Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnLine 56-2 (2023) 1229–1234

Automatic Scenario Generation for Robust Optimal Control Problems ⋆

M. Zagorowska * **P. Falugi** **,† **E. O'Dwyer** *** **E. C. Kerrigan** **,****

* *Automatic Control Laboratory, ETH Zurich, Switzerland (e-mail: mzagorowska@ethz.ch)*
** *Department of Electrical and Electronic Engineering, Imperial College London, UK (e-mail: p.falugi@imperial.ac.uk, e.kerrigan@imperial.ac.uk).*
*** *Department of Chemical Engineering, Imperial College London, UK (e-mail: e.odwyer@imperial.ac.uk)*
**** *Department of Aeronautics, Imperial College London, UK*
† *Department of Engineering & Construction, University of East London, UK.*

Abstract: Existing methods for nonlinear robust control often use scenario-based approaches to formulate the control problem as nonlinear optimization problems. Increasing the number of scenarios improves robustness, while increasing the size of the optimization problems. Mitigating the size of the problem by reducing the number of scenarios requires knowledge about how the uncertainty affects the system. This paper draws from local reduction methods used in semi-infinite optimization to solve robust optimal control problems with parametric uncertainty. We show that nonlinear robust optimal control problems are equivalent to semi-infinite optimization problems and can be solved by local reduction. By iteratively adding interim globally worst-case scenarios to the problem, methods based on local reduction provide a way to manage the total number of scenarios. In particular, we show that local reduction methods find worst case scenarios that are not on the boundary of the uncertainty set. The proposed approach is illustrated with a case study with both parametric and additive time-varying uncertainty. The number of scenarios obtained from local reduction is 101, smaller than in the case when all $2^{14+3\times192}$ boundary scenarios are considered. A validation with randomly drawn scenarios shows that our proposed approach reduces the number of scenarios and ensures robustness even if local solvers are used.

Copyright © 2023 The Authors. This is an open access article under the CC BY-NC-ND license (https://creativecommons.org/licenses/by-nc-nd/4.0/)

Keywords: Optimization, Mathematical programming, Trajectory optimization, Uncertainty, Iterative methods, Numerical simulation, Robust control

1. INTRODUCTION

Robust nonlinear optimal control problems are often solved using a scenario-based approach, where each *scenario* corresponds to a separate realization of uncertainty. This paper draws from approaches used in semi-infinite optimization to solve robust optimal control problems in an efficient way.

To ensure that the optimization problems resulting from scenario-based approaches to robust control are tractable, the number of scenarios must be limited (Calafiore and Campi, 2006). Usually, the choice of scenarios is done from experience (Grammatico et al., 2015; Puschke et al., 2018) and requires knowledge about both the controlled system and the uncertainty to ensure that the chosen scenarios guarantee robustness. A recent review of scenario-based methods was done by Campi et al. (2021) who indicated that scenario selection is highly affected by the knowledge about the uncertainty distribution. In practice, to tackle problems with limited knowledge about the uncertainty, it is often assumed that the worst-case scenarios lie on the boundary of the uncertainty set (Mutapcic and Boyd, 2009; Lucia et al., 2014; Vuffray et al., 2015). As indicated by Thombre et al. (2021), the worst-case scenario in nonlinear systems may lie in the interior of the uncertainty range. In this paper, we present a method for choosing potential worst-case scenarios that is derived from semi-infinite optimization and is independent from the uncertainty distribution.

An in-depth review of semi-infinite optimization methods has been done by Hettich and Kortanek (1993); Hettich et al. (2009); Hettich (1983) and recently by Seidel and Küfer (2020); Djelassi et al. (2021). Semi-infinite optimization methods have been used for optimal control by Hauser (2018) to find optimal trajectories for robotic arms. However, they considered only exogenous uncer-

⋆ This work has received funding from the EPSRC (Engineering and Physical Sciences) under the Active Building Centre project (reference number: EP/V012053/1). M. Zagorowska also acknowledges NCCR Automation, a National Centre of Competence in Research, funded by the Swiss National Science Foundation (grant number: 180545).

2405-8963 Copyright © 2023 The Authors. This is an open access article under the CC BY-NC-ND license.
Peer review under responsibility of International Federation of Automatic Control.
10.1016/j.ifacol.2023.10.1743

tainty due to obstacles that did not affect the dynamics of the controlled systems. Puschke et al. (2018) used semi-infinite optimization methods to solve an optimal control problem with only parametric uncertainty. Time-varying uncertainty was considered by Thombre et al. (2021) who used local reduction to find the interim worst-case scenarios. However, they assumed that at every time step the number of possible scenarios was finite.

The main contribution of the current work is a formulation of robust nonlinear optimal control problems as semi-infinite optimization problems. We then numerically demonstrate that local reduction methods enable more robust handling of significant parametric and time-varying uncertainty than existing approaches.

The rest of the paper is structured as follows. Section 2 introduces robust optimal control problems. Section 3 presents the new method for solving robust optimal control problems. The numerical results are shown in Section 4. The paper ends with conclusions in Section 5.

