

Constraint Removal for MPC with Performance Preservation and a Hyperthermia Cancer Treatment Case Study

S.A.N. Nouwens¹, B. de Jager¹, M.M. Paulides^{2,3}, W.P.M.H. Heemels¹

Abstract—Model predictive control (MPC) is an optimization-based control strategy with broad industrial adoption. Unfortunately, the required computation time to solve the receding-horizon MPC optimization problem can become prohibitively large for many applications with a large number of state constraints. This large number of state constraints can, for instance, originate from spatially discretizing a partial differential equation of which the solution has to satisfy constraints over the full spatial domain. This is particularly the case in MPC for RF-based hyperthermia cancer treatments, which forms a strong motivation for this study. To address this problem, we propose a novel constraint-adaptive MPC framework for linear discrete-time systems. In this framework, we select at each time-step a subset of the state constraints that are included in the optimization problem, thereby reducing the online computational burden. Critically, our framework guarantees the same closed-loop performance, recursive feasibility, and constraint satisfaction properties as the original (non-reduced) MPC scheme. We achieve this result by efficiently exploiting reachable set computations and the MPC cost function. We will demonstrate our novel method using a hyperthermia cancer treatment case study showing a two-orders of magnitude improvement in computation time, with identical closed-loop performance as the original (non-reduced) MPC scheme.

I. INTRODUCTION

Model predictive control (MPC) is a successful, optimization-based, control technology [1]. Generally, the control inputs are obtained by recursively optimizing a finite-horizon cost function subject to input and state constraints and system dynamics. The explicit inclusion of state and inputs constraints and the capability to handle multi-input multi-output systems are distinct advantages of MPC. However, solving an optimization problem at each time-step can prohibit the real-time feasibility of the controller for computationally complex scenarios. Particularly, in applications requiring the control of systems modeled by partial differential equations with constraints on the continuous spatial domain, such as the heat equation for hyperthermia cancer treatments, the real-time application of MPC can be problematic.

Efforts to improve the computational aspects of MPC are commonplace with among others, explicit MPC, model

reduction, and tailored numerical solvers, see, e.g., [2]–[6], and the references therein. However, our focus in this paper is on systems with large-scale constraints. For such systems, the aforementioned techniques are typically not highly effective. In particular, interior-point methods are reported to scale linearly with the number of inequality constraints [5]. As a result, systems with many constraints motivate so-called constraint removal techniques, which are roughly separated into offline [7], and online methods [8]–[10]. Offline methods can be prohibitively complex to compute and possibly do not enable real-time MPC, as constraints can only be removed if they are redundant with respect to all feasible states. In contrast, online techniques can typically remove more constraints as they exploit the knowledge of the current state of the system. However, online methods use a so-called pre-solve step whose *additional* computation time cannot be ignored and should be limited. Although the existing online constraint removal techniques are effective in various scenarios, they still require extensive computational effort for large-scale problems, or do not provide performance guarantees.

In this paper, we propose a novel online constraint removal method for MPC of linear discrete-time systems that is computationally tractable and guarantees the same closed-loop performance, recursive feasibility, and constraint satisfaction as the original MPC scheme. To this end, we exploit properties inherent to the system by using fast reachability analysis and the optimal control sequence from the previous time step to systematically remove constraints from the optimization problem. We will show the strengths of our new framework using a hyperthermia cancer treatment case study [11]–[14], where we obtain a two-orders of magnitude reduction in computational time. Note that existing work on MPC for hyperthermia is largely focused on ultrasound heating and localized 2D models, which limits their application to small tumors [11], [14]. In contrast, radio frequency based (RF) hyperthermia, used for larger tumors, requires large-scale 3D models due to the non-localized heating from the actuators [12]. The non-local heating also introduces the need for temperature constraints on the whole spatial domain to ensure patient safety. For these reasons, MPC for RF-based hyperthermia is computationally challenging.

The remainder of this paper is structured as follows. We start by introducing the preliminaries in Section II. In Section III, we provide our ca-MPC scheme that exploits reachable sets and the cost function to remove constraints without changing the optimizer of the MPC problem, after which, we provide computationally cheap methods to remove state constraints in Section IV. The hyperthermia cancer

This research is supported by KWF Kankerbestrijding and NWO Domain AES, as part of their joint strategic research programme: Technology for Oncology II. The collaboration project is co-funded by the PPP Allowance made available by Health~Holland, Top Sector Life Sciences & Health, to stimulate public-private partnerships.

¹Control Systems Technology, Department of Mechanical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands

²Electromagnetics for Care & Cure, Department of Electrical Engineering, Eindhoven University of Technology, Eindhoven, The Netherlands

³Department of Radiotherapy, Erasmus University Medical Center Cancer Institute, Rotterdam, The Netherlands

treatment case study is presented in Section V. Finally, conclusions are stated in Section VI.

II. PRELIMINARIES

In this section, we introduce the linear system to be controlled, a typical MPC setup, the forward and backward reachable sets and the cost function level sets. These sets form the basis of our constraint removal framework.

A. System

In this paper, we consider plants that are described by the discrete-time linear time-invariant (LTI) system

$$\mathbf{x}_{k+1} = \mathbf{A}\mathbf{x}_k + \mathbf{B}\mathbf{u}_k. \quad (1)$$

Here, $\mathbf{x}_k \in \mathbb{R}^n$ and $\mathbf{u}_k \in \mathbb{R}^m$ denote the plant states and the inputs, respectively, at discrete time $k \in \mathbb{N}$. Furthermore, \mathbf{A} and \mathbf{B} are real matrices of appropriate dimensions. The system (1) is subject to polyhedral state and input constraints that are given by

$$\mathbf{x}_k \in \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{C}\mathbf{x} \leq \mathbf{b}\} =: \mathbb{X}, \quad (2a)$$

$$\mathbf{u}_k \in \{\mathbf{u} \in \mathbb{R}^m \mid \mathbf{C}_u\mathbf{u} \leq \mathbf{b}_u\} =: \mathbb{U}. \quad (2b)$$

Here, \mathbb{X} and \mathbb{U} are assumed to be non-empty polyhedral sets with $\mathbf{C} \in \mathbb{R}^{n_x \times n}$, $\mathbf{C}_u \in \mathbb{R}^{n_u \times m}$, $\mathbf{b} \in \mathbb{R}^{n_x}$, and $\mathbf{b}_u \in \mathbb{R}^{n_u}$. In this paper we are interested in systems constrained by many state constraints, i.e., $n_x \gg 1$.

