Worst-Case Optimization Under Uncertainty With Hybrid Models for Robust Process Systems

Matthew D. Stuber, Robert X. Gottlieb, Dimitri Alston, Matthew E. Wilhelm, and Chenyu Wang

Wednesday, August 16, 2023
Outline

1. Worst-case perspective of robust design and operations, optimization under uncertainty
2. Problem formulation
3. Hybrid modeling and optimization of hybrid models
4. Semi-infinite programming with hybrid models
5. Examples
6. Conclusion
Worst-Case Perspective

- Think “safety-critical” systems: must ensure feasibility in the face of the worst-case realization(s) of uncertainty
Worst-Case Perspective

- Think “safety-critical” systems: must ensure feasibility in the face of the worst-case realization(s) of uncertainty
  - Model validation
Worst-Case Perspective

- Think “safety-critical” systems: must ensure feasibility in the face of the worst-case realization(s) of uncertainty
  - Model validation
  - Fault detection and isolation
Worst-Case Perspective

• Think “safety-critical” systems: must ensure feasibility in the face of the worst-case realization(s) of uncertainty
  
  – Model validation
  – Fault detection and isolation
  – Process design under uncertainty
Worst-Case Perspective

• Think “safety-critical” systems: must ensure feasibility in the face of the worst-case realization(s) of uncertainty

  – Model validation
  – Fault detection and isolation
  – Process design under uncertainty
  – Process flexibility
Worst-Case Perspective

- Think “safety-critical” systems: must ensure feasibility in the face of the worst-case realization(s) of uncertainty
  - Model validation
  - Fault detection and isolation
  - Process design under uncertainty
  - Process flexibility
  - Formal verification
Worst-Case Perspective

- Think “safety-critical” systems: must ensure feasibility in the face of the worst-case realization(s) of uncertainty
  - Model validation
  - Fault detection and isolation
  - Process design under uncertainty
  - Process flexibility
  - Formal verification
  - Among others…
Worst-Case Perspective

- Think “safety-critical” systems: must ensure feasibility in the face of the worst-case realization(s) of uncertainty $\mathbf{p}$

\[
\begin{align*}
\min_{\mathbf{x} \in \mathbb{R}^{n_x}} & \quad \phi(\mathbf{x}) \\
\text{s.t.} & \quad \max_{\mathbf{p} \in \mathbb{R}^{n_p}} \max_{\mathbf{i} \in \mathbb{N}} \{ g_i(\mathbf{x}, \mathbf{p}) \} \leq 0
\end{align*}
\]

Bilevel program
Worst-Case Perspective

• Think “safety-critical” systems: must ensure feasibility in the face of the worst-case realization(s) of uncertainty \( p \)

\[
\begin{align*}
\min_{x \in X \subseteq \mathbb{R}^n_x} & \quad \phi(x) \\
\text{s.t.} & \quad \max_{p \in P \subseteq \mathbb{R}^{n_p}} \max_{i} \{ g_i(x, p) \} \leq 0
\end{align*}
\]

Bilevel program

\[
\begin{align*}
\min_{x \in X \subseteq \mathbb{R}^n_x} & \quad \phi(x) \\
\text{s.t.} & \quad \max_{\gamma \in \mathbb{R}, p \in P \subseteq \mathbb{R}^{n_p}, \hat{z} \in \mathbb{R}^{n_z}} \gamma \\
& \quad \text{s.t.} \quad h(\hat{z}, x, p) = 0 \\
& \quad \gamma \leq \max_{i} \{ g_i(\hat{z}, x, p) \}
\end{align*}
\]

Bilevel program with coupling equality constraints
Problem Formulation

- Main challenge: most algorithms do not apply to problems with coupling equality constraints

\[
\begin{align*}
\min_{x \in \mathbb{R}^n_x} & \quad \phi(x) \\
\text{s.t.} & \quad \max_{\gamma \in \mathbb{R}, p \in \mathbb{R}^n_p, z \in \mathbb{R}^n_z} & & \gamma \\
& \quad \text{s.t.} \quad h(\hat{z}, x, p) = 0 \\
& & \gamma \leq \max \{ g_i(\hat{z}, x, p) \}
\end{align*}
\]
Problem Formulation

