Introduction to System Optimization: Part 1

James Allison

ME 555
March 7, 2007

System Optimization

- Subsystem optimization results $\Rightarrow$ optimal system?
- Objective: provide tools for developing system optimization strategy

System Optimization

- Subsystem optimization results $\Rightarrow$ optimal system?
- Objective: provide tools for developing system optimization strategy

Overview

1. Systems and Interactions
   - Definitions
   - Interaction Examples
2. System Consistency and Optimality
3. System Optimization Methods
   - AiO
   - IDF
   - ATC

System Definition

What makes something a system?

System Definition

What makes something a system?

- Comprised of several components (or subsystems)

System Definition

What makes something a system?

- Comprised of several components (or subsystems)
- Interactions exist between the components

System Definition

What makes something a system?

- Comprised of several components (or subsystems)
- Interactions exist between the components
    - Some aspect of one component influences the effect of changes in another component

Explicit Analysis Interactions

Analysis: evaluate f, $\mathbf{g}$, and $\mathbf{h}$ in design problem for a given $\mathbf{x}$

$\mathbf{y}_{ij}$: analysis output passed from subsystem j to i
(coupling variable)

Example: deflection and pressures in *aeroelastic* analysis

Interaction Example (First Type)

System objective function:

$$f(y_{12}, y_{13}) = c_1(y_{12} - c_2)^2 + c_3(y_{13} - c_4)^2 + c_5 y_{12} y_{13}$$

Inputs to subsystem 1:

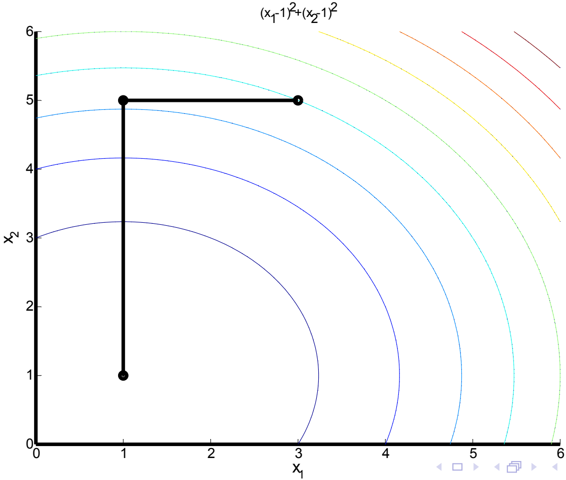
- y_{12}: output (response) of subsystem 2
- y_{13}: output (response) of subsystem 3

Three objective function terms:

1. Depends only on SS2 response
2. Depends only on SS3 response
3. Depends on a combination of the responses (interaction)

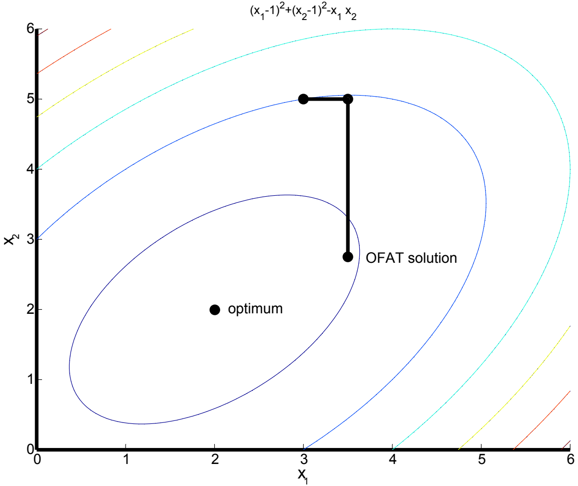
Interaction Example (First Type)

Case I: No Interaction $\mathbf{c} = (1, 1, 1, 1, 0)^T$



Interaction Example (First Type)

Case II: Interaction Present $\mathbf{c} = (1, 1, 1, 1, -1)^T$



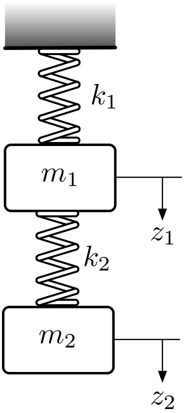
Interaction Example (Second Type)