2. PROBLEM FORMULATION

2.1 Semi-infinite optimization problem

$$\mathcal{Q}: \quad \min_{\theta\in\mathcal{A}} \quad Q(\theta) \tag{1a}$$
$$\text{subject to } R(\theta,\rho)\le 0 \text{ for all } \rho\in\mathcal{B} \tag{1b}$$

where $\mathcal{A}\subset\mathbb{R}^{n_\theta}$ and $\mathcal{B}\subset\mathbb{R}^{n_\rho}$ are nonempty and compact sets, and Q and R are continuous functions of their respective arguments (Blankenship and Falk, 1976). The problem (1) has a finite number of variables θ but includes an infinite number of constraints if $\mathcal{B}$ has an infinite number of points. In particular, $\mathcal{B}$ may be uncountable.

One approach to remove the infinite number of constraints consists in rewriting the constraint (1b) as:

$$S(\theta) := \max_{\rho\in\mathcal{B}} R(\theta,\rho)\le 0. \tag{2}$$

The challenge in solving the equivalent problem with constraint (2) is in non-differentiability of the function $S(\cdot)$. The local reduction method proposed by Blankenship and Falk (1976) allows overcoming the non-differentiability of $S(\cdot)$ by sequentially solving (1) with finite subsets of constraints taken from $\mathcal{B}$. In this paper, we show that optimal control problems can be formulated as semi-infinite optimization problems and solved using the method proposed by Blankenship and Falk (1976).

2.2 Dynamic system with uncertainty

The system to be controlled is described by a nonlinear difference equation with time-varying uncertainty $w_k\in\mathbb{W}\subset\mathbb{R}^{n_w}$ and constant uncertainty $d\in\mathbb{D}\subset\mathbb{R}^{n_d}$:

$$x_{k+1}=f_k(x_k,u_k,w_k,d) \tag{3}$$

where f_k is continuously differentiable. The state x_0 at time zero is w.l.o.g. assumed to be equal to a given $\hat{x}$.

The control trajectory $\mathbf{u}:=(u_0,\dots,u_{N-1})$ is generated by a causal dynamic feedback policy

$$u_k:=\pi_k(x_0,\dots,x_k;q_0,\dots,q_k,r)$$

that is parameterised by $\mathbf{q}:=(q_0,q_1,\dots,q_{N-1})\in\mathbb{R}^{n_q}$ and $r\in\mathbb{R}^{n_r}$. The state trajectory $\mathbf{x}:=(x_0,\dots,x_N)$. The time-varying uncertainty w_k at time k and the constant uncertainty d affect the dynamics in both an additive and non-additive way, and take on values from compact and uncountable (infinite cardinality) sets. Uncertainty in the measured value of x_k can be modelled by a suitably-defined choice of f_k, π_k and w_k.

A trajectory $(\mathbf{x},\mathbf{u})$ satisfying the dynamics (3) and control policy for a given parameterization $(\mathbf{q},r)$ and realisation of uncertainty $(\mathbf{w},\ d)$, where the trajectory $\mathbf{w}:=(w_0,\dots,w_{N-1})\in\mathbb{W}^N:=\mathbb{W}\times\cdots\times\mathbb{W}$, is defined as:

$$\begin{aligned}\mathbf{z}(\mathbf{q},r,\mathbf{w},d):=\Big\{(\mathbf{x},\mathbf{u})\mid\ & x_0=\hat{x}\\ & x_{k+1}=f_k(x_k,u_k,w_k,d)\\ & u_k=\pi_k(x_0,\dots,x_k;q_0,\dots,q_k,r)\\ & k=0,1,\dots,N-1\Big\}.\end{aligned} \tag{4}$$

2.3 Robust optimal control problem

Objective function and constraints The cost function for the optimal control problem over a horizon of length N is:

$$J_N(\mathbf{x},\mathbf{u},\mathbf{w},d):=J_f(x_N,w_N,d)+\sum_{k=0}^{N-1}\ell_k(x_k,u_k,w_k,d). \tag{5}$$

Both the terminal cost function $J_f(\cdot,\cdot,\cdot)$ and stage cost $\ell_k(\cdot,\cdot,\cdot,\cdot)$ are continuously differentiable and depend on the uncertainty w and d. The objective of the optimal control problem is to find a feedback policy π for system (3) such that the worst-case cost in (5) is minimized and the constraints

$$g_k(x_k,u_k,w_k,d)\le 0 \tag{6}$$

are satisfied for all time instants $k=0,\dots,N-1$, all states $\mathbf{x}$, control $\mathbf{u}$, uncertainty $\mathbf{w}$ and d. The vector function of n_g components, $g_k(\cdot,\cdot,\cdot,\cdot)$, is continuously differentiable and depends on uncertainty $\mathbf{w}$ and d. Note that a constraint on x_N can be included by incorporating f_{N-1} in a suitable definition of g_{N-1}.

