely, so each of the $n_t - 1$ defect equations and path constraints are independent, enabling fine-grained parallel computing. The DT problem can be implemented such that the problem structure is nearly diagonal, enabling sparse finite difference gradient evaluations with as few as two perturbations regardless of problem dimension [8]. The DT problem is arithmetically consistent, enhancing numerical properties. Often analytical derivatives can be derived, further improving solution speed and accuracy. Variable step and order techniques for DT are available to provide error control [6, 8, 15].

Direct transcription is especially useful for problems with path constraints as it avoids another level of nesting within simulation to solve algebraic constraints; it is capable of handling higher-index DAEs [15, 16]. Inequality path constraints may be added directly, a problem element that often is impossible to solve using any other means [8].

If quadrature is performed using Euler's method, the defect constraints are:

$$\boldsymbol{\zeta}_i(\mathbf{\Xi}, \mathbf{x}_c) = \boldsymbol{\xi}_i - \boldsymbol{\xi}_{i-1} - h_i \mathbf{f}_d(\boldsymbol{\xi}_{i-1}, \mathbf{x}_c, t_{i-1})$$

Implicit quadrature methods for general nonlinear systems can be used with DT without requiring inner iterations to solve for the implicit state values. For example, the defect constraints for the implicit trapezoidal method are:

$$\boldsymbol{\zeta}_i(\mathbf{\Xi}, \mathbf{x}_c) = \boldsymbol{\xi}_i - \boldsymbol{\xi}_{i-1} - \frac{h_i}{2} (\mathbf{f}_d(\boldsymbol{\xi}_{i-1}, \mathbf{x}_c, t_{i-1}) + \mathbf{f}_d(\boldsymbol{\xi}_i, \mathbf{x}_c, t_i))$$

$\boldsymbol{\xi}_i$ is known a priori (it is a row of $\mathbf{\Xi}$), so defect equations may be evaluated without inner iteration. This concept extends to high-order implicit methods. While these methods are very accurate, they typically are impractical for forward simulation. Higher-order implicit methods can be used with DT, allowing fewer time steps (reducing problem dimension) while maintaining the required precision [5, 17, 18].

Helpful insights can be gained by considering the different paths through the combined state and design solution space traced by the nested and DT methods. Figure 2 is an abstraction of these subspaces, each of which may have high cardinality.

The nested approach begins with an initial design point \mathbf{x}_0 ,

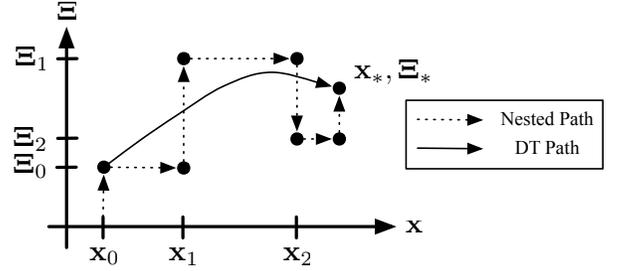


Figure 2: Conceptual solution trajectories through design and state subspaces.

which might include physical system design variables \mathbf{x}_p in addition to \mathbf{x}_c if a co-design problem is being solved. Forward simulation is then used to solve for state values $\boldsymbol{\xi}_0$ that are consistent with physics. The optimization algorithm chooses a new point \mathbf{x}_1 to reduce J , and the process repeats until convergence to \mathbf{x}_* , $\boldsymbol{\xi}_*$. The nested method can move only in one subspace at a time, whereas DT can move in both simultaneously, tracing a more direct path to the solution.

1.2 Relationship to MDO

In this section we clarify the relationship between the three formulations for optimal control discussed above (nested, multiple shooting (MS), and DT), and single-level formulations for multidisciplinary design optimization (MDO) [19, 20]. The nested solution approach in Eqn. (7) can be viewed as a special case of the multidisciplinary feasible (MDF) formulation, where a complete system analysis is performed at every optimization iteration. In other words, the system is consistent with physics (i.e., feasible) during the entire solution process because the differential equations are satisfied at each optimization step. As with MDF, optimization variables for the nested approach involve only independent design variables, and the analysis task is consolidated into a single monolithic process.

The multiple shooting approach in Eqn. (8) is a special case of the individual feasible formulation (IDF), where the analysis problem is partitioned into several smaller analysis tasks. Feasibility is maintained within analysis subproblems via simulation, but feasibility across subproblems is only guaranteed at convergence by defect constraints. In addition to independent design variables, the set of optimization variables includes interface (coupling) variables that quantify interactions between analysis subproblems. Auxiliary defect constraints ensure consistency across subproblems at convergence.

Direct transcription (Eqn. (9)) is a special case of the all-at-once (AAO) formulation, where the optimization algorithm manages both design and all analysis tasks. Analysis equations are embedded in optimization equality constraints, and state variables are part of the optimization variable set. These relation-

Table 1: Relationship between direct transcription and MDO.

	MDF/Nested	IDF/MS	AAO/DT
Solution Process	Entire analysis feasible at every optimization iteration	Individual analysis components feasible at every optimization iteration	Analysis feasible only at optimization convergence
Optimization Variables	Design variables	Design and interface variables	Design and state variables
Analysis Type	Consolidated analysis	Partially distributed analysis	Fully distributed analysis

ships are summarized in Table 1.

2 Direct Transcription for Co-Design

Co-design methods seek to identify system-optimal designs for controlled engineering systems by considering simultaneously physical artifact (plant) and control design. Conventional sequential (plant then control) design processes produce sub-optimal results because the two design domains are coupled. The sequential process is in essence a single iteration of block-coordinate descent [21], which even if iterated does not guarantee system optimality.