Governing Equations of Motion:

$$\begin{bmatrix} m_1 & 0 \\ 0 & m_2 \end{bmatrix} \begin{Bmatrix} \ddot{z}_1 \\ \ddot{z}_2 \end{Bmatrix} +$$

$$\begin{bmatrix} k_1 + k_2 & -k_2 \\ -k_2 & k_2 \end{bmatrix} \begin{Bmatrix} z_1 \\ z_2 \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix}$$

Coupled in stiffness: Motion of m_1 (i.e. $x_1(t)$) is dependent of the value of k_2 ($\therefore z_1(t)$ depends on the state of z_2), and visa-versa.



Interaction Representation

Example system:

$$
\begin{aligned}
&a_1(x_1, x_2, x_4)\\
&a_2(x_6, y_{21})\\
&a_3(x_2, x_3, x_4, y_{31}, y_{34})\\
&a_4(x_4, x_5, y_{41})
\end{aligned}
$$

Interaction Identification

Example: Interaction between automotive subsystems

- structure
- powertrain
- suspension
- steering
- braking
- cabin (interior geometry, HVAC, etc.)

Analysis Interactions

What might result if interactions are ignored in system design?

Analysis Interactions

What might result if interactions are ignored in system design?

- Missed opportunity to improve performance (not system optimal)
- Incompatibility between subsystems (inconsistent system)

Analysis Interactions

- What are some interactions between your subsystems?
- Specific effects of ignoring interactions?

System Consistency

A system has coupling variable consistency if for every coupling variable,

$$\mathbf{y}_{ij} - \mathbf{a}_j(\mathbf{x}_j, \mathbf{y}_j) = \mathbf{0}$$

is satisfied.

System Consistency

A system has coupling variable consistency if for every coupling variable,

$$\mathbf{y}_{ij} - \mathbf{a}_j(\mathbf{x}_j, \mathbf{y}_j) = \mathbf{0}$$

is satisfied.

System Analysis: task of solving system analysis equations for $\mathbf{y}_p$, given $\mathbf{x}$.

System Optimality

Subsystem Optimality $\neq$ System Optimality

From Subsystem to System Formulation

Subsystem i optimization formulation:

$$\begin{aligned} \min_{\mathbf{x}_i} \quad & f_i(\mathbf{x}_i, \mathbf{y}_p) \\ \text{subject to} \quad & \mathbf{g}_i(\mathbf{x}_i, \mathbf{y}_p) \leq \mathbf{0} \\ & \mathbf{h}_i(\mathbf{x}_i, \mathbf{y}_p) = \mathbf{0} \end{aligned}$$

System optimization formulation:

$$\begin{aligned} \min_{\mathbf{x}} \quad & f(\mathbf{x}, \mathbf{y}_p) \\ \text{subject to} \quad & \mathbf{g}(\mathbf{x}) = [\mathbf{g}_1, \mathbf{g}_2, \ldots, \mathbf{g}_N] \leq \mathbf{0} \\ & \mathbf{h}(\mathbf{x}) = [\mathbf{h}_1, \mathbf{h}_2, \ldots, \mathbf{h}_N] = \mathbf{0} \end{aligned}$$

where:

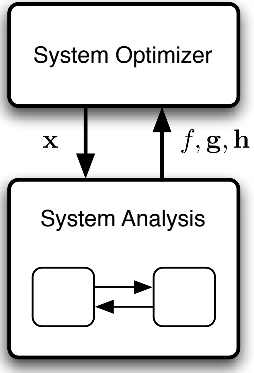
$$f = \begin{cases} f_k & \text{select one subsystem objective} \\ \sum_{i=1}^{N} w_i f_i(\mathbf{x}, \mathbf{y}_p) & \text{weighted sum of subsystem objectives} \\ f(\mathbf{x}, \mathbf{y}_p) & \text{define new function} \end{cases}$$

All-in-One (AiO) Design Approach

- Nest system analysis within system optimization
- Can identify system optimum

but:

- Can be very computationally expensive
- Requires a complete system analysis for every design iteration
- May not converge



Example Problem—Single Element Aeroelasticity

- **Aeroelasticity:** Requires both aerodynamic and structural analysis
- **Application:** Air-flow sensor design

Air-flow Sensor Analysis

▷**Structural Analysis:**

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

Given a design (ℓ, w) and a drag force F, solve for the corresponding deflection θ.

