Introduction to Geometric Programming Based Aircraft Design

Warren Hoburg

University of California, Berkeley
Electrical Engineering and Computer Science Department

8th Research Consortium for Multidisciplinary System Design Workshop
July 16, 2013
Specifications

Baseline design

Evaluate objective and constraints

Change design

Is the design optimal?

No

Yes

Optimal Design

[Adapted from Alonso 2012]
Specifications

Baseline design

Evaluate objective and constraints

Change design

Is the design optimal?

No

Yes

Optimal Design

Challenge #1: Coupled Models

- Significant disciplinary separation
- Expensive function evaluations

[Adapted from Alonso 2012]
Specifications

Baseline design

Evaluate objective and constraints

Is the design optimal?

Optimal Design

Change design

Challenge #1: Coupled Models

Challenge #2: Competing Objectives

minimize $w_1 \dot{m}_{\text{fuel}} + \frac{w_2}{V_{\text{max}}} + \frac{w_3}{m_{\text{pay}}}$

[Adapted from Alonso 2012]
Challenge #1: Coupled Models

Challenge #2: Competing Objectives

minimize $w_1 \dot{m}_{fuel} + \frac{w_2}{V_{max}} + \frac{w_3}{m_{pay}}$

[Adapted from Alonso 2012]
Specifications

Baseline design

Evaluate objective and constraints

Change design

Is the design optimal?

Change specifications

Optimal Design

[Adapted from Alonso 2012]

Challenge #1: Coupled Models

Challenge #2: Competing Objectives

- Want Pareto frontier
  \[ \text{must solve many times} \]
Specifications

Baseline design

Evaluate objective and constraints

Is the design optimal?

Change design

Optimal Design

Change specifications

Challenge #1: Coupled Models

Challenge #2: Competing Objectives

- Want Pareto frontier
  → must solve many times

Challenge #3: Solution Quality

- Local vs. global optima
Specifications

Baseline design

Evaluate objective and constraints

Is the design optimal?

Optimal Design

Change design

Change specifications

Challenge #1: Coupled Models

Challenge #2: Competing Objectives

- Want Pareto frontier
  → must solve many times

Challenge #3: Solution Quality

- Local vs. global optima
- Sensitivity to initial guess

[Adapted from Alonso 2012]
Specifications

Baseline design

Evaluate objective and constraints

Is the design optimal?

Optimal Design

Change design

Change specifications

Surprisingly, many relationships in engineering design have an underlying convex structure.

Benefits

- Globally optimal solutions
- Robust algorithms - no initial guesses; no parameters to tune
- Extremely fast solutions, even for large problems

[Adapted from Alonso 2012]
Insight

- Surprisingly, many relationships in engineering design have an underlying convex structure.
Insight

- Surprisingly, many relationships in engineering design have an underlying convex structure.
Insight

- Surprisingly, many relationships in engineering design have an underlying **convex structure**.

Benefits

- *Globally* optimal solutions

[Adapted from Alonso 2012]
Insight

- Surprisingly, many relationships in engineering design have an underlying convex structure.

Benefits

- *Globally* optimal solutions
- *Robust* algorithms - no initial guesses; no parameters to tune
Insight

- Surprisingly, many relationships in engineering design have an underlying convex structure.

Benefits

- *Globally* optimal solutions
- *Robust* algorithms - no initial guesses; no parameters to tune
- Extremely fast solutions, even for large problems

[Adapted from Alonso 2012]
Today’s Talk

Approach Overview

The Power of Lagrange Duality

GP-compatible Modeling for Aircraft Design
Today’s Talk

Approach Overview

The Power of Lagrange Duality

GP-compatible Modeling for Aircraft Design
Optimization Compromizes

least squares vs. simulated annealing
Optimization Compromizes

least squares vs. simulated annealing

- fast
- reliable
- extremely specific
- still, widely used
Optimization Compromises

least squares vs. simulated annealing

- fast
- reliable
- extremely specific
- still, widely used

- Extremely general
- Slow convergence
- Requires hand-holding
Optimization Compromizes

least squares                 simulated annealing
Optimization Compromizes

convex programs

least squares  LP  GP  SDP

simulated annealing

Extremely general

Slow convergence

Requires hand-holding

fast

reliable

extremely specific

still, widely used
Optimization Compromizes

convex programs

least squares | LP | GP | SDP | SQP, NLP | simulated annealing
General Nonlinear Program

Decision Variables  $x \in \mathbb{R}^{n_V}$

Objective Function  $f_0(x) : \mathbb{R}^{n_V} \rightarrow \mathbb{R}$

Constraints  $0 \geq f_i(x) : \mathbb{R}^{n_V} \rightarrow \mathbb{R}$

$0 = h_i(x) : \mathbb{R}^{n_V} \rightarrow \mathbb{R}$
General Nonlinear Program

Decision Variables $x \in \mathbb{R}^{n_{\nu}}$

Objective Function $f_0(x): \mathbb{R}^{n_{\nu}} \rightarrow \mathbb{R}$

Constraints $0 \geq f_i(x): \mathbb{R}^{n_{\nu}} \rightarrow \mathbb{R}$
$0 = h_i(x): \mathbb{R}^{n_{\nu}} \rightarrow \mathbb{R}$

- In general, extremely difficult to solve
General Nonlinear Program

Decision Variables $x \in \mathbb{R}^{n_v}$

Objective Function $f_0(x) : \mathbb{R}^{n_v} \rightarrow \mathbb{R}$

Constraints $0 \geq f_i(x) : \mathbb{R}^{n_v} \rightarrow \mathbb{R}$

$0 = h_i(x) : \mathbb{R}^{n_v} \rightarrow \mathbb{R}$

- In general, extremely difficult to solve

Convex Program

Same as nonlinear program, except

- $f_i(x)$ must be convex
- $h_i(x)$ must be affine
General Nonlinear Program

Decision Variables \( x \in \mathbb{R}^{n_v} \)

Objective Function \( f_0(x) : \mathbb{R}^{n_v} \rightarrow \mathbb{R} \)

Constraints \( 0 \geq f_i(x) : \mathbb{R}^{n_v} \rightarrow \mathbb{R} \)
\( 0 = h_i(x) : \mathbb{R}^{n_v} \rightarrow \mathbb{R} \)

- In general, extremely difficult to solve

Convex Program

Same as nonlinear program, except
- \( f_i(x) \) must be convex
- \( h_i(x) \) must be affine

- Very efficient to solve
Geometric Program: Definition

Monomial Function

\[ m(x) = c^n \prod_{i=1}^{n} x^{a_i}, \quad c > 0 \] (e.g., \( 1^2 \rho V^2 C_L S \))

Posynomial Function: sum of monomials

\[ p(x) = K \sum_{k=1}^{K} c_k^n \prod_{i=1}^{n} x^{a_{ik}}, \quad c_k > 0 \] (e.g., \( C_D 0 + C_2 L \pi e A \))