It is the experience of the authors that with the increasing complexity and demands associated with modern engineering systems, control engineers often struggle to meet system requirements when using sequential design. Additionally, many new system capabilities are possible only when plant and control design are considered together, including areas such as automotive suspension [22], bio-mimetic robotic locomotion [23, 24], controlled compliant mechanisms [25, 26], and structural design [27, 28].

New approaches are needed that preserve design freedom of both plant and control design domains throughout the process. Reyer et al. provide a review of such strategies [29]. Fathy proposed a co-design technique where the control design problem is nested within the plant design problem, producing system optimal results [30]. Allison and Nazari introduced a co-design method with distinct plant and control design subproblems that are guided toward system optimality using augmented Lagrangian coordination [31].

A fundamental challenge of solving co-design problems is managing a heterogeneous set of design variables; control variables either depend directly on time (e.g., $\mathbf{u}(t)$ in Eqn. (5)) or are proxies for time dependent variables, while plant design variables typically are time independent. Fathy's nested approach manages this by employing established optimal control methods to solve the inner control problem, and nonlinear programming algorithms to solve the outer plant design problem. Allison and Nazari derived an extension of optimal control theory to account for time-independent linking variables in the control design subproblem.

Direct transcription has been applied successfully to parameter estimation problems with time-independent variables

[5, 8, 32], and it is tempting to leap directly from applying DT to parameter estimation to co-design. The structures of the two problems are very different, however, and can be analyzed using graph-theoretic techniques [33]. Figure 3 illustrates a simplified directed-graph view of the standard DT problem and DT augmented to handle co-design problems. The plant analysis depends on both plant design variables and state variable values, and calculates plant design constraints \mathbf{g}_p and intermediate variables used in computing the objective function and dynamic constraints.

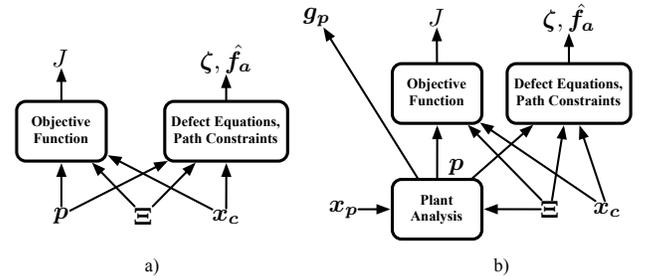


Figure 3: Analysis structure of a) DT and b) DT for co-design.

Dependence of plant analysis on Ξ and influence on J , ζ , and \mathbf{f}_a is a complication. This may preclude the use of analytical derivatives, and impacts the density of the Jacobian matrix, possibly precluding the use of large-scale optimization methods. These issues may be ameliorated via creative application of path constraints, or addition of states to the system problem. Nonetheless, DT applied to co-design problems still enables massive parallelization, and the solution of problems with ‘wicked’ elements, such as singular control problems or inequality path constraints. The DT formulation for co-design used here is:

$$\begin{aligned}
 \min_{\Xi, \mathbf{x}_c, \mathbf{x}_p, t_F} \quad & J = \phi(\Xi, \mathbf{x}_c, \mathbf{x}_p, t_F) \\
 \text{s.t.} \quad & \mathbf{g}_p(\Xi, \mathbf{x}_p) \leq \mathbf{0} \\
 & \zeta_i(\Xi, \mathbf{x}_c, \mathbf{x}_p, t_F) = \mathbf{0} \\
 & \mathbf{f}_{ai}(\Xi, \mathbf{x}_c, \mathbf{x}_p, t_F) = \mathbf{0}
 \end{aligned} \tag{10}$$

where $i = 2, \dots, n_t - 1, n_t$

3 Active Suspension Case Study

In this section we introduce a new model for an active suspension system, in fully reproducible detail, that includes a model of important physical system design considerations in addition to a dynamic model of the suspension. Effort was made to maintain linearity of system dynamics to preserve the usefulness of this model in other studies that are limited to linear time-invariant systems.

Consider the quarter-car model of a vehicle suspension illustrated in Fig. 4. The the sprung mass m_s (325 kg) and the unsprung mass m_{us} (65 kg) vertical positions are given by z_s and z_{us} , respectively. The system is excited by variations in road elevation z_0 as the vehicle travels at speed v .

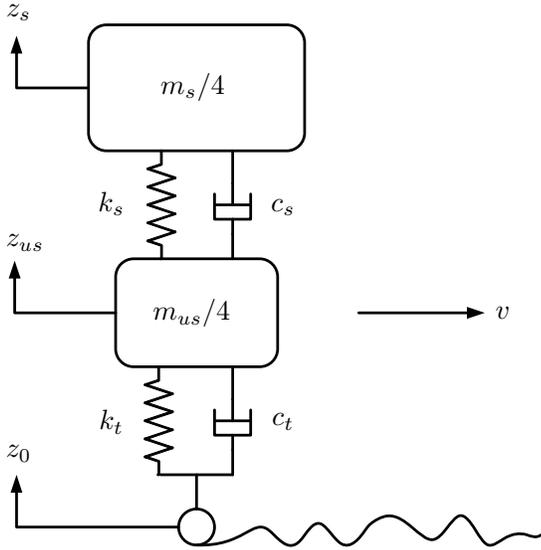


Figure 4: Quarter-car vehicle suspension model.