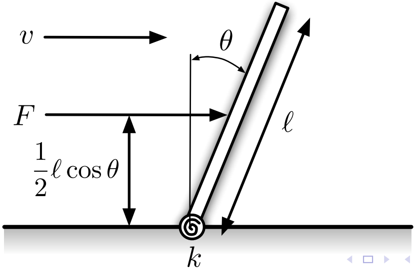
▷**Aerodynamic Analysis:**

$$F = CA_f v^2 = C\ell w \cos\theta v^2$$

Given a design (ℓ, w) and a deflection θ, find the drag force F.

▷**System Analysis:**

Given a design (ℓ, w), find the equilibrium values F and θ.



Air-flow Sensor Design

(Sensor Calibration Problem)

AiO Formulation:

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

Design Parameters: k, A, F_{max}, C, v

Solution by Monotonicity Analysis

- For cases where meeting the deflection target requires a drag force greater than F_{max}, the target cannot be met, and the inequality constraint will be active.

Solution by Monotonicity Analysis

- For cases where meeting the deflection target requires a drag force greater than F_{max}, the target cannot be met, and the inequality constraint will be active.

$$F_{max} = C\ell w \cos\theta v^2 \quad \Rightarrow \theta = \cos^{-1}\left(\frac{F_{max}}{CAv^2}\right)$$

$$k\theta - \frac{1}{2}F_{max}\ell\cos\theta = 0 \quad \Rightarrow \ell^* = \frac{2k\cos^{-1}\left(\frac{F_{max}}{CAv^2}\right)CAv^2}{F_{max}^2}$$

Solution by Monotonicity Analysis

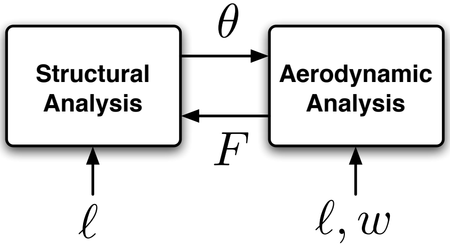
- For cases where meeting the deflection target requires a drag force greater than F_{max}, the target cannot be met, and the inequality constraint will be active.

$$F_{max} = C\ell w \cos\theta v^2 \quad \Rightarrow \theta = \cos^{-1}\left(\frac{F_{max}}{CAv^2}\right)$$

$$k\theta - \frac{1}{2}F_{max}\ell\cos\theta = 0 \quad \Rightarrow \ell^* = \frac{2k\cos^{-1}\left(\frac{F_{max}}{CAv^2}\right)CAv^2}{F_{max}^2}$$

- F, θ and ℓ^* known $\Rightarrow$ solve for w^* using $w^* = A/\ell^*$

Shared Quantities and System Analysis



Shared variable: ℓ
Coupling variables: θ and F
System analysis requires finding θ and F such that:

$$
\begin{aligned}
\theta &= \theta(F) \\
F &= F(\theta)
\end{aligned}
$$

System Analysis Options

- Solve the system analysis equations at each optimization iteration (AiO)
- Could we use the optimization algorithm to solve these equations?

System Analysis Options

- Solve the system analysis equations at each optimization iteration (AiO)
- Could we use the optimization algorithm to solve these equations?
- Yes!
  - Make the system analysis equations constraints
  - Let the optimizer choose values for the coupling variables, in addition to design variables

Why Use the Optimizer for Analysis?

