Check for updates

Received: 16 March 2023 | Revised: 21 August 2023 | Accepted: 28 September 2023

DOI: 10.1002/rnc.7038

RESEARCH ARTICLE

WILEY

Automatic scenario generation for efficient solution of robust optimal control problems

Marta Zagorowska[1,2] | **Paola Falugi**[2,3] | **Edward O'Dwyer**[4] | **Eric C. Kerrigan**[2,5]

[1]Automatic Control Laboratory, ETH Zörich, Zörich, Switzerland

[2]Department of Electrical and Electronic Engineering, Imperial College London, London, UK

[3]Department of Engineering & Construction, University of East London, London, UK

[4]Department of Chemical Engineering, Imperial College London, London, UK

[5]Department of Aeronautics, Imperial College London, London, UK

Correspondence
Marta Zagorowska, Department of Engineering Cybernetics, NTNU, Trondheim, Norway.
Email: marta.zagorowska@ntnu.no

Funding information
EPSRC, Grant/Award Number: EP/V012053/1

Abstract

Existing methods for nonlinear robust control often use scenario-based approaches to formulate the control problem as large nonlinear optimization problems. The optimization problems are challenging to solve due to their size, especially if the control problems include time-varying uncertainty. This paper draws from local reduction methods used in semi-infinite optimization to solve robust optimal control problems with parametric and time-varying uncertainty. By iteratively adding interim worst-case scenarios to the problem, methods based on local reduction provide a way to manage the total number of scenarios. We show that the local reduction method for optimal control problems consists of solving a series of simplified optimal control problems to find worst-case constraint violations. In particular, we present examples where local reduction methods find worst-case scenarios that are not on the boundary of the uncertainty set. We also provide bounds on the error if local solvers are used. The proposed approach is illustrated with two case studies with parametric and additive time-varying uncertainty. In the first case study, the number of scenarios obtained from local reduction is 101, smaller than in the case when all $2^{14+3\times192}$ extreme scenarios are considered. In the second case study, the number of scenarios obtained from the local reduction is two compared to 512 extreme scenarios. Our approach was able to satisfy the constraints both for parametric uncertainty and time-varying disturbances, whereas approaches from literature either violated the constraints or became computationally expensive.

KEYWORDS

local reduction, optimization, robust control, scenario

1 | INTRODUCTION

Robust optimal control problems are often solved using a scenario-based approach, where each *scenario* corresponds to a separate realization of uncertainty. 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 which is a challenge especially if uncertainty varies with time. This paper

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.
© 2023 The Authors. *International Journal of Robust and Nonlinear Control* published by John Wiley & Sons Ltd.

draws from approaches used in semi-infinite optimisation to solve robust optimal control problems by selecting scenarios in an efficient way.

1.1 | Background

To ensure that the optimization problems resulting from scenario-based approaches to robust control are tractable, the number of scenarios must be limited.[1] Usually, the choice of scenarios is done from experience[2,3] 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 indicated that scenario selection is highly affected by the knowledge about the uncertainty distribution.[4] 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.[5-8] The worst-case scenario in nonlinear systems may lie in the interior of the uncertainty range.[9,10] In this paper we show that the worst-case scenarios may be inside the uncertainty set, even for a linear dynamic system and present a method for choosing potential worst-case scenarios assuming limited knowledge about uncertainty.

Systematic approaches to choosing scenarios for time-varying uncertainty are usually based on creating *scenario trees*.[8,11] In these approaches, a large set of scenarios is chosen at the beginning of the time horizon. The number of elements in the set of scenarios increases combinatorially.[12] However, there is no guarantee that the chosen set of scenarios includes the actual worst-case scenarios.[10] To overcome this drawback, we propose to use a method derived from semi-infinite optimisation to iteratively add new scenarios to the current set, to provide more flexibility in adjusting the set of scenarios and finding worst-case scenarios.

Previous studies provided an in-depth review of semi-infinite optimization methods.[13-17] In particular, it has been indicated[13] that *local reduction* methods allow overcoming the dependence on the initial choice of scenarios. In these methods, the set of scenarios is created iteratively by alternating between solving an optimisation problem with the current set of scenarios and solving interim optimisation problems to find the maximal violation of constraints and extend the set of scenarios.[18] Therefore local reduction methods enable adding scenarios that may not have been considered at the beginning, such as scenarios from the interior of the uncertainty set.

The iterative approach based on alternating between optimization problems has been already used for robust control, which suggests potential usefulness of local reduction. The D-K iteration found in μ-synthesis consists in alternating between the synthesis of an $\mathcal{H}_\infty$ controller and minimization of the singular value for the corresponding controller.[19] However, the D-K iteration method depends on the optimality of each step, which indicates that the local reduction method will also be affected by how the interim optimization problems are solved. In particular, due to the need to solve to global optimality the interim problems,[17] the complexity of local reduction methods prevented their use in robust optimal control problems. An approach based on simulated annealing has been proposed to facilitate finding a global solution.[20] We show that the local reduction method provides good results even if local solvers are used and provide bounds on the solution.

Existing applications of semi-infinite optimization methods in control systems with uncertainty are limited. Semi-infinite optimization methods have been used for optimal control[21] to find optimal trajectories for robotic arms. However, the authors considered only exogenous uncertainty due to obstacles that did not affect the dynamics of the controlled systems. Parametric uncertainty for linear systems was considered[22,23] to apply semi-infinite optimization methods for model identification. However, these works did not consider time-varying uncertainty. Similarly, semi-infinite optimisation methods has been used to solve an optimal control problem with only parametric uncertainty.[10] Time-varying uncertainty was considered in works where local reduction was applied to find the interim worst-case scenarios.[8] However, the authors assumed that at every time step the number of possible scenarios was finite.

1.2 | Contributions

The main contributions of the current work are:

1. Formulation of a local reduction method as a way of automatically generating scenarios in robust nonlinear control problems if time-varying and constant uncertainties are present,
2. Formulation of error bounds on constraint violation if local optimization solvers are used in local reduction,

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

3. Numerical demonstration of local reduction methods for solving nonlinear robust optimal control problems with parametric uncertainty and linear robust optimal control problems with both parametric and time-varying uncertainty.

A preliminary version of the application of local reduction for scenario generation is available.[24]

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 where we compared the results obtained with the scenarios from local reduction to three commonly used approaches: nominal case with no uncertainty, random case with randomly drawn realizations of uncertainty, boundary case with only extreme values. The paper ends with conclusions in Section 5.

2 | PROBLEM FORMULATION

2.1 | Semi-infinite optimization problem

A semi-infinite optimization problem is formulated as:

$$\mathcal{Q}: \quad \min_{\theta \in \mathcal{A}} \quad Q(\theta) \tag{1a}$$

$$\text{subject to} \;\; R(\theta, \rho) \leq 0 \;\; \text{for all} \;\; \rho \in \mathcal{B} \tag{1b}$$

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.[18] 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:

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

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 Reference 18 allows overcoming the non-differentiability of $S(\cdot)$ by sequentially solving (1) with finite subsets of constraints taken from $\mathcal{B}$.

The main challenge in formulating robust optimal control problems as semi-infinite optimization lies in inclusion of system dynamics in the form of equality constraints. In this paper, we show that optimal control problems can be formulated as semi-infinite optimization problems and solved using the local reduction method.[18]

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 $\mathbf{u} := (u_0, \ldots, u_{N-1})$ is generated by a causal dynamic feedback policy $u_k := \pi_k(x_0, \ldots, x_k; q_0, \ldots, q_k, r)$ that is parameterized by $\mathbf{q} := (q_0, q_1, \ldots, q_{N-1}) \in \mathbb{R}^{n_q}$ and $r \in \mathbb{R}^{n_r}$. The state trajectory $\mathbf{x} := (x_0, \ldots, x_N)$. 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 modeled by a suitably-defined choice of f_k, π_k and w_k. In this work, we assume that the structure of the dynamic feedback policy, and hence the parameterization, is already defined.

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

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

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

2.3 | Robust optimal control problem

2.3.1 | 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 $\mathbf{w}$ 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

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

are satisfied for all time instants $k = 0, \ldots, N-1$, 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}.

To ensure that the optimal control problem with the objective (5) and the constraints (6) is well-defined over the horizon N, we introduce the following assumption on the trajectories (4):

Assumption 1. The trajectories (4) are bounded over a finite horizon N:

$$\forall N \in \mathbb{R}, \mathbf{q} \in \mathbb{R}^{n_q}, r \in \mathbb{R}^{n_r}, \mathbf{w} \in \mathbb{W}^N, d \in \mathbb{D}\ \exists \varsigma_z \in \mathbb{R} : \ \|\mathbf{z}(\mathbf{q}, r, \mathbf{w}, d)\|_2 \leq \varsigma_z \tag{7}$$

We note that the boundedness of (4) need not imply stability of the dynamics (3).

2.3.2 | 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:

$$\mathcal{P}_N(\mathbb{H}) : \min_{\substack{\mathbf{q}, r, \\ \mathbf{x}^i, \mathbf{u}^i, \\ i \in \mathbb{J}}} \ \max_{\substack{\mathbf{w}^i, d^i \\ i \in \mathbb{J}}} J_N\left(\mathbf{x}^i, \mathbf{u}^i, \mathbf{w}^i, d^i\right) \tag{8a}$$

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

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

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

$$\mathbb{H} = \bigcup_{i \in \mathbb{J}} \left\{\left(\mathbf{w}^i, d^i\right)\right\}.$$

If $\mathbf{z}(\cdot)$ in (8c) is linear jointly in all arguments, the problem (8) can often be solved using scenario-based methods for robust control,[1,12] 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. Moreover, the set $\mathbb{H}$ is assumed uncountable, which is a common case if the disturbances belong to a polytope.

Theorem 1. *The robust optimal control problem (8) 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 $\mathcal{A} := \mathbb{R}^{n_q} \times \mathbb{R}^{n_r} \times \mathbb{R}$, $\mathcal{B} := \mathbb{H}$.*

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Proof. In contrast to (1a), the objective function in (8a) contains uncertainty. Introducing $\gamma \in \mathbb{R}$, we rewrite (8) as:

$$\mathcal{P}_N(\mathbb{H}): \quad \min_{\gamma \mathbf{q} r, \mathbf{x}^i, \mathbf{u}^i, i \in \mathbb{J}} \quad \gamma \tag{9a}$$

$$\text{s.t. } g_k\left(x_k^i, u_k^i, w_k^i, d^i\right) \leq 0, \forall i \in \mathbb{J}, k = 0, \ldots, N-1 \tag{9b}$$

$$(\mathbf{x}^i, \mathbf{u}^i) = \mathbf{z}(\mathbf{q}, r, \mathbf{w}^i, d^i), \forall i \in \mathbb{J} \tag{9c}$$

$$J_N(\mathbf{x}^i, \mathbf{u}^i, \mathbf{w}^i, d^i) \leq \gamma, \forall i \in \mathbb{J} \tag{9d}$$

The problem (9) has uncertainty exclusively in the constraints. If $card\mathbb{H}$ is finite, then the problem (9) 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 (9) needs to be further reformulated to become (1). Noticing that the constraint (9b) is equivalent to

$$\max_k \; g_k(x_k^i, u_k^i, w_k^i, d^i) \leq 0, \; \forall i \in \mathbb{J}, \tag{10}$$

we introduce

$$G(\mathbf{x}^i, \mathbf{u}^i, \mathbf{w}^i, d^i, \gamma) := \max\left\{ \max_{h,k} e_h^T g_k\left(x_k^i, u_k^i, w_k^i, d^i\right), J_N\left(\mathbf{x}^i, \mathbf{u}^i, \mathbf{w}^i, d^i\right) - \gamma \right\} \tag{11}$$

In (11), e_h is the h^{th} column of an identity matrix $\mathbb{I}_{n_g}$. Using (4) and (11), we can write (9) as:

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

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

The problem (12) is equivalent to

$$\mathcal{P}_N(\mathbb{H}): \quad \min_{\mathbf{q}, r, \gamma} \quad \gamma \tag{13a}$$

$$\text{s.t. } \max_{(\mathbf{w}, d) \in \mathbb{H}} G(\mathbf{z}(\mathbf{q}, r, \mathbf{w}, d), \mathbf{w}, d, \gamma) \leq 0. \tag{13b}$$

Taking $\theta := (\mathbf{q}, r, \gamma)$ and $\rho := (\mathbf{w}, d)$ in (12) (similarly in (13)) we obtain the form of (1) (similarly (2)). ∎

Theorem 1 makes no assumptions on the cardinality of the set $\mathbb{H}$, which is uncountable in general. As a result, problem (12) has an infinite number of constraints, in general, in a similar way to (1). Using the fact that (12) is equivalent to (1), and (8) is equivalent to (12), we note that solving (12) using methods developed for semi-infinite optimization of the form (1) is equivalent to solving the optimal control problem (8). Thus, Theorem 1 allows one to solve the optimal control problem (8) as a semi-infinite optimization problem of the form (1) using local reduction.[18]

2.3.3 | Robust solution

We now introduce definitions of robust solutions that we are going to use in the remainder of the paper. First, we notice that the left-hand side of (13b) is equivalent to:

$$G_{\max}(\mathbf{q}, r, \gamma, \mathbb{H}) := \max_{\substack{(\mathbf{w}, d) \in \mathbb{H} \\ (\mathbf{x}, \mathbf{u}) = \mathbf{z}(\mathbf{q}, r, \mathbf{w}, d)}} G(\mathbf{x}, \mathbf{u}, \mathbf{w}, d, \gamma) \tag{14}$$

Using (14) allows us to introduce the necessary definitions.

