IMPERIAL

LoReSIO.jl: Local Reduction for Semi-Infinite Optimization

Jad Wehbeh and Eric C. Kerrigan
19/07/2024

$$\min \quad \max \quad J(\cdot)$$

- Control gains
- Estimator Parameters
- Design variables

- Disturbances
- Measurement noise
- Model uncertainty

Subject to:

- System dynamics: $f(\cdot) = 0$
- Inequality constraints: $g(\cdot) \leq 0$
- Measurements $h(\cdot) = 0$

Semi-infinite program

Imperial College London

What is Semi-Infinite Programming?

Infinite set of constraints

Finite set of decision variables

$$\min_x \; f(x)$$

$$\text{s.t.} \quad g(x,y) \le 0 \qquad \forall y \in \mathcal{Y}$$

Min-Max problems are Semi-Infinite Programs

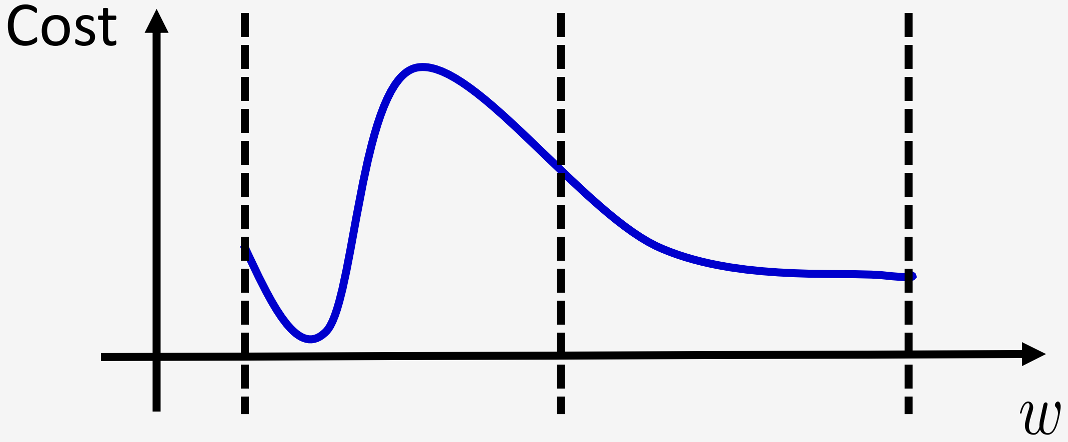
$$\min_x \max_y \; f(x,y) \quad \Longleftrightarrow \quad \begin{aligned} &\min_{x,\gamma} \; \gamma \\ &\text{s.t.} \quad f(x,y) - \gamma \le 0 \qquad \forall y \in \mathcal{Y} \end{aligned}$$

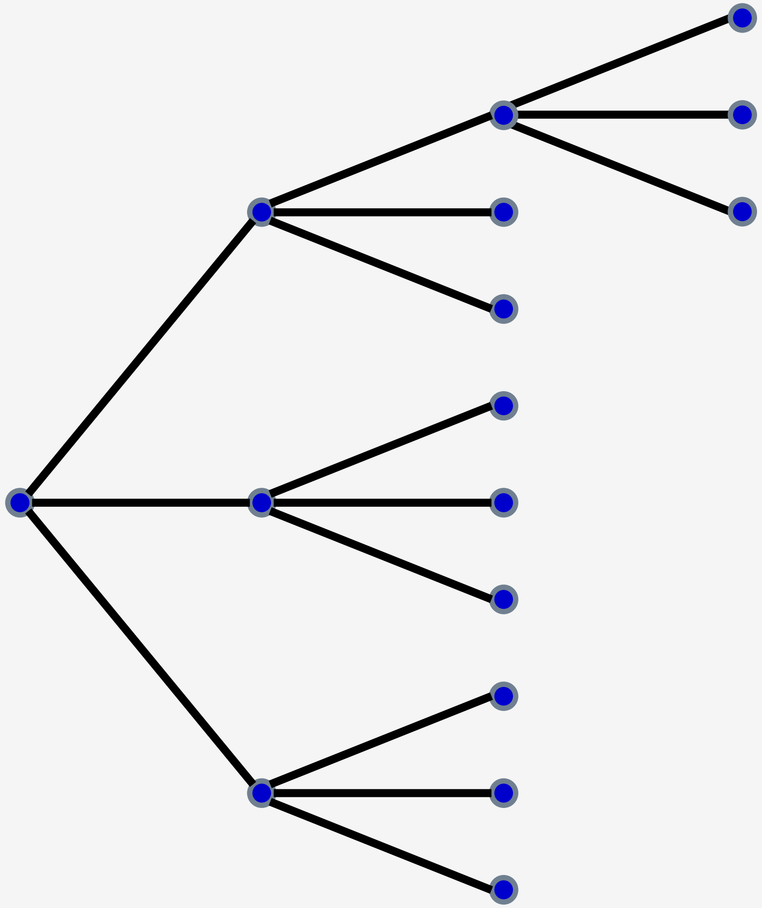
Why Robust Control?