Semi-infinite formulation Given a set of uncertainties $\mathbb{H}\subseteq\mathbb{W}^N\times\mathbb{D}$, the problem in this work is stated as:

$$\mathcal{P}_N(\mathbb{H}): \min_{\substack{\mathbf{q},r\\ \mathbf{x}^i,\mathbf{u}^i,i\in\mathbb{J}}} \max_{i\in\mathbb{J}} J_N(\mathbf{x}^i,\mathbf{u}^i,\mathbf{w}^i,d^i) \tag{7a}$$
$$\text{s.t. } g_k(x_k^i,u_k^i,w_k^i,d^i)\le 0,\ \forall i\in\mathbb{J},k=0,\dots,N-1 \tag{7b}$$
$$(\mathbf{x}^i,\mathbf{u}^i)=\mathbf{z}(\mathbf{q},r,\mathbf{w}^i,d^i),\ \forall i\in\mathbb{J} \tag{7c}$$

where $\mathbb{J}:=\{1,\dots,\operatorname{card}\mathbb{H}\}$ and $(\mathbf{x}^i,\mathbf{u}^i)$ is the state and input trajectory associated with the i^{th} disturbance realisation $(\mathbf{w}^i,d^i)$ such that $\mathbb{H}=\bigcup_{i\in\mathbb{J}}\{(\mathbf{w}^i,d^i)\}$.

If $\mathbf{z}(\cdot)$ in (7c) is linear jointly in all arguments, the problem (7) can often be solved using scenario-based methods for robust control from Calafiore and Campi (2006); Scokaert and Mayne (1998), provided additional convexity assumptions are satisfied by the uncertainty set $\mathbb{W}$. In this work, the dynamics from (3) are nonlinear and $\mathbb{W}$ is only non-empty and compact.

Theorem 1. The robust optimal control problem (7) is equivalent to the semi-infinite optimization problem (1) with $\theta:=(\mathbf{q},r,\gamma)$, where γ is an additional scalar parameter characterizing the cost upper-bound, $\rho:=(\mathbf{w},d)$ and the sets $\mathcal{A}:=\mathbb{R}^{n_q}\times\mathbb{R}^{n_r}\times\mathbb{R}$, $\mathcal{B}:=\mathbb{H}$.

Proof 1. In contrast to (1a), the objective function in (7a) contains uncertainty. Introducing $\gamma \in \mathbb{R}$, we rewrite (7) as:

$$\mathcal{P}_N(\mathbb{H}): \quad \min_{\substack{\gamma,\mathbf{q},r \\ \mathbf{x}^i,\mathbf{u}^i,i\in\mathbb{J}}} \quad \gamma \tag{8a}$$

$$\text{s.t. } g_k(x_k^i, u_k^i, w_k^i, d^i) \leq 0,\ \forall i \in \mathbb{J}, k = 0,\dots,N-1 \tag{8b}$$

$$(\mathbf{x}^i, \mathbf{u}^i) = \mathbf{z}(\mathbf{q}, r, \mathbf{w}^i, d^i),\ \forall i \in \mathbb{J} \tag{8c}$$

$$J_N(\mathbf{x}^i, \mathbf{u}^i, \mathbf{w}^i, d^i) \leq \gamma,\ \forall i \in \mathbb{J} \tag{8d}$$

The problem (8) has uncertainty exclusively in the constraints. If $\operatorname{card}\mathbb{H}$ is finite, then the problem (8) is convenient to solve numerically using tailored efficient finite-dimensional optimization methods that exploit the sparsity in the relevant Jacobians and Hessians. However, infinite cardinality of $\mathbb{H}$ yields an infinite number of both constraints and variables, which means that the problem (8) needs to be further reformulated to become (1). Noticing that the constraint (8b) is equivalent to

$$\max_k g_k(x_k^i, u_k^i, w_k^i, d^i) \leq 0,\ \forall i \in \mathbb{J}, \tag{9}$$

we introduce

$$G(\mathbf{x}^i, \mathbf{u}^i, \mathbf{w}^i, d^i, \gamma) := \max\{\max_{h,k}\ e_h^T g_k(x_k^i, u_k^i, w_k^i, d^i), J_N(\mathbf{x}^i, \mathbf{u}^i, \mathbf{w}^i, d^i) - \gamma\} \tag{10}$$

In (10), e_h is the h^{th} column of an identity matrix $\mathbb{I}_{n_g}$. Using (4) and (10), we can write (8) as:

$$\mathcal{P}_N(\mathbb{H}): \quad \min_{\mathbf{q},r,\gamma} \quad \gamma \tag{11a}$$

$$\text{s.t. } G(\mathbf{z}(\mathbf{q}, r, \mathbf{w}, d), \mathbf{w}, d, \gamma) \leq 0,\ \forall (\mathbf{w}, d) \in \mathbb{H}, \tag{11b}$$

The problem (11) is equivalent to

$$\mathcal{P}_N(\mathbb{H}): \quad \min_{\mathbf{q},r,\gamma} \quad \gamma \tag{12a}$$

$$\text{s.t.} \max_{(\mathbf{w},d)\in\mathbb{H}} G(\mathbf{z}(\mathbf{q}, r, \mathbf{w}, d), \mathbf{w}, d, \gamma) \leq 0. \tag{12b}$$