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Definition 1 (Scenario). A *scenario* is a realization of the uncertainty $(\mathbf{w}, d) \in \mathbb{W}^N \times \mathbb{D}$.

Definition 2 (Worst-case scenario). A *worst-case scenario* for a given $(\mathbf{q}, r, \gamma)$ is a realization of the uncertainty $(\mathbf{w}^*, d^*) \in \mathbb{W}^N \times \mathbb{D}$ that maximises constraint violation:

$$(\mathbf{x}^*, \mathbf{u}^*, \mathbf{w}^*, d^*) \in \underset{\substack{(\mathbf{w},d)\in\mathbb{W}^N\times\mathbb{D} \\ (\mathbf{x},\mathbf{u})=\mathbf{z}(\mathbf{q},r,\mathbf{w},d)}}{\arg\max} \; G(\mathbf{x}, \mathbf{u}, \mathbf{w}, d, \gamma) \tag{15}$$

Definition 3 (Robust solution). A triple $(\mathbf{q}^*, r^*, \gamma^*)$ is called a *robust solution* if $G_{\max}(\mathbf{q}^*, r^*, \gamma^*, \mathbf{w}, d) \leq 0$ for all $(\mathbf{w}, d) \in \mathbb{W}^N \times \mathbb{D}$.

Definition 4 (Solution robust to s scenarios). A triple $(\mathbf{q}^*, r^*, \gamma^*)$ is called a *robust solution to the s scenarios in* $\mathbb{H}$ if $G_{\max}(\mathbf{q}^*, r^*, \gamma^*, \mathbf{w}, d) \leq 0$ for all $(\mathbf{w}, d) \in \mathbb{H}$ and $card\mathbb{H} = s$.

3 | LOCAL REDUCTION FOR OPTIMAL CONTROL

We will now extend the local reduction methods[18] to robust nonlinear optimal control.

3.1 | Algorithm

The local reduction method[18] consists in iteratively solving finite-dimensional optimization problems. We use the local reduction methods for the problem (9) or (12) by iteratively solving optimal control problems parametrized by scenarios. The iterations alternate between solving minimization and maximization steps that will now be described.

3.2 | Minimization step

The local reduction algorithm for robust optimal control is shown in Algorithm 1. The proposed algorithm in iteration j solves an optimal control problem of the form (9) or (12) assuming that the number of scenarios $card\mathbb{H}_j$ 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

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 $\mathbb{H}^*$

Set $\mathbf{q}^1 \leftarrow \mathbf{q}$, $r^1 \leftarrow r$, $\gamma^1 \leftarrow \gamma$, $j \leftarrow 1$
repeat
  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 (14) with $\mathbb{H} = \mathbb{W}^N \times \mathbb{D}$.
  if $G_{\max}(\mathbf{q}^j, r^j, \gamma^j, \mathbb{W}^N \times \mathbb{D}) \leq 0$ **then**

$$\mathbb{H}_{j+1} \leftarrow \mathbb{H}_j$$

  else
    Add new scenario

$$\mathbb{H}_{j+1} \leftarrow \mathbb{H}_j \cup \{(\mathbf{w}^j, d^j)\} \tag{16}$$

    Find a $(\mathbf{q}^{j+1}, r^{j+1}, \gamma^{j+1})$ that solves $\mathcal{P}_N(\mathbb{H}_{j+1})$ using (9) or (12).
  end
  Set $j \leftarrow j + 1$
  Set $(\mathbf{q}^*, r^*, \gamma^*) \leftarrow (\mathbf{q}^j, r^j, \gamma^j)$ and $\mathbb{H}^* \leftarrow \mathbb{H}_j$.
until $\operatorname{card} \mathbb{H}_j = \operatorname{card} \mathbb{H}_{j-1}$;

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

solving (12) for one scenario, that is, card $\mathbb{H}_1 = 1$. Alternatively, the initial guess can be found by solving (12) for a small number of scenarios, obtained for example from a coarse discretization of the uncertainty set.[25]

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 (11). 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, that is, card $\mathbb{H}_j$ = 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.[26]

3.3 | Maximization step

The maximization step consists in solving (14) with $\mathbb{H} = \mathbb{W}^N \times \mathbb{D}$. Solving (14) 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 (lines 4 to 8 in Algorithm 2). Note that the problem corresponding to the objective (line 2) and all the problems corresponding to the constraints (lines 4 to 8) can be solved in parallel.

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:

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

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:

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

| | | Set $\mathbb{K} \leftarrow \mathbb{K} \cup (\mathbf{w}^*, d^*, e_h^T g_k(x_k^*, u_k^*, w_k^*, d^*))$

6 | **end**

7 **end**

8 Set $v^* \leftarrow \max\{v_3 \mid (v_1, v_2, v_3) \in \mathbb{K}\}$;

9 Choose any $(\mathbf{w}^j, d^j) \in \{(v_1, v_2) \mid (v_1, v_2, v^*) \in \mathbb{K}\}$

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 any off-the-shelf solver for optimal control problems.

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Solving (14) with $\mathbb{H} = \mathbb{W}^N \times \mathbb{D}$ corresponds to lines 4 to 8 in Algorithm 2 and can be done by solving a number of finite-dimensional optimization problems in parallel.[23]

3.4 | Analysis

3.4.1 | Convergence of Algorithm 1

The convergence of local reduction method in the case of the form (1) was shown in several previous works.[18,27,28] The authors 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. 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). A discussion on convergence rate of local reduction methods also follows.[16] We show in Theorem 2 when the Algorithm 1 solves problem (8).

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 (8) if the set* $\mathbb{F} \subset \mathbb{R}^{n_q} \times \mathbb{R}^{n_r} \times \mathbb{R}$ *such that* $G : \mathbb{F} \times \mathbb{W}^N \times \mathbb{D} \to \mathbb{R}^{n_g}$ *is non-empty and compact*.

Proof. From Theorem 1, we have $\theta := (\mathbf{q}, r, \gamma)$ and $\rho := (\mathbf{w}, d)$, $\mathcal{A} := \mathbb{F}$, $\mathcal{B} := \mathbb{H}$. In (12), we take $Q(\theta) := \gamma$ which is linear and thus continuous. Then we have $R(\theta, \rho) := G(\mathbf{q}, r, \gamma, \mathbf{w}, d)$ 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 Lemma 2.2 from a previous study.[28] ∎

Theorem 2 requires the constraints in G to be defined over a compact set $\mathbb{F} \times \mathbb{W}^N \times \mathbb{D}$. As a direct consequence, we obtain Remark 1.

Remark 1 (Boundedness of constraints). $G_{\max}$ is bounded on $\mathbb{F} \times \mathbb{W}^N \times \mathbb{D}$, that is,

$$\exists m, M \in \mathbb{R} : \forall (\mathbf{q}, r, \gamma, \mathbf{w}, d) \in \mathbb{F} \times \mathbb{W}^N \times \mathbb{D} : \; m \leq G_{\max}(\mathbf{q}, r, \gamma, \mathbf{w}, d) \leq M \tag{17}$$

Proof. The proof follows directly from the extreme value theorem because $G_{\max}(\mathbf{q}, r, \gamma, \mathbf{w}, d)$ is continuous over a compact set $\mathbb{F} \times \mathbb{W}^N \times \mathbb{D}$. ∎

Boundedness of G ensures that the maximization step presented in Algorithm 2 is well-posed. Section 3.5.1 will further demonstrate the impact of boundedness on the solution obtained from Algorithm 1.

We also note that similarly to Assumption 1, the requirement of boundedness of G need not imply stability of the dynamics (3). In particular, the method can be used for solving finite horizon optimal control problems with unstable linear dynamics affected by uncertainty, as will be demonstrated in Section 4.

3.4.2 | Constraint dropping

The method presented in Algorithm 1 assumes that the cardinality of the sets $\mathbb{H}_j$ is increasing with j, that is, $\mathbb{H}_j \subset \mathbb{H}_{j+1}$ for all j. The increasing cardinality corresponds to an increase in the size of the optimization problem in line 9 in Algorithm 1. The authors[18] provide additional convexity conditions allowing one to drop elements from the set $\mathbb{H}_j$. In particular, they require (1a) to be strictly convex with respect to θ and $R(\theta)$ to be convex with respect to θ for $\rho \in \mathcal{B}$. Following Theorem 1, the conditions provided in previous works[18] correspond to strict convexity of (12a), and convexity of $G(\mathbf{z}(\cdot,\cdot,\mathbf{w},d),\mathbf{w},d,\cdot)$ for any $(\mathbf{w}, d) \in \mathbb{W}^N \times \mathbb{D}$. In the current work, we do not assume convexity of (11) and (12a) is only convex, not strictly convex, so dropping constraints from the set $\mathbb{H}_j$ does not guarantee convergence of the local reduction algorithm. To enable constraint dropping, we consider a special case of problem (8) where the cost is independent of the uncertainty:

$$\mathcal{P}_N(\mathbb{H}) : \quad \min_{\substack{\mathbf{q}, r \\ \mathbf{x}^i, \mathbf{u}^i, \\ \forall i \in \mathbb{J}}} \quad J_N(\mathbf{q}, r) \tag{18a}$$

$$\text{s.t. } g_k(x_k^i, u_k^i, w_k^i, d^i) \leq 0, \, \forall i \in \mathbb{J}, k = 0, \, \dots \, , N-1 \tag{18b}$$

$$(\mathbf{x}^i, \mathbf{u}^i) = \mathbf{z}(\mathbf{q}, r, \mathbf{w}^i, d^i), \, \forall i \in \mathbb{J} \tag{18c}$$

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Then we can adjust Algorithm 1 to Problem (18) to enable dropping constraints. First let us rewrite (11) as:

$$G(\mathbf{x}^i, \mathbf{u}^i, \mathbf{w}^i, d^i) := \max_{h,k} e_h^T g_k(x_k^i, u_k^i, w_k^i, d^i) \tag{19}$$

and (14) as:

$$G_{\max,\text{new}}(\mathbf{q}, r, \mathbb{H}) := \max_{\substack{(\mathbf{w},d)\in\mathbb{H} \\ (\mathbf{x},\mathbf{u})=\mathbf{z}(\mathbf{q},r,\mathbf{w},d)}} G(\mathbf{x}, \mathbf{u}, \mathbf{w}, d) \tag{20}$$

where $(\mathbf{q}, r) \in \mathbb{F}_{\text{new}} \subset \mathbb{R}^{n_q} \times \mathbb{R}^{n_r}$ are found in a subset of the whole search space.

Theorem 3 (Adapted[18]). *If $J_N(\cdot,\cdot)$ is strictly convex, (19) is convex w.r.t. x_k^i and u_k^i for any $(w_k^i, d^i) \in \mathbb{W} \times \mathbb{D}$, and $\mathbb{F}_{\text{new}}$ is convex, then (16) can be replaced by*:

$$\mathbb{H}_{j+1} \leftarrow \mathbb{H}_j \cup \left(\mathbf{w}^j, d^j\right) \setminus \mathbb{Z}_j \tag{21}$$

where

$$\mathbb{Z}_j := \left\{(\mathbf{w}, d) \in \mathbb{H}_j | G_{\max,\text{new}}\left(\mathbf{q}^j, r^j, \mathbb{H}_j\right) < 0\right\} \tag{22}$$

and the solution of the modified algorithm will converge to the solution of (18).