The passive dynamic response of this system can be characterized by the following system of linear differential equations:

$$\dot{\xi} = \mathbf{A}\xi + \begin{bmatrix} -1 \\ \frac{4c_t}{m_{us}} \\ 0 \\ 0 \end{bmatrix} \dot{z}_0, \quad (11)$$

$$\text{where } \xi = \begin{bmatrix} z_{us} - z_0 \\ \dot{z}_{us} \\ z_s - z_{us} \\ \dot{z}_s \end{bmatrix} \text{ and } \mathbf{A} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{4k_t}{m_{us}} & -\frac{4(c_s+c_t)}{m_{us}} & \frac{4k_s}{m_{us}} & \frac{4c_s}{m_{us}} \\ 0 & -1 & 0 & 1 \\ 0 & \frac{4c_s}{m_s} & -\frac{4k_s}{m_s} & -\frac{4c_s}{m_s} \end{bmatrix}$$

The tire and sprung mass spring stiffnesses are k_t ($232.5 \cdot 10^3$ N/m) and k_s , respectively, and the tire and spring mass damping rates are c_t and c_s , respectively. Here we assume $c_t = 0$. This canonical model has been used as an example in numerous design studies [34–36], including the design of active con-

trol systems [37–39], where an additional control input term $\mathbf{B}u$ is appended to Eqn. (11). Often k_s and c_s are treated as independent design variables [36, 37, 40], but are in fact dependent on geometric design and are subject to stress, fatigue, packaging, thermal, and other constraints. Here we introduce an extension to the basic quarter-car model that treats k_s and c_s as dependent variables, and incorporates a plant model that computes stiffness and damping coefficients as a function of independent geometric spring and damper design variables. The detailed spring and damper models are presented, followed by a demonstration of active suspension co-design using DT.

3.1 Spring Design

The vehicle suspension in this model utilizes a helical compression spring with squared and ground ends (Fig. 5). The suspension has a coil-over configuration; the coil spring surrounds the damper and they share the same axis. The model presented here is derived from [41]. See also [42] and [43] for alternative spring design optimization formulations. The independent spring design variables here are the helix diameter D , wire diameter d , spring pitch p , and the number of active coils N_a , which is relaxed to a continuous variable. These are components of the complete vector of plant design variables \mathbf{x}_p , along with other variables yet to be discussed. The formula for stiffness and a collection of spring design constraints are presented below.

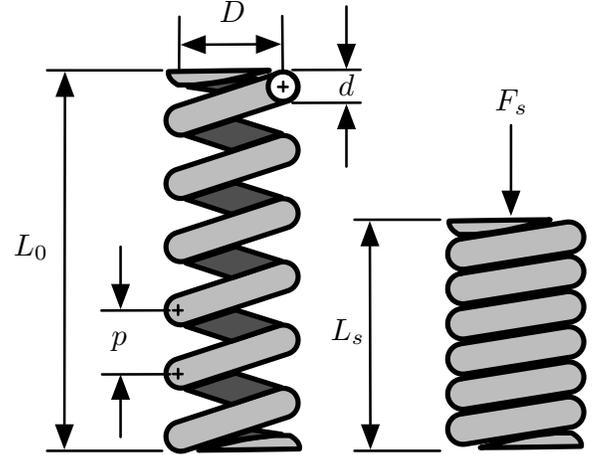


Figure 5: Helical compression spring with squared ground ends.

The free length of the spring is $L_0 = pN_a + 2d$, and the solid height is $L_s = d(N_a + Q - 1)$, where $Q = 1.75$ for squared and ground ends. F_s is the axial force at the solid height, and the spring constant is:

$$k_s = \frac{d^4 G}{8D^3 N_a \left(1 + \frac{1}{2C^2}\right)} \quad (12)$$

where G is the shear modulus (ASTM A401, $G = 77.2$ MPa),

and $C = D/d$ is the spring index. Springs with $C < 4$ are difficult to form, and springs with $C > 12$ can tangle. These requirements provide our first two plant design constraints:

$$g_1(\mathbf{x}_p) = 4 - C \leq 0 \quad (13)$$

$$g_2(\mathbf{x}_p) = C - 12 \leq 0 \quad (14)$$

The following constraint prevents buckling:

$$g_3(\mathbf{x}_p) = L_0 - 5.26D \leq 0 \quad (15)$$

The uncompressed spring must fit within the specified pocket length ($L_{0\max} = 0.40$ m) for the vehicle:

$$g_4(\mathbf{x}_p) = L_0 - L_{0\max} \leq 0 \quad (16)$$

The outer spring diameter must not exceed $D_{o\max} = 0.25$ m to prevent interference with vehicle components:

$$g_5(\mathbf{x}_p) = d + D - D_{o\max} \leq 0 \quad (17)$$

The spring inner diameter must be large enough to fit around the damper with at least $\delta_{dc} = 9.0$ mm clearance:

$$g_6(\mathbf{x}_p) = d - D + D_p + 2(\delta_{dc} + t_d) \leq 0 \quad (18)$$

where D_p is the damper piston diameter and $t_d = 2.0$ mm is the damper wall thickness.

While suspension rattlepace (permissible peak-to-peak displacement [34]) is often treated as an objective function [35,37], it is natural to formulate it as a constraint. Here peak displacement δ_{\max} is calculated using a ramp input (road grade of 25% at a speed of 10 m/s) to check for maximum rattlepace violation:

$$g_7(\mathbf{x}_p) = \delta_{\max} - L_0 + L_s + L_B + \delta_g \leq 0 \quad (19)$$

This test requires specification of damper design, discussed in the next section. L_B is bumpstop thickness (0.02 m), and $\delta_g = m_s/4k_s$ is static suspension deflection.