Taking $\theta := (\mathbf{q}, r, \gamma)$ and $\rho := (\mathbf{w}, d)$ in (11) (similarly in (12)) we obtain the form of (1) (similarly (2)). □

We also notice that (12b) is equivalent to:

$$G_{\max}(\mathbf{q}, r, \gamma, \mathbb{H}) := \max_{\substack{(\mathbf{w},d)\in\mathbb{H} \\ (\mathbf{x},\mathbf{u})=\mathbf{z}(\mathbf{q},r,\mathbf{w},d)}} G(\mathbf{x}, \mathbf{u}, \mathbf{w}, d, \gamma) \tag{13}$$

Theorem 1 allows solving problem (7) as a semi-infinite optimization problem of the form (1) using the local reduction from Blankenship and Falk (1976).

3. LOCAL REDUCTION FOR OPTIMAL CONTROL

The local reduction method from Blankenship and Falk (1976) consists in iteratively solving finite-dimensional optimization problems. We use the local reduction methods for the problem (11) by iteratively solving optimal control problems parametrised by scenarios. A scenario is a realisation of the uncertainty $(\mathbf{w}^*, d^*) \in \mathbb{W}^N \times \mathbb{D}$.

3.1 Minimization step

Algorithm 1 is the local reduction algorithm for robust optimal control. The algorithm in iteration j solves an optimal control problem of the form (8) or (11) assuming that the number of scenarios $\operatorname{card}\mathbb{H}_j$ at step j is finite. The algorithm needs an initial guess for the parameters of the controller. For instance, the initial guess can be obtained by solving (11) for one scenario, $\operatorname{card}\mathbb{H}_1 = 1$.

Algorithm 1: Exact local reduction method

Input: Initial guess for $\mathbf{q}$, r, γ and $\mathbb{H}_1 \neq \emptyset$
Output: Optimal $\mathbf{q}^*$, r^*, γ^*, set of scenarios $\mathbb{H}^*$

1 Set $\mathbf{q}^1 \leftarrow \mathbf{q}$, $r^1 \leftarrow r$, $\gamma^1 \leftarrow \gamma$, $j \leftarrow 1$
2 **repeat**
3 Compute $G_{\max}(\mathbf{q}^j, r^j, \gamma^j, \mathbb{W}^N \times \mathbb{D})$ and a maximizer $(\mathbf{x}^j, \mathbf{u}^j, \mathbf{w}^j, d^j)$ by solving (13) with $\mathbb{H} = \mathbb{W}^N \times \mathbb{D}$.
4 **if** $G_{\max}(\mathbf{q}^j, r^j, \gamma^j, \mathbb{W}^N \times \mathbb{D}) \leq 0$ **then**
5 $\mathbb{H}_{j+1} \leftarrow \mathbb{H}_j$
6 **else**
7 Add new scenario $\mathbb{H}_{j+1} \leftarrow \mathbb{H}_j \cup \{(\mathbf{w}^j, d^j)\}$
8 Find a $(\mathbf{q}^{j+1}, r^{j+1}, \gamma^{j+1})$ that solves $\mathcal{P}_N(\mathbb{H}_{j+1})$ using (8) or (11).
9 **end**
10 Set $j \leftarrow j + 1$
11 Set $(\mathbf{q}^*, r^*, \gamma^*) \leftarrow (\mathbf{q}^j, r^j, \gamma^j)$ and $\mathbb{H}^* \leftarrow \mathbb{H}_j$.
12 **until** $\operatorname{card}\mathbb{H}_j = \operatorname{card}\mathbb{H}_{j-1}$;

In the first step of Algorithm 1 (line 3), the algorithm checks whether worst-case scenarios exist that would lead to a violation of constraints (10). If no constraints are violated (line 4), the current parameters give a robust solution to the current set of scenarios $\mathbb{H}_j$. If there exists at least one violated constraint, then a scenario corresponding to the maximum constraint violation is added to the scenario set $\mathbb{H}_{j+1}$ in the next iteration (line 7). The new set $\mathbb{H}_{j+1}$ is then used to find a new set of control parameters (line 9). The algorithm ends if no new scenarios are added, i.e. $\operatorname{card}\mathbb{H}_j = \operatorname{card}\mathbb{H}_{j-1}$.

In this work, any scenario corresponding to the maximum constraint violation can be added to the set of scenarios. However, it has been shown that computational performance may be improved if multiple scenarios are added (Tsoukalas et al., 2008).

The convergence of local reduction method in the case of the form (1) was first shown by Blankenship and Falk (1976), generalised by Reemtsen and Rückmann (1998), and recently by Mitsos (2011). They required that the sets $\mathcal{A}$ and $\mathcal{B}$ in (1) are non-empty and compact, and that the functions Q and R are continuous with respect to all their arguments. Then they showed that the sequence of solutions obtained for a sequence of finite and countable subsets of $\mathcal{B}$ converges to the solution of (1), provided that the minimization and the maximization steps are solved to global optimality. We show in Theorem 2 when the Algorithm 1 solves the problem (7). A discussion on convergence rate of local reduction methods was done by Seidel and Küfer (2020).