- For many applications, overall performance is secondary to constraint satisfaction.
- While obtaining a distribution for the uncertainties affecting a system can be difficult in practice, bounds are generally much simpler to determine.
- Robust control ensures that constraints are satisfied while the uncertainties remain within the prescribed bounds.

Solving Semi-Infinite Programs

- Global methods
- Random sampling
- Scenario trees

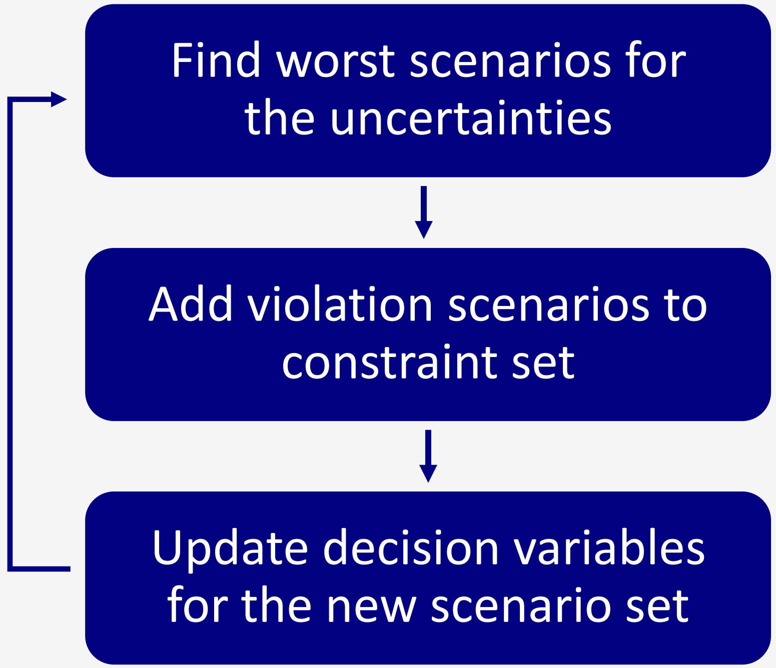




Local Reduction: Algorithm Description

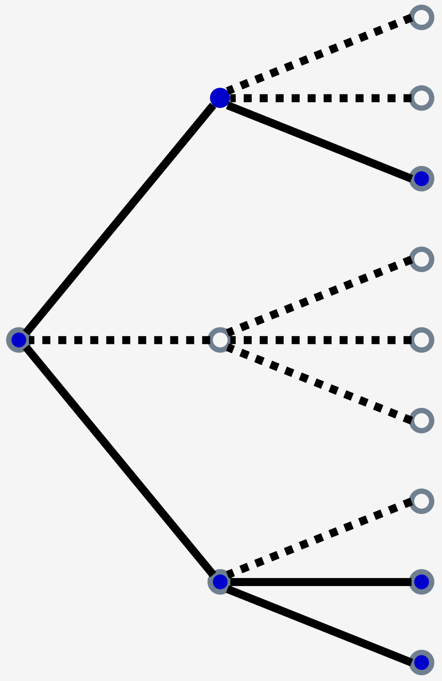
$$\min_{x} \; f(x)$$

$$\text{s.t.} \quad g(x,y) \leq 0 \qquad \forall y \in \mathcal{Y}$$

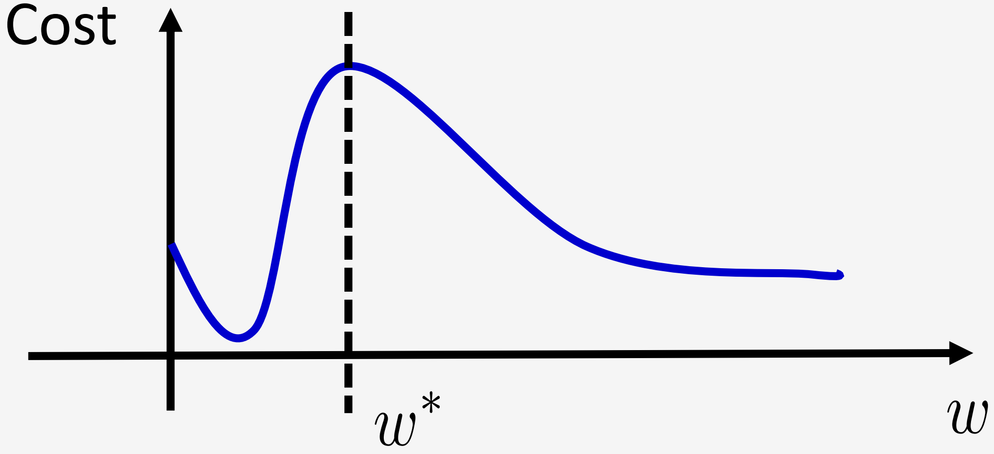


Local Reduction: Advantages

Reduce the overall number of scenarios