B. MPC setup

Based on the system dynamics (1) and constraints (2), a common MPC setup, given state \mathbf{x}_k at time $k \in \mathbb{N}$, can be formulated as

$$\underset{\mathbf{U}_k}{\text{minimize}} \quad J(\mathbf{x}_k, \mathbf{U}_k), \quad (3a)$$

$$\text{subject to} \quad \mathbf{X}_k = \Phi\mathbf{x}_k + \Gamma\mathbf{U}_k, \quad (3b)$$

$$\mathbf{X}_k \in \underbrace{\mathbb{X} \times \dots \times \mathbb{X}}_{N-1 \text{ times}} \times \mathbb{X}_T =: \mathcal{X}_N, \quad (3c)$$

$$\mathbf{U}_k \in \underbrace{\mathbb{U} \times \dots \times \mathbb{U}}_{N \text{ times}} =: \mathcal{U}_N, \quad (3d)$$

where

$$J(\mathbf{x}_k, \mathbf{U}_k) := \ell_T(\mathbf{x}_{N|k}) + \sum_{i=0}^{N-1} \ell(\mathbf{x}_{i|k}, \mathbf{u}_{i|k}), \quad (4a)$$

$$\mathbf{X}_k := [\mathbf{x}_{1|k}^\top \dots \mathbf{x}_{N|k}^\top]^\top, \quad (4b)$$

$$\mathbf{U}_k := [\mathbf{u}_{0|k}^\top \dots \mathbf{u}_{N-1|k}^\top]^\top, \quad (4c)$$

$$\Phi := \begin{bmatrix} \mathbf{A} \\ \mathbf{A}^2 \\ \vdots \\ \mathbf{A}^N \end{bmatrix}, \quad \Gamma := \begin{bmatrix} \mathbf{B} & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{A}\mathbf{B} & \mathbf{B} & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}^{N-1}\mathbf{B} & \mathbf{A}^{N-2}\mathbf{B} & \dots & \mathbf{B} \end{bmatrix}. \quad (4d)$$

Here, $\ell(\cdot)$, $\ell_T(\cdot)$, $\mathbf{x}_{i|k}$, $\mathbf{u}_{i|k}$, and \mathbb{X}_T denote the stage cost, the terminal cost, the predicted state, the predicted input, and the polyhedral terminal set, respectively. The terminal set satisfies $\mathbb{X}_T := \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{C}_T\mathbf{x} \leq \mathbf{b}_T\} \subseteq \mathbb{X}$, where $\mathbf{C}_T \in \mathbb{R}^{n_T \times n}$ and $\mathbf{b}_T \in \mathbb{R}^{n_T}$, which is positively invariant with respect to \mathbb{U} for some auxiliary control law $\mathbf{u} = \mathbf{K}\mathbf{x}$, i.e., $(\mathbf{A} + \mathbf{B}\mathbf{K})\mathbb{X}_T \subseteq \mathbb{X}_T$ and $\mathbf{K}\mathbb{X}_T \subseteq \mathbb{U}$. The $i|k$ subscript is used to denote the i -th prediction at time k . For example, $\mathbf{x}_{i|k}$ denotes the prediction of \mathbf{x}_{k+i} made at time k for $i = 1, 2, \dots, N$, where N denotes the prediction horizon. Additionally, the stage and terminal cost quantify

the performance in terms of the predicted states $\mathbf{x}_{i|k}$ and inputs $\mathbf{u}_{i-1|k}$ for $i = 1, 2, \dots, N$.

For the optimization problem (3), we denote the set of feasible inputs parameterized by the state \mathbf{x}_k by

$$\mathcal{U}_{f,N}(\mathbf{x}_k) := \{\mathbf{U}_k \in \mathcal{U}_N \mid (3b) - (3d)\}. \quad (5)$$

Moreover, we denote the set of feasible states by

$$\mathbb{X}_f := \{\mathbf{x} \in \mathbb{X} \mid \mathcal{U}_{f,N}(\mathbf{x}) \neq \emptyset\}. \quad (6)$$

Finally, note that under suitable assumptions on $\ell(\cdot)$ and $\ell_T(\cdot)$, for all $\mathbf{x}_k \in \mathbb{X}_f$ a minimizer of (3) exists and we denote by $\mathbf{U}_k^* := [\mathbf{u}_{0|k}^{*\top} \dots \mathbf{u}_{N-1|k}^{*\top}]^\top$ a particular one at time $k \in \mathbb{N}$, i.e.,

$$\mathbf{U}_k^* \in \underset{\mathbf{U}_k \in \mathcal{U}_{f,N}(\mathbf{x}_k)}{\text{arg min}} \quad J(\mathbf{x}_k, \mathbf{U}_k). \quad (7)$$

The optimal predicted state sequence corresponding to \mathbf{U}_k^* is denoted by $\mathbf{X}_k^* = [\mathbf{x}_{1|k}^{*\top} \dots \mathbf{x}_{N|k}^{*\top}]^\top = \Phi\mathbf{x}_k + \Gamma\mathbf{U}_k^*$.

Using a receding horizon implementation, the MPC problem (3) is turned into a feedback law $K_{\text{MPC}} : \mathbb{X}_f \rightarrow \mathbb{U}$ by applying the first computed input in \mathbf{U}_k^* on the real plant (1), i.e., $\mathbf{u}_k := K_{\text{MPC}}(\mathbf{x}_k) := \mathbf{u}_{0|k}^*$.