Geometric Program (GP)

\[
\begin{align*}
\text{minimize} & \quad p_0(x) \\
\text{subject to} & \quad p_i(x) \leq 1, \quad i = 1, \ldots, N_p, \\
& \quad m_j(x) = 1, \quad j = 1, \ldots, N_m,
\end{align*}
\]

with \( p_i \) posynomial, \( m_i \) monomial

\[ x = (x_1, x_2, \ldots, x_n) > 0 \]
Geometric Program: Definition

Monomial Function

\[ m(x) = c \prod_{i=1}^{n} x_i^{a_i}, \quad c > 0 \quad \text{ (e.g., } \frac{1}{2} \rho V^2 C_L S) \]
Geometric Program: Definition

Monomial Function

\[ m(x) = c \prod_{i=1}^{n} x_i^{a_i}, \quad c > 0 \quad \text{(e.g.,} \frac{1}{2} \rho V^2 C_L S) \]

Posynomial Function: sum of monomials

\[ p(x) = \sum_{k=1}^{K} c_k \prod_{i=1}^{n} x_i^{a_{ik}}, \quad c_k > 0 \quad \text{(e.g.,} C_D_0 + \frac{C_i^2}{\pi e A}) \]
Geometric Program: Definition

Monomial Function

\[ m(x) = c \prod_{i=1}^{n} x_i^{a_i}, \quad c > 0 \quad \text{(e.g., } \frac{1}{2} \rho V^2 C_L S) \]

Posynomial Function: sum of monomials

\[ p(x) = \sum_{k=1}^{K} c_k \prod_{i=1}^{n} x_i^{a_{ik}}, \quad c_k > 0 \quad \text{(e.g., } C_D_0 + \frac{c_i^2}{\pi e A}) \]

Geometric Program (GP)

\[
\begin{align*}
\text{minimize} & \quad p_0(x) \\
\text{subject to} & \quad p_i(x) \leq 1, \quad i = 1, \ldots, N_p; \\
& \quad m_j(x) = 1, \quad j = 1, \ldots, N_m
\end{align*}
\]

with \( p_i \) posynomial, \( m_j \) monomial
\[ x = (x_1, x_2, \ldots, x_n) > 0 \]
variable change: \( y_i := \log x_i \)
Geometric Program: Convex Formulation

variable change: \( y_i := \log x_i \)

- Monomials \( m(x) = c \prod_{i=1}^{n} x_i^{a_i} \): affine in \( y \) after log transform

\[
\log m = b + a^T y \quad (b = \log c)
\]
Geometric Program: Convex Formulation

variable change: \( y_i := \log x_i \)

- Monomials \( m(x) = c \prod_{i=1}^{n} x_i^{a_i} \): affine in \( y \) after log transform
  \[
  \log m = b + a^T y \quad (b = \log c)
  \]

- Posynomials \( \sum_{k=1}^{K} c_k \prod_{i=1}^{n} x_i^{a_{ik}} \): convex in \( y \) after log transform
  \[
  \log p = \log \left( \sum_{k=1}^{K} e^{b_k + a_k^T y} \right)
  \]
variable change: $y_i := \log x_i$

- **Monomials** $m(x) = c \prod_{i=1}^{n} x_i^{a_i}$: affine in $y$ after log transform
  \[
  \log m = b + a^T y \quad (b = \log c)
  \]

- **Posynomials** $\sum_{k=1}^{K} c_k \prod_{i=1}^{n} x_i^{a_{ik}}$: convex in $y$ after log transform
  \[
  \log p = \log \left( \sum_{k=1}^{K} e^{b_k + a_k^T y} \right)
  \]

- **GP in convex form**
  \[
  \begin{align*}
  \text{minimize} & \quad \log \left( \sum_{k=1}^{K_0} \exp(b_{0k} + a_{0k}^T y) \right) \\
  \text{subject to} & \quad \log \left( \sum_{k=1}^{K_i} \exp(b_{ik} + a_{ik}^T y) \right) \leq 0, \quad i = 1, \ldots, N_p \\
  & \quad Gy + h = 0
  \end{align*}
  \]
Geometric Program: Convex Formulation

- \( a < 0 \)
- \( a = 0 \)
- \( 0 < a < 1 \)
- \( a = 1 \)
- \( a > 1 \)

Monomials

- \( 0.01x^{-2} + 0.2x^{0.2} + 0.00006x^4 \)
- \( 0.03x^{-1} + 0.8x^{0.2} + 0.2x^{1.5} \)

Posynomials

- \( 0.01x^{-2} + 0.2x^{0.2} + 0.00006x^4 \)
- \( 0.03x^{-1} + 0.8x^{0.2} + 0.2x^{1.5} \)

Posynomials (log-space)

- \( 0.01x^{-2} + 0.2x^{0.2} + 0.00006x^4 \)
- \( 0.03x^{-1} + 0.8x^{0.2} + 0.2x^{1.5} \)
Geometric Program: Convex Formulation

monomials

monomials (log-space)
Geometric Program: Convex Formulation

- **Monomials**:
  - $a < 0$
  - $a = 1$
  - $0 < a < 1$
  - $a = 0$
  - $a > 1$

- **Posynomials**:
  - $2x^2$
  - $0.01x^2 + 0.00006x$
  - $0.03x^{-1} + 0.8x^{0.2}$

- **Monomials (log-space)**:
  - $a < 0$
  - $a = 1$
  - $0 < a < 1$
  - $a = 0$
  - $a > 1$
Geometric Program: Convex Formulation

- **Monomials**
  - $a < 0$
  - $0 < a < 1$
  - $a = 1$
  - $a = 0$
  - $a > 1$

- **Posynomials**
  - $a < 0$
  - $0 < a < 1$
  - $a = 1$
  - $a = 0$
  - $a > 1$

- **Monomials (log-space)**
  - $a < 0$
  - $0 < a < 1$
  - $a = 1$
  - $a = 0$
  - $a > 1$

- **Posynomials (log-space)**
  - $a < 0$
  - $0 < a < 1$
  - $a = 1$
  - $a = 0$
  - $a > 1$
Solution of Geometric Programs

*Interior-point* methods

![Graph showing solution of geometric programs](image)

**Benefits:**
- **Globally** optimal solution, guaranteed
- Robust: no initial guesses or parameter tuning
- Off-the-shelf solvers

Figures: [Boyd 2004]
Solution of Geometric Programs

Interior-point methods

Benefits:
- **Globally** optimal solution, guaranteed
- Robust: no initial guesses or parameter tuning
- Off-the-shelf solvers

Boyd GP benchmarks (2005) [1]
- dense GP: 1,000 variables; 10,000 constraints: less than 1 minute
- sparse GP: 10,000 variables; 1,000,000 constraints: “minutes”