Identify non-boundary extremes



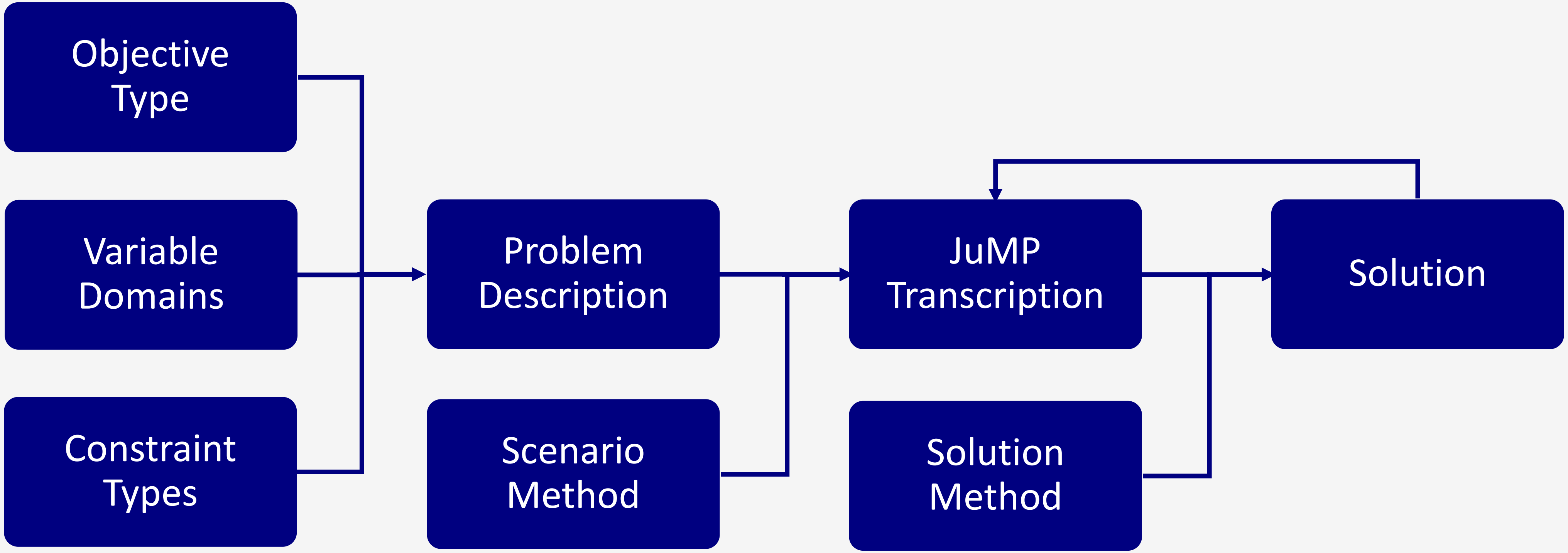
Julia Implementation

```julia
struct LoReSIOProblem
    objective::LoReSIOObjective
    u_vars::Vector{LoReSIODiscVariableDomain}
    w_vars::Vector{AugmentedDomain}
    z_vars::Vector{EqualityDomain}
    s_vars::Vector{LoReSIODiscVariableDomain}
    ineqs::Vector{LoReSIOConstraints}
    opts::LoReSIOProblemOptions
    info::LoReSIOProblemInfo
    transcription::LoReSIOProblemTranscription
end
```

```julia
struct ParamDomain <: LoReSIODiscVariableDomain
    nvars::Int
    lims::Array{Real,2}
    eqs::VariableConstraints
    ineqs::VariableConstraints
end
```

```julia
struct UncertConstraint <: LoReSIOConstraints
    nconstr::Int
    constr!::Vector{Function}
end
```