C. Forward and backward reachable sets

We define the forward reachable set $\vec{\mathcal{R}}^k(\mathbb{X}_0, \mathbb{U})$ for initial state set $\mathbb{X}_0 \subseteq \mathbb{R}^n$ as the collection of all states reachable from states in \mathbb{X}_0 in $k \in \mathbb{N}$ time-steps subject to the system dynamics (1) and input constraints (2b). More precisely, the forward reachable set for (1), (2b), and initial state set \mathbb{X}_0 can be computed using the recursion

$$\vec{\mathcal{R}}^{k+1}(\mathbb{X}_0, \mathbb{U}) := \{\mathbf{x}^+ \in \mathbb{R}^n \mid \mathbf{x}^+ = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}, \quad (8)$$

for some $\mathbf{x} \in \vec{\mathcal{R}}^k(\mathbb{X}_0, \mathbb{U})$, and $\mathbf{u} \in \mathbb{U}\}$, $k \in \mathbb{N}$,

where $\vec{\mathcal{R}}^0(\mathbb{X}_0, \mathbb{U}) := \mathbb{X}_0$. Similarly, the backward reachable set $\overleftarrow{\mathcal{R}}^k(\mathbb{X}_T, \mathbb{U})$ for a given terminal set at time $N \in \mathbb{N}$, $\mathbb{X}_T \subseteq \mathbb{X}$, is obtained through a backwards recursion

$$\overleftarrow{\mathcal{R}}^{k-1}(\mathbb{X}_T, \mathbb{U}) := \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \in \overleftarrow{\mathcal{R}}^k(\mathbb{X}_T, \mathbb{U}) \quad (9)$$

for some $\mathbf{u} \in \mathbb{U}\}$, $k \in \mathbb{N}_{[1,N]}$,

where $\overleftarrow{\mathcal{R}}^N(\mathbb{X}_T, \mathbb{U}) := \mathbb{X}_T$. Note that the definitions for the forward and backward reachable sets *do not include* the state constraints, but do include the input constraints.

Last, for ease of notation, we define forward and backward reachable tubes as

$$\vec{\mathcal{R}}(\mathbb{X}_0, \mathbb{U}) := \vec{\mathcal{R}}^1(\mathbb{X}_0, \mathbb{U}) \times \dots \times \vec{\mathcal{R}}^N(\mathbb{X}_0, \mathbb{U}), \quad (10a)$$

$$\overleftarrow{\mathcal{R}}(\mathbb{X}_T, \mathbb{U}) := \overleftarrow{\mathcal{R}}^1(\mathbb{X}_T, \mathbb{U}) \times \dots \times \overleftarrow{\mathcal{R}}^{N-1}(\mathbb{X}_T, \mathbb{U}) \times \mathbb{R}^n. \quad (10b)$$

Note that, in (10b) we replace $\overleftarrow{\mathcal{R}}^N(\mathbb{X}_T, \mathbb{U})$ with \mathbb{R}^n to ensure we do not remove terminal constraints with \mathbb{X}_T itself.

D. Cost function level set

Besides the forward and backward reachable sets, we will also exploit the cost function level set to remove constraints from (3c). The key observation is that for state $\mathbf{x}_k \in \mathbb{X}_f$ and any given input sequence $\tilde{\mathbf{U}}_k \in \mathcal{U}_{f,N}(\mathbf{x}_k)$, an upper bound for the MPC cost function can be easily obtained as $J(\mathbf{x}_k, \mathbf{U}_k^*) \leq J(\mathbf{x}_k, \tilde{\mathbf{U}}_k)$. More precisely, this upper bound defines a cost function level set

$$\mathcal{J}_u(\mathbf{x}_k, \tilde{\mathbf{U}}_k) := \{\mathbf{U} \in \mathbb{R}^{Nm} \mid J(\mathbf{x}_k, \mathbf{U}) \leq J(\mathbf{x}_k, \tilde{\mathbf{U}}_k)\}. \quad (11)$$

For the well-known case where $J(\mathbf{x}, \mathbf{U})$ is a quadratic cost function such that

$$\ell(\mathbf{x}, \mathbf{u}) = \mathbf{x}^\top \mathbf{Q} \mathbf{x} + \mathbf{u}^\top \mathbf{R} \mathbf{u}, \quad \ell_T(\mathbf{x}) = \mathbf{x}^\top \mathbf{P} \mathbf{x}, \quad (12)$$

with \mathbf{Q} , \mathbf{R} , and \mathbf{P} positive-definite matrices, the resulting level set (11) is an ellipse in \mathbb{R}^{Nm} as a function of $\mathbf{x}_k, \tilde{\mathbf{U}}_k$.

Using (3b), it is also possible to define a cost function level set for state trajectories in \mathbb{R}^{Nn} ,

$$\mathcal{J}_x(\mathbf{x}_k, \tilde{\mathbf{U}}_k) := \{\Phi \mathbf{x}_k + \Gamma \mathbf{U} \mid \mathbf{U} \in \mathcal{J}_u(\mathbf{x}_k, \tilde{\mathbf{U}}_k)\}. \quad (13)$$

III. CONSTRAINT REMOVAL USING REACHABLE SETS AND COST FUNCTION LEVEL SETS

Generally, given a convex MPC problem, it is known that there exist exactly at most Nm so-called support constraints whose removal changes the minimizer [15]. However, removing all but these state-dependent support constraints is time-consuming. Nevertheless, it is possible to improve the computational complexity of (3) by removing only a subset of the non-support constraints. Clearly, this speeds up the MPC problem without impacting the closed-loop performance, recursive feasibility, and constraint satisfaction.