• Main challenge: most algorithms do not apply to problems with coupling equality constraints

\[
\begin{align*}
\min_{x \in X \subseteq \mathbb{R}^n} \phi(x) \\
\text{s.t.} \quad \max_{\gamma \in \mathbb{R}, p \in P \subseteq \mathbb{R}^p} \gamma \\
\text{s.t.} \quad h(\hat{z}, x, p) = 0 \\
\gamma \leq \max \{ g_i(\hat{z}, x, p) \}
\end{align*}
\]

\[
\exists z : X \times P \rightarrow Z \\
h(z(x, p), x, p) = 0
\]
Problem Formulation

• Main challenge: most algorithms do not apply to problems with coupling equality constraints

\[
\min_{x \in X \subseteq \mathbb{R}^n_x} \phi(x)
\]

s.t. \( g(z(x,p), x, p) \leq 0, \ \forall p \in P \)

Semi-infinite program with implicit functions embedded
Problem Formulation

• Main challenge: most algorithms do not apply to problems with coupling equality constraints

\[
\begin{align*}
\min_{x \in X \subseteq \mathbb{R}^n_x} & \quad \phi(x) \\
\text{s.t.} & \quad \max_{\gamma \in \mathbb{R}, p \in P \subseteq \mathbb{R}^n_p, \hat{z} \in \mathbb{R}^n_{\hat{z}}} & \quad \gamma \\
& \quad \text{s.t.} & \quad h(\hat{z}, x, p) = 0 \\
& \quad \gamma \leq \max \{g_i(\hat{z}, x, p)\}
\end{align*}
\]

\[
\dot{z}(u, p, t) = f(z(x, p, t), x, p, t), \ t \in I \\
z(x, p, 0) = z_0(x, p)
\]
Problem Formulation

- Main challenge: most algorithms do not apply to problems with coupling equality constraints

\[
\begin{align*}
\min_{x \in X \subseteq \mathbb{R}^n} & \quad \phi(x) \\
\text{s.t.} & \quad \max_{\gamma \in \mathbb{R}, p \in P \subseteq \mathbb{R}^n, \zeta \in \mathbb{R}^n} \gamma \\
& \quad \text{s.t.} \quad h(\zeta, x, p) = 0 \\
& \quad \gamma \leq \max \{ g_i(\zeta, x, p) \}
\end{align*}
\]

\[
\dot{z}(u, p, t) = f(z(x, p, t), x, p, t), \quad t \in I \\
z(x, p, 0) = z_0(x, p)
\]
Hybrid Modeling

First-Principles Models
- Process mechanism is known
- Laws of nature are applied
- Idealized by assumptions

Hybrid Models
- Process knowledge is partially unknown
- Takes advantages of both

Data-Driven Models
- Input-output data inference
- No need for mechanism
- Valid within the domain

Hybrid Modeling

- Hybrid model architectures

\[
\begin{align*}
\text{(a)} & & h_{FPM}(\hat{z}_{FPM}, \cdot, y) = 0 \\
& & \hat{z}_{FPM} \\
& & y \\
& & \mu \\
& & \hat{z}_{DDM}^{(n)} = o(a^{(1)}) \\
& & a^{(1)} = y \\
\text{(b)} & & \hat{z}_{DDM}^{(n)} = o(a^{(1)}) \\
& & a^{(1)} = y \\
& & (\hat{z}_{DDM}, y) \\
& & h_{FPM}(\hat{z}_{FPM}, \hat{z}_{DDM}, y) = 0 \\
& & \hat{z}_{FPM} \\
\text{(c)} & & h_{FPM}(\hat{z}_{FPM}, \cdot, y) = 0 \\
& & (\hat{z}_{FPM}, y) \\
& & \hat{z}_{DDM}^{(n)} = o(a^{(1)}) \\
& & a^{(1)} = (\hat{z}_{FPM}, y) \\
& & \hat{z}_{DDM}
\end{align*}
\]
DGO of ANNs

- Hybrid models with artificial neural networks are still nonconvex
DGO of ANNs

- Hybrid models with artificial neural networks are still nonconvex.
**SIPs with Hybrid Models**

- SIP structure is the same, but some decomposition and simplification may be exploited

\[
\begin{align*}
\min_{x \in X, \hat{z}, \hat{p}} & \quad \phi(x) \\
\text{s.t.} & \quad \max \{ g_i(\hat{z}, x, p) \} \leq 0, \quad \forall p \in P \\
& \quad h(\hat{z}, x, p) = 0, \quad \forall p \in P
\end{align*}
\]