JuMP Underbelly




```
mutable struct LoReSIOProblemTranscription
    min_model::Model
    max_models::Vector{Union{Model,LoReSIOProblemTranscription}}
end
```

Results: Robust Obstacle Avoidance

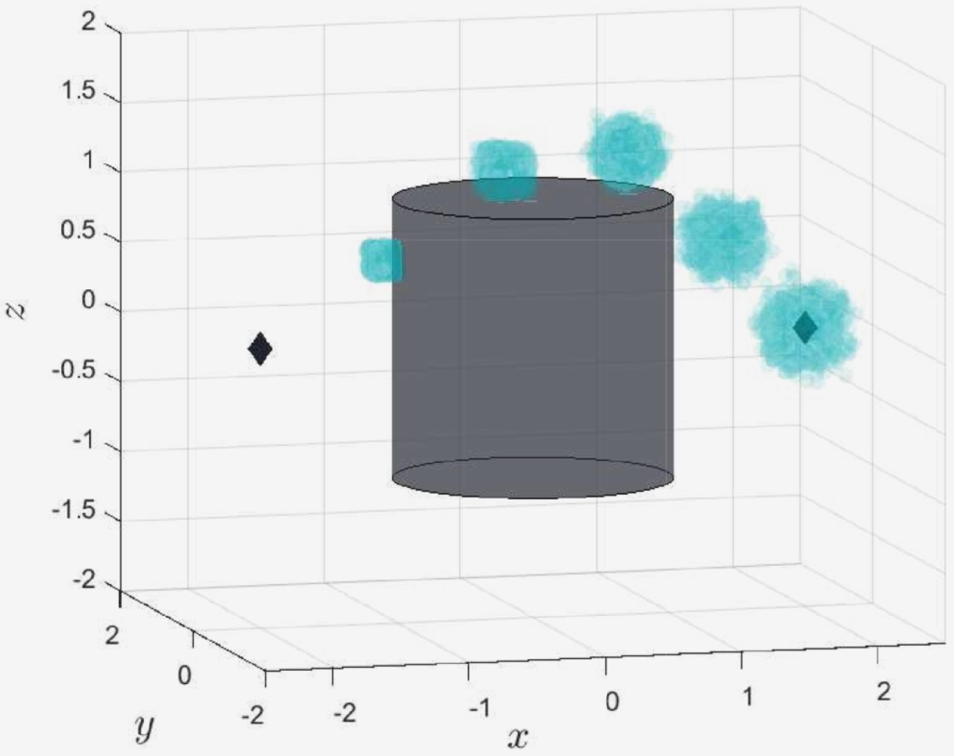
System Description

$$x_{i,k+1} = x_{i,k} + u_{i,k} + w_{i,k} \qquad \forall i \in \{1,2,3\}$$

$$\min_{\mathbf{u}} \max_{\mathbf{x},\mathbf{w}} \left(\sum_{k=0}^{n-1} \|\mathbf{u}_k\|_R^2 + \|\mathbf{x}_n - \mathbf{x}_{ref}\|_Q^2 \right)$$