We start with the observation that we can remove state constraints that are either, not forward- or backward-reachable, or correspond to state trajectories that do not lower the MPC cost below $J(\mathbf{x}, \tilde{\mathbf{U}}_k)$. To start exploiting this observation, we define the set

$$\mathcal{H}(\mathbf{x}_k, \tilde{\mathbf{U}}_k) := \vec{\mathcal{R}}(\mathbf{x}_k, \mathbf{U}) \cap \overleftarrow{\mathcal{R}}(\mathbb{X}_T, \mathbf{U}) \cap \mathcal{J}_x(\mathbf{x}_k, \tilde{\mathbf{U}}_k). \quad (14)$$

Definition 1: A constraint $\mathbf{c}_j \mathbf{x} \leq b_j$ is called $(\mathbf{x}_k, \mathbf{U}, \mathbb{X}_T, \tilde{\mathbf{U}}_k)$ -redundant (or *redundant* for short) at time $i \in \mathbb{N}_{[1, N]}$ for system (1), if

$$[\mathbf{x}_{1|k}^\top \dots \mathbf{x}_{N|k}^\top]^\top \in \mathcal{H}(\mathbf{x}_k, \tilde{\mathbf{U}}_k) \implies \mathbf{c}_j \mathbf{x}_{i|k} \leq b_j. \quad (15)$$

Loosely speaking, the constraint $\mathbf{c}_j \mathbf{x}_{i|k} \leq b_j$, where $\mathbf{C} := [\mathbf{c}_1^\top, \dots, \mathbf{c}_{n_x}^\top]^\top$ and $\mathbf{b} := [b_1, \dots, b_{n_x}]^\top$, is redundant at time i , if system dynamics, input constraints, terminal set, and MPC cost function already guarantee its satisfaction. Clearly, if $\mathbf{c}_j \mathbf{x}_{i|k} \leq b_j$ is redundant at time i , it can be removed from the MPC problem without changing the minimizer.

Definition 2: A set $\mathcal{X}_N^{\text{red}} := \mathbb{X}_1^{\text{red}} \times \dots \times \mathbb{X}_N^{\text{red}}$ is called a *reduced constraint set* for \mathcal{X}_N , if there are index sets $\mathbb{A}_i \subseteq \mathbb{N}_{[1, n_x]}$, for $i = 1, 2, \dots, N-1$, such that

$$\mathbb{X}_i^{\text{red}} = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{c}_j \mathbf{x} \leq b_j, \text{ for } j \in \mathbb{A}_i\}. \quad (16a)$$

For $i = N$, we define index sets $\mathbb{A}_N \subseteq \mathbb{N}_{[1, n_T]}$ such that

$$\mathbb{X}_N^{\text{red}} = \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{c}_{T,j} \mathbf{x} \leq b_{T,j}, \text{ for } j \in \mathbb{A}_N\}, \quad (16b)$$

where $\mathbf{C}_T := [\mathbf{c}_{T,1}^\top, \dots, \mathbf{c}_{T,n_T}^\top]^\top$ and $\mathbf{b}_T := [b_{T,1}, \dots, b_{T,n_T}]^\top$.

From the above reasoning the next result follows.

Proposition 1: Consider a reduced constraint set $\mathcal{X}_N^{\text{red}}$ for \mathcal{X}_N . If for initial state $\mathbf{x}_k \in \mathbb{X}_f$ and $\tilde{\mathbf{U}}_k \in \mathcal{U}_{f,N}(\mathbf{x}_k)$ it holds that

$$\mathcal{H}(\mathbf{x}_k, \tilde{\mathbf{U}}_k) \cap \mathcal{X}_N = \mathcal{H}(\mathbf{x}_k, \tilde{\mathbf{U}}_k) \cap \mathcal{X}_N^{\text{red}}, \quad (17)$$

then the MPC problem, where (3c) is replaced by $\mathbf{X}_k \in \mathcal{X}_N^{\text{red}}$, has the same minimizer as the original MPC problem, i.e.,

$$\arg \min_{\mathbf{U}_k \in \mathcal{U}_{f,N}(\mathbf{x}_k)} J(\mathbf{x}_k, \mathbf{U}_k) = \arg \min_{\substack{(3b), \\ \mathbf{U}_k \in \mathcal{U}_N \mathbf{X}_k \in \mathcal{X}_N^{\text{red}}}} J(\mathbf{x}_k, \mathbf{U}_k). \quad (18)$$

IV. NUMERICALLY EFFICIENT CONSTRAINT REDUCTION

In this section, we provide numerically efficient procedures to remove state constraints based on the observations in Section III. While Section III describes conditions that can be used to remove redundant constraints, we need efficient and fast techniques to carry out the corresponding calculations online. To this end, we first show that *outer* approximations of the reachable sets and the cost function level set can be exploited in our ca-MPC framework. Second, we provide numerically efficient methods for computing such outer approximations of reachable sets. Finally, we present a method to efficiently construct suitable reduced constraints sets $\mathcal{X}_N^{\text{red}}$ based on these computational methods.

A. Outer approximations in constraint removal

In this section, we show that using outer approximations of $\mathcal{H}(\mathbf{x}_k, \tilde{\mathbf{U}}_k)$ can be exploited in our framework.

Corollary 1: Consider an outer approximation of $\mathcal{H}(\mathbf{x}_k, \tilde{\mathbf{U}}_k)$ in the sense that $\mathcal{H}(\mathbf{x}_k, \tilde{\mathbf{U}}_k) \subseteq \mathcal{H}_+(\mathbf{x}_k, \tilde{\mathbf{U}}_k)$. If

$$\mathcal{H}_+(\mathbf{x}_k, \tilde{\mathbf{U}}_k) \cap \mathcal{X}_N = \mathcal{H}_+(\mathbf{x}_k, \tilde{\mathbf{U}}_k) \cap \mathcal{X}_N^{\text{red}} \quad (19)$$

holds for a reduced constraint set $\mathcal{X}_N^{\text{red}}$ of \mathcal{X}_N , then (17) holds and thus (18).

Clearly, $\mathcal{X}_N^{\text{red}}$ obtained by removing constraints based on the outer approximation $\mathcal{H}_+(\mathbf{x}_k, \tilde{\mathbf{U}}_k)$ has an equal or larger number of constraints than a reduced constraint set that is computed without outer approximations. However, outer approximations can be exploited in our framework by providing a beneficial trade-off between the computational effort of removing redundant constraints and the associated speedup of the MPC problem.

B. Reachable set outer approximations

Computing reachable sets is known to be a computationally intensive task. However, we will exploit an important feature of reachable sets for LTI systems that will make these computations tractable. For ease of exposition, we limit ourselves to the forward reachable set in the explanation below, but the same properties apply to the backward reachable set, *mutatis mutandis*.