Algorithm 3: Inexact local reduction method

Input: Initial guess for $\mathbf{q}$, r, γ, $\mathbb{H}_1 \neq \emptyset$, and the tolerances ϵ_w and ϵ_d
Output: Optimal $\mathbf{q}^*$, r^*, γ^*, set of scenarios $\mathbb{H}^*$ that includes the worst-case

Set $\mathbf{q}^1 \leftarrow \mathbf{q}$, $r^1 \leftarrow r$, $\gamma^1 \leftarrow \gamma$, $j \leftarrow 1$
repeat
  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 (14) with $\mathbb{H} = \mathbb{W}^N \times \mathbb{D}$.
  if $G_{\max}(\mathbf{q}^j, r^j, \gamma^j, \mathbb{W}^N \times \mathbb{D}) \leq 0$ **then**
    $$\mathbb{H}_{j+1} \leftarrow \mathbb{H}_j \tag{23}$$
  else
    forall $(\mathbf{w}, d) \in \mathbb{H}_j$ **do**
      if $\frac{1}{N}\|\mathbf{w}^j - \mathbf{w}\|_2^2 > \epsilon_w$ **or** $\|d^j - d\|_2^2 > \epsilon_d$ **then**
        $$\mathbb{H}_{j+1} \leftarrow \mathbb{H}_j \cup (\mathbf{w}^j, d^j) \tag{24}$$
      else
        $$\mathbb{H}_{j+1} \leftarrow \mathbb{H}_j \tag{25}$$
      end
    end
    Find a $(\mathbf{q}^{j+1}, r^{j+1}, \gamma^{j+1})$ that solves $\mathcal{P}_N(\mathbb{H}_{j+1})$ using (9) or (12).
  end
  Set $(\mathbf{q}^*, r^*, \gamma^*) \leftarrow (\mathbf{q}^j, r^j, \gamma^j)$ and $\mathbb{H}^* \leftarrow \mathbb{H}_j$.
  Set $j \leftarrow j + 1$.
until $\operatorname{card} \mathbb{H}_j = \operatorname{card} \mathbb{H}_{j-1}$;

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Proof. Taking $\theta := (\mathbf{q}, r)$, $\rho := (\mathbf{w}, d)$, $Q(\theta) := J_N(\mathbf{q}, r)$, $R(\theta, \rho) := G(\mathbf{z}(\mathbf{q}, r, \mathbf{w}, d), \mathbf{w}, d)$, $\mathcal{A} := \mathbb{F}_{\text{new}} \subset \mathbb{R}^{n_q} \times \mathbb{R}^{n_r}$ in (18) we obtain the form of (1). Then the proof follows from theorems 2.2 to 2.4 in Reference 18. ∎

3.5 | Inexact local reduction

To simplify the exact local reduction from Algorithm 1 we propose an inexact formulation of the algorithm focusing on numerical properties of the solvers used for the optimal control problems. The exact algorithm for local reduction presented in Algorithm 1 assumes that the maximization step finds the global solutions to the maximization problem and only one scenario obtained in this step is then added to the scenario set. These assumptions are often difficult to satisfy. In practice, there are two possible cases:

1. The maximization step in a given iteration has multiple solutions in general, but only a limited number is used,
2. The maximization step is solved approximately.

In particular, we will focus on analyzing the case when local solvers are used.

If a global solver is used, but only a limited number of scenarios is added, the local reduction algorithm needs more iterations to find a solution than in the case of adding all the scenarios.[5] Therefore there exists a trade-off between the speed of convergence of the local reduction method and the size of the problem solved in the minimization step. We show the impact of approximate solutions by considering *similarity of scenarios*, that is, when the interim worst-case scenarios from line 3 in Algorithm 1 are considered similar.

Definition 5 (Similar scenarios). Let $(\mathbf{w}^1, d^1)$ and $(\mathbf{w}^2, d^2)$ be two scenarios and let $\epsilon_w \geq 0$ and $\epsilon_d \geq 0$ be fixed parameters. The two scenarios are similar if

$$\frac{1}{N}\|\mathbf{w}^1 - \mathbf{w}^2\|_2^2 \leq \epsilon_w \tag{26}$$

and

$$\|d^1 - d^2\|_2^2 \leq \epsilon_d. \tag{27}$$

Using Definition 5, we modify line 7 in Algorithm 1 so that the scenario $(\mathbf{w}^j, d^j)$ in iteration j is added to the current set of scenarios if it is not similar to any of the scenarios in $\mathbb{H}_j$. Algorithm 3 summarizes the inexact local reduction method with the evaluation of when the scenarios are similar.

3.5.1 | Impact of similarity of scenarios

The algorithm provides a solution that is robust to s scenarios in the sense of Definition 4, where $s = \text{card}\ \mathbb{H}^*$. From Remark 1 and (11), we already have that for any two scenarios $(\mathbf{w}^1, d^1)$, $(\mathbf{w}^2, d^2)$, the constraints g_k are bounded and hence:

$$\|g_k(z_k^1, w_k^1, d_k^1) - g_k(z_k^2, w_k^2, d_k^2)\|_2 \leq \|g_k(z_k^1, w_k^1, d_k^1)\|_2 + \|g_k(z_k^2, w_k^2, d_k^2)\|_2 \leq 2M. \tag{28}$$

A tighter bound can be obtained if we use Assumption 1 and note that for a chosen parameterization $\mathbf{q}^*, r^*, \gamma^*$ the trajectories $\mathbf{z}$ are BIBO-stable w.r.t. disturbances, that is,

$$\forall \varsigma_w > 0 : \|(\mathbf{w}, d)\|_2 \leq \varsigma_w \Rightarrow \exists \varsigma_z > 0 : \|\mathbf{z}\|_2 \leq \varsigma_z \tag{29}$$

Theorem 4 uses (29) to show the impact of the similarity of scenarios on the constraint satisfaction.

Theorem 4. *Let* $(\mathbf{w}^1, d^1)$, $(\mathbf{w}^2, d^2)$ *be two identical scenarios with* $\epsilon_w = \frac{1}{N}\epsilon_w^*$ *and* $\epsilon_d = \epsilon_d^*$, *and* $\|(\mathbf{w}^i, d^i)\|_2 \leq \varsigma_w$ *for* $i = 1, 2$. *Let* $\mathbf{q}^*, r^*, \gamma^*$ *be the solution of (8) obtained for* $(\mathbf{w}^1, d^1)$ *and let* $\mathbf{z}^1 = \mathbf{z}(\mathbf{q}^*, r^*, \mathbf{w}^1, d^1)$,

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

$\mathbf{z}^2 = \mathbf{z}(\mathbf{q}^*, r^*, \mathbf{w}^2, d^2)$. *Then the constraint violation for* $(\mathbf{w}^2, d^2)$ *is bounded and*

$$\|g_k\left(z_k^1, w_k^1, d_k^1\right) - g_k\left(z_k^2, w_k^2, d_k^2\right)\|_2^2 \leq L^2\left(\epsilon_w^* + \epsilon_d^* + 4\varsigma_z^2\right) \tag{30}$$

where z_k^i *is the trajectory* $\mathbf{z}^i$ *at time* k, $i = 1, 2$, *and* L *is a local Lipschitz constant for a given* k.

Proof. We have

$$\|\mathbf{w}^1 - \mathbf{w}^2\|_2^2 \leq \epsilon_w^* \tag{31}$$

and

$$\|d^1 - d^2\|_2^2 \leq \epsilon_d^*. \tag{32}$$

We can write (31) as:

$$\begin{aligned}\|\mathbf{w}^1 - \mathbf{w}^2\|_2^2 &= \|\left(w_1^1 - w_1^2, \ldots, w_N^1 - w_N^2\right)\|_2^2 \\ &= \left\|\sum_{k=1}^{N} e_k(w_k^1 - w_k^2)\right\|_2^2 \\ &\leq \epsilon_w^*\end{aligned} \tag{33}$$

where e_k is the k^{th} row of an identity matrix $\mathbb{I}_N$. Without loss of generality, we can assume that $w_k^1 - w_k^2 \neq 0$ for $k = 1, \ldots, N$. Then from orthogonality of the set $\left\{e_k\left(w_k^1 - w_k^2\right)\right\}_{k=1,\ldots,N}$, and using (31), we get from the Pythagorean theorem:

$$\epsilon_w^* \geq \left\|\sum_{k=1}^{N} e_k\left(w_k^1 - w_k^2\right)\right\|_2^2 = \sum_{k=1}^{N}\|e_k\left(w_k^1 - w_k^2\right)\|_2^2. \tag{34}$$

Taking into account that $\|e_k\left(w_k^1 - w_k^2\right)\|_2^2 \geq 0$ for all k and $\|e_k\left(w_k^1 - w_k^2\right)\|_2^2 = \|\left(w_k^1 - w_k^2\right)\|_2^2$ in (34), we get:

$$\|w_k^1 - w_k^2\|_2^2 \leq \epsilon_w^* \tag{35}$$

for all $k = 1, \ldots, N$. Summing up (32) and (35) gives:

$$\|w_k^1 - w_k^2\|_2^2 + \|d^1 - d^2\|_2^2 \leq \epsilon_d^* + \epsilon_w^* \tag{36}$$

Then we have:

$$\|(w_k^1 - w_k^2, 0)\|_2^2 + \|(0, d^1 - d^2)\|_2^2 = \|\left(w_k^1 - w_k^2, d^1 - d^2\right)\|_2^2 \tag{37}$$

where 0 denotes the origin of $\mathbb{R}^{n_d}$ and we used the Pythagorean theorem in $\mathbb{R}^{n_d+1}$. Then we have:

$$\|\left(w_k^1 - w_k^2, d^1 - d^2\right)\|_2^2 = \|\left(0, w_k^1 - w_k^2, d^1 - d^2\right)\|_2^2 \tag{38}$$

where 0 denotes the origin of $\mathbb{R}^{n_x+n_u}$.

At the same time, from the assumption that g_k is continuously differentiable with respect to all its arguments, we get the local Lipschitz condition [Chap. 2]:[29]

$$\|g_k\left(z_k^1, w_k^1, d^1\right) - g_k\left(z_k^2, w_k^2, d^2\right)\|_2 \leq L\|\left(z_k^1, w_k^1, d^1\right) - \left(z_k^2, w_k^2, d^2\right)\|_2 \tag{39}$$

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

where L is the local Lipschitz constant of g_k. Taking a square in (39) and using (38), we obtain:

$$\begin{aligned}\|g_k(z_k^1, w_k^1, d^1) - g_k(z_k^2, w_k^2, d^2)\|_2^2 &\leq L^2\|(z_k^1, w_k^1, d^1) - (z_k^2, w_k^2, d^2)\|_2^2 \\ &= L^2\|(z_k^1 - z_k^2, 0, 0) + (0, w_k^1 - w_k^2, d^1 - d^2)\|_2^2 \\ &= L^2\left(\|z_k^1 - z_k^2\|^2 + \|(w_k^1 - w_k^2, d^1 - d^2)\|_2^2\right) \\ &\leq L^2\left(\left(\|z_k^1\| + \|z_k^2\|\right)^2 + \|(w_k^1 - w_k^2, d^1 - d^2)\|_2^2\right) \\ &\leq L^2\left(\epsilon_d^* + \epsilon_w^* + 4\varsigma_z^2\right) \end{aligned} \tag{40}$$

which concludes the proof. ∎

In a similar way the proof can be done for the constraint from (9d) because the cost J_N is also continuously differentiable.

Theorem 4 shows that the satisfaction of constraints depends on:

1. The local Lipschitz constant of the constraints,
2. The response of the system to disturbances,
3. The choice of similarity of scenarios.

The local Lipschitz constant and the response to disturbances are inherent to the system. We note from Theorem 4 that the assumption about having a global solution to the maximization problem in Algorithm 1 is crucial to ensure no constraint violation. Even if the maximization step is solved exactly and $\epsilon_w = \epsilon_d = 0$, the constraint violation in (30) is defined by $4L^2\varsigma_z^2$. At the same time, the choice of parameters in Definition 5 affects the constraints violations.

We also note that (29) makes no assumptions on ς_z. For unstable systems, ς_z may be large, thus making the bound in Theorem 4 uninformative. However, the value of ς_z depends also on the chosen control policy π_k which can be used to modify the control invariant set[30] and thus tighten the bound.

Furthermore, the impact of the similarity of scenarios provides information about the solution if local, instead of global, optimization solvers are used to solve the maximization problems. To make this precise, let $(\mathbf{w}^1, d^1)$ in Theorem 4 be a local solution from Algorithm 2, and let $(\mathbf{w}^2, d^2)$ be a global solution. Taking ϵ_w^* and ϵ_z^* such that $\|\mathbf{w}^1 - \mathbf{w}^2\|_2^2 \leq \epsilon_w^*$, $\|d^1 - d^2\|_2^2 \leq \epsilon_d^*$, and $\|(\mathbf{w}^i, d^i)\|_2 \leq \varsigma_w$ for $i = 1, 2$, the maximal constraint violation is bounded by (30).

As a direct consequence of Theorem 4 we obtain the following result:

Theorem 5. *Let us assume that Algorithm 3 finished with scenario set* $\mathbb{H}^*$, card $\mathbb{H}^* = H$. *The constraint violation for scenario* $(\mathbf{w}^*, d^*) \notin \mathbb{H}^*$ *is bounded by* (30) *with* $\epsilon_w^* = \max_{i=1,\dots,H}\|\mathbf{w}^i - \mathbf{w}^*\|_2^2$ *and* $\epsilon_d^* = \max_{i=1,\dots,H}\|d^i - d^*\|_2^2$.