Theorem 2. The solution $(\mathbf{q}^*, r^*, \gamma^*)$ obtained from Algorithm 1 for a non-empty and compact set $\mathbb{W}^N \times \mathbb{D}$ converges to the solution of (7) if the set $\mathbb{F}$ such that $(\mathbf{q}, r, \gamma) \in \mathbb{F} \subset \mathbb{R}^{n_q} \times \mathbb{R}^{n_r} \times \mathbb{R}$ is non-empty and compact.

Proof 2. From Theorem 1, we have $\theta := (\mathbf{q}, r, \gamma)$ and $\rho := (\mathbf{w}, d)$, $\mathcal{A} := \mathbb{F}$, $\mathcal{B} := \mathbb{H}$. In (11), we take $Q(\theta) := \gamma$ which

Algorithm 2: Maximization - line 3 in Algorithm 1

Input: Current values of $\mathbf{q}^j$, r^j, γ^j
Output: Worst case scenario $(\mathbf{w}^j, d^j)$ in iteration j

1 Find any $\mathbf{x}^*, \mathbf{u}^*, \mathbf{w}^*, d^*$ that solves:

$$\max_{\mathbf{x},\mathbf{u},\mathbf{w},d} \quad J_N(\mathbf{x},\mathbf{u},\mathbf{w},d) - \gamma^j$$
$$\text{s.t. } (\mathbf{x},\mathbf{u}) = \mathbf{z}(\mathbf{q}^j, r^j, \mathbf{w}, d)$$
$$(\mathbf{w}, d) \in \mathbb{W}^N \times \mathbb{D}$$

2 Set $\mathbb{K} \leftarrow (\mathbf{w}^*, d^*, J_N(\mathbf{x}^*, \mathbf{u}^*, \mathbf{w}^*, d^*) - \gamma^j)$
3 **for** $h = 1, \dots, n_g$ **do**
4 **for** $k = 1, \dots, N-1$ **do**
5 Find any $\mathbf{x}^*, \mathbf{u}^*, \mathbf{w}^*, d^*$ that solves:

$$\max_{\mathbf{x},\mathbf{u},\mathbf{w},d} \quad e_h g_k(x_k, u_k, w_k, d)$$
$$\text{s.t. } (\mathbf{x},\mathbf{u}) = \mathbf{z}(\mathbf{q}^j, r^j, \mathbf{w}, d)$$
$$(\mathbf{w}, d) \in \mathbb{W}^N \times \mathbb{D}$$

  Set $\mathbb{K} \leftarrow \mathbb{K} \cup (\mathbf{w}^*, d^*, e_h g_k(x_k^*, u_k^*, w_k^*, d^*))$
6 **end**
7 **end**
8 Set $v^* \leftarrow \max\{v_3 \mid (v_1, v_2, v_3) \in \mathbb{K}\}$;
9 Choose any $(\mathbf{w}^j, d^j) \in \{(v_1, v_2) \mid (v_1, v_2, v^*) \in \mathbb{K}\}$

is linear and thus continuous. Then we have $R(\theta, \rho) := G(\mathbf{z}(\mathbf{q}, d, \mathbf{w}, d), \mathbf{w}, d)$ which is continuous because both $\max_{h,k} e_h^T g_k(\cdot,\cdot,\cdot,\cdot)$ and $J_N(\cdot,\cdot,\cdot,\cdot)$ are continuous. The proof follows from Lemma 2.2 by Mitsos (2011). □

3.2 Maximization step

The maximization step consists in solving (13) with $\mathbb{H} = \mathbb{W}^N \times \mathbb{D}$. Solving (13) is equivalent to solving $n_g \cdot (N-1)+1$ optimization problems, where n_g denotes the number of elements in the vector function $g(\cdot)$ from constraints in (6). The algorithm is presented in Algorithm 2. Without loss of generality, we assume that the first constraint to include in the maximization problem corresponds to the reformulated objective function (5). A scenario that corresponds to maximal value of this constraint is added to an auxiliary set $\mathbb{K}$. The remaining $n_g \cdot (N-1)$ constraints are included as objectives in the respective maximization problems (line four to eight in Algorithm 2). Note that the problem corresponding to the objective (line 2) and all the problems corresponding to the constraints (line four to eight) can be solved in parallel.

All maximization problems are subject to the same equality constraints capturing the dynamics. This formulation allows us to treat the maximization problems as optimal control problems and preserve the sparsity of the relevant Jacobians and Hessians. We solve the maximization problems as optimal control problems where $\mathbf{q}$, r, and γ are known parameters whereas $\mathbf{w}$ and d are treated as unknown inputs. Thus, the maximization problems can be solved using off-the-shelf optimal control solvers.