A key feature of reachable sets for LTI systems is the decomposition $\vec{\mathcal{R}}(\mathbf{x}, \mathbf{U}) = \Phi \mathbf{x} + \vec{\mathcal{R}}(\mathbf{0}, \mathbf{U})$. Here, $\Phi \mathbf{x}$ is the free state response with initial condition \mathbf{x} and zero input sequence, while $\vec{\mathcal{R}}(\mathbf{0}, \mathbf{U})$ is the forward reachable tube starting at the origin. Crucially, $\vec{\mathcal{R}}(\mathbf{0}, \mathbf{U})$ does *not* depend on the current state and can therefore be computed offline, thereby lowering the online computational burden drastically. Even though the forward reachable tube can be computed offline, it is not necessarily easy and the complexity of the set itself can be intractable in some cases. In such cases, the use of outer approximations of reachable sets can be useful and render the computations tractable while simultaneously lowering the complexity of the set itself. The reduced complexity of the sets themselves will be extensively used to adaptively remove redundant constraints in our ca-MPC framework detailed in Section IV-C.

In this paper, we limit ourselves to bounding boxes and outer ellipsoidal approximations of the reachable sets, although the main ideas also apply to other shapes. A bounding

box approximates the forward reachable tube such that

$$\mathcal{B}(\mathbf{P}_1, \mathbf{q}_1, \mathbf{l}_1) \times \cdots \times \mathcal{B}(\mathbf{P}_N, \mathbf{q}_N, \mathbf{l}_N) \supseteq \vec{\mathcal{R}}(\mathbf{0}, \mathbb{U}), \quad (20a)$$

$$\mathcal{B}(\mathbf{P}, \mathbf{q}, \mathbf{l}) := \{\mathbf{x} \in \mathbb{R}^n \mid -\mathbf{l} \leq \mathbf{P}(\mathbf{x} - \mathbf{q}) \leq \mathbf{l}\}. \quad (20b)$$

Here, $\mathbf{P}_1, \dots, \mathbf{P}_N$ are invertible matrices and $\mathbf{l}_1, \dots, \mathbf{l}_N$ non-negative vectors of appropriate dimension. Similarly, an outer ellipsoidal approximation satisfies

$$\mathcal{E}(\mathbf{L}_1, \mathbf{q}_1) \times \cdots \times \mathcal{E}(\mathbf{L}_N, \mathbf{q}_N) \supseteq \vec{\mathcal{R}}(\mathbf{0}, \mathbb{U}), \quad (21a)$$

$$\mathcal{E}(\mathbf{L}, \mathbf{q}) := \{\mathbf{x} \in \mathbb{R}^n \mid \|\mathbf{L}^\top(\mathbf{x} - \mathbf{q})\|_2 \leq 1\}. \quad (21b)$$

Here, $\mathbf{L}_1, \dots, \mathbf{L}_N$ are positive-definite matrices and $\mathbf{q}_1, \dots, \mathbf{q}_N$ vectors of appropriate dimension. These outer approximations allow for simple set descriptions that can be used to remove redundant constraints efficiently as we will show. In particular, recent developments in ellipsoidal methods allow for numerically efficient reachable set computations, even for large-scale systems [16].

C. Numerically efficient constraint removal

In this section, we use the previously introduced outer approximations to derive efficient methods to identify redundant state constraints. Recall, that redundant constraints can be identified by verifying if the reduced constraint set satisfies (19). One approach to identify and remove a redundant constraint is by solving an optimization problem for each inequality constraint [7]. However, these optimization problems are computationally expensive, therefore, we limit ourselves to numerically efficient, albeit more conservative tests. More specifically, we remove an inequality constraint given by $\mathbf{c}\mathbf{X} \leq b$, when

$$\{\mathbf{X} \in \mathbb{R}^{Nn} \mid \mathbf{c}\mathbf{X} \leq b\} \cap \mathcal{H}(\mathbf{x}_k, \tilde{\mathbf{U}}_k) = \mathcal{H}(\mathbf{x}_k, \tilde{\mathbf{U}}_k) \quad (22)$$

Checking (22) is equivalent to verifying

$$\mathbf{X} \in \mathcal{H}(\mathbf{x}_k, \tilde{\mathbf{U}}_k) \implies \mathbf{c}\mathbf{X} \leq b. \quad (23)$$

Next, we elaborate on (23) for box-shaped and ellipsoidal sets, respectively. At the end of this section, we also discuss how a finite intersections and products of boxes and ellipses can be handled efficiently.

In the box case, we would like to test if

$$\mathbf{X} \in \mathcal{B}(\mathbf{P}, \mathbf{q}, \mathbf{l}) \implies \mathbf{c}\mathbf{X} \leq b, \quad (24)$$

where $\mathcal{B}(\mathbf{P}, \mathbf{q}, \mathbf{l})$ is an Nn -dimensional box-shaped set. In case (24) holds, we set the test outcome of $\mathcal{M}_B(\mathbf{c}, b, \mathcal{B}(\mathbf{P}, \mathbf{q}, \mathbf{l}))$ to true (the constraint can be removed), otherwise we set it to false. Elementary calculations show that (24) is true if and only if

$$\mathbf{c}\mathbf{P}^{-1}(\text{sign}(\mathbf{P}^{-\top} \mathbf{c}^\top) \circ \mathbf{l}) \leq |b - \mathbf{c}\mathbf{q}|, \quad (25)$$

where $\text{sign}(\cdot)$ denotes the element-wise sign and \circ the element wise product. Similar, for ellipses we have

$$\mathbf{X} \in \mathcal{E}(\mathbf{L}, \mathbf{q}) \implies \mathbf{c}\mathbf{X} \leq b, \quad (26)$$

and thus the constraint can be removed if (26) holds. Again, elementary calculations show that (26) holds if and only if

$$\|\mathbf{c}\mathbf{L}^{-\top}\|_2 \leq |b - \mathbf{c}\mathbf{q}|. \quad (27)$$

Hence, we define $\mathcal{M}_E(\mathbf{c}, b, \mathcal{E}(\mathbf{L}, \mathbf{q})) \iff (27)$.