Proof. From Theorem 4 we have that for all $i = 1, \dots, H$:

$$\|g_k(z_k^i, w_k^i, d^i) - g_k(z_k^*, w_k^*, d^*)\|_2^2 \leq L^2\left(\epsilon_{w,i}^* + \epsilon_{d,i}^* + 4\varsigma_z^2\right) \tag{41}$$

where we assumed that $\|\mathbf{w}^i - \mathbf{w}^*\|_2^2 \leq \epsilon_{w,i}^*$ and $\|d^i - d^*\|_2^2 \leq \epsilon_{d,i}^*$. Thus, we have:

$$\begin{aligned}\|g_k(z_k^i, w_k^i, d^i) - g_k(z_k^*, w_k^*, d^*)\|_2^2 &\leq \max_{i=1,\dots,H} L^2\left(\epsilon_{w,i}^* + \epsilon_{d,i}^* + 4\varsigma_z^2\right) \\ &= L^2\left(\max_{i=1,\dots,H}\epsilon_{w,i}^* + \max_{i=1,\dots,H}\epsilon_{d,i}^* + 4\varsigma_z^2\right) \end{aligned} \tag{42}$$

Taking in Theorem 4

$$\epsilon_w^* = \max_{i=1,\dots,H}\epsilon_{w,i}^* = \max_{i=1,\dots,H}\|\mathbf{w}^i - \mathbf{w}^*\|_2^2$$

and

$$\epsilon_d^* = \max_{i=1,\dots,H}\epsilon_{d,i}^* = \max_{i=1,\dots,H}\|d^i - d^*\|_2^2$$

concludes the proof. ∎

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

3.5.2 | Impact of number of scenarios

The bounds obtained in Theorem 5 allow inference about constraint violation in a practical implementation of Algorithm 3. In particular, we will now analyze how fixing the number of scenarios H affects the constraint violation. For simplicity, we focus on the constraint from (9b) but an analogous reasoning can be done for (9d). Let us assume that the function g_k from (9b) is such that:

$$g_k : \mathbb{R}^{n_x} \times \mathbb{R}^{n_u} \times \mathbb{R}^{n_w} \times \mathbb{R}^{n_d} \to \mathbb{Q}_k \subset \mathbb{R}^{n_g} \tag{43}$$

where $\mathbb{Q}_k$ is Lebesgue-measurable. The Lebesgue measure of the set will be called its *n-dimensional volume* [Chap. 21][31] and denoted Vol $\mathbb{Q}_k = \Omega_k$. We assume that the values of g_k are from a uniform distribution over $\mathbb{Q}_k$ for any realization of uncertainty. Let us also introduce for every scenario $(\mathbf{w}^i, d^i) \in \mathbb{H}^*$ the following subsets:

$$\mathbb{S}^i_{k,\delta} := \left\{\bar{g}_k \in \mathbb{Q}_k : \|\bar{g}_k - g_k\left(z^i_k, w^i_k, d^i\right)\|_2 \leq \delta\right\} \tag{44}$$

where $\delta \geq 0$ is a constant such that $\mathbb{S}^i_{k,\delta} \subseteq \mathbb{Q}_k$. Then we can introduce

$$\mathbb{S}_{k,\delta} := \bigcup_i^{H_q} \mathbb{S}^i_{k,\delta} \tag{45}$$

such that $\mathbb{S}^i_{k,\delta} \neq \mathbb{S}^j_{k,\delta}$ for $i,j = 1, \ldots, H_q, i \neq j$. The set from (45) collects H_q sets $\mathbb{S}^i_{k,\delta}$ corresponding to distinct values of constraints $g_k\left(z^i_k, w^i_k, d^i\right)$ obtained from scenarios $\left(w^i_k, d^i\right)$ and trajectories z^i_k. We have that $H_\delta \leq H$, with strict inequality if at least two different realizations of scenarios $\left(w^i_k, d^i\right)$, $\left(w^j_k, d^j\right)$ and the corresponding trajectories $z^i_k, z^j_k, i \neq j, i,j = 1, \ldots, H$, lead to the same value of the constraint, that is, $g_k\left(z^i_k, w^i_k, d^i\right) = g_k\left(z^j_k, w^j_k, d^j\right)$. The n-dimensional volume of $\mathbb{S}_{k,\delta}$ is now:

$$\text{Vol}\, \mathbb{S}_{k,\delta} = \text{Vol} \bigcup_i^{H_q} \mathbb{S}^i_{k,\delta} \tag{46}$$

From countable subadditivity of Lebesgue measure,[32] we get Vol $\bigcup_i \mathbb{S}^i_{k,\delta} \leq \sum_i$ Vol $\mathbb{S}^i_{k,\delta}$, and then:

$$\text{Vol}\, \mathbb{S}^1_{k,\delta} \leq \text{Vol}\, \mathbb{S}_{k,\delta} \leq \sum_i^{H_q} \text{Vol}\, \mathbb{S}^i_{k,\delta} \tag{47}$$

Noticing that $\mathbb{S}^i_{k,\delta}$ is a ball in $\mathbb{R}^{n_g}$ with radius δ, we have [p. 135]:[33]

$$\text{Vol}\, \mathbb{S}^i_{k,\delta} = \frac{\pi^{\frac{n_g}{2}} \delta^{n_g}}{\Gamma\left(\frac{n_g}{2} + 1\right)} \tag{48}$$

where Γ is the gamma function, and then:

$$\frac{\pi^{\frac{n_g}{2}} \delta^{n_g}}{\Gamma\left(\frac{n_g}{2} + 1\right)} \leq \text{Vol}\, \mathbb{S}_{k,\delta} \leq \frac{H_\delta \pi^{\frac{n_g}{2}} \delta^{n_g}}{\Gamma\left(\frac{n_g}{2} + 1\right)}. \tag{49}$$

We can now state:

Theorem 6. *Let us assume that Algorithm 3 is used to solve (8) where g_k satisfies (43) for all $k = 0, \ldots, N-1$ with* Vol $\mathbb{Q}_k = \Omega_k$, *and the values of g_k are uniformly distributed over $\mathbb{Q}_k$. Let us assume that Algorithm 3 finished with scenario set $\mathbb{H}^*$, card $\mathbb{H}^* = H$. Let us assume that a threshold $\delta \in \mathbb{R}$ has been chosen so that the sets $\mathbb{S}^i_{k,\delta}$ from (44) satisfy $\mathbb{S}^i_{k,\delta} \subset \mathbb{Q}_k$ for all $k = 0, \ldots, N-1, i = 1, \ldots, H_q$. The probability that the constraint violation*

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

exceeds the threshold δ for scenario $(\mathbf{w}^*, d^*) \notin \mathbb{H}^*$ such that $g_k(z_k^*, w_k^*, d^*) = \overline{g}_k^*$ is bounded as follows:

$$1 - \frac{\pi^{\frac{n_g}{2}} \delta^{n_g}}{\Omega_k \Gamma\left(\frac{n_g}{2}+1\right)} \geq P\left(\overline{g}_k^* \in \mathbb{S}_{k,\delta}^c\right) \geq 1 - \frac{H_\delta \pi^{\frac{n_g}{2}} \delta^{n_g}}{\Omega_k \Gamma\left(\frac{n_g}{2}+1\right)} \tag{50}$$

where $\mathbb{S}_{k,\delta}^c$ is the complement of the set $\mathbb{S}_{k,\delta}$, that is, $\mathbb{S}_{k,\delta}^c := \mathbb{Q}_k \setminus \mathbb{S}_{k,\delta}$.

Proof. Dividing by $\Omega_k > 0$ all the terms in (49) we obtain

$$\frac{\pi^{\frac{n_g}{2}} \delta^{n_g}}{\Omega_k \Gamma\left(\frac{n_g}{2}+1\right)} \leq P\left(\overline{g}_k^* \in \mathbb{S}_{k,\delta}\right) \leq \frac{H_\delta \pi^{\frac{n_g}{2}} \delta^{n_g}}{\Omega_k \Gamma\left(\frac{n_g}{2}+1\right)} \tag{51}$$

because $P\left(\overline{g}_k^* \in \mathbb{S}_{k,\delta}\right) := \frac{\text{Vol } \mathbb{S}_{k,\delta}}{\Omega_k}$. Substituting $P\left(\overline{g}_k^* \in \mathbb{S}_{k,\delta}\right) = 1 - P\left(\overline{g}_k^* \in \mathbb{S}_{k,\delta}^c\right)$ into (51) and rearranging, we obtain (50). ∎

Theorem 6 combines the number of scenarios H_δ obtained from local reduction with the threshold for constraint violation δ. Assuming that g has been normalized so that $\Omega_k = 1$, we rewrite (50) as:

$$P\left(\overline{g}_k^* \in \mathbb{S}_{k,\delta}^c\right) \geq 1 - \kappa H_\delta \delta^{n_g} \tag{52}$$

where $\kappa := \frac{\pi^{\frac{n_g}{2}}}{\Gamma\left(\frac{n_g}{2}+1\right)}$ is a constant depending on the number of constraints n_g. We see in (52) that there is a trade-off between the threshold δ and the number of scenarios H_δ. For a given threshold δ, increasing the number of scenarios will decrease the probability of violation. Conversely, if the number of scenarios is constant, a small threshold may be exceeded with high probability. An example of the trade-off is shown in Figure 1 as a function of the dimensions n_g.

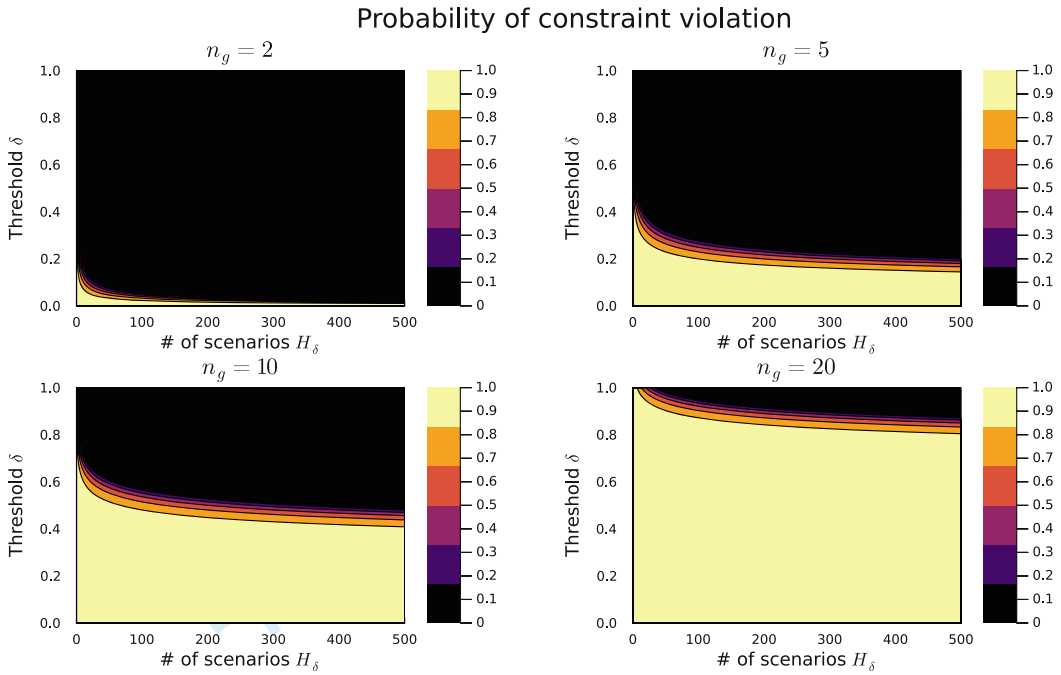


FIGURE 1 Trade-off between the number of scenarios H_δ and the desired threshold δ as a function of dimensionality n_g of the uncertainty space.

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

Finally, Theorem 6 has a practical interpretation. Assuming that the allowed probability of constraint violation is p_{des}, from (52) we obtain a bound on the number of scenarios H_δ to satisfy this requirement for a given δ:

$$H_\delta \geq \frac{1 - p_{\text{des}}}{\kappa \delta^{n_g}}. \tag{53}$$

The number of scenarios obtained from (53) can be used as an additional stopping criterion (line 18 in Algorithm 3).

Finally, should the probability distributions of the uncertainty be available, our approach can be extended to explicitly include chance constraints and probability distributions a priori in the problem formulation by using an appropriate problem structure.[34]

4 | EXAMPLES

We will first show that the realization of uncertainty leading to the maximal constraint violation can be anywhere, not necessarily on the boundary of the uncertainty set even for a linear system. We also show the performance of the local reduction applied to an unstable system.