Solving (13) with $\mathbb{H} = \mathbb{W}^N \times \mathbb{D}$ corresponds to lines four to eight in Algorithm 2 and can be done by solving a number of finite-dimensional optimization problems in parallel as indicated by Žaković et al. (2003).

4. NUMERICAL RESULTS

We show that the local reduction method described in Section 3 finds scenarios from inside the uncertainty sets and provides robust solutions to optimal control problems with uncertainty in a case study of temperature control in a residential building, adapted from Lian and Jones (2021). The example was implemented in Julia 1.6 (Bezanson et al., 2017) using JuMP 0.21.4 (Dunning et al., 2017). The problems were then solved with Ipopt version 3.12.10 (Thierry and Biegler, 2020). All tests were performed on a laptop with an Intel® Core™ i7-7500U with 16 GB of RAM.

4.1 Example

The example shows a single zone building affected by time-varying internal heat gain, solar radiation, and external temperature. The objective is to keep the internal temperature x_k^{temp} within time-varying bounds:

$$T_{\min} \le x_k^{\text{temp}} \le T_{\max}. \tag{16}$$

During the day, the indoor temperature must be kept above 23 °C and during the night it can drop down to 17 °C. The maximal temperature is the same during the day and night, $T_{\max} = 26\,^\circ\text{C}$. The dynamics are discrete and linear:

$$x_{k+1} = A(d)x_k + B(d)u_k^{\text{sat}} + Ww_k. \tag{17}$$

The states x describe the indoor temperature x^{temp}, wall temperature x^{wall}, and the corridor temperature x^{corr}. The initial condition was chosen as

$$x_0 = [25\,^\circ\text{C}\ \ 24\,^\circ\text{C}\ \ 24\,^\circ\text{C}]^\intercal.$$

The control u represents the amount of heating and cooling delivered to the building. The nominal matrices are taken from Lian and Jones (2021).

There are 14 uncertain parameters affecting the matrices A, B, and the initial condition for the wall and corridor temperatures, $d \in \mathbb{R}^{14}$, $d = [\lambda_{\text{wall}}, \lambda_{\text{corr}}, \delta_{i,j}, \eta_j]_{i,j=1,\dots,3}$. We assume that the wall temperature and the corridor temperature can only be measured approximately. We have $x_0^i = 24 + \lambda_i$, $i = \text{wall}, \text{corr}$, where $\lambda_i \in [-0.5, 0.5]$. We also assume that the matrices $A = [a_{i,j}]$ and $B = [b_j]$, $i, j = 1, 2, 3$ are affected by uncertainty $a_{i,j} \cdot \delta_{i,j}$ and $b_j \cdot \eta_j$ where $\delta_{i,j}, \eta_j$ are uncertain parameters, $\delta_{i,j}, \eta_j \in [0.96, 1.03]$.

The minimal control effort is ensured by the objective function

$$J = \frac{1}{N} \sum_{k=1}^{N} u_k^2.$$

It is assumed that the day starts at 6.00 am and lasts 12 hours. The optimal control problem is solved over a period of 48 hours starting at 6.00 am the first day, with $N = 192$. The three uncertain parameters, internal gain, solar radiation, and external temperature, vary with time within the limits provided by Lian and Jones (2021).

The control variables are parameterised as:

$$u_k = Kx_k^{\text{temp}} + q_k \tag{18}$$

where $r := \text{vec}(K)$ and q_k are decision variables. We include saturation of the control inputs:

$$u_k^{\text{sat}} = \begin{cases} -500\,\text{W} & \text{for } u_k < -500\,\text{W} \\ u_k & \text{for } -500\,\text{W} \leq u_k \leq 1200\,\text{W} \\ 1200\,\text{W} & \text{for } u_k > 1200\,\text{W} \end{cases} \quad (19)$$

The saturation was approximated by a smooth function:

$$u_k^{\text{sat}} = \frac{\beta_0}{\beta_1 + \exp(\beta_2 u_k)} + \beta_3 \quad (20)$$

where β_i are constants. Here $\beta_0 = -5030$, $\beta_1 = 2.937$, $\beta_2 = 0.003$, $\beta_3 = 1207$.

The results obtained from local reduction are then compared with three scenario-based approaches from the literature (Thombre et al., 2021): nominal, with a controller obtained assuming there is no uncertainty ("Nominal"), randomised, with a controller obtained for a number of scenarios chosen from a uniform distribution ("Random"), extreme, with a controller obtained for three cases: nominal, lower bound, and upper bound for all uncertainties ("Nominal+two extreme").

4.2 Overall performance

The local reduction method reduced the number of scenarios to 101. The resulting controller obtained for the interim worst-case scenarios was then validated for 500 random realisations from a uniform distribution of uncertainty. The validation of the controller is shown in top plot in Fig. 1. The black curves stay within the green bounds corresponding to constraints (16). The results suggest that local reduction was able to find a robust solution despite using a local solver for maximizations.

The results also indicate that the local reduction method handles time-varying uncertainty without specifying the scenarios over a shorter time horizon. Algorithm 2 treats a scenario of time-varying uncertainty as a single realization of the uncertainty over the whole horizon, removing the necessity of defining all possible uncertainty realizations on short horizons.