Based on the definition of $\mathcal{H}(\mathbf{x}_k, \tilde{\mathbf{U}}_k)$ (14), it is convenient to extend our constraint removal framework to intersections of boxes and ellipses,

$$\mathcal{H}(\mathbf{x}_k, \tilde{\mathbf{U}}_k) = \bigcap_{p \in \mathbb{N}_{[1, M]}} \mathcal{H}(p, \mathbf{x}_k, \tilde{\mathbf{U}}_k), \quad (28)$$

where $\mathcal{H}(p, \mathbf{x}_k, \tilde{\mathbf{U}}_k)$ is either a box or an ellipse for $p \in \mathbb{N}_{[1, M]}$. Observe that when a constraint $\mathbf{c}\mathbf{X} \leq b$ is redundant for some $p \in \mathbb{N}_{[1, M]}$, i.e., $\mathbf{X} \in \mathcal{H}(p, \mathbf{x}_k, \tilde{\mathbf{U}}_k) \implies \mathbf{c}\mathbf{X} \leq b$, then (23) also holds. Hence, the constraint can be removed.

Crucially, (28) enables the removal of redundant constraints using simpler and faster, sequential redundancy checks based on $\mathcal{H}(p, \mathbf{x}_k, \tilde{\mathbf{U}}_k)$. Additionally, when $\mathcal{H}(\mathbf{x}_k, \tilde{\mathbf{U}}_k)$ is given by

$$\mathcal{H}(\mathbf{x}_k, \tilde{\mathbf{U}}_k) = \mathcal{H}^1(\mathbf{x}_k, \tilde{\mathbf{U}}_k) \times \cdots \times \mathcal{H}^N(\mathbf{x}_k, \tilde{\mathbf{U}}_k), \quad (29)$$

as is for the reachable tubes (10), then it is also possible to process $\mathcal{H}^i(\mathbf{x}_k, \tilde{\mathbf{U}}_k)$ and corresponding constraint set $\mathbb{X}_i^{\text{red}}$ sequentially for each time $i = 1, 2, \dots, N$.

V. CASE STUDY: HYPERTHERMIA TREATMENT

In this section, we demonstrate our proposed ca-MPC framework for numerically efficient constraint removal with a hyperthermia cancer treatment case study. A hyperthermia treatment involves heating a tumor thereby inducing a local fever. As a result, the tumor is sensitized to radio and chemo therapies, making these therapies more effective [11]–[14]. Critically, tissue temperature constraints ensure the safety and comfort of the patient [17]. The temperature constraint is defined on the *continuous* spatial domain and therefore results in many state constraints after spatial discretization, which is a typical scenario for MPC-based PDE control.

The remainder of this section is structured as follows. First, the system dynamics and MPC problem are introduced. Second, the ca-MPC relevant sets are computed. Last, we compare the results obtained using ca-MPC and the original non-reduced MPC setup.

A. System description

We model the system dynamics in the hyperthermia case study using the 1D heat equation with distributed control, convective boundary conditions, and additional convection on the spatial domain to model the body's thermoregulatory response. This leads for the normalized spatial coordinates $r \in [0, 1]$ and time $t \in \mathbb{R}_{\geq 0}$ to

$$\dot{T}(r, t) = \alpha \nabla^2 T(r, t) - \beta T(r, t) + Q(r, t), \quad (30a)$$

$$Q(r, t) = B_1(r)u_1(t) + B_2(r)u_2(t) \quad (30b)$$

with boundary conditions

$$\nabla T(0, t) = \gamma T(0, t), \quad \nabla T(1, t) = -\gamma T(1, t). \quad (31)$$

The parameters are given by $\alpha = 2.5 \cdot 10^{-4}$, $\beta = 10^{-2}$, and $\gamma = 2.5 \cdot 10^{-3}$. Note that the 1D setup enables the comparison with a non-reduced MPC setup, which would become intractable for 2D and 3D cases.

To ensure patient safety and efficacy of the treatment, we constrain the healthy tissue temperature increase to five degrees with respect to the body temperature and seven degrees for the tumor on $r \in [0.6, 0.9]$. This constraint is illustrated in Fig. 1 by the function $T_{\max}(r)$.

Heating is applied to the patient using RF-applcator [12], which is modeled through the actuator profiles $B_1(r)$ and $B_2(r)$, see Fig. 1. Observe that the heat deposition is not limited to the tumor region, this is also typically observed in real-RF-based hyperthermia treatments. Healthy tissue regions with high power deposition are typically referred to

as hot spots. These hot spots also motivate the use of control strategies that explicitly include state constraints to ensure patient safety. To ensure that the actuator applies no negative heat load and respects power limits, the input constraints are given by $0 \leq u_1 \leq 1$ and $0 \leq u_2 \leq 1$.

Next, we spatially and temporally discretize (30) and (31), using a central-difference scheme and zero-order hold, respectively, to obtain a model as in (1). Here, each element of $\mathbf{x}_k \in \mathbb{R}^n$ denotes a nodal temperature defined on an equidistant grid of n points, i.e., $\{0, \frac{1}{n-1}, \frac{2}{n-1}, \dots, 1\}$. Note that the case study will use a varying number of discretization points n to demonstrate our ca-MPC framework.

In order to heat the tumor, we use a quadratic cost function to track the reference \mathbf{x}_r illustrated in Fig. 1. The reference \mathbf{x}_r and \mathbf{u}_r are obtained through an offline optimization procedure to maximize the tumor temperature. The nominal MPC setup is now given as

$$\underset{\mathbf{U}_k}{\text{minimize}} \quad \sum_{i=1}^{10} \|\mathbf{x}_{i|k} - \mathbf{x}_r\|_2^2 + \|\mathbf{u}_{i-1|k} - \mathbf{u}_r\|_2^2, \quad (32a)$$

$$\text{subject to} \quad \mathbf{X}_k = \Phi \mathbf{x}_k + \Gamma \mathbf{U}_k, \quad (32b)$$

$$\mathbf{x}_{i|k} \leq \mathbf{T}_{\max}, \quad i = 1, \dots, N-1, \quad (32c)$$

$$\mathbf{x}_{N|k} \leq \mathbf{T}_{\text{terminal}}, \quad (32d)$$

$$\mathbf{0} \leq \mathbf{u}_{i|k} \leq \mathbf{1}, \quad i = 0, \dots, N-1. \quad (32e)$$

Here, $\mathbf{T}_{\max} \in \mathbb{R}^n$ denotes the spatially discrete temperature constraint, see Fig. 1. The terminal set constraint (32d) maximizes $\|\mathbf{T}_{\text{terminal}}\|_1$ while satisfying $(\mathbf{A} - \mathbf{I})\mathbf{T}_{\text{terminal}} \leq \mathbf{0}$ and $\mathbf{T}_{\text{terminal}} \leq \mathbf{T}_{\max}$ to ensure positive invariance with auxiliary input $\mathbf{u} = \mathbf{0}$. For technical details we refer the interested reader to [18].