We then 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 two numerical examples: temperature control in a residential building and flow control in a centrifugal compressor. The solution provided by local reduction is then compared with the case obtained for boundary scenarios and for scenarios chosen randomly from a uniform distribution.

The examples were implemented in Julia 1.6[35] using JuMP 0.21.4.[36] The problems were then solved using Ipopt version 3.12.10.[37] All tests were performed on an Intel® Core™ i7-7500U with 16 GB of RAM.

4.1 | Scenarios not on the boundary

An example of scenarios not on the boundary for a nonlinear system was provided in previous works.[9,10] We show that a linear system with parametric uncertainty may have interim worst-case scenario in the interior of the uncertainty range. The worst-case scenario in the sense of Definition 2 for a robust optimal control problem of the form (12) may be in the interior of the set $\mathbb{W} \times \mathbb{D}$.

Let us assume that we have a system with dynamics affected by parametric uncertainty d:

$$x_{k+1} = (A + d)x_k + Bu_k \tag{54}$$

where $k = 1, \ldots, N$ and $d \in [\underline{d}, \overline{d}]$, A, B are constant scalar matrices, x_0 is known. Let us assume further that the optimal control problem includes a constraint of the form:

$$x_k \leq 0 \tag{55}$$

The constraint (55) must be satisfied for all k. The maximization step in the local reduction method consists in solving a series of optimization problems with the objective for every k:

$$\max_d \quad x_k \tag{56}$$

$$\text{subject to} \quad (54) \tag{57}$$

Every k corresponds to a different optimization problem of the form (57).

Let us now take $k = 4, x_0 = 0, A = -0.5, B = 1$, and we are looking for the maximal constraint violation for a constant $d \in [-0.5, 0.5]$. Let us assume that the current optimal control input $u_k, k = 0, \ldots, 4$ is $u_0 = u_2 = u_3 = -1$ and $u_1 = u_4 = 1$.

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

The maximization problem from (57) becomes a maximization of a fourth order polynomial of d:

$$\max_d \sum_{j=0}^{4} u_{4-j}(-0.5+d)^j \tag{58}$$

Looking for the maximum of the polynomial (58) yields $d \approx 0.2$ which is not on the boundary of the interval $[-0.5, 0.5]$.

This example confirms that considering boundary scenarios would miss the actual worst-case scenario. In a similar way, selecting a priori a number of scenarios would result in adding unnecessary scenarios that may or may not be the worst-case scenario.

4.2 | Unstable system

To show that the boundedness of constraints required in Theorem 2 need not imply stability, we analysed a system with dynamics:

$$x_{k+1} = adx_k + u_k \tag{59}$$

where $a = 2.1$, the uncertainty $d \in [0.9, 1.1]$, and with $x_1 = 0.5$. The constraints to be satisfied were $0 \le x_k \le 1$, for all $k = 1, \dots, N$ with $N = 10$. The controller u_k was parameterised as an affine function of the state:

$$u_k = Kx_k + q_k \tag{60}$$

and the constraints $u_k \in [-1, 1]$ were enforced by a smooth saturation function

$$u_k^{\text{sat}} = \frac{\beta_0}{\beta_1 + \exp(\beta_2 u_k)} + \beta_3 \tag{61}$$

where β_i are constants. Here $\beta_0 = -2.0229$, $\beta_1 = 1$, $\beta_2 = 1.2963$, $\beta_3 = 1.01145$. The objective was to minimize the square of the control over the whole horizon N:

$$J = \sum_{k=0}^{N-1} u_k^2. \tag{62}$$

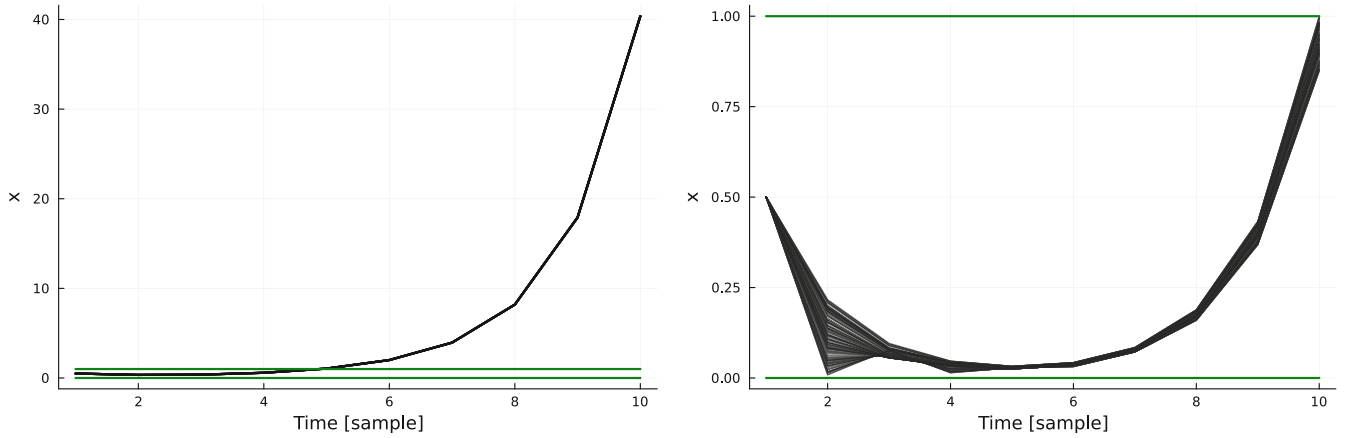
The trajectory obtained from maximization of the violation of the constraint $x_{10} \le 1$ is shown in Figure 2A. The time horizon is finite, $N = 10$, so the trajectory $\mathbf{x}$ is bounded for any value of d and thus the local reduction can be used. Using the local reduction algorithm resulted in three scenarios: $d_1 = 1$, $d_2 = 0.9$, $d_3 = 1.1$ that ensure robustness, as indicated in Figure 2B. The trajectories in Figure 2B were obtained for 500 randomly chosen scenarios uniformly distributed in $[0.9, 1.1]$.

The example shows that stability of the dynamics is optional, provided the constraints are bounded. Nevertheless, it is recommended to take instability into account when numerically solving the minimization problems in Algorithm 2. To show the influence of instability on the numerical performance, we computed the violation for six different values of the horizon N and the results are collected in Table 1. For instance, the maximization problems can be solved approximately, terminating as soon as a violation has been found. Such approaches correspond to choosing large values of ϵ_w and ϵ_d in Algorithm 3, so the analysis in Section 3.5.1 holds.

4.3 | Linear system with parametric uncertainty

This numerical example consists in a linear system with both parametric and additive time-varying uncertainty. The example describes a single zone building affected by time-varying internal heat gain, solar radiation, and external temperature.[38] The objective is to follow a time-varying set-point for internal temperature x_k^{temp}. The dynamics are discrete

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License



(A) Trajectory x corresponding to maximization of upper bound (black) with the respective bounds (green)

(B) Validation of the scenarios from local reduction for 500 randomly chosen scenarios uniformly distributed in [0.9, 1.1]

FIGURE 2 Application of local reduction for the unstable system (59).

TABLE 1 Constraint violation obtained for the unstable system (59) with different time horizon N.

N	5	10	15	20	25	30
Violation	3.75	39.37	49829.0	2.63163e6	2.00637e8	1.35451e9

and linear:

$$x_{k+1} = Ax_k + Bu_k^{\text{sat}} + Ww_k \tag{63}$$

with matrices:

$$A = \begin{bmatrix} 0.8511 & 0.0541 & 0.0707 \\ 0.1293 & 0.8635 & 0.0055 \\ 0.0989 & 0.0032 & 0.7541 \end{bmatrix}, \quad B = 10^{-3}\begin{bmatrix} 3.5 \\ 0.3 \\ 0.2 \end{bmatrix}, \quad W = 10^{-3}\begin{bmatrix} 22.217 & 1.7912 & 42.2123 \\ 1.5376 & 0.6944 & 2.29214 \\ 103.1813 & 0.1032 & 196.0444 \end{bmatrix}.$$

The states x describe the indoor temperature x^{temp}, wall temperature x^{wall}, and the corridor temperature x^{corr}. The control u represents the amount of heating and cooling delivered to the building. The initial condition was chosen as $x_0 = \begin{bmatrix} 25.0 & 24.0 & 24.0 \end{bmatrix}^{\top}$°C. Moreover, we assume that the wall temperature and the corridor temperature can only be measured approximately, so there are two additional sources of uncertainty in the initial condition for these two states. We assume $x_0^i = 24 + d^i$, $i = \text{wall, corr}$, where $d^i \in [-0.5, 0.5]$. We also assume that the matrices $A = [a_{i,j}]$ and $B = [b_j]$, $i, j = 1, 2, 3$ are affected by uncertainty:

$$a_{i,j} \cdot \delta_{i,j} \text{ and } b_j \cdot \eta_j$$

where $\delta_{i,j}, \eta_j$ are uncertain parameters. Two cases will be considered: Case A with $\delta_{i,j}, \eta_j \in [0.98, 1.02]$ and Case B with $\delta_{i,j}, \eta_j \in [0.96, 1.03]$. The minimal control effort is ensured by the objective function:

$$J = \frac{1}{N}\sum_{k=0}^{N-1} u_k^2 \tag{64}$$

It is assumed that the day starts at 6:00 a.m. and lasts 12 h. The temperature indoors must stay within limits:

$$T_{\min} \le x_k^{\text{temp}} \le T_{\max} \tag{65}$$

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

TABLE 2 Ranges of uncertain parameters throughout the day.

	Day	Night
Internal heat gain	[4, 6]	[0, 2]
Solar radiation	[4, 6]	0
External temperature	[6, 8]°C	[2, 4]°C

During the day, the indoor temperature must be kept above 23°C and during the night can drop down to 17°C:

$$T_{\min} = \begin{cases} 17°\text{C during night time} \\ 23°\text{C during day time} \end{cases} \tag{66}$$

The maximal temperature is the same during the day and night, $T_{\max} = 26°\text{C}$.

The optimal control problem is solved over a period of 48 h starting at 6:00 a.m. the first day, with $N = 192$. As a result, the trajectory constraints (65) impose $192 \cdot 2$ constraints corresponding to every sampling time. The three uncertain parameters, internal gain, solar radiation, and external temperature, vary with time within the limits provided in Table 2.

The control variables are parameterised as:

$$u_k = Kx_k^{\text{temp}} + q_k \tag{67}$$

where K and q_k are decision variables. Furthermore, we include saturation of the control inputs:

$$u_k^{\text{sat}} = \text{sat}(u_k) = \begin{cases} -500\ \text{W} & \text{for } u_k < -500\ \text{W} \\ u_k & \text{for } -500\ \text{W} \le u_k \le 1200\ \text{W} \\ 1200\ \text{W} & \text{for } u_k > 1200\ \text{W} \end{cases} \tag{68}$$

The saturation was approximated by a smooth function:

$$u_k^{\text{sat}} = \frac{\beta_0}{\beta_1 + \exp(\beta_2 u_k)} + \beta_3 \tag{69}$$

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

In total, there are 14 uncertain parameters affecting the matrices A, B, and the initial condition for the wall and corridor temperatures. We assume no knowledge about the scenarios, except the ranges of uncertainty.

4.3.1 | Results–Case A

In case A, we ran Algorithm 1 with the time-varying uncertainties from Table 2 and parametric uncertainties $\delta_{i,j}, \eta_j \in [0.98, 1.02]$.