Spring shear stress τ at maximum deflection must not exceed shear yield stress S_{sy} . The following model of shear stress incorporates the Bergsträsser augmentation factor:

$$\tau = \left(\frac{4C+2}{4C-2} \right) \frac{8F_s D}{\pi d^3}$$

where the axial force at the spring solid height is calculated using $F_s = k_s(L_0 - L_s)$. The scaled constraint is:

$$g_8(\mathbf{x}_p) = (\tau n_d - S_{sy})/S_{sy} \leq 0 \quad (20)$$

with a design factor of $n_d = 1.2$. The shear yield strength is assumed proportional to the ultimate tensile strength for the spring material: $S_{sy} = 0.65S_{ut}$, where $S_{ut} = A \cdot 10^6/(d^m)$, $A = 1974$ MPa·mm^m, and $m = 0.108$. To ensure spring linearity and validity of Eqn. (11), the following constraint must be satisfied:

$$g_9(\mathbf{x}_p) = 0.15 + 1 - \frac{L_0 - L_s}{\delta_g + 1.1\delta_{\max f}} \quad (21)$$

where $\delta_{\max f}$ is the maximum deflection during a simulation of the quarter-car over a rough road (IRI = 7.37) [36, 44] at $v = 20$ m/s. Under these rough road conditions the maximum axial

spring force is $F_{\max} = k_s(\delta_{\max f} + \delta_g)$, and the minimum force is $F_{\min} = k_s(\delta_g - \delta_{\max f})$. The mean axial force and force amplitude are $F_m = (F_{\max} + F_{\min})/2$ and $F_a = (F_{\max} - F_{\min})/2$, respectively. The shear stress mean and amplitude are:

$$\tau_m = \left(\frac{4C+2}{4C-2} \right) \frac{8F_m D}{\pi d^3}, \tau_a = \left(\frac{4C+2}{4C-2} \right) \frac{8F_a D}{\pi d^3}$$

Soderberg and Zimmerli fatigue criterion are applied here:

$$g_{10}(\mathbf{x}_p) = \frac{\tau_a}{S_{e2}} + \frac{\tau_m}{S_{sy}} - 1 \leq 0, \quad \text{where } S_{e2} = 0.24S_{ut}/n_d \quad (22)$$

$$g_{11}(\mathbf{x}_p) = \frac{\tau_a n_d - 241 \cdot 10^6}{241 \cdot 10^6} \leq 0 \quad (23)$$

3.2 Damper Design

Equation (11) assumes linear damping, i.e., the damping force F_D is proportional to damper piston velocity \dot{x}_3 . Real suspension dampers are highly nonlinear due to both design intent and limitations of practical physical implementations (Fig. 6). Here we explore an approach to constructing a linear damper.

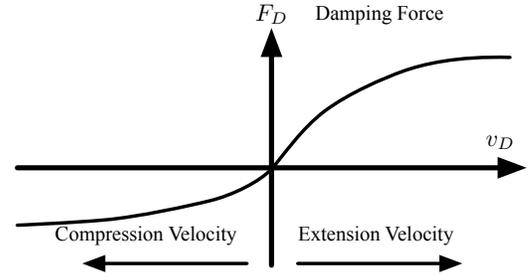


Figure 6: Typical nonlinear damping curve.

Figure 7 illustrates a single tube telescopic damper. On jounce (compression) hydraulic fluid flows from the compression chamber to the extension chamber through the compression valve, and on rebound (extension) fluid flows through the extension valve in the reverse direction. The pressurized gas chamber, separated from the hydraulic fluid by a floating piston, compensates for volume change from rod movement. Here we neglect the effect of the foot valve, and assume the piston valves are spring-biased spool valves, as shown in Fig. 8. Other valve types exist and are in common use, including disc, rod, and shim valves [45]. As the damper is compressed, the compression valve opens and fluid flows through ports in the side of the spool valve.

3.2.1 Damping Properties Exposed port area (A_v) variation with valve lift (x_v) is a key design element that controls damping curve shape. Under standard assumptions, the damper

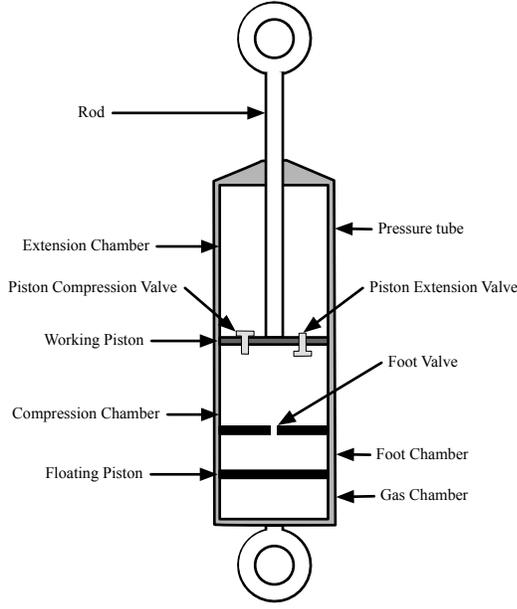


Figure 7: Sectional view of a single-tube telescopic damper.

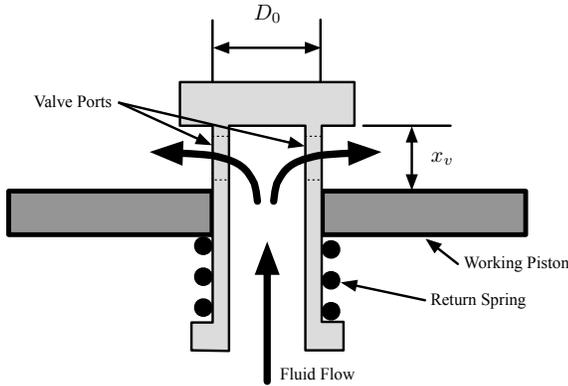


Figure 8: Sectional view of the piston compression valve.