Figures: [Boyd 2004]
Running Example

minimize \[ A, S, C_D, C_L, W, W_w, V \]
subject to

CD breakdown
1 \geq \frac{(CDA_0)}{C_D S} + \frac{C_{Dp}}{2W} + \frac{C_L^2}{C_D \pi A e}

CL definition
1 \geq \frac{\rho V^2 C_L S}{W_0} + \frac{W_w}{W}

weight breakdown
1 \geq \frac{45.42 S}{W_w} + 8.71 \times 10^{-5} \frac{N_{lift}}{W_0 WS} \frac{A^{3/2} \sqrt{W_0 W S}}{W_w T}

wing weight
1 \geq \frac{2W}{\rho V^2 \min S C_{L, \max}}

stall speed
1 \geq \frac{\rho V^2}{\min S C_{L, \max}}
Running Example

\[
\begin{align*}
\text{minimize} \quad & \frac{1}{2} \rho V^2 C_D S \\
\text{subject to} \quad & 1 \geq \frac{(CDA_0)}{C_D S} + \frac{C_{D_D}}{C_D} + \frac{C_L^2}{C_D \pi A e} \\
\text{CD breakdown} \quad & 1 \geq \frac{\rho V^2 C_L S}{W_0 W_w} \\
\text{CL definition} \quad & 1 \geq \frac{45.42 S}{W_w} + 8.71 \times 10^{-5} \frac{N_{lift} A^{3/2} \sqrt{W_0 W_S}}{W_w^2} \\
\text{weight breakdown} \quad & 1 \geq \frac{W}{W} + \frac{W_w}{W} \\
\text{wing weight} \quad & 1 \geq \frac{45.42 S}{W_w} + 8.71 \times 10^{-5} \frac{N_{lift} A^{3/2} \sqrt{W_0 W_S}}{W_w^2} \\
\text{stall speed} \quad & 1 \geq \frac{2W}{\rho V_{\text{min}}^2 S C_{L,max}}
\end{align*}
\]

\text{CONSTANTS} \quad \\
CDA_0 \quad 0.031 \\
CDp \quad 0.0095 \\
CLmax \quad 1.5 \\
Nult \quad 3.8 \\
Vmin \quad 22 \\
W0 \quad 4940 \\
e \quad 0.95 \\
rho \quad 1.23 \\
tau \quad 0.12
GP Parameterization

<table>
<thead>
<tr>
<th>map</th>
<th>$CDA_0$</th>
<th>$\rho$</th>
<th>$C_Dp$</th>
<th>$e$</th>
<th>$W_0$</th>
<th>$N_{lift}$</th>
<th>$\tau$</th>
<th>$V_{min}$</th>
<th>$C_{L_{max}}$</th>
<th>$A$</th>
<th>$S$</th>
<th>$C_D$</th>
<th>$C_L$</th>
<th>$W$</th>
<th>$W_w$</th>
<th>$V$</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>$1/2$</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>1</td>
<td>1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-2</td>
<td></td>
</tr>
<tr>
<td>1</td>
<td>1</td>
<td>1</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td>-2</td>
</tr>
<tr>
<td>1</td>
<td>$1/\pi$</td>
<td>1</td>
<td></td>
<td>-1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-1</td>
<td>-1</td>
<td>2</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>2</td>
<td>-1</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>1</td>
<td></td>
<td></td>
<td>-1</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
<td>-1</td>
<td>-1</td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>45.42</td>
<td></td>
<td></td>
<td>1/2</td>
<td>1</td>
<td>-1</td>
<td></td>
<td></td>
<td></td>
<td>3/2</td>
<td>1/2</td>
<td>1/2</td>
<td>-1</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4</td>
<td>$8.7e-5$</td>
<td></td>
<td></td>
<td>-1</td>
<td></td>
<td>-2</td>
<td>-1</td>
<td>-1</td>
<td></td>
<td>-1</td>
<td>1</td>
<td>-1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>2</td>
<td></td>
<td></td>
<td>-1</td>
<td></td>
<td>-2</td>
<td>-1</td>
<td>-1</td>
<td></td>
<td>-1</td>
<td></td>
<td>1</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Globally Optimal Solution

A  8.792
S  16.79
CD 0.02269
CL 0.5456
W  7495
Ww 2555
V  36.48

OBJECTIVE VALUE: 311.7159
# Globally Optimal Solution

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>8.792</td>
</tr>
<tr>
<td>S</td>
<td>16.79</td>
</tr>
<tr>
<td>CD</td>
<td>0.02269</td>
</tr>
<tr>
<td>CL</td>
<td>0.5456</td>
</tr>
<tr>
<td>W</td>
<td>7495</td>
</tr>
<tr>
<td>Ww</td>
<td>2555</td>
</tr>
<tr>
<td>V</td>
<td>36.48</td>
</tr>
</tbody>
</table>

**Solution is:**

- Feasible (satisfies all constraints)

**OBJECTIVE VALUE:** 311.7159
Globally Optimal Solution

Solution is:
- *Feasible* (satisfies all constraints)
- *Globally optimal* (no other feasible solution has better objective value)

<table>
<thead>
<tr>
<th>Variable</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>8.792</td>
</tr>
<tr>
<td>S</td>
<td>16.79</td>
</tr>
<tr>
<td>CD</td>
<td>0.02269</td>
</tr>
<tr>
<td>CL</td>
<td>0.5456</td>
</tr>
<tr>
<td>W</td>
<td>7495</td>
</tr>
<tr>
<td>Ww</td>
<td>2555</td>
</tr>
<tr>
<td>V</td>
<td>36.48</td>
</tr>
</tbody>
</table>

OBJECTIVE VALUE: 311.7159
Today’s Talk

Approach Overview

The Power of Lagrange Duality

GP-compatible Modeling for Aircraft Design
Lagrange Dual of GP

Primal problem (in convex form):

\[
\text{minimize} \quad \log \sum_{k=1}^{K_0} \exp(a_{0k}^T y + b_{0k}) \\
\text{subject to} \quad \log \sum_{k=1}^{K_i} \exp(a_{ik}^T y + b_{ik}) \leq 0, \quad i = 1, ... , m, \tag{1}
\]
Lagrange Dual of GP

Primal problem (in convex form):

\[
\begin{align*}
\text{minimize} & \quad \log \sum_{k=1}^{K_0} \exp(a_{0k}^T y + b_{0k}) \\
\text{subject to} & \quad \log \sum_{k=1}^{K_i} \exp(a_{ik}^T y + b_{ik}) \leq 0, \quad i = 1, \ldots, m, \quad (1)
\end{align*}
\]

Lagrangian and dual function:

\[
\begin{align*}
L(y, z, \lambda, \nu) &= \log \sum_{k=1}^{K_0} \exp z_{0k} + \sum_{i=1}^{m} \lambda_i \log \sum_{k=1}^{K_i} \exp z_{ik} + \sum_{i=0}^{m} \nu_i^T (A_i y + b_i - z_i) \\
g(\lambda, \nu) &= \inf_{y, z} L(y, z, \lambda, \nu).
\end{align*}
\]
Lagrange Dual of GP

maximize \[ \sum_{i=0}^{m} \left[ \nu_i^T b_i - \sum_{k=1}^{K_i} \nu_{ik} \log \frac{\nu_{ik}}{1^T \nu_i} \right] \]

subject to \[ \sum_{i=0}^{m} \nu_i^T A_i = 0 \]

\[ \nu_i \geq 0, \quad i = 0, ..., m \]

\[ 1^T \nu_0 = 1. \]
Lagrange Dual of GP

 maximize $\sum_{i=0}^{m} \left[ \nu_i^T b_i - \sum_{k=1}^{K_i} \nu_{ik} \log \frac{\nu_{ik}}{1^T \nu_i} \right]$

 subject to $\sum_{i=0}^{m} \nu_i^T A_i = 0$

 $\nu_i \geq 0, \quad i = 0, \ldots, m$

 $1^T \nu_0 = 1$.