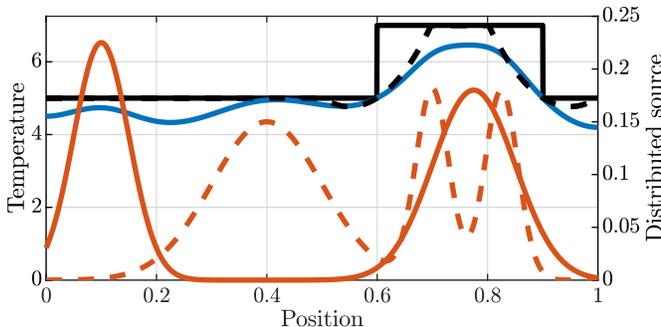


Fig. 1. Temperature constraint $T_{\max}(r)$ (—), terminal constraint $T_{\text{terminal}}(r)$ (---), the state reference \mathbf{x}_r (—), and the distributed source $B_1(r)$ (—), and $B_2(r)$ (---), for $r \in [0, 1]$.

B. ca-MPC setup

We approximate the forward and backward reachable sets using axis-aligned boxes and exploit the observation that our model (30) is a positive system. As a result, the forward reachable sets are outer-approximated by

$$\vec{\mathcal{R}}^i(\mathbf{0}, \mathbf{U}) := \{\mathbf{x} \in \mathbb{R}^n \mid \mathbf{0} \leq \mathbf{x} \leq \sum_{k=0}^{i-1} \mathbf{A}^k \mathbf{B} \mathbf{1}\}. \quad (33)$$

Here, $\mathbf{1} \in \mathbb{R}^2$ denotes a column of ones. Moreover, note that (33) can be trivially transformed to (20b). Conceptually, (33) denotes a set based on the maximum temperature that can be achieved with both inputs. Next, the backward reachable set is approximated by

$$\overleftarrow{\mathcal{R}}^i(\mathbb{X}_T, \mathbf{0}) := \{\mathbf{x} \in \mathbb{R}^n \mid 0 \leq x_j \leq T_{j,\text{terminal}} + \delta_{i,j}, \quad (34)$$

$$\text{for } j = 1, \dots, n\},$$

where $\delta_{i,j} := \max_{\mathbf{A}^{N-i} \mathbf{e}_j \delta \leq (\mathbf{I} - \mathbf{A}^{N-i}) \mathbf{T}_{\text{terminal}}}$ and \mathbf{e}_j denotes the j -th standard basis vector. Conceptually, (34) is the union of maximum single-state perturbations that satisfy the terminal set after $N - i$ time-steps with zero input, i.e., $\mathbf{u} = \mathbf{0}$. Again, (34) can be trivially transformed to (20b).

To obtain the cost function level set, we substitute (3b) into the MPC cost function (32a), which yields

$$J(\mathbf{x}_k, \mathbf{U}_k) = \frac{1}{2} \mathbf{U}_k^\top \mathbf{G} \mathbf{G}^\top \mathbf{U}_k + \mathbf{f}^\top(\mathbf{x}_k) \mathbf{U}_k + c(\mathbf{x}_k), \quad (35)$$

where \mathbf{G} , $\mathbf{f}(\mathbf{x}_k)$, and $c(\mathbf{x}_k)$ denote the respective matrix, vector, and scalar obtained by rewriting (32). Given an initial state $\mathbf{x}_k \in \mathbb{X}_f$, and input sequence $\tilde{\mathbf{U}}_k \in \mathcal{U}_{f,N}(\mathbf{x}_k)$, the cost function level set is given by the ellipse $\mathcal{J}_u(\mathbf{x}_k, \tilde{\mathbf{U}}_k) := \mathcal{E}(\mathbf{L}_k, \mathbf{q}_k)$ with $\mathbf{L}_k := \mathbf{G} \|\mathbf{G}^\top (\tilde{\mathbf{U}}_k - \mathbf{q}_k)\|_2^{-1}$ and $\mathbf{q}_k := -(\mathbf{G} \mathbf{G}^\top)^{-1} \mathbf{f}(\mathbf{x}_k)$.

Our framework allows for any feasible input sequence $\tilde{\mathbf{U}}_k \in \mathcal{U}_{f,N}(\mathbf{x}_k)$, but a natural choice is a shifted version of the previous optimal sequence and appended by the auxiliary control law, $\tilde{\mathbf{U}}_k = [\mathbf{u}_{1|k-1}^* \dots \mathbf{u}_{N-1|k-1}^* \mathbf{0}^\top]^\top$. Crucially, note that $\tilde{\mathbf{U}}_k$ ensures feasibility as the terminal set was chosen to be positively invariant for $\mathbf{u} = \mathbf{0}$.