Overall performance

The local reduction method in Case A reduced the number of scenarios to five. The resulting controller obtained for the interim worst-case scenarios was then validated for 500 random realizations from a uniform distribution of uncertainty. The validation of the controller is shown in top left plot in Figure 3. The black curves stay within the green bounds corresponding to constraints (65). 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 the whole time horizon. This is because there is no need to specify time-varying scenarios as they will be found in the maximization step in Algorithm 2. Moreover, Algorithm 2 treats time-varying uncertainty as one realization over the whole horizon, thus overcoming the limitations of separate robust horizon.[8]

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

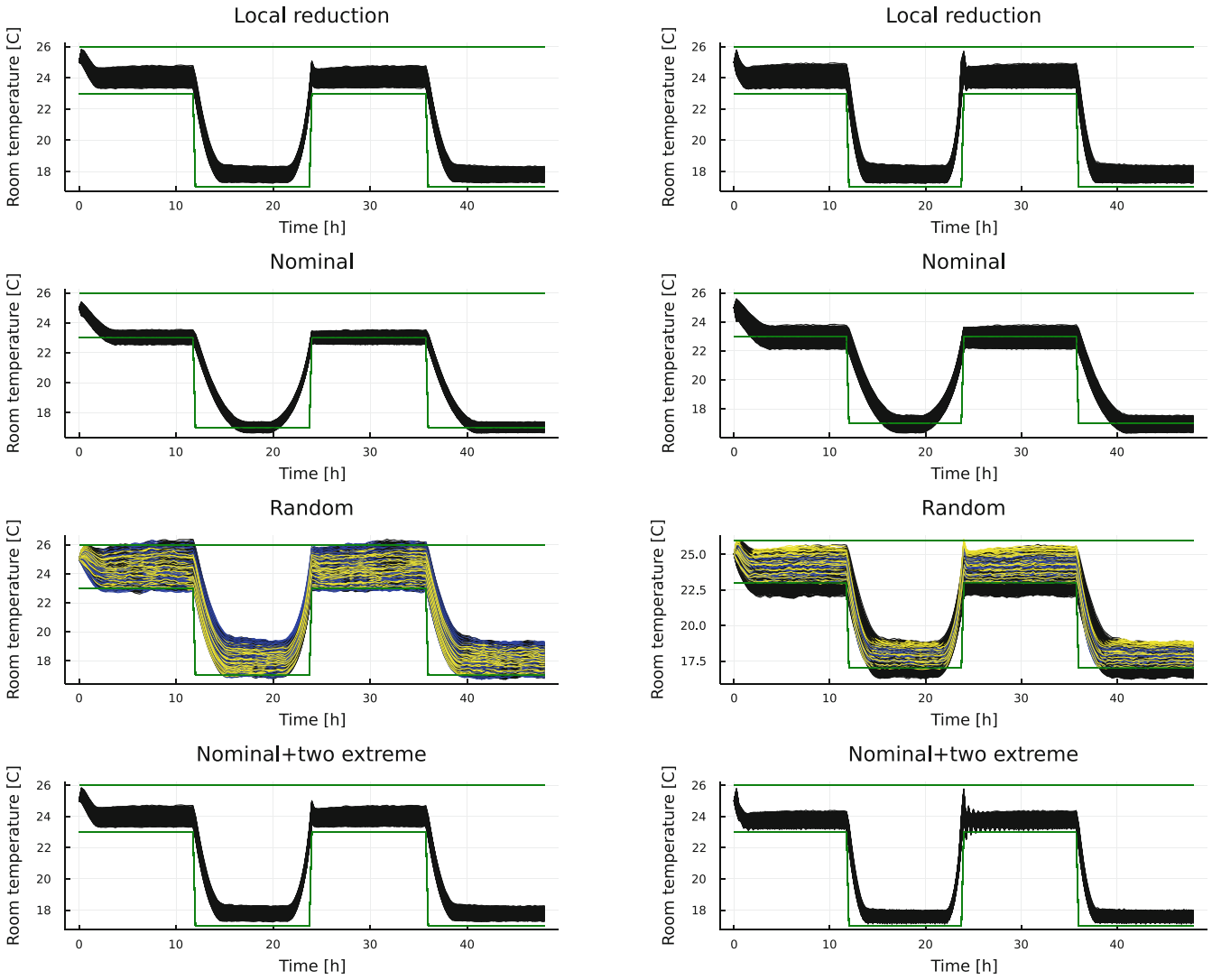


FIGURE 3 Comparison of local reduction with scenario based approaches in Case A (left) and Case B (right).

An example of an interim worst-case scenarios obtained in the maximisation step is shown in Figure 4.

Comparison with other approaches

The results obtained from local reduction are then compared with three scenario-based approaches from the literature:[4]

1. Nominal approach, with a controller obtained assuming there is no uncertainty, that is, $(\mathbf{w}, d) = 0$ (further denoted as "Nominal")
2. Randomised approach, with a controller obtained for a number of randomly chosen scenarios (further denoted as "Random")
3. Extreme approach, with a controller obtained for three scenarios: nominal, lower bound, and upper bound for all uncertainties[8] (further denoted as "Nominal+two extreme")

Validating the nominal controller with 500 random scenarios shows that the approach based on nominal values leads to violation of constraints as shown in the plot 'Nominal' in the left column of Figure 3.

The second set of controllers we used was derived using three sets of random scenarios: five scenarios because five scenarios were found in local reduction, 100 scenarios, and 250 scenarios. The results are shown in the plot 'Random' in the left column of Figure 3, with black corresponding to the controller obtained from five scenarios, yellow to the

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

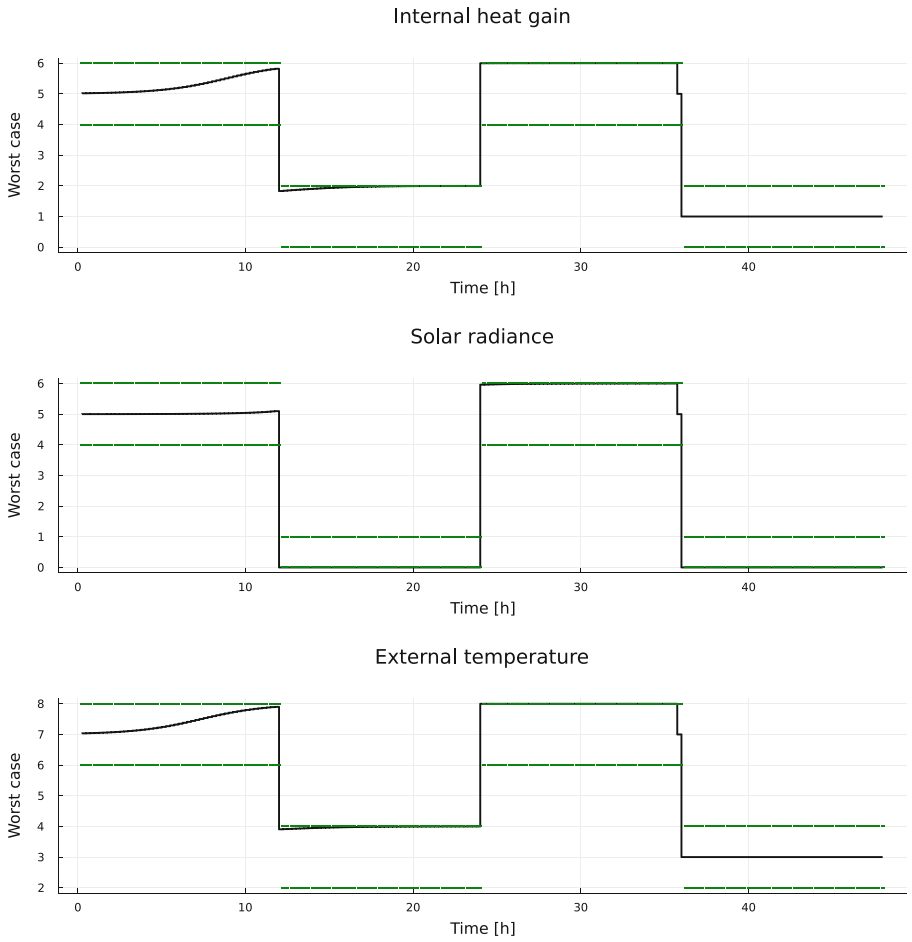


FIGURE 4 An example of a time-varying scenario found by the local reduction in Case A.

controller with 100 scenarios, and blue to the controller with 250 scenarios. In all the cases the controller violated at least one of the bounds (100 scenarios gave 0.2°C, 250 scenarios gave 0.1°C), with the controller based on five scenarios violating both the lower and upper bound (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 on the computer.

A possible reason for the random controller being unable to satisfy the constraints is due to not including extreme scenarios in the scenario set. If we were to take only extreme values for every uncertainty and consider all extreme scenarios, we would need to solve a problem with $2^{14+3\times192}$ scenarios, which is intractable. 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 the left column of Figure 3. The controller based on the extreme scenarios was also able to avoid constraint violations with three scenarios.

The results of the comparison show that the local reduction method provides better results than the nominal control or a controller based on a random choice of scenarios. At the same time, the number of scenarios found in the maximization step from Algorithm 2 is comparable to the controller based only on nominal and two extreme scenarios. Further analysis of the performance comparison will be done for Case B in Section 4.3.2 to show that other controllers fail if parametric uncertainty is more significant.

Time performance

The results from Figure 3 indicate that the local reduction method enables reducing the number of scenarios compared to approaches based on random choice or on time-varying extreme scenarios. Figure 5 shows in the left column the time necessary to solve each step of the local reduction method. The plot in the top left shows the time to solve the minimization

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

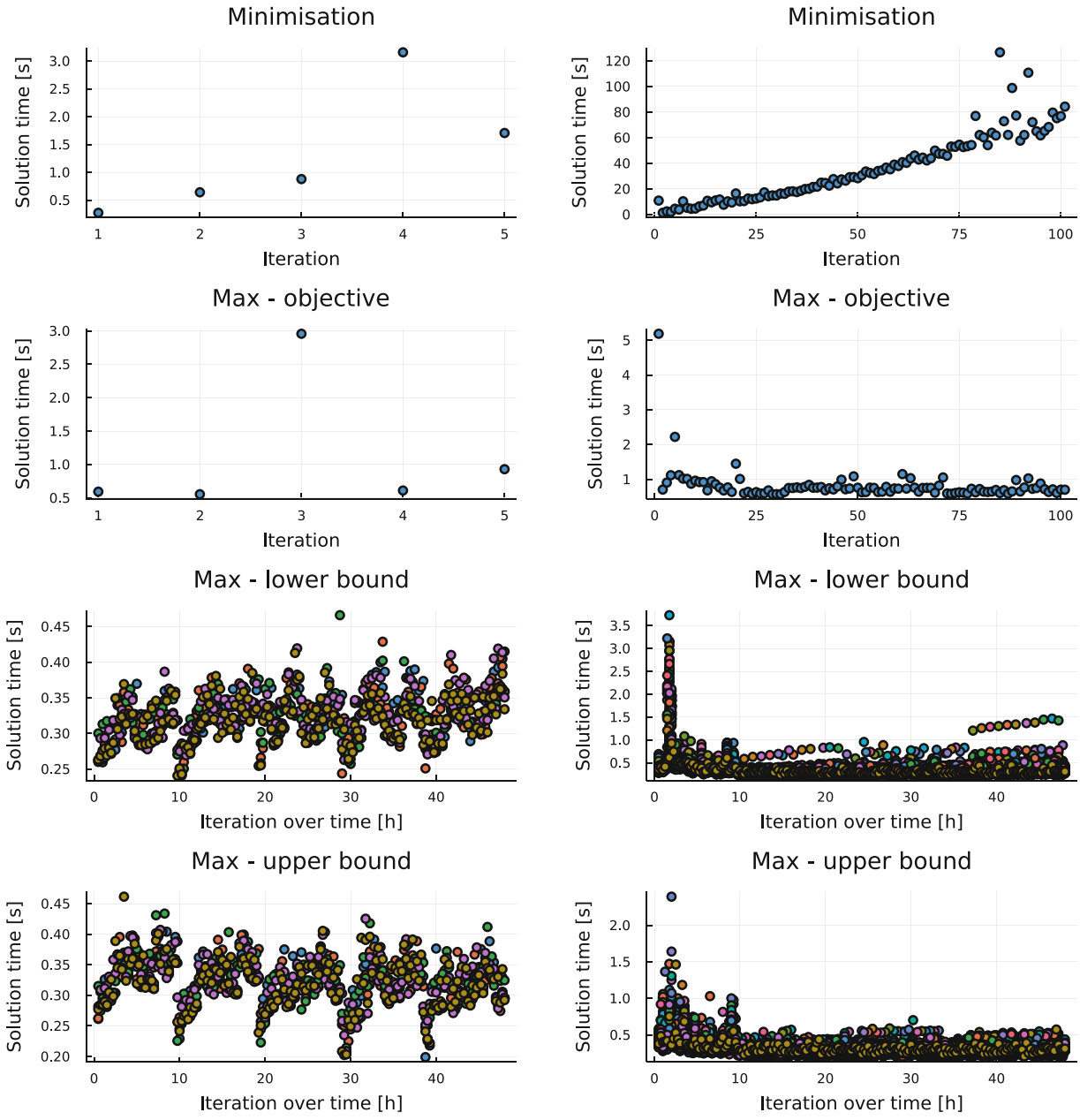


FIGURE 5 Impact of significant parametric uncertainty on local reduction for Case A (left) and Case B (right). The first row shows the time necessary for solving the minimization problem (line 3 in Algorithm 3) as a function of iteration of local reduction, corresponding to the number of scenarios in the current set $\mathbb{H}$. The second row shows the time necessary to solve the first maximization problem related to the objective as a function of iteration of local reduction (line 1 in Algorithm 2). The two bottom rows show the time necessary to solve the maximization problems corresponding to $n_g = 2$ trajectory constraints (65) as a function of samples over the horizon of 48 h, $N = 192$ (lines 4–8 in Algorithm 2).

problem as a function of iterations of the local reduction. The iterations correspond to the number of scenarios included in the minimization problem. The number of scenarios is relatively small (five scenarios), so the time for each iteration is below 3.5 s.