- An equality-constrained entropy maximization
Lagrange Dual of GP

maximize \[ \sum_{i=0}^{m} \left[ \nu_i^T b_i - \sum_{k=1}^{K_i} \nu_{ik} \log \frac{\nu_{ik}}{1^T \nu_i} \right] \]

subject to \[ \sum_{i=0}^{m} \nu_i^T A_i = 0 \]

\[ \nu_i \geq 0, \quad i = 0, \ldots, m \]

\[ 1^T \nu_0 = 1. \]

▶ An equality-constrained entropy maximization

▶ (unnormalized) probability distributions \( \nu_i \) satisfy \( 1^T \nu_i = \lambda_i \)
Constraint Sensitivities

Consider perturbed GP:

\[
\begin{align*}
\text{minimize} & \quad \sum_{k=1}^{K_0} c_{0k}x^{a_{0k}} \\
\text{subject to} & \quad \sum_{k=1}^{K_i} c_{ik}x^{a_{ik}} \leq u_i, \quad i = 1, \ldots, m.
\end{align*}
\]

Define \( p^* (u) \equiv \text{optimal objective value of perturbed GP} \)

Extremely useful fact:

\[
\left. \frac{\partial \log p^*(u)}{\partial \log u_i} \right|_{u=1} = \left. \frac{\partial (p^*(u) - p^*(1))}{\partial u_i} \right|_{u=1} = -\lambda_i
\]

Best of all, modern solvers determine \( \lambda_i \)’s for free
Constraint Sensitivities

- Consider perturbed GP:

\[
\begin{align*}
\text{minimize} & \quad \sum_{k=1}^{K_0} c_{0k} x^{a_{0k}} \\
\text{subject to} & \quad \sum_{k=1}^{K_i} c_{ik} x^{a_{ik}} \leq u_i, \quad i = 1, \ldots, m.
\end{align*}
\]

- Define \( p^*(u) \equiv \) optimal objective value of perturbed GP
Constraint Sensitivities

- Consider perturbed GP:

\[
\begin{align*}
\text{minimize} & \quad \sum_{k=1}^{K_0} c_{0k} x^{a_{0k}} \\
\text{subject to} & \quad \sum_{k=1}^{K_i} c_{ik} x^{a_{ik}} \leq u_i, \quad i = 1, \ldots, m.
\end{align*}
\]

- Define \( p^*(u) \equiv \text{optimal objective value of perturbed GP} \)

- Extremely useful fact:

\[
\left. \frac{\partial \log p^*(u)}{\partial \log u_i} \right|_{u=1} = \left. \frac{\partial \left( \frac{p^*(u)}{p^*(1)} \right)}{\partial \left( \frac{u_i}{1} \right)} \right|_{u=1} = -\lambda_i
\]
Constraint Sensitivities

- Consider perturbed GP:

\[
\begin{align*}
\text{minimize} & \quad \sum_{k=1}^{K_0} c_{0k} x^{a_{0k}} \\
\text{subject to} & \quad \sum_{k=1}^{K_i} c_{ik} x^{a_{ik}} \leq u_i, \ i = 1, \ldots, m.
\end{align*}
\]

- Define \( p^*(u) \equiv \) optimal objective value of perturbed GP

- Extremely useful fact:

\[
\frac{\partial \log p^*(u)}{\partial \log u_i} \bigg|_{u=1} = \frac{\partial \left( \frac{p^*(u)}{p^*(1)} \right)}{\partial \left( \frac{u_i}{1} \right)} \bigg|_{u=1} = -\lambda_i
\]

- Best of all, modern solvers determine \( \lambda_i \)'s for free
Running Example – Constraint Sensitivities

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>8.792</td>
</tr>
<tr>
<td>S</td>
<td>16.79</td>
</tr>
<tr>
<td>CD</td>
<td>0.02269</td>
</tr>
<tr>
<td>CL</td>
<td>0.5456</td>
</tr>
<tr>
<td>W</td>
<td>7495</td>
</tr>
<tr>
<td>Ww</td>
<td>2555</td>
</tr>
<tr>
<td>V</td>
<td>36.48</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Breakdown</th>
<th>Percentage</th>
</tr>
</thead>
<tbody>
<tr>
<td>weight breakdown</td>
<td>139.37%</td>
</tr>
<tr>
<td>CD breakdown</td>
<td>100.00%</td>
</tr>
<tr>
<td>CL definition</td>
<td>100.00%</td>
</tr>
<tr>
<td>induced drag model</td>
<td>50.00%</td>
</tr>
<tr>
<td>wing weight model</td>
<td>47.51%</td>
</tr>
<tr>
<td>landing stall speed</td>
<td>22.71%</td>
</tr>
<tr>
<td>fuselage drag model</td>
<td>8.14%</td>
</tr>
</tbody>
</table>
### Running Example – Constraint Sensitivities

<table>
<thead>
<tr>
<th>OBJECTIVE</th>
<th>D</th>
<th>DV</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>8.792</td>
<td>8.572</td>
</tr>
<tr>
<td>S</td>
<td>16.79</td>
<td>30.55</td>
</tr>
<tr>
<td>CD</td>
<td>0.02269</td>
<td>0.04206</td>
</tr>
<tr>
<td>CL</td>
<td>0.5456</td>
<td>0.8983</td>
</tr>
<tr>
<td>W</td>
<td>7495</td>
<td>8859</td>
</tr>
<tr>
<td>Ww</td>
<td>2555</td>
<td>3919</td>
</tr>
<tr>
<td>V</td>
<td>36.48</td>
<td>22.91</td>
</tr>
</tbody>
</table>

- **weight breakdown**: 139.37% / 175.00%
- **CD breakdown**: 100.00% / 100.00%
- **CL definition**: 100.00% / 150.00%
- **induced drag model**: 50.00% / 75.00%
- **wing weight model**: 47.51% / 77.41%
- **landing stall speed**: 22.71% / 0.00%
- **fuselage drag model**: 8.14% / 2.41%
## Running Example – Constraint Sensitivities