will be linear if either $A_v \propto \sqrt{x_v}$ or $x_v \propto \sqrt{F_v}$, where F_v is the valve spring axial force. The latter relationship requires a non-linear valve spring. The former relationship (used here) requires that the valve port shape is designed such that A_v increases proportionately with $\sqrt{x_v}$. More precisely, the ideal valve port area function is:

$$\tilde{A}_v(x_v) = C_2 C_0 \sqrt{x_v} \quad (24)$$

where C_2 is the damper valve coefficient, and $C_0 = \pi D_0$ is the outer circumference of the spool valve. Port shape can be characterized by the arc length exposed by the ports at the top of the working piston as a function of valve lift: $C_e(x_v)$. The ideal valve

port area function can be re-written as:

$$\tilde{A}_v(x_v) = \int_0^{x_v} C_e(\tau) d\tau \quad (25)$$

Solving Eqns. (24) and (25) for $C_e(x_v)$ reveals that linear damping requires the following exposed arc function:

$$\tilde{C}_e(x_v) = \frac{d}{dx_v} \tilde{A}_v(x_v) = \frac{1}{2} C_2 C_0 x_v^{-1/2} \quad (26)$$

Unfortunately as $x_v \rightarrow 0$, $\tilde{C}_e(x_v) \rightarrow \infty$, which is physically unrealizable; $C_e(x_v)$ must remain less than the spool valve outer circumference C_0 . Several candidate functions for approximating $\tilde{C}_e(x_v)$ were evaluated using a least squares approach with the requirement $C_e(x_v) \leq \eta C_0$ for all $x_v \geq 0$. The coefficient $0 < \eta < 1$ defines the upper limit of the proportion of C_0 that may be exposed ($\eta = 0.9$ here). The following exposed arc function was selected:

$$C_e(x_v) = a_1 (x_v + a_1^2)^{-1/2} \quad (27)$$

where $a_1 = C_3(x_m - C_3)^{-1/2}$, $x_m = A_0 P_m / k_v$ is the maximum valve lift at the maximum allowed damper pressure P_m , k_v is the spool valve spring constant, $A_0 = \pi D_0^2 / 4$ is the spool valve frontal area, and

$$C_3 = \frac{C_2 D_0}{2\eta} \sqrt{\frac{\pi P_m}{k_v}}$$

The damper valve coefficient is defined here as $C_2 = \eta A_f \sqrt{x_m}$, where A_f is an area factor that can be used to tune port shape. Here $A_f = 0.1$. Decreasing A_f brings $C_e(x_v)$ closer to ideal for linear damping, but increases the valve diameter required to deliver a particular suspension damping coefficient c_s , which is calculated using:

$$c_s = \frac{D_p^4}{8C_d C_2 D_0^2} \sqrt{\frac{\pi k_v \rho_1}{2}} \quad (28)$$

where D_p is the working piston diameter, ρ_1 is the damper fluid density (850 kg/m^3), and C_d is the discharge coefficient (≈ 0.7 for spool valves). Damper stroke (total axial displacement) D_s is chosen here as an independent design variable. The complete plant design vector can now be defined:

$$\mathbf{x}_p = [d, D, p, N_a, D_o, D_p, D_s]$$

3.2.2 Thermal Properties Fluid temperature increase due to energy dissipation is an important consideration in damper design as it can induce damper fade. Heat generation in the damper is calculated based on the damper coefficient and suspension stroke velocity, i.e., $q_{\text{gen}} = c_s \xi_2^2$. The same rough road profile is used that was specified for the spring fatigue calculations.

Figure 9 illustrates the thermal model for the damper, similar to the model found in [46], except here constant viscosity is assumed. Heat is generated in the damper fluid, which has heat capacity c_{p1} (2500 J/kgK), density ρ_1 , and volume v_1 . Heat is

conducted through the steel damper tube (with conduction coefficient $k_2 = 60.5$ W/mK) to the atmosphere at a constant temperature $T_4 = 300$ K. The convection coefficient between the damper and the atmosphere is h . Heat capacity in the steel shell is small compared to that of the damper fluid (c_{p1}), and is assumed to be zero here, so $T_2 = T_3$.

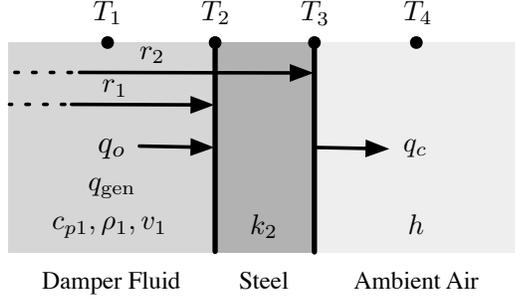


Figure 9: Damper heat transfer model.

Heat flow from the damper fluid through the shell is:

$$q_0 = q_{gen} - \rho_1 v_1 c_{p1} \frac{\partial T_1}{\partial t} = \frac{2\pi L_2 k_2}{\ln r_2/r_1} (T_1 - T_3) = h A_4 (T_3 - T_4)$$

where v_1 is the fluid volume, r_1 and r_2 are the inner and outer shell radii, L_2 is the shell height, and $A_4 = 2\pi r_2 L_2$ is the external surface area of the shell. This DAE can be converted to a single ordinary differential equation via substitution. For convenience we can rewrite this system as:

$$q_{gen} - b_1 \dot{T}_1 = b_2 (T_1 - T_3) = b_3 (T_3 - T_4)$$

where the appropriate constants are replaced with b_1 , b_2 , and b_3 . Choosing T_1 as the state variable, the resulting ODE is:

$$\dot{T}_1 = -\frac{b_2 b_3}{b_1 b_2 + b_1 b_3} T_1 + \frac{b_2 b_3}{b_1 b_2 + b_1 b_3} T_4 + \frac{q_{gen}}{b_1} \quad (29)$$

which can be simulated using the time history of ξ_2 as input to predict the resulting damper fluid temperature $T_1(t)$.