4.3 Comparison with other approaches

Validating the nominal controller with 500 random scenarios shows that the approach based on nominal values leads to violation of constraints (plot 'Nominal' in Fig. 1). The second set of controllers we used was derived using three random cases: five scenarios, 100 scenarios, and 250 scenarios. The results are shown in the plot 'Random' in Fig. 1, with black corresponding to the controller obtained from five scenarios, yellow to the controller with 100 scenarios, and blue to the controller with 250 scenarios. As shown in Fig. 2, in all the cases the controller violated at least one of the bounds (100 scenarios by 0.2 °C, 250 scenarios by 0.1 °C), with the controller based on five scenarios violating both the lower and upper bound (by 1.1 °C). Even though the violation decreased with increasing the number of scenarios, further increasing the number of random scenarios to 600 proved unsuccessful in avoiding the violation. Larger problems could not be solved with the given computer in a reasonable amount of time.

If we were to take only extreme values for every uncertainty and consider all the scenarios, we would need to solve a problem with $2^{14+3\times192}$ scenarios, which is too

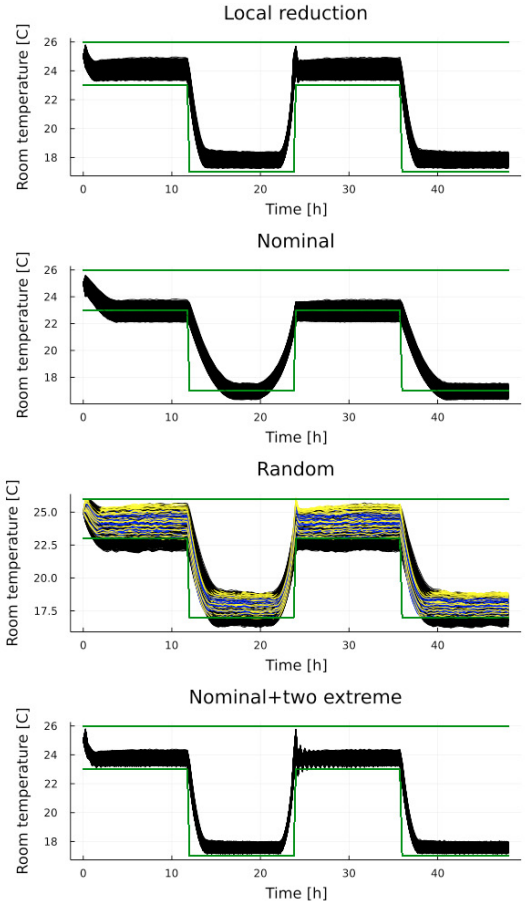


Fig. 1. Comparison with other scenario-based approaches

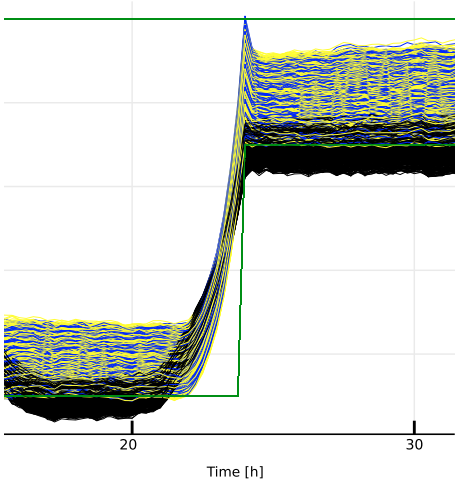


Fig. 2. Zoomed in part showing how random scenarios violate constraints

many. To reduce the number of scenarios, we chose to use the nominal scenario, combined with two extreme scenarios. The extreme scenarios were taken as all uncertainties on their lower or upper bound simultaneously. The results of validating the controller for 500 scenarios are shown in the plot 'Nominal+two extreme' in Fig. 1. The controller based on extreme scenarios was also unable to satisfy the constraints. The black lines after 24 hours cross the green lines so that the lower bound on the temperature is violated (by 0.5° C).

5. CONCLUSIONS

Solving robust nonlinear optimal control problems is challenging, especially if the knowledge about the uncertainty is limited. Scenario-based approaches provide a way of reformulating the optimal control problems as nonlinear optimization problems. However, the choice of scenarios and their number is non-trivial, because the scenarios must ensure robustness while keeping the size of the optimization problem manageable. In particular, the size of the resulting optimization problems increases computationally if time-varying uncertainties are involved. In this paper, we showed that a class of robust optimal control problems is equivalent to semi-infinite optimization problems. We then demonstrated how a local reduction method derived from semi-infinite optimization provides flexibility in choosing scenarios for nonlinear robust control problems. By adding interim worst-case scenarios, the local reduction method enables finding a trade-off between the size of the resulting optimization problem and robustness of the solution to the original optimal control problem.