<table>
<thead>
<tr>
<th>OBJECTIVE --&gt;</th>
<th>D</th>
<th>DV</th>
<th>D/V</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>8.792</td>
<td>8.572</td>
<td>6.307</td>
</tr>
<tr>
<td>S</td>
<td>16.79</td>
<td>30.55</td>
<td>14.69</td>
</tr>
<tr>
<td>CD</td>
<td>0.02269</td>
<td>0.04206</td>
<td>0.01548</td>
</tr>
<tr>
<td>CL</td>
<td>0.5456</td>
<td>0.8983</td>
<td>0.2699</td>
</tr>
<tr>
<td>W</td>
<td>7495</td>
<td>8859</td>
<td>6561</td>
</tr>
<tr>
<td>Ww</td>
<td>2555</td>
<td>3919</td>
<td>1621</td>
</tr>
<tr>
<td>V</td>
<td>36.48</td>
<td>22.91</td>
<td>51.87</td>
</tr>
</tbody>
</table>

weight breakdown 139.37% 175.00%  114.71%
CD breakdown 100.00% 100.00%  100.00%
CL definition 100.00% 150.00%  50.00%
induced drag model 50.00% 75.00%  25.00%
wing weight model 47.51% 77.41%  28.34%
landing stall speed 22.71% 0.00%  56.37%
fuselage drag model 8.14% 2.41%  13.63%
Constant Sensitivities

GP with fixed variables \( \bar{x} \)

minimize \[ \sum_{k=1}^{K_0} c_{0k} \bar{x}^{\bar{a}_{0k}} x^{a_{0k}} \]

subject to \[ \sum_{k=1}^{K_i} c_{ik} \bar{x}^{\bar{a}_{ik}} x^{a_{ik}} \leq 1, \quad i = 1, \ldots, m, \quad (2) \]

Dual variables encode sensitivity of optimum to fixed variables:

\[ \frac{\partial \log p^*}{\partial \log \bar{x}_j} = \sum_{i=0}^{m} \nu_i^T \bar{a}^{(j)}_i \]
Constant Sensitivities

GP with fixed variables $\bar{x}$

$$\text{minimize} \quad \sum_{k=1}^{K_0} c_{0k} \bar{x}^{a_{0k}} x^{a_{0k}}$$

$$\text{subject to} \quad \sum_{k=1}^{K_i} c_{ik} \bar{x}^{a_{ik}} x^{a_{ik}} \leq 1, \quad i = 1, \ldots, m, \quad (2)$$

Dual variables encode sensitivity of optimum to fixed variables:

$$\frac{\partial \log p^*}{\partial \log \bar{x}_j} = \sum_{i=0}^{m} \nu_i^T \bar{a}_i$$

Running Example

<table>
<thead>
<tr>
<th>SNSTVTY</th>
<th>CONST</th>
<th>VALUE</th>
</tr>
</thead>
<tbody>
<tr>
<td>(-108.50%)</td>
<td>W0</td>
<td>4940</td>
</tr>
<tr>
<td>50.00%</td>
<td>e</td>
<td>0.95</td>
</tr>
<tr>
<td>45.41%</td>
<td>Vmin</td>
<td>22</td>
</tr>
<tr>
<td>(-41.86%)</td>
<td>CDp</td>
<td>0.0095</td>
</tr>
<tr>
<td>(-33.33%)</td>
<td>Nult</td>
<td>3.8</td>
</tr>
<tr>
<td>33.33%</td>
<td>tau</td>
<td>0.12</td>
</tr>
<tr>
<td>22.71%</td>
<td>CLmax</td>
<td>1.5</td>
</tr>
<tr>
<td>22.71%</td>
<td>rho</td>
<td>1.23</td>
</tr>
<tr>
<td>(-8.138%)</td>
<td>CDA0</td>
<td>0.031</td>
</tr>
</tbody>
</table>
Some Useful Bounds

Dual Sensitivity Analysis

- Start with feasible solution $p^*(u = 1)$
Some Useful Bounds

Dual Sensitivity Analysis

- Start with feasible solution $p^*(u = 1)$
- Perturb design constraints (via $u$)

Performance bound:

$$\log p^*(u) \geq \log p^*(1) + \lambda T$$

An optimistic estimate

Design Averaging

- Consider two designs $\theta_1, \theta_2$, with objective values $p^*_1, p^*_2$
- Form geometric mean design $\theta(i) = \sqrt{\theta_1(i) \theta_2(i)}$
- Performance bound:
  $$p^*_{\text{3}} \leq \sqrt{p^*_1 p^*_2}$$

A pessimistic estimate
Dual Sensitivity Analysis

- Start with feasible solution $p^*(u = 1)$
- Perturb design constraints (via $u$)
- Performance bound:

$$\log p^*(u) \geq \log p^*(1) + \lambda^T u$$

Some Useful Bounds
Some Useful Bounds

Dual Sensitivity Analysis

- Start with feasible solution $p^*(u = 1)$
- Perturb design constraints (via $u$)
- Performance bound:
  \[
  \log p^*(u) \geq \log p^*(1) + \lambda^T u
  \]
- An optimistic estimate
Some Useful Bounds

## Dual Sensitivity Analysis
- Start with feasible solution $p^*(u = 1)$
- Perturb design constraints (via $u$)
- Performance bound:
  \[
  \log p^*(u) \geq \log p^*(1) + \lambda^T u
  \]
- An optimistic estimate

## Design Averaging
- Consider two designs $\theta_1, \theta_2$, with objective values $p_1^*, p_2^*$
Some Useful Bounds

Dual Sensitivity Analysis

- Start with feasible solution \( p^*(u = 1) \)
- Perturb design constraints (via \( u \))
- Performance bound:
  \[
  \log p^*(u) \geq \log p^*(1) + \lambda^T u
  \]
- An optimistic estimate

Design Averaging

- Consider two designs \( \theta_1, \theta_2 \), with objective values \( p_1^*, p_2^* \)
- Form geometric mean design
  \[
  \theta_3^{(i)} = \sqrt{\theta_1^{(i)} \theta_2^{(i)}}
  \]
Some Useful Bounds

Dual Sensitivity Analysis

- Start with feasible solution $p^*(u = 1)$
- Perturb design constraints (via $u$)
- Performance bound:
  \[
  \log p^*(u) \geq \log p^*(1) + \lambda^T u
  \]
- An optimistic estimate

Design Averaging

- Consider two designs $\theta_1, \theta_2$, with objective values $p_1^*, p_2^*$
- Form geometric mean design
  \[
  \theta_3^{(i)} = \sqrt{\theta_1^{(i)} \theta_2^{(i)}}
  \]
- Performance bound:
  \[
  p_3^* \leq \sqrt{p_1^* p_2^*}
  \]
Some Useful Bounds

Dual Sensitivity Analysis
- Start with feasible solution $p^*(u = 1)$
- Perturb design constraints (via $u$)
- Performance bound:
  \[ \log p^*(u) \geq \log p^*(1) + \lambda^T u \]
- An optimistic estimate

Design Averaging
- Consider two designs $\theta_1, \theta_2$, with objective values $p_1^*, p_2^*$
- Form geometric mean design
  \[ \theta_3^{(i)} = \sqrt{\theta_1^{(i)} \theta_2^{(i)}} \]
- Performance bound:
  \[ p^*_3 \leq \sqrt{p_1^* p_2^*} \]
- A pessimistic estimate
Feasibility Analysis

When constraints cannot all be satisfied, GP solvers provide a mathematical certificate that no feasible point exists.
When constraints cannot all be satisfied, GP solvers provide a mathematical certificate that no feasible point exists.