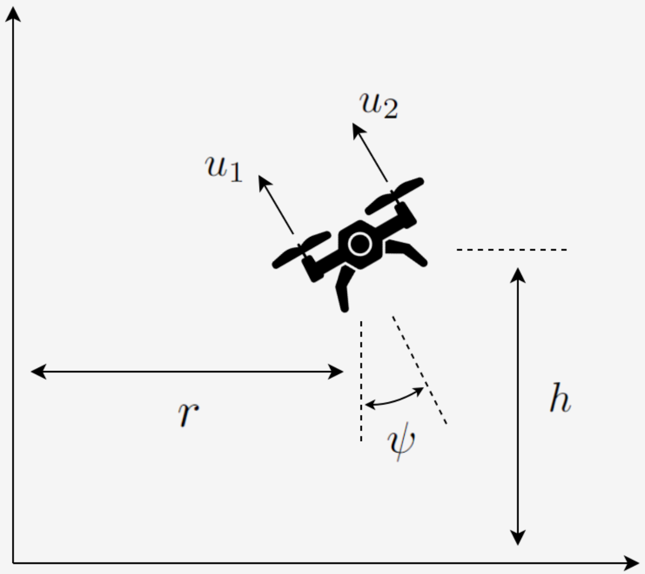
$$x_{1,k}^2 + x_{2,k}^2 \geq 1 \quad \vee \quad x_{3,k} \geq 1 \quad \vee \quad x_{3,k} \leq -1$$

Implementation Results (29 Scenario Termination)

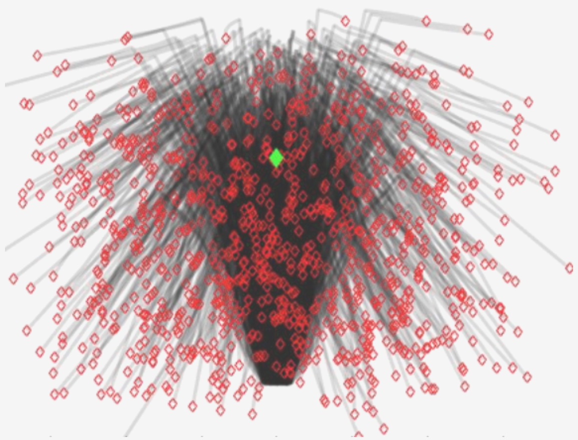


Results: Robust Output Feedback

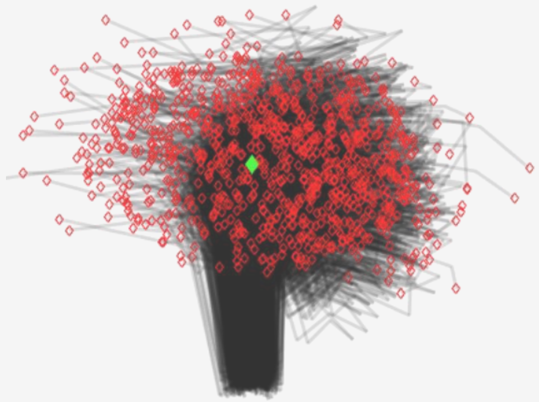
System Description



Implementation Results (50 Scenario Cap)



Open-Loop Performance



Optimized Performance

IMPERIAL

Questions?

References:

- Blankenship, J.W. and Falk, J.E., 1976. Infinitely constrained optimization problems. *Journal of Optimization Theory and Applications*, *19*, pp.261-281.
- Zagorowska, M., Falugi, P., O'Dwyer, E. and Kerrigan, E.C., 2024. Automatic scenario generation for efficient solution of robust optimal control problems. *International Journal of Robust and Nonlinear Control*, *34*(2), pp.1370-1396.
- Wehbeh, J. and Kerrigan, E.C., 2024. Semi-Infinite Programs for Robust Control and Optimization: Efficient Solutions and Extensions to Existence Constraints. *arXiv preprint arXiv:2404.05635*.

Email: jad.wehbeh22@imperial.ac.uk