3.2.3 Damper Constraints

$$g_{12}(\mathbf{x}_p) = L_0 - L_s - D_s \leq 0 \quad (30)$$

ensures adequate damper range of motion.

$$g_{13}(\mathbf{x}_p) = 2D_s + \ell_{d1} + \ell_{d2} - L_{0max} \leq 0 \quad (31)$$

requires the damper to fit within the pocket length $L_{0max} = 0.4$ m; $\ell_{d1} = 0.02$ m and $\ell_{d2} = 0.04$ m quantify the space required for damper components above and below the working piston range, respectively. The damper fluid temperature constraint prevents fade and seal damage:

$$g_{14}(\mathbf{x}_p) = T_{1max} - T_{1allow} \leq 0 \quad (32)$$

where T_{1max} is the maximum damper temperature achieved

across all vehicle simulations and $T_{1allow} = 390$ K.

$$g_{15}(\mathbf{x}_p) = P_{max} - P_{allow} \leq 0 \quad (33)$$

ensures the maximum damper pressure achieved during simulation (P_{max}) does not exceed the seal maximum pressure ($P_{allow} = 4.75 \cdot 10^6$ Pa).

$$g_{16}(\mathbf{x}_p) = \dot{\xi}_{3max} - \dot{\xi}_{3allow} \leq 0 \quad (34)$$

protects damper components from excessive velocity ($\dot{\xi}_{3allow} = 5.0$ m/s). The following is a clearance requirement for the maximum amount of spool valve lift, where $x_{vallow} = 0.01$ m:

$$g_{17}(\mathbf{x}_p) = x_{vmax} - x_{vallow} \leq 0 \quad (35)$$

3.3 Active Suspension Co-Design

The control input to the quarter-car active suspension system is an active force added between the sprung mass and unsprung masses. Refer to [30] for a detailed model of the control system. The active suspension model with the control input is described as:

$$\dot{\xi} = \mathbf{A}\xi + \begin{bmatrix} -1 \\ \frac{4c_t}{m_{us}} \\ 0 \\ 0 \end{bmatrix} \dot{z}_0 + \begin{bmatrix} 0 \\ -\frac{1}{m_{us}} \\ 0 \\ \frac{1}{m_s} \end{bmatrix} u \quad (36)$$

where u is the control input.

The objective function incorporates handling, comfort, and control cost:

$$J = \int_0^{t_f} (r_1 (z_{us} - z_0)^2 + r_2 \dot{z}_s^2 + r_3 u^2) dt$$

where the weights are $r_1 = 3 \times 10^4$, $r_2 = 3 \times 10^4$ and $r_3 = 10^{-6}$. The plant design variable bounds $\mathbf{x}_p \leq \bar{\mathbf{x}}_p$ used here are:

$$\mathbf{x}_p = [0.005, 0.05, 0.05, 3, 0.005, 0.03, 0.1]$$

$$\bar{\mathbf{x}}_p = [0.02, 0.4, 0.5, 12, 0.008, 0.06, 0.3]$$

An initial simulation of the open-loop system is performed to determine the time steps h_i used in solving Eqn. (10) using DT. A large-scale interior-point algorithm from the MATLAB[®] optimization toolbox was used, taking advantage of problem structure by using sparse Hessian and Jacobian matrices and analytical derivatives. The trapezoidal ODE solver is used here to form defect constraints. As a comparison Fathy's nested co-design approach was used as well, where \mathbf{x}_p was managed in the outer loop, and the inner loop solved the control design problem (with fixed \mathbf{x}_p) using algebraic Riccati equations [12] with the infinite-horizon linear quadratic regulator as the control law. Because LQR in the MATLAB control system toolbox solves algebraic Riccati equation, which is much simpler than the finite horizon Riccati equations, comparison of computational expense between these two methods cannot be made directly.

3.3.1 Active Suspension Co-Design: Ramp Input

For illustration the co-design problem is solved first with ramp road input only using DT (no thermal or fatigue constraints) to validate DT as a solution approach against LQR. The optimal plant design is:

$$\mathbf{x}_{p*} = [0.0184, 0.1432, 0.0537, 3.2979, 0.0057, 0.0538, 0.1560]$$

When using the nested method, the outer optimization loop often terminates at an inferior local optimum with respect to \mathbf{x}_p compared to the DT solution. Figure 10 compares system response of the DT and LQR control designs, using the DT result for plant design. For ramp input, LQR is the optimal solution to the infinite horizon control problem, which is usually very close to the finite horizon optimum. The DT response is nearly identical to LQR, indicating it is an accurate solution approach.

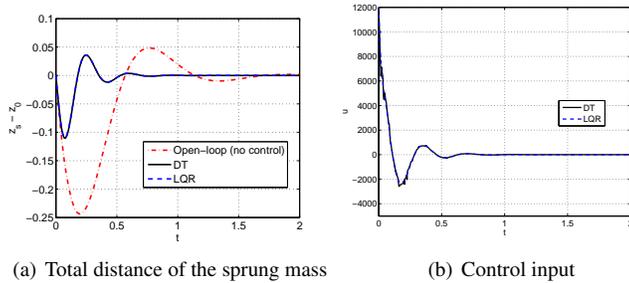


Figure 10: Comparison of open-loop and closed-loop system responses for the ramp road input

The objective function value for DT and LQR control solutions (with the same plant design) are compared against the objective for the optimal passive suspension below. Both techniques improve the objective significantly, and arrive at similar cost values. DT solution requires 733 SQP iterations and 738 function evaluations.

No active control	DT	LQR
442.4556	25.9698	26.3486

3.3.2 Combined Active Suspension Co-Design

The active suspension problem was also solved using the complete road input (both ramp and rough road). The objective function is scaled differently for these two inputs: $J = 1 \times 10^{-2} \cdot J_{ramp} + J_{rough}$. The DT plant design results are:

$$\mathbf{x}_{p*} = [0.0168, 0.168, 0.0536, 3.3317, 0.0066, 0.0415, 0.1577]$$

Figure 11 compares system response for DT and LQR control solutions, again using the plant design produced by DT in both cases. The objective function values are:

	No active control	DT	LQR
J_{ramp}	460.1687	34.6450	34.0477
J_{rough}	2.4673	0.9399	1.6023

DT produces an accurate result over the ramp input portion using the LQR solution as a benchmark. For the rough road input, LQR gives a reasonable performance improvement compared to the passive case, while DT produces the best design. This is as expected as DT implements an open-loop controller that incorporates more system input detail. The DT solution requires 1180 SQP iterations and 1218 function evaluations.