In this case, look for closest feasible point:

### Original GP

<table>
<thead>
<tr>
<th>minimize</th>
<th>$p_0(x)$</th>
</tr>
</thead>
<tbody>
<tr>
<td>subject to</td>
<td>$p_i(x) \leq 1, \quad i = 1, ..., N_p,$</td>
</tr>
<tr>
<td></td>
<td>$m_j(x) = 1, \quad j = 1, ..., N_m$</td>
</tr>
</tbody>
</table>

### Closest Feasible Point GP

<table>
<thead>
<tr>
<th>minimize</th>
<th>$s$</th>
</tr>
</thead>
<tbody>
<tr>
<td>subject to</td>
<td>$p_i(x) \leq s, \quad i = 1, ..., N_p,$</td>
</tr>
<tr>
<td></td>
<td>$m_j(x) = 1, \quad j = 1, ..., N_m$</td>
</tr>
</tbody>
</table>

The closest feasible point GP is always feasible, and its optimal point is within $100(s - 1)$% of satisfying the original inequality constraints.
When constraints cannot all be satisfied, GP solvers provide a mathematical certificate that no feasible point exists.

In this case, look for closest feasible point:

**Original GP**

\[
\begin{align*}
\text{minimize} & \quad p_0(x) \\
\text{subject to} & \quad p_i(x) \leq 1, \quad i = 1, \ldots, N_p, \\
& \quad m_j(x) = 1, \quad j = 1, \ldots, N_m
\end{align*}
\]

**Closest Feasible Point GP**

\[
\begin{align*}
\text{minimize} & \quad s \\
\text{subject to} & \quad p_i(x) \leq s, \quad i = 1, \ldots, N_p, \\
& \quad m_j(x) = 1, \quad j = 1, \ldots, N_m
\end{align*}
\]

The closest feasible point GP is always feasible, and its optimal point is within 100(s − 1)% of satisfying the original inequality constraints.
Today’s Talk

Approach Overview

The Power of Lagrange Duality

GP-compatible Modeling for Aircraft Design
Conceptual Design – Modeling Summary

- Fuselage Pressure Loads
- Fuselage Bending Loads
- Fuselage Weight
- Steady Level Flight Relations
- Wing Moments and Stresses
- Wing Weight
- Stability
- Tail Moments and Stresses
- Tail Weight
- Engine Weight
- Turbine Cycle Analysis
- Noise
- CG Envelope
- Active Gust Response
- Wing Profile Drag
- V-speeds and critical loading cases
- Wing Induced Drag
- Tail Drag
- Fuselage Drag
- Interference Drags
- Airfoil Shape Optimization
- Laminar Flow Control
- Compressibility Effects
- Propulsive Efficiency
- Blade Element Momentum Theory
- APU Sizing
- Hydraulic, Fuel, & Electrical System Weights
- Mission Breakdown and Fuel Burn
- Cruise Climb
- Loiter Performance/Endurance
- Takeoff Distance & 50’ obstacle clearance
- Landing Distance
- Spoiler Sizing
- Climb Performance
- Engine-Out Operation
- Windmilling Drag
- Maneuverability
- High Lift System Sizing
- Control Surface Sizing
- Landing Gear Sizing
- Engine Ground Clearance
- Tail Strike Clearance
- Maintenance Costs
- Material Costs
- Manufacturability
- Assembly/Integration Time and Cost
- Fastener Count
- Supply Chain Dynamics
Conceptual Design – Modeling Summary

- Fuselage Pressure Loads
- Fuselage Bending Loads
- Fuselage Weight
- Steady Level Flight Relations
- Wing Moments and Stresses
- Wing Weight
- Stability
- Tail Moments and Stresses
- Tail Weight
- Engine Weight
- Turbine Cycle Analysis
- Noise
- CG Envelope
- Active Gust Response
- Wing Profile Drag
- V-speeds and critical loading cases
- Wing Induced Drag
- Tail Drag
- Fuselage Drag
- Interference Drags
- Airfoil Shape Optimization
- Laminar Flow Control
- Compressibility Effects
- Propulsive Efficiency
- Blade Element Momentum Theory
- APU Sizing
- Hydraulic, Fuel, & Electrical System Weights
- Mission Breakdown and Fuel Burn
- Cruise Climb
- Loiter Performance/Endurance
- Takeoff Distance & 50’ obstacle clearance
- Landing Distance
- Spoiler Sizing
- Climb Performance
- Engine-Out Operation
- Windmilling Drag
- Maneuverability
- High Lift System Sizing
- Control Surface Sizing
- Landing Gear Sizing
- Engine Ground Clearance
- Tail Strike Clearance
- Maintenance Costs
- Material Costs
- Manufacturability
- Assembly/Integration Time and Cost
- Fastener Count
- Supply Chain Dynamics
Conceptual Design – Modeling Summary

- Fuselage Pressure Loads
- Fuselage Bending Loads
- Fuselage Weight
- Steady Level Flight Relations
- Wing Moments and Stresses
- Wing Weight
- Stability
- Tail Moments and Stresses
- Tail Weight
- Engine Weight
- Turbine Cycle Analysis
- Noise
- CG Envelope
- Active Gust Response
- Wing Profile Drag
- V-speeds and critical loading cases
- Wing Induced Drag
- Tail Drag
- Fuselage Drag
- Interference Drags
- Airfoil Shape Optimization
- Laminar Flow Control
- Compressibility Effects
- Propulsive Efficiency
- Blade Element Momentum Theory
- APU Sizing
- Hydraulic, Fuel, & Electrical System Weights
- Mission Breakdown and Fuel Burn
- Cruise Climb
- Loiter Performance/Endurance
- Takeoff Distance & 50’ obstacle clearance
- Landing Distance
- Spoiler Sizing
- Climb Performance
- Engine-Out Operation
- Windmilling Drag
- Maneuverability
- High Lift System Sizing
- Control Surface Sizing
- Landing Gear Sizing
- Engine Ground Clearance
- Tail Strike Clearance
- Takeoff Distance & 50’ obstacle clearance
- Material Costs
- Manufacturability
- Assembly/Integration Time and Cost
- Fastener Count
- Supply Chain Dynamics
Conceptual Design – Modeling Summary