This paper extended the original local reduction method to optimal control problems with time-varying uncertainty. The performance of our approach was evaluated in a case study with both additive and parametric uncertainty. A comparison with common approaches based on random choice of scenarios and on boundary scenarios indicates that local reduction has potential for solving robust optimal control problems in an efficient way while ensuring robustness.

Future work will include theoretical analysis of the local reduction method as well as numerical improvements needed if local solvers are used. Furthermore, applications in predictive control settings, including analysis of time to find a solution, can also be considered.

REFERENCES

Bezanson, J., Edelman, A., Karpinski, S., and Shah, V.B. (2017). Julia: A fresh approach to numerical computing. *SIAM review*, 59(1), 65–98.

Blankenship, J.W. and Falk, J.E. (1976). Infinitely constrained optimization problems. *Journal of Optimization Theory and Applications*, 19(2), 261–281.

Calafiore, G.C. and Campi, M.C. (2006). The scenario approach to robust control design. *IEEE Transactions on Automatic Control*, 51(5), 742–753.

Campi, M., Carè, A., and Garatti, S. (2021). The scenario approach: A tool at the service of data-driven decision making. *Annual Reviews in Control*, 52, 1–17.

Djelassi, H., Mitsos, A., and Stein, O. (2021). Recent advances in nonconvex semi-infinite programming: Applications and algorithms. *EURO Journal on Computational Optimization*, 100006.

Dunning, I., Huchette, J., and Lubin, M. (2017). JuMP: A modeling language for mathematical optimization. *SIAM Review*, 59(2), 295–320.

Grammatico, S., Zhang, X., Margellos, K., Goulart, P., and Lygeros, J. (2015). A scenario approach for nonconvex control design. *IEEE Transactions on Automatic Control*, 61(2), 334–345.

Hauser, K. (2018). Semi-infinite programming for trajectory optimization with nonconvex obstacles. In *International Workshop on the Algorithmic Foundations of Robotics*, 565–580. Springer.

Hettich, R. (1983). A review of numerical methods for semi-infinite optimization. *Semi-infinite programming and applications*, 158–178.

Hettich, R., Kaplan, A., and Tichatschke, R. (2009). Semi-infinite programming: Numerical methods. *Encyclopedia of Optimization*, 5, 112–117.

Hettich, R. and Kortanek, K.O. (1993). Semi-infinite programming: theory, methods, and applications. *SIAM review*, 35(3), 380–429.

Lian, Y. and Jones, C.N. (2021). From system level synthesis to robust closed-loop data-enabled predictive control. In *2021 60th IEEE Conference on Decision and Control (CDC)*, 1478–1483.

Lucia, S., Andersson, J.A., Brandt, H., Diehl, M., and Engell, S. (2014). Handling uncertainty in economic nonlinear model predictive control: A comparative case study. *Journal of Process Control*, 24(8), 1247–1259.

Mitsos, A. (2011). Global optimization of semi-infinite programs via restriction of the right-hand side. *Optimization*, 60(10-11), 1291–1308.

Mutapcic, A. and Boyd, S. (2009). Cutting-set methods for robust convex optimization with pessimizing oracles. *Optimization Methods & Software*, 24(3), 381–406.

Puschke, J., Djelassi, H., Kleinekorte, J., Hannemann-Tamás, R., and Mitsos, A. (2018). Robust dynamic optimization of batch processes under parametric uncertainty: Utilizing approaches from semi-infinite programs. *Computers & Chemical Engineering*, 116, 253–267.

Reemtsen, R. and Rückmann, J.J. (1998). *Semi-infinite programming*, volume 25. Springer Science & Business Media.

Scokaert, P.O.M. and Mayne, D.Q. (1998). Min-max feedback model predictive control for constrained linear systems. *IEEE Transactions on Automatic Control*, 43(8), 1136–1142.

Seidel, T. and Küfer, K.H. (2020). An adaptive discretization method solving semi-infinite optimization problems with quadratic rate of convergence. *Optimization*, 71(8), 2211–2239.

Thierry, D. and Biegler, L. (2020). The l_1 exact penalty-barrier phase for degenerate nonlinear programming problems in Ipopt.

Thombre, M., Yu, Z.J., Jäschke, J., and Biegler, L.T. (2021). Sensitivity-assisted multistage nonlinear model predictive control: Robustness, stability and computational efficiency. *Computers & Chemical Engineering*, 107269.

Tsoukalas, A., Rustem, B., and Pistikopoulos, E.N. (2008). A global optimization algorithm for generalized semi-infinite, continuous minimax with coupled constraints and bi-level problems. *Journal of Global Optimization*, 44(2), 235–250.

Vuffray, M., Misra, S., and Chertkov, M. (2015). Monotonicity of dissipative flow networks renders robust maximum profit problem tractable: General analysis and application to natural gas flows. In *2015 54th IEEE Conference on Decision and Control (CDC)*. IEEE.

Žaković, S., Rustem, B., and Asprey, S. (2003). A parallel algorithm for semi-infinite programming. *Computational Statistics and Data Analysis*, 44(1-2), 377–390.