The second plot in the left column shows the time for solving the maximization problem if the objective is considered (line 2 in Algorithm 2). The time to solve the maximization problem in every iteration is comparable to the time to solve the minimization problem. The subsequent two plots show the time for the maximization of the constraint violation corresponding to the lower and upper bounds over the overall time horizon (one maximization per time step, lines 4–8 in Algorithm 2). In both cases, the average time to solve a single maximization problem was 0.35 s. The total time for finding

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

the five scenarios was 10 min 48 s. The algorithm can be parallelized so that the maximization problems are solved simultaneously.[23] Therefore, it can be expected that the time to find a solution in a single iteration of the local reduction method will be equivalent to the solution of the minimization problem and the maximal time needed to solve the maximization problems.

4.3.2 | Results–Case B

In case B, we ran Algorithm 1 with the time-varying uncertainties from Table 2 and parametric uncertainties $\delta_{i,j}, \eta_j \in [0.96, 1.03]$. The range for parameters $\delta_{i,j}, \eta_j$ was chosen to increase the parametric uncertainty in the dynamics while still ensuring that a controller of the form (67) exists.

Overall performance

In contrast to Case A, which has given only five scenarios, the local reduction method in Case B found 101 scenarios. A validation for 500 scenarios is shown in the top right plot in Figure 3. The plot shows that the controller obtained for 101 scenarios from the local reduction avoided violating constraints. This result indicates that local reduction can handle parametric uncertainty.

Comparison with other approaches

The controller in Case B has also been compared with the same set of controllers as in Case A, obtained for the new range of uncertainty. As expected, the nominal controller and the random controller were unable to satisfy the constraints (middle plots in the right column in Figure 3). In contrast to Case A, the controller based on nominal and two extreme scenarios was also unable to satisfy the constraints, as shown in the bottom right plot in Figure 3. The black lines after 24 h cross the green lines so that the lower bound on the temperature is violated (0.5°C). Therefore, taking extreme scenarios may be insufficient, as shown in Section 4.1.

The performance of local reduction in handling parametric uncertainty will be also confirmed in the nonlinear case study in Section 4.4.

Computational time performance

Figure 5 shows in the right column the time performance of the elements of the local reduction method in terms of time needed to solve them. As expected, the time to obtain the solution to the minimization problems increases with iterations. The increase is due to the fact that the number of scenarios considered in every iteration is greater than in the previous one. At the same time, the time necessary to solve a single maximization problem remained similar across the iterations. This result indicates the potential for parallelization to improve performance.

In this work, we also assume that the structure of the dynamic feedback policy in (3) is known. The improved computational time of the local reduction can be used to validate whether the chosen control parametrization is suitable for robustness, because it enables obtaining a solution more quickly. Thus, if the results for a given parametrization are unsatisfactory, a different parametrization can be evaluated.

4.3.3 | Results–Properties of interim scenarios

Finally, we show the impact of the choice of when two scenarios are considered similar in Algorithm 3. The results of varying ϵ are collected in Table 3. The time was obtained using BenchmarkTools.jl.[39]

As expected, a high threshold for similarity of scenarios leads to fewer scenarios added to the problem. This is visible in particular in Case B with more significant parametric uncertainty, where the high threshold $\epsilon = 0.1$ led to two scenarios, whereas a lower threshold $\epsilon = 0.001$ led to 101 scenarios. The middle column in Table 3 shows that robustness to the three scenarios in Case B for $\epsilon = 0.1$ is insufficient to robustify the system against random realizations of uncertainties. Conversely, both $\epsilon = 0.01$ and $\epsilon = 0.001$ seem to robustify the system against the random realizations. Unless explicitly stated, the paper considered $\epsilon = 0.001$.

The number of scenarios also affects the time necessary to solve the resulting optimization problem corresponding to all the scenarios (right column of Table 3).

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

TABLE 3 Influence of the tolerance for checking the similarity of scenarios on the resulting number of scenarios, maximal constraint violation over 500 random scenarios, and the time to obtain a solution for the scenarios obtained

	ϵ	# Scenarios	Max violation	Time
Case A	0.1	3	0	6.4 min
	0.01	5	0	10.8 min
	0.001	5	0	10.5 min
Case B	0.1	2	1.5°C	5.3 min
	0.01	101	0	5 h
	0.001	101	0	5 h

4.4 | Nonlinear system with a dynamic controller

4.4.1 | Dynamics

A further case study is presented to show how our proposed method can be used in nonlinear systems. We want to design a flow controller for a centrifugal compressor. The dynamics for a compressor are nonlinear:[40]

$$\begin{aligned}
\dot{p}_s &= \frac{a_{01}^2}{V_s}(m_{\text{in}} - m + m_r),\\
\dot{p}_d &= \frac{a_{01}^2}{V_d}(m - m_{\text{out}} - m_r),\\
\dot{m} &= \frac{A_1}{L_c}(\Pi(m,\omega)p_s - p_d),\\
\dot{\omega} &= \frac{1}{J}(\tau - \tau_c),\\
\dot{m}_r &= \frac{1}{\tau_r}(m_{\text{SP}} - m_r),
\end{aligned} \tag{70}$$

where p_s and p_d are the suction and discharge suction pressures, a_{01}, V_s, A_1, L_c, J are constant parameters defining the geometry of the compressor, the piping, and the shaft, m_{SP} is the controller for the recycle valve, m_r is the mass flow through the recycle valve, m is the mass flow through the compressor, ω is the speed of the shaft of the compressor in rad s^{-1}, τ is torque provided by a flow controller, τ_c is the reaction torque of the compressor. The function $\Pi(\cdot,\cdot)$ gives the pressure ratio across a compressor as a function of compressor mass flow and speed:

$$\Pi(m,\omega) = \alpha_0 + \alpha_1 m + \alpha_2\omega + \alpha_3 m\omega + \alpha_4 m^2 + \alpha_5\omega^2. \tag{71}$$

The coefficients α_i, $i = 0, \ldots, 5$ are usually estimated from operating data. Here we assume $\alpha_0 = 2.691$, $\alpha_1 = -0.014$, $\alpha_2 = -0.041$, $\alpha_3 = 0.0009$, $\alpha_4 = 0.0002$, $\alpha_5 = 0.00002$. The uncertainty in α_i is described in Section 4.4.3.

The value of m_{in} and m_{out} captures the external mass flows on the suction and discharge side, respectively. The mass flows depend on the pressures p_s and p_d, and external pressures p_{in} and p_{out}:

$$m_{\text{in}} = 0.4k_{\text{in}}A_{\text{in}}\sqrt{p_{\text{in}} - p_s}, \tag{72a}$$

$$m_{\text{out}} = 0.8k_{\text{out}}A_{\text{out}}\sqrt{p_d - p_{\text{out}}}, \tag{72b}$$

$$m_{\text{SP}} = k_{\text{rec}}u_{\text{rec}}A_{\text{rec}}\sqrt{p_d - p_s}, \tag{72c}$$

where $A_{\text{in}}, A_{\text{out}}, A_{\text{rec}}$ represent the inlet, outlet and recycle valve orifice areas and $k_{\text{in}}, k_{\text{out}}, k_{\text{rec}}$ the respective valve gains. The values of constant parameters were taken from Reference 41. The value of u_{rec} is obtained from an auxiliary PI

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

controller and can take values between 0 and 1, ensured by a smoothed saturation function of the form (69) with $\beta_0 = 0.072$, $\beta_1 = 0.071$, $\beta_2 = 5.279$, $\beta_3 = -0.001$.

4.4.2 | Optimal control

The objective is to reach the desired flow level $m_d = 100$ kg s^{-1} without violating speed and flow constraints, imposed due to safety. The objective function was formulated as:

$$J(\tau) = \int_0^{t_f} 100 m_r^2(s) + 0.1\omega^2(s) + 1000(m(s) - m_d)^2 \mathrm{d}s \tag{73}$$

where $t_f = 100$ s.

The constraints on the mass flow and the speed are:

$$m \in [65,105] \ \text{kg s}^{-1} \tag{74a}$$

$$\omega \in [550,876] \ \text{rad s}^{-1} \tag{74b}$$

The control input τ is a PI controller parametrised by K_p and K_i:

$$\tau(t) = K_p(m(t) - m_d) + K_i \int_0^t (m(s) - m_d)\mathrm{d}s \tag{75}$$

The parametrization from (75) is typical for centrifugal compressors.[41] The torque that can be applied to the compressor must be between zero and 1000 Nm. The bounds on the torque were ensured by a smoothed saturation function of the form (69) with $\beta_0 = 73.324$, $\beta_1 = 0.072$, $\beta_2 = 0.005$, $\beta_3 = 0$.

4.4.3 | Uncertainties

The uncertainties we considered in this case study are in the valve gains k_{in}, k_{out}, k_{rec}, and correspond to $\pm 5\%$, and in the parameters α_i in the polynomial compressor map (71), and correspond to $\pm 2\%$. Thus, there are nine uncertain parameters.

4.4.4 | Results

To find the flow controller from (75), the dynamics were discretized using the trapezoidal collocation method with time step 0.5 s.

Overall performance

The local reduction method applied to the compressor case study resulted in two scenarios. Figure 6 shows the results of the validation for the two scenarios obtained from the local reduction method with $\epsilon = 10^{-6}$. The controller obtained from the local reduction did not violate the constraints on either the mass flow (top left) or the speed of the compressor (top right).

Comparison with other approaches

If we were to consider all extreme realizations of the uncertainties, we would obtain $2^9 = 512$ scenarios. The local reduction method we propose in this paper reduced the number of scenarios to two. Conversely, the nominal controller was not able to satisfy the constraints and both the mass flow and the speed violated their upper limits (second row of plots in

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

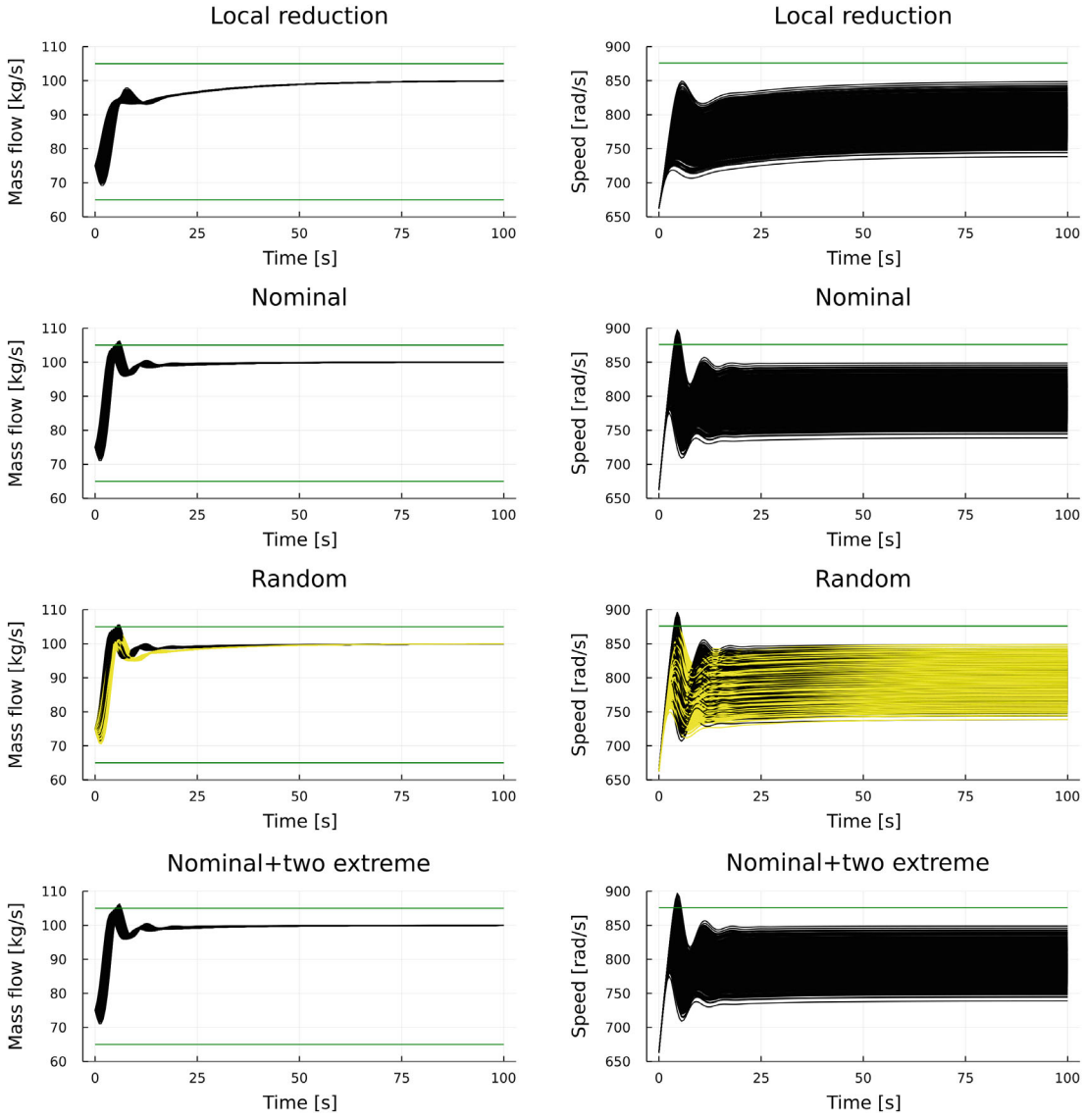


FIGURE 6 Compressor case study–comparison of four approaches.