- Fuselage Pressure Loads
- Fuselage Bending Loads
- Fuselage Weight
- Steady Level Flight Relations
- Wing Moments and Stresses
- Wing Weight
- Stability
- Tail Moments and Stresses
- Tail Weight
- Engine Weight
- Turbine Cycle Analysis
- Noise
- CG Envelope
- Active Gust Response
- Wing Profile Drag
- V-speeds and critical loading cases
- Wing Induced Drag
- Tail Drag
- Fuselage Drag
- Interference Drags
- Airfoil Shape Optimization
- Laminar Flow Control
- Compressibility Effects
- Propulsive Efficiency
- Blade Element Momentum Theory
- APU Sizing
- Hydraulic, Fuel, & Electrical System Weights
- Mission Breakdown and Fuel Burn
- Cruise Climb
- Loiter Performance/Endurance
- Takeoff Distance & 50’ obstacle clearance
- Landing Distance
- Spoiler Sizing
- Climb Performance
- Engine-Out Operation
- Windmilling Drag
- Maneuverability
- High Lift System Sizing
- Control Surface Sizing
- Landing Gear Sizing
- Engine Ground Clearance
- Tail Strike Clearance
- Maintenance Costs
- Material Costs
- Manufacturability
- Assembly/Integration Time and Cost
- Fastener Count
- Supply Chain Dynamics
Fitting Reduced-Order GP-compatible Models

GP-compatible models can approximate any log-convex data [Boyd 2007]

Given set of data points \((x_1, y_1), \ldots, (x_m, y_m) \in \mathbb{R}^n \times \mathbb{R}^n\)

Minimize fitting error \(||y - f(x)||\), subject to \(f \in F\)

Several choices for \(F\), e.g.

- Max-affine functions [Magnani and Boyd 2008]
- Softmax affine functions [Hoburg et al. 2013]
- Implicit posynomials [Hoburg et al. 2013]

Fitting problem solved offline using trust region Newton methods

Many extensions, e.g. conservative fitting, sparse fitting

\[
\begin{align*}
\text{max-affine: } & \quad \text{RMS error} = 0.17764 \\
\text{softmax-affine: } & \quad \text{RMS error} = 0.12729 \\
\text{scaled softmax: } & \quad \text{RMS error} = 0.12657 \\
\text{implicit softmax: } & \quad \text{RMS error} = 0.011219 
\end{align*}
\]
Fitting Reduced-Order GP-compatible Models

GP-compatible models can approximate any log-convex data [Boyd 2007]
GP-compatible models can approximate any log-convex data [Boyd 2007].

Given set of data points

\((x_1, y_1), \ldots, (x_m, y_m) \in \mathbb{R}^n \times \mathbb{R}\)
GP-compatible models can approximate any log-convex data [Boyd 2007]

Given set of data points
\((x_1, y_1), \ldots, (x_m, y_m) \in \mathbb{R}^n \times \mathbb{R}\)

Minimize fitting error \(\|y - f(x)\|\), subject to \(f \in \mathcal{F}\)
GP-compatible models can approximate any log-convex data [Boyd 2007]

Given set of data points

\[(x_1, y_1), \ldots, (x_m, y_m) \in \mathbb{R}^n \times \mathbb{R}\]

Minimize fitting error \[||y - f(x)||\], subject to \(f \in \mathcal{F}\)

Several choices for \(\mathcal{F}\), e.g.

- Max-affine functions [Magnani and Boyd 2008]
- Softmax affine functions [Hoburg et al. 2013]
- Implicit posynomials [Hoburg et al. 2013]

RMS error comparison:

- Max-affine: \(0.17764\)
- Softmax-affine: \(0.12729\)
- Scaled softmax: \(0.12657\)
- Implicit softmax: \(0.011219\)
Fitting Reduced-Order GP-compatible Models

- GP-compatible models can approximate any log-convex data [Boyd 2007]
- Given set of data points \((x_1, y_1), \ldots, (x_m, y_m) \in \mathbb{R}^n \times \mathbb{R}\)
- Minimize fitting error \(||y - f(x)||\), subject to \(f \in F\)
- Several choices for \(F\), e.g.
  - Max-affine functions [Magnani and Boyd 2008]
  - Softmax affine functions [Hoburg et al. 2013]
  - Implicit posynomials [Hoburg et al. 2013]
- Fitting problem solved offline using trust region Newton methods

![Graph showing RMS errors for different function types](image.png)
Fitting Reduced-Order GP-compatible Models

GP-compatible models can approximate any log-convex data [Boyd 2007]

Given set of data points 
\((x_1, y_1), \ldots, (x_m, y_m) \in \mathbb{R}^n \times \mathbb{R}\)

Minimize fitting error \(||y - f(x)||\), subject to \(f \in F\)

Several choices for \(F\), e.g.
- Max-affine functions [Magnani and Boyd 2008]
- Softmax affine functions [Hoburg et al. 2013]
- Implicit posynomials [Hoburg et al. 2013]

Fitting problem solved offline using trust region Newton methods

Many extensions, e.g. conservative fitting, sparse fitting
Handling non-GP-compatible models

- Primary limitation of GP approach: models must be log-convex
Handling non-GP-compatible models

- Primary limitation of GP approach: models must be log-convex
- Can handle more general models using:
  - Nonlinear change of variables
  - Signomial programming
  - Sequential convex programming
- Similar to solving a nonlinear program, but much work offloaded as GP
Handling non-GP-compatible models

▶ Primary limitation of GP approach: models must be log-convex
▶ Can handle more general models using:
  ▶ Nonlinear change of variables
  ▶ Signomial programming
  ▶ Sequential convex programming
▶ Similar to solving a nonlinear program, but much work offloaded as GP

\[
\begin{align*}
\text{minimize} & \quad p_0(x) \\
\text{subject to} & \quad p_i(x) \leq 1, \quad i = 1, \ldots, N_p, \\
& \quad m_j(x) = 1, \quad j = 1, \ldots, N_m \\
& \quad q(x) \leq 1
\end{align*}
\]
Handling non-GP-compatible models

- Primary limitation of GP approach: models must be log-convex
- Can handle more general models using:
  - Nonlinear change of variables
  - Signomial programming
  - Sequential convex programming
- Similar to solving a nonlinear program, but much work offloaded as GP

\[
\begin{align*}
\text{minimize} & \quad p_0(x) \\
\text{subject to} & \quad p_i(x) \leq 1, \quad i = 1, \ldots, N_p, \\
& \quad m_j(x) = 1, \quad j = 1, \ldots, N_m \\
& \quad q(x) \leq 1
\end{align*}
\]
Handling non-GP-compatible models