For efficiency reasons, we sequentially remove state constraints for each time step and the forward-, backward-, and cost function level set, as mentioned in (28) and (29). More specifically, we compute each reduced constraint set $\mathbb{X}_i^{\text{red}}$ for $i = 1, \dots, N-1$ using

$$\mathbb{A}_i = \{j \in \mathbb{N}_{[1, n_x]} \mid \neg [\mathcal{M}_B(\mathbf{c}_j, b_j, \vec{\mathcal{R}}^i) \vee \quad (36a)$$

$$\mathcal{M}_E(\hat{\mathbf{c}}_{j,i}, \hat{b}_{j,i}, \mathcal{J}_u(\mathbf{x}_k, \tilde{\mathbf{U}}_k))]\}, \quad (36b)$$

and for $i = N$,

$$\mathbb{A}_N = \{j \in \mathbb{N}_{[1, n_T]} \mid \neg [\mathcal{M}_B(\mathbf{c}_{T,j}, b_{T,j}, \vec{\mathcal{R}}^N) \vee \quad (36c)$$

$$\mathcal{M}_E(\hat{\mathbf{c}}_{T,j}, \hat{b}_{T,j}, \mathcal{J}_u(\mathbf{x}_k, \tilde{\mathbf{U}}_k))]\}. \quad (36d)$$

Here $\mathbf{c}_j, b_j, \mathbf{c}_{T,j}, b_{T,j}$ denote the state constraints $\mathbf{x} \leq \mathbf{T}_{\max}$ and $\mathbf{x} \leq \mathbf{T}_{\text{terminal}}$ written as $\mathbf{C} \mathbf{x} \leq \mathbf{b}$ and $\mathbf{C}_T \mathbf{x} \leq \mathbf{b}_T$, respectively. The operator \vee , denotes the logical OR, and \neg the logical NOT. For (36b) and (36d), $\hat{\mathbf{c}}_{j,i}$ and $\hat{b}_{j,i}$ denote the state constraint $\mathbf{c}_j \mathbf{x}_{i|k} \leq b_j$ at time i as $\hat{\mathbf{c}}_{j,i} \mathbf{U}_k \leq \hat{b}_{j,i}$ using (3b). This allows for $\mathcal{M}_E(\cdot)$ to be evaluated using the Nm -dimensional \mathbf{U}_k -space in contrast to the Nn -dimensional \mathbf{X}_k -space, which is typically faster when $m < n$.

C. Results

To illustrate the obtained computational improvements we compare the maximum computation time of our proposed ca-MPC framework with respect to a standard MPC setup, with an increasing number of states $n \in [100, 2000]$. Additionally, for our proposed method, we differentiate between the computation time of the pre-solve and quadratic program to provide insight in the overhead of our method. In Fig. 2, the resulting computation time is shown. Clearly, our framework can achieve speedups of two-orders of magnitude. Moreover, the overhead from the pre-solve step scales well with an increasing number of constraints.

Finally, in Fig. 3 the computation time as well as the percentage of total constraints are shown for a single ca-MPC simulation with $n = 2000$. Here, the adaptive nature of the method is clearly seen as the number of included constraints

varies during the simulation. During the first 30 seconds, barely any constraints are included as the state cannot reach the constraints within the prediction horizon. Hereafter, the system approaches the temperature constraints, which results in the need to include more constraints as seen in Fig. 3. Healthy tissue temperatures approaching the temperature constraints are also typical for real hyperthermia treatments due to the presence of hot spots. After approximately 60 seconds, the cost function level set starts to shrink as the system approaches the steady-state reference. As result, more constraints can be removed (due to the cost level set) even though the constraints are approached even closer.

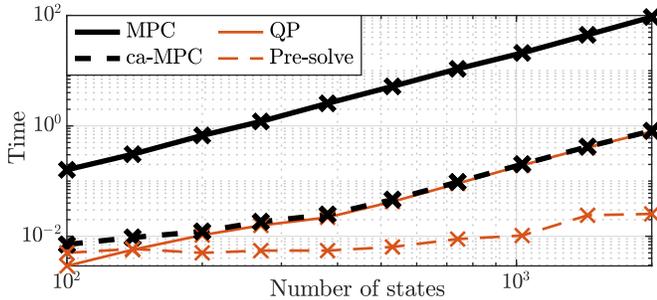


Fig. 2. The maximum computation time of our constraint-adaptive MPC method and a standard MPC for increasingly fine spatial discretizations. For our framework the solver and pre-solve time are also shown.

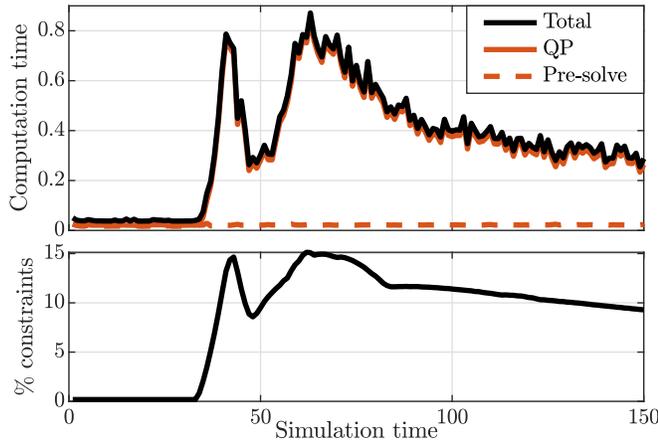


Fig. 3. Computation time for ca-MPC and the percentage of constraints that are included in the MPC problem for $n = 2000$. The computation time correlates well with the number of constraints. Moreover, the pre-solve overhead remains approximately constant throughout the simulation.

VI. CONCLUSION AND OUTLOOK

In this paper, we presented a novel *constraint-adaptive* MPC scheme that can enable real-time feasibility for systems with many state constraints. Crucially, our framework guarantees the same closed-loop performance, recursive feasibility and constraints satisfaction properties as the non-reduced MPC scheme. The presented method exploits fast outer approximation calculations of reachable sets as well as the cost function level set, to efficiently remove redundant constraints in an online setting. Moreover, the majority of the computational complexity is offline by exploiting key properties of reachable sets for LTI systems. Hence, we believe the method is applicable to a large variety of systems

with many state constraints. A hyperthermia cancer treatment case study demonstrated the benefits of ca-MPC showing up to two orders of magnitude reduction in computation time with respect to a standard MPC setup without loss of closed-loop performance.

This paper focused on computationally cheap methods in order to obtain low computational overhead for the constraint removal pre-solve step. Clearly, less conservative checks can be performed to remove more redundant constraints. For example, state constraints that are not removed are currently not used to identify more redundant constraints. Furthermore, specific solver interactions with, e.g., interior-point and active-set solvers are of interest. These are topics for future work next to applying our framework to a full (3D