Figure 6). The controllers based on randomly chosen scenarios (from a uniform distribution) are shown in the third row, with black indicating a controller based on two random scenarios. The controller based on two random scenarios was insufficient to ensure constraint satisfaction and both the mass flow and the speed of the compressor violated their upper limits. The minimal number of random scenarios needed to ensure constraint satisfaction was nine (yellow). Finally, the bottom row in Figure 6 shows the performance of the nominal + extreme controller obtained for three scenarios (one scenario on the lower bound, one scenario on the upper bound, and one scenario with no uncertainty). The controller based on the nominal and two extreme scenarios was insufficient to satisfy constraints. Thus, the comparison with other scenario-based approaches confirms the potential of local reduction for solving robust optimal control problems with parametric uncertainty.

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. The choice of scenarios and their number affects the robustness of the solution as well as computational complexity of the resulting optimization problems. In this work, we formulated robust optimal control problems with

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

time-varying and parametric uncertainty as semi-infinite optimization problems to facilitate the choice of scenarios. The new formulation enabled usage of semi-infinite optimization algorithms, such as local reduction methods. 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. We overcome the dependence on global solvers in the original local reduction formulation by proposing inexact local reduction and providing theoretical bounds on possible constraint violation. The new method consists in solving multiple optimal control problems of reduced size compared to the full scenario-based optimization. In particular, the small control problems can be solved in parallel, further improving the computational speed.

The performance of our approach was evaluated in two case studies with both additive and parametric uncertainty: thermal comfort control in a residential building and mass flow control in a centrifugal compressor. A comparison with common approaches based on a random choice of scenarios and on extreme scenarios indicates that local reduction allows solving robust optimal control problems in an efficient way while ensuring robustness. In particular, the case studies confirm that the proposed inexact local reduction method allows finding worst-case scenarios in the interior of the uncertainty sets. As a result, the new method was able to handle larger parametric uncertainty than other scenario-based approaches.

In this work we required that the constraints must be satisfied for all realisations of the uncertainty. In the future, it would be advisable to look at the conservatism of the obtained solutions and possible relaxations of this requirement. In particular, tighter bounds on constraint violation can be derived if a distribution of the uncertainty is available. Future work could include numerical improvements of approximate local reduction, including warm-starting and use of custom nonlinear optimization solvers, as well as explicit parallelization of the optimal control problems.

ACKNOWLEDGMENTS

This work has received funding from the EPSRC (Engineering and Physical Sciences) under the Active Building Centre project (reference number: EP/V012053/1).

CONFLICT OF INTEREST STATEMENT

The authors declare no potential conflict of interest.

DATA AVAILABILITY STATEMENT

The data that support the findings of this study are available from the corresponding author upon reasonable request.

REFERENCES

1. Calafiore GC, Campi MC. The scenario approach to robust control design. *IEEE Trans Automat Contr*. 2006;51(5):742-753. doi:10.1109/tac.2006.875041
2. Grammatico S, Zhang X, Margellos K, Goulart P, Lygeros J. A scenario approach for non-convex control design. *IEEE Trans Automat Contr*. 2015;61(2):334-345. doi:10.1109/tac.2015.2433591
3. Håberg M. Fundamentals and recent developments in stochastic unit commitment. *Int J Electr Power Energy Syst*. 2019;109:38-48. doi:10.1016/j.ijepes.2019.01.037
4. Campi M, Carè A, Garatti S. The scenario approach: a tool at the service of data-driven decision making. *Ann Rev Control*. 2021;52:1-17. doi:10.1016/j.arcontrol.2021.10.004
5. Mutapcic A, Boyd S. Cutting-set methods for robust convex optimization with pessimizing oracles. *Optim Methods Softw*. 2009;24(3):381-406. doi:10.1080/10556780802712889
6. Lucia S, Andersson JA, Brandt H, Diehl M, Engell S. Handling uncertainty in economic nonlinear model predictive control: a comparative case study. *J Process Control*. 2014;24(8):1247-1259. doi:10.1016/j.jprocont.2014.05.008
7. Vuffray M, Misra S, Chertkov M. Monotonicity of dissipative flow networks renders robust maximum profit problem tractable: general analysis and application to natural gas flows. Paper presented at: 2015 54th IEEE Conference on Decision and Control (CDC), IEEE. 2015.
8. Thombre M, Yu ZJ, Jäschke J, Biegler LT. Sensitivity-assisted multistage nonlinear model predictive control: robustness, stability and computational efficiency. *Comput Chem Eng*. 2021;148:107269.
9. Krasnochtanova I, Rauh A, Kletting M, Aschemann H, Hofer EP, Schoop KM. Interval methods as a simulation tool for the dynamics of biological wastewater treatment processes with parameter uncertainties. *App Math Model*. 2010;34(3):744-762. doi:10.1016/j.apm.2009.06.019
10. Puschke J, Djelassi H, Kleinekorte J, Hannemann-Tamás R, Mitsos A. Robust dynamic optimization of batch processes under parametric uncertainty: utilizing approaches from semi-infinite programs. *Comput Chem Eng*. 2018;116:253-267. doi:10.1016/j.compchemeng.2018.05.025
11. Heitsch H, Römisch W. Scenario tree reduction for multistage stochastic programs. *Comput Manag Sci*. 2009;6(2):117-133. doi:10.1007/s10287-008-0087-y

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License

12. Scokaert POM, Mayne DQ. Min-max feedback model predictive control for constrained linear systems. *IEEE Trans Automat Contr*. 1998;43(8):1136-1142. doi:10.1109/9.704989
13. Hettich R, Kortanek KO. Semi-infinite programming: theory, methods, and applications. *SIAM Rev*. 1993;35(3):380-429. doi:10.1137/1035089
14. Hettich R, Kaplan A, Tichatschke R. Semi-infinite programming: numerical methods. *Encyclopedia Optim*. 2009;5:112-117. doi:10.1007/978-0-387-74759-0_588
15. Hettich R. A review of numerical methods for semi-infinite optimization. In: Fiacco AV, Kortanek KO, eds. *Semi-Infinite Programming and Applications*. Lecture Notes in Economics and Mathematical Systems. Vol 215. Springer. 1983. doi:10.1007/978-3-642-46477-5_11
16. Seidel T, Küfer KH. An adaptive discretization method solving semi-infinite optimization problems with quadratic rate of convergence. *Optimization*. 2020;71(8):2211-2239. doi:10.1080/02331934.2020.1804566
17. Djelassi H, Mitsos A, Stein O. Recent advances in nonconvex semi-infinite programming: applications and algorithms. *EURO J Comput Optim*. 2021;9:100006. doi:10.1016/j.ejco.2021.100006
18. Blankenship JW, Falk JE. Infinitely constrained optimization problems. *J Optim Theory Appl*. 1976;19(2):261-281. doi:10.1007/bf00934096
19. Skogestad S, Postlethwaite I. *Multivariable Feedback Control: Analysis and Design*. Vol 2. Wiley; 2007.
20. Pereira AIPN, Fernandes EMGP. A reduction method for semi-infinite programming by means of a global stochastic approach. *Optimization*. 2009;58(6):713-726. doi:10.1080/02331930701764072
21. Hauser K. Semi-infinite programming for trajectory optimization with nonconvex obstacles. Paper presented at: International Workshop on the Algorithmic Foundations of Robotics, Springer. 2018 565–580.
22. Katselis D, Rojas CR, Welsh JS, Hjalmarsson H. Robust experiment design for system identification via semi-infinite programming techniques. *IFAC Proc Vol*. 2012;45(16):680-685. doi:10.3182/20120711-3-be-2027.00346
23. Žaković S, Rustem B, Asprey S. A parallel algorithm for semi-infinite programming. *Comput Stat Data Anal*. 2003;44(1-2):377-390. doi:10.1016/s0167-9473(03)00152-x
24. Zagorowska M, Falugi P, O'Dwyer E, Kerrigan EC. Efficient solution of robust optimal control problems using local reduction. 2022. Accepted at IFAC World Congress 2023. doi:10.48550/ARXIV.2204.14145
25. Schwientek J, Seidel T, Küfer KH. A transformation-based discretization method for solving general semi-infinite optimization problems. *Math Methods Oper Res*. 2020;93(1):83-114. doi:10.1007/s00186-020-00724-8
26. Tsoukalas A, Rustem B, Pistikopoulos EN. A global optimization algorithm for generalized semi-infinite, continuous minimax with coupled constraints and bi-level problems. *J Global Optim*. 2008;44(2):235-250. doi:10.1007/s10898-008-9321-y
27. Reemtsen R, Rückmann JJ. *Semi-infinite Programming*. Vol 25. Springer Science & Business Media; 1998.
28. Mitsos A. Global optimization of semi-infinite programs via restriction of the right-hand side. *Optimization*. 2011;60(10-11):1291-1308. doi:10.1080/02331934.2010.527970
29. Wheeden RL, Zygmund A. *Measure and Integral: an Introduction to Real Analysis*. Pure and Applied Mathematics. Chapman & Hall/CRC; Taylor & Francis; 1977.
30. Blanchini F. Set invariance in control. *Automatica*. 1999;35(11):1747-1767. doi:10.1016/S0005-1098(99)00113-2
31. Schechter E. *Handbook of Analysis and its Foundations*. Elsevier; 1997.
32. Royden H, Fitzpatrick P. *Real Analysis*. Prentice Hall; 2010.
33. Sommerville DMY. *Introduction to the Geometry of N Dimensions*. Dover Books on Mathematics. Dover Publications; 2020.
34. Bertsimas D, Brown DB, Caramanis C. Theory and applications of robust optimization. *SIAM Rev*. 2011;53(3):464-501. doi:10.1137/080734510
35. Bezanson J, Edelman A, Karpinski S, Shah VB. Julia: a fresh approach to numerical computing. *SIAM Rev*. 2017;59(1):65-98. doi:10.1137/141000671
36. Dunning I, Huchette J, Lubin M. JuMP: a Modeling language for mathematical optimization. *SIAM Rev*. 2017;59(2):295-320. doi:10.1137/15m1020575
37. Thierry D, Biegler L. The ℓ_1-exact penalty-barrier phase for degenerate nonlinear programming problems in Ipopt. *IFAC-PapersOnLine*. 2020;53(2):6496-6501. doi:10.1016/j.ifacol.2020.12.1798
38. Lian Y, Jones CN. From system level synthesis to robust closed-loop data-enabled predictive control. Paper presented at: 2021 60th IEEE Conference on Decision and Control (CDC). 2021 1478-1483.
39. Chen J, Revels J. Robust benchmarking in noisy environments. *arXiv e-prints*. 2016.
40. Cortinovis A, Ferreau HJ, Lewandowski D, Mercangöz M. Experimental evaluation of MPC-based anti-surge and process control for electric driven centrifugal gas compressors. *J Process Control*. 2015;34:13-25. doi:10.1016/j.jprocont.2015.07.001
41. Milosavljevic P, Marchetti AG, Cortinovis A, Faulwasser T, Mercangöz M, Bonvin D. Real-time optimization of load sharing for gas compressors in the presence of uncertainty. *Appl Energy*. 2020;272:114883. doi:10.1016/j.apenergy.2020.114883

How to cite this article: Zagorowska M, Falugi P, O'Dwyer E, Kerrigan EC. Automatic scenario generation for efficient solution of robust optimal control problems. *Int J Robust Nonlinear Control*. 2024;34(2):1370-1396. doi: 10.1002/rnc.7038

10991239, 2024, 2, Downloaded from https://onlinelibrary.wiley.com/doi/10.1002/rnc.7038 by Imperial College London, Wiley Online Library on [03/01/2024]. See the Terms and Conditions (https://onlinelibrary.wiley.com/terms-and-conditions) on Wiley Online Library for rules of use; OA articles are governed by the applicable Creative Commons License