Where:

\[
\hat{z} = (\hat{z}^{\text{FPM}}, \hat{z}^{\text{DDM}})
\]

\[
h = (h^{\text{FPM}}, h^{\text{DDM}})
\]
SIPs with Hybrid Models

• SIP structure is the same, but some decomposition and simplification may be exploited.

\[ \min_{x \in X \in \mathbb{R}^n, \dot{z}} \]

s.t. \[ \max_{p \in \Phi} h(\dot{z}, x, p) \leq 0 \]

Where:

Semi-Infinite Optimization with Hybrid Models

Chenyu Wang, Matthew E. Wilhelm, and Matthew D. Stuber*

ABSTRACT: The robust design of performance/safety-critical process systems, from a model-based perspective, remains an existing challenge. Hybrid first-principles data-driven models offer the potential to dramatically improve model prediction accuracy, stepping closer to the digital twin concept. Within this context, worst-case engineering design feasibility and reliability problems give rise to a class of semi-infinite program (SIP) formulations with hybrid models as coupling equality constraints. Reduced-space deterministic global optimization methods are exploited to solve this class of SIPs to ϵ-global optimality in finitely many iterations. This approach is demonstrated on two challenging case studies: a nitrification reactor for a wastewater treatment system to address worst-case feasibility verification of dynamical systems and a three-phase separation system plagued by numerical domain violations to demonstrate how they can be overcome using a nonsmooth SIP formulation with hybrid models and a validity constraint incorporated.
SIPs with Hybrid Models

Solution Method

Backbone:
deterministic global optimization solver

EAGO

EAGO.jl: easy advanced global optimization in Julia

M. E. Wilhelm and M. D. Stuber
Process Systems and Operations Research Laboratory, Department of Chemical and Biomolecular Engineering, University of Connecticut, Storrs, CT, USA

ABSTRACT
An extensible open-source deterministic global optimizer (EAGO) programmed entirely in the Julia language is presented. EAGO was developed to serve the need for supporting higher-complexity user-defined functions (e.g., functions defined implicitly via algorithms) within optimization models. EAGO embeds a first-of-its-kind implementation of McCormick arithmetic in an Evaluator structure allowing for the construction of convex/concave relaxations using a combination of source code transformation, multiple dispatch, and context-specific approaches. Utilities are included to parse user-defined functions into a directed acyclic graph representation and perform symbolic transformations enabling dramatically improved solution speed. EAGO is compatible with a wide variety of local optimizers, the most exhaustive library of transcendental functions, and allows for easy accessibility through the JuMP modeling language. Together with Julia’s minimalistic syntax and competitive speed, these powerful features make EAGO a versatile research platform enabling easy construction of novel meta-solvers, incorporation and utilization of new relaxations, and extension to advanced problem formulations encountered in engineering and operations research (e.g., multilevel problems, user-defined functions). The aplicability and flexibility of this novel software is demonstrated on a diverse set of examples. Lastly, EAGO is demonstrated to perform comparably to state-of-the-art commercial optimizers on a benchmarking test set.
SIPs with Hybrid Models

Solution Method

Backbone: deterministic global optimization solver

SIP Solver: SIPres cutting-plane algorithm (Mitsos, 2011)
SIPs with Hybrid Models

Solution Method

Backbone: deterministic global optimization solver

SIP Solver: SIPres cutting-plane algorithm (Mitsos, 2011)

Data-Driven Model: ANN with supported activation functions (NNlib.jl)
SIPs with Hybrid Models

Solution Method

Backbone:
deterministic global optimization solver

```
# Objective function
f(x) = (x[1]-3.5)^4 - 5.0*(x[1]-3.5)^3 - 2.0*(x[1]-3.5)^2 + 15.0*(x[1]-3.5)

# SIP constraint
# g(x, y, p) = y + cos(x - p/90) - p
function gSIP(x, p)
    # Call surrogate model to solve for y at the given x and p
    y = scale_output(ANN_model, x[1], p[1])

    # Return g
end

# Variable bounds for SIP
x_lo_SIP = Float64[0.5]
x_hi_SIP = Float64[8.0]
p_lo_SIP = Float64[80.0]
p_hi_SIP = Float64[120.0]

# Solve SIP
sip_result = sip_solve(SIPRes(), x_lo_SIP, x_hi_SIP, p_lo_SIP, p_hi_SIP,
                        f, Any[gSIP], abs_tolerance = 1E-4)
```
Ex: Simple