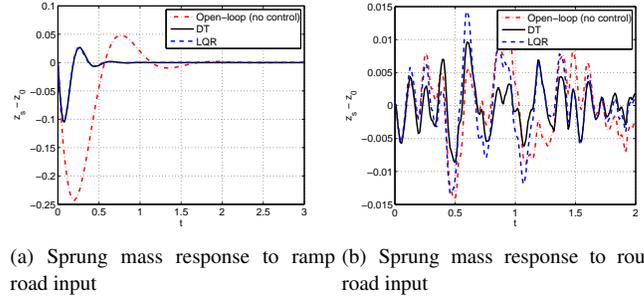


Figure 11: Comparison of open-loop and closed-loop system responses

4 Conclusion

Direct transcription was reviewed, and an extension of DT to co-design problems was introduced and demonstrated using an active suspension design problem. The case study built upon existing active suspension design problems by incorporating detailed physical system design considerations and physically independent design variables. The system model maintains linearity of the differential equations, making it suitable for a range of other co-design studies. While DT results in large problem dimension, it is arithmetically consistent, enables fine-grained parallelism and the use of higher-order implicit methods, and can be applied to challenging design problems that are otherwise unsolvable (e.g., inequality path constraints, open-loop instabilities). The case study illustrates that DT can be advantageous compared to the nested co-design approach. DT offers a path forward for solving especially difficult co-design problems. The large size and density of the DT optimization problem remains a challenge for its applicability in co-design. The new problem structure in some cases makes efficient DT implementation difficult, at least with currently known techniques. Solution can also be complicated when the range of plant design variables can cause both stiff and non-stiff dynamics. Exploration of this new problem structure and development of more efficient DT co-design solution implementations are opportunities for further contribution to the design of modern engineering systems.