- No longer have to completely solve system analysis when you are far from the system optimum
- Breaks feedback loops (eliminating nested iterations)
- Enables coarse-grained parallel computation
- AiO can overlook global optimum in some cases, or even fail

Air Flow Sensor Reformulation

AiO Formulation:

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

System analysis is implicitly required in the calculation of F and θ.

New Formulation:

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

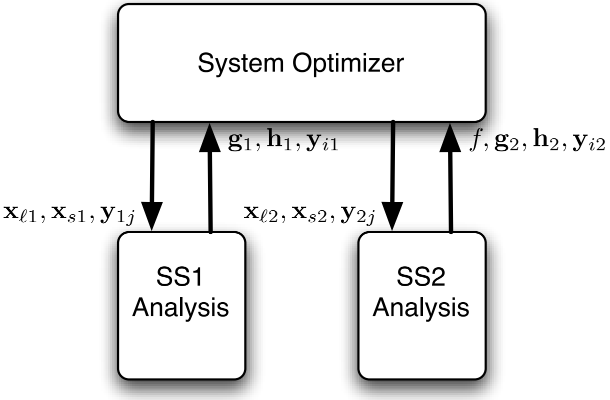
Individual Disciplinary Feasible (IDF) Method

IDF: simplest formal approach to combining analysis and optimization tasks.

IDF General Formulation:

$$
\begin{aligned}
\min_{\mathbf{x},\mathbf{y}} \quad & f(\mathbf{x}, \mathbf{y}) \\
\text{subject to} \quad & \mathbf{g}(\mathbf{x}, \mathbf{y}) = [\mathbf{g_1}, \mathbf{g_2}, \ldots, \mathbf{g_N}] \leq \mathbf{0} \\
& \mathbf{h}(\mathbf{x}, \mathbf{y}) = [\mathbf{h_1}, \mathbf{h_2}, \ldots, \mathbf{h_N}] = \mathbf{0} \\
& \mathbf{h_{aux}}(\mathbf{x}, \mathbf{y}) = \mathbf{Sa}(\mathbf{x}, \mathbf{y}) - \mathbf{y} = \mathbf{0}
\end{aligned}
$$

IDF Architecture



IDF Formulation Example: Electric Water Pump

$$
\begin{aligned}
\min_{\mathbf{x}} \quad & P_e = VI \\
\text{subject to} \quad & P \geq P_{min} = 100 \text{ kPa} \\
& T \leq T_{\max} = 428 \text{ K} \\
& L + \ell_c \leq 0.2 \text{ m} \\
& Q = 1.55 \cdot 10^{-3} \text{ m}^3/\text{sec}
\end{aligned}
$$

Analysis Functions

$T = a_1(I, \omega, d, d_2, d_3, L, \ell_c)$	Mot. winding temp. (K)
$I = a_2(\tau, T, d, d_2, d_3, L)$	Motor current (amps)
$\omega = a_3(I, T, d, d_2, d_3, L, \ell_c)$	Motor speed (rad/sec)
$\tau = a_4(\omega, D_2, b, \beta_1, \beta_2, \beta_3)$	Pump drive torque (Nm)
$P = a_5(\omega, D_2, b, \beta_1, \beta_2, \beta_3)$	Pressure differential (kPa)

Other Options

IDF seems helpful for many problems, but what if I have more design variables than my optimization algorithm can handle?

Other Options

IDF seems helpful for many problems, but what if I have more design variables than my optimization algorithm can handle?

Multilevel Methods:

- Use multiple optimization algorithms to share the load
- Distributed decision making reduces individual problem dimension

Multi-level Methods

- Optimization algorithm coupled with every element of the system
    - Local optimizers make local decisions (distributed decision making)
    - Can utilized specialized optimization algorithms
- Best for sparse problem structures
    - Many local decisions required, but relatively few subsystem connections
    - Possible to reduce individual problem dimension w/ multilevel methods

Analytical Target Cascading (ATC)

- Multi-level system design method
- Developed based on needs in the automotive industry
- Intended for hierarchical problems with object-based decomposition

(covered in detail in a later lecture)

Individual Disciplinary Feasible (IDF) Method