- Primary limitation of GP approach: models must be log-convex
- Can handle more general models using:
  - Nonlinear change of variables
  - Signomial programming
  - Sequential convex programming
- Similar to solving a nonlinear program, but much work offloaded as GP

\[
\begin{align*}
\text{minimize} & \quad p_0(x) \\
\text{subject to} & \quad p_i(x) \leq 1, \quad i = 1, \ldots, N_p, \\
& \quad m_j(x) = 1, \quad j = 1, \ldots, N_m \\
& \quad q(x) \leq 1
\end{align*}
\]

Sketch of sequential GP approach
Handling non-GP-compatible models

- Primary limitation of GP approach: models must be log-convex
- Can handle more general models using:
  - Nonlinear change of variables
  - Signomial programming
  - Sequential convex programming
- Similar to solving a nonlinear program, but much work offloaded as GP

Sketch of sequential GP approach

\[
\begin{align*}
\text{minimize} & \quad p_0(x) \\
\text{subject to} & \quad p_i(x) \leq 1, \quad i = 1, \ldots, N_p, \\
& \quad m_j(x) = 1, \quad j = 1, \ldots, N_m \\
& \quad q(x) \leq 1
\end{align*}
\]
Handling non-GP-compatible models

- Primary limitation of GP approach: models must be log-convex
- Can handle more general models using:
  - Nonlinear change of variables
  - Signomial programming
  - Sequential convex programming
- Similar to solving a nonlinear program, but much work offloaded as GP

\[
\begin{align*}
\text{minimize} & \quad p_0(x) \\
\text{subject to} & \quad p_i(x) \leq 1, \quad i = 1, \ldots, N_p, \\
& \quad m_j(x) = 1, \quad j = 1, \ldots, N_m \\
& \quad q(x) \leq 1
\end{align*}
\]
Handling non-GP-compatible models

- Primary limitation of GP approach: models must be log-convex
- Can handle more general models using:
  - Nonlinear change of variables
  - Signomial programming
  - Sequential convex programming
- Similar to solving a nonlinear program, but much work offloaded as GP

\[
\begin{align*}
\text{minimize} & \quad p_0(x) \\
\text{subject to} & \quad p_i(x) \leq 1, \quad i = 1, \ldots, N_p, \\
& \quad m_j(x) = 1, \quad j = 1, \ldots, N_m \\
& \quad q(x) \leq 1
\end{align*}
\]

Sketch of sequential GP approach
Handling non-GP-compatible models

- Primary limitation of GP approach: models must be log-convex
- Can handle more general models using:
  - Nonlinear change of variables
  - Signomial programming
  - Sequential convex programming
- Similar to solving a nonlinear program, but much work offloaded as GP

![Sketch of sequential GP approach](image)

\[
\begin{align*}
\text{minimize} & \quad p_0(x) \\
\text{subject to} & \quad p_i(x) \leq 1, \quad i = 1, \ldots, N_p, \\
& \quad m_j(x) = 1, \quad j = 1, \ldots, N_m \\
& \quad q(x) \leq 1
\end{align*}
\]
Handling non-GP-compatible models

- Primary limitation of GP approach: models must be log-convex
- Can handle more general models using:
  - Nonlinear change of variables
  - Signomial programming
  - Sequential convex programming
- Similar to solving a nonlinear program, but much work offloaded as GP

\[
\begin{align*}
\text{minimize} & \quad p_0(x) \\
\text{subject to} & \quad p_i(x) \leq 1, \quad i = 1, \ldots, N_p, \\
& \quad m_j(x) = 1, \quad j = 1, \ldots, N_m \\
& \quad q(x) \leq 1
\end{align*}
\]

Sketch of sequential GP approach
Handling non-GP-compatible models

- Primary limitation of GP approach: models must be log-convex
- Can handle more general models using:
  - Nonlinear change of variables
  - Signomial programming
  - Sequential convex programming
- Similar to solving a nonlinear program, but much work offloaded as GP

\[
\begin{align*}
\text{minimize} & \quad p_0(x) \\
\text{subject to} & \quad p_i(x) \leq 1, \quad i = 1, \ldots, N_p, \\
& \quad m_j(x) = 1, \quad j = 1, \ldots, N_m \\
& \quad q(x) \leq 1
\end{align*}
\]

Sketch of sequential GP approach
Handling non-GP-compatible models

- Primary limitation of GP approach: models must be log-convex
- Can handle more general models using:
  - Nonlinear change of variables
  - Signomial programming
  - Sequential convex programming
- Similar to solving a nonlinear program, but much work offloaded as GP

\[
\begin{align*}
\text{minimize} & \quad p_0(x) \\
\text{subject to} & \quad p_i(x) \leq 1, \quad i = 1, \ldots, N_p, \\
& \quad m_j(x) = 1, \quad j = 1, \ldots, N_m \\
& \quad q(x) \leq 1
\end{align*}
\]

Sketch of sequential GP approach
Take-aways

- Importance of mathematical structure
**Take-aways**

- Importance of mathematical structure
- Key to tractability:

**Looking Ahead**

- Automatic identification of convexity in data
- Understanding reparameterizations
- (Community) development of standard model libraries
Take-aways

- Importance of mathematical structure
- Key to tractability: **convexity**
Take-aways

- Importance of mathematical structure
- Key to tractability: convexity
- Result: reliable and efficient optimization that scales to large problems
Take-aways

- Importance of mathematical structure
- Key to tractability: convexity
- Result: reliable and efficient optimization that scales to large problems

Looking Ahead

- Automatic identification of convexity in data
Take-aways

- Importance of mathematical structure
- Key to tractability: convexity
- Result: reliable and efficient optimization that scales to large problems

Looking Ahead

- Automatic identification of convexity in data
- Understanding reparameterizations
Take-aways

▶ Importance of mathematical structure
▶ Key to tractability: convexity
▶ Result: reliable and efficient optimization that scales to large problems

Looking Ahead

▶ Automatic identification of convexity in data
▶ Understanding reparameterizations
▶ (Community) development of standard model libraries
Thank you
References

Stephen Boyd, Seung-Jean Kim, Lieven Vandenberghe, and Arash Hassibi.
A tutorial on geometric programming.

Stephen Boyd and Lieven Vandenberghe.
Convex Optimization.

D.J. Wilde.
Globally optimal design.

Richard James Duffin, Elmor L Peterson, and Clarence Zener.
*Geometric programming: theory and application.*

Charles S Beightler and Don T Phillips.
*Applied geometric programming*, volume 150.

Mosek-ApS.
Mosek version 6.0.0.148.

Yu Nesterov and A Nemirovsky.
Interior-point polynomial methods in convex programming, volume 13 of studies in applied mathematics.

Sanjay Mehrotra.
On the implementation of a primal-dual interior point method.

T. Bui-Thanh, K. Willcox, and O. Ghattas.
Model reduction for large-scale systems with high-dimensional parametric input space.