REFERENCES

- [1] Friedland, B., 1996. *Advanced Control System Design*. Prentice-Hall.
- [2] Fathy, H. K., Reyer, J. A., Papalambros, P. Y., and Ulsoy, A. G., 2001. "On the coupling between the plant and controller optimization problems". *Proceedings of the 2001 American Control Conference*, pp. 1864–1869.
- [3] Peters, D., Papalambros, P., and Ulsoy, A., 2009. "On measures of coupling between the artifact and controller optimal design problems". In 2009 ASME Design Engineering Technical Conference, ASME, ASME.
- [4] Cervantes, A., and Biegler, L., 2001. "Optimization Strategies for Dynamic Systems". *Encyclopedia of Optimization*, 4(2), p. 216.
- [5] Biegler, L., 2007. "An overview of simultaneous strategies for dynamic optimization". *Chemical Engineering and Processing: Process Intensification*, 46(11), pp. 1043–1053.
- [6] Hargraves, C. R., and Paris, S. W., 1987. "Direct trajectory optimization using nonlinear programming and collocation". *Journal of Guidance, Control, and Dynamics*, 10(4), July, pp. 338–342.
- [7] Ozimek, M. T., Grebow, D. J., and Howell, K. C., 2008. "Solar Sails and Lunar South Pole Coverage". In AIAA/AAS Astrodynamics Specialist Conference and Exhibit, AIAA, pp. 1–25.
- [8] Betts, J. T., 2010. *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*. SIAM, Philadelphia, PA.
- [9] Xiang, Y., Arora, J. S., and Abdel-Malek, K., 2010. "Physics-based modeling and simulation of human walking: a review of optimization-based and other approaches". *Structural and Multidisciplinary Optimization*, 42(1), Mar., pp. 1–23.
- [10] Jung, E., Lenhart, S., and Feng, Z., 1996. "Optimal Control of Treatments in a Two-Strain Tuberculosis Model". *Discrete and Continuous Dynamical Systems—Series B*, 19(2), pp. 476–482.
- [11] Brenan, K., Campbell, S., and Petzold, L., 1996. *Numerical solution of initial-value problems in differential-algebraic equations*. Society for Industrial Mathematics.
- [12] Athans, M., and Falb, P., 1966. *Optimal control: an introduction to the theory and its applications*. McGraw-Hill, New York.
- [13] Powell, M., 1978. "Algorithms for Nonlinear Constraints That Use Lagrangian Functions". *Mathematical Programming*, 14, pp. 224–248.
- [14] MathWorks, Inc. Simulink Design Optimization Product Reference. <http://www.mathworks.com/products/sl-design-optimization/>. Accessed February 22, 2011.
- [15] Vasantharajan, S., and Biegler, L., 1990. "Simultaneous Strategies for Optimization of Differential-Algebraic Systems with Enforcement of Error Criteria". *Computers & chemical engineering*, 14(10), pp. 1083–1100.
- [16] Laird, C. D., Wong, A. V., and Akesson, J., 2011. "Parallel Solution of Large-Scale Dynamic Optimization Problems". In 21st European Symposium on Computer Aided Process Engineering — ESCAPE 21.
- [17] Herman, A., and Conway, B., 1996. "Direct optimization using collocation based on high-order Gauss-Lobatto quadrature rules". *Journal of Guidance Control and Dynamics*, 19(3), pp. 592–599.
- [18] Williams, P., 2009. "Hermite–Legendre–Gauss–Lobatto Direct Transcription in Trajectory Optimization". *Journal of Guidance, Control, and Dynamics*, 32(4), pp. 1392–1395.
- [19] Cramer, E., Dennis Jr, J., Frank, P., Lewis, R., and Shubin, G., 1994. "Problem Formulation for Multidisciplinary Optimization". *SIAM Journal on Optimization*, 4(4), p. 754.
- [20] Allison, J. T., Kokkolaras, M., and Papalambros, P. Y., 2007. "On Selecting Single-Level Formulations for Complex System Design Optimization". *Journal of Mechanical Design*, 129(9), p. 898.
- [21] Bertsekas, D. P., and Tsitsiklis, J. N., 1997. *Parallel and Distributed Computation: Numerical Methods*. Athena Scientific, Belmont, MA.
- [22] Smith, M. C., and Walker, G. W., 2005. "Interconnected vehicle suspension". *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 219(3), Jan., pp. 295–307.
- [23] Schroer, R., Boggess, M., Bachmann, R., Quinn, R., and Ritzmann, R., 2004. "Comparing cockroach and Whegs robot body motions". *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, pp. 3288–3293 Vol.4.
- [24] Lee, J., Lamperski, A., Schmitt, J., and Cowan, N., 2006. "Task-level control of the lateral leg spring model of cockroach locomotion". *Fast Motions in Biomechanics and Robotics*, 340, pp. 167–188.
- [25] Nishigaki, H., and Kawashima, K., 1998. "Motion control and shape optimization of a suitlike flexible arm". *Structural Optimization*, 15(3-4), June, pp. 163–171.
- [26] Trease, B., 2008. "Topology Synthesis of Compliant Systems with Embedded Actuators and Sensors". Ph.D. dissertation, University of Michigan.
- [27] Carmichael, D., 1990. "Structural Optimization and System Dynamics". *Structural and Multidisciplinary Optimization*, 2(2), pp. 105–108.
- [28] Fares, M., Youssif, Y., and Hafiz, M., 2005. "Multiobjective design and control optimization for minimum thermal postbuckling dynamic response and maximum buckling temperature of composite laminates". *Structural and Multidisciplinary Optimization*, 30(2), Mar., pp. 89–100.
- [29] Reyer, J., Fathy, H., Papalambros, P., and Ulsoy, A., 2001. "Comparison of combined embodiment design and control optimization strategies using optimality conditions". *Proceedings of the ASME Design Engineering Technical Conferences, JE Renaud, ed*, 2, pp. 1023–1032.
- [30] Fathy, H., 2003. "Combined plant and control optimization: Theory, strategies and applications". Ph.D. dissertation, University of Michigan.
- [31] Allison, J., and Nazari, S., 2010. "Combined Plant and Controller Design Using Decomposition-Based Design Optimization and the Minimum Principle". In the Proceedings of the 2010 ASME Design Engineering Technical Conference, ASME.
- [32] Williams, P., and Trivailoa, P., 2005. "Optimal parameter estimation of dynamical systems using direct transcription methods". *Inverse Problems in Science and Engineering*, 13(4), pp. 377–409.
- [33] Reyer, J., and Papalambros, P., 2000. "An investigation into modeling and solution strategies for optimal design and control". In ASME Design Engineering Technical Conferences.
- [34] Kasturi, P., and Dupont, P., 1998. "Constrained Optimal Control of Vibration Dampers". *Journal of Sound and Vibration*, 215(3), Aug., pp. 499–509.
- [35] Gobbi, M., 2001. "Analytical Description and Optimization of the Dynamic Behaviour of Passively Suspended Road Vehicles". *Journal of Sound and Vibration*, 245(3), Aug., pp. 457–481.
- [36] Allison, J., 2008. "Optimal partitioning and coordination decisions in decomposition-based design optimization". Ph.D. dissertation, University of Michigan.
- [37] Fathy, H., Papalambros, P., Ulsoy, A., and Hrovat, D., 2003. "Nested Plant / Controller Optimization with Application to Combined Passive / Active Automotive Suspensions". In American Control Conference, 2003., Vol. 4, IEEE, pp. 3375–3380.
- [38] Bourmistrova, A., Storey, I., and Subic, A., 2005. "Multiobjective Optimisation of Active and Semi-Active Suspension Systems with Application of Evolutionary Algorithm". In International Conference on Modeling and Simulation, Melbourne, pp. 12–15.
- [39] Alyaouq, S., Papalambros, P., and Ulsoy, A., 2007. "Combined design and robust control of a vehicle passive/active suspension". In European Control Conference.
- [40] Verros, G., Natsiavas, S., and Papadimitriou, C., 2005. "Design

Optimization of Quarter-car Models with Passive and Semi-active Suspensions under Random Road Excitation”. *Journal of Vibration and Control*, **11**(5), pp. 581–606.

- [41] Shigley, J., Mischke, C., and Budynas, R., 2003. *Mechanical Engineering Design*. McGraw-Hill, New York.
- [42] Azarm, S., 1982. An Interactive Design Procedure for Optimization of Helical Compression Springs. Technical Report UM-MEAM-82-7, University of Michigan.
- [43] Stoicescu, A., 2009. “On the Optimal Design of Helical Springs of an Automobile Suspension”. *U.P.B. Sci. Bull., Series D*, **71**(1).
- [44] Sayers, M., and Karamihas, S., 1998. The Little Book of Profiling. Technical Report University of Michigan Transportation Research Institute.
- [45] Dixon, J. C., 2007. *The Shock Absorber Handbook*, second ed. John Wiley & Sons Ltd, Chichester, UK.
- [46] Lion, A., and Loose, S., 2002. “A Thermomechanically Coupled Model for Automotive Shock Absorbers: Theory, Experiments and Vehicle Simulations on Test Tracks”. *Vehicle System Dynamics*, **37**(4), Apr., pp. 241–261.