• Consider the SIP:

\[
\min_{\hat{z} \in \mathbb{Z}, x \in X} (x - 3.5)^4 - 5(x - 3.5)^3 - 2(x - 3.5)^2 + 15(x - 3.5)
\]

s.t. \( \hat{z} + \cos(x - p / 90) - p \leq 0, \forall p \in P \)

\( \hat{z} - \left( \frac{x - x^3}{6} + \frac{x^5}{120} \right) / \sqrt{\hat{z}} = 0, \forall p \in P \)
Ex: Simple

• Consider the SIP:

\[
\min_{\hat{z} \in Z, x \in X} \ (x - 3.5)^4 - 5(x - 3.5)^3 - 2(x - 3.5)^2 + 15(x - 3.5)
\]

s.t. \( z(x, p) + \cos(x - p / 90) - p \leq 0, \forall p \in P \)

\[
\phi(x) = -7.8985
\]

\[
z(x, p) : h(z(x, p), x, p) = 0
\]
Ex: Simple

- Consider the SIP:

\[
\min_{\hat{x} \in Z, x \in X} (x - 3.5)^4 - 5(x - 3.5)^3 - 2(x - 3.5)^2 + 15(x - 3.5)
\]

s.t. \( z_s(x, p) + \cos(x - p / 90) - p \leq 0, \forall p \in P \)

\[
\phi^* = -7.6961
\]

\[
z_s(x, p) \approx z(x, p) : h(z(x, p), x, p) = 0
\]

\[
g(z_s(x, p), x, p)
\]
Ex: Nitrification CSTR

Robust operation: \( \phi^* = \max_{\pi \in \Pi, \eta \in \Theta} \eta \)

s.t. \( \eta \leq \max\{C_{NH}(t_f, \pi, u) - USL_{NH}, C_O(t_f, \pi, u) - USL_O\}; \forall u \in U \)
**Ex: Nitrification CSTR**

**Model:**

\[
\begin{align*}
\frac{dC_{NH}}{dt} &= \frac{1}{V}(m_{in} C_{in} - m_{out} C_{NH} - r_{AO} X_{AO}) \\
\frac{dC_{NO}}{dt} &= r_{AO} X_{AO} - r_{NO} X_{NO} \\
\frac{dC_{NO}}{dt} &= r_{NO} X_{NO} \\
\frac{dC_{O}}{dt} &= -r_{AO} X_{AO} + r_{NO} \psi_{NO} X_{NO} + r_{in} (C^*-C_0)
\end{align*}
\]

\[\pi = C_{in} \in \Pi\]

\[u^* = 40\]

\[u^* = 521.76\]

\[\eta^* = 0.288\]

Robust operation:

\[\phi^* = \max_{\pi \in \Pi, \eta \in \mathcal{H}} \eta \]

s.t. \[\eta \leq \max\{C_{NH}(t_f, \pi, u) - \text{USL}_{NH}, C_{O}(t_f, \pi, u) - \text{USL}_O\}; \forall u \in U\]
Ex: Remote Separations
Ex: Remote Separations

\[ f(\hat{z}) = \cos^{-1}\left(1 - \frac{\hat{z}}{r}\right), \quad r, \hat{z} > 0 \]

\[ Z_{\text{valid}} = [0, 2r] \]

```
julia> acos(-1.1)
ERROR: DomainError with -1.1: 
acos(x) not defined for |x| > 1
Stacktrace:
 [1] acos_domain_error(x::Float64)
     @ Base.Math .\special\trig.jl:669
 [2] acos(x::Float64)
     @ Base.Math .\special\trig.jl:699
 [3] top-level scope
     @ REPL[5]:1
```
Conclusions

• Worst-case optimization under uncertainty problems are nonconvex SIPs with coupling equality constraints
• Hybrid modeling approaches can be used to potentially reduce the complexity of the optimization problem
• Toolchain based on EAGO for solving SIPs (with hybrid models)
• Examples illustrate some usage cases for relevant applications in process systems engineering, including:
  – dynamic optimization
  – validity domains
Acknowledgements

Members of the Process Systems and Operations Research Laboratory at the University of Connecticut (https://psor.uconn.edu/)

Funding:
National Science Foundation, Award No.: 1932723

Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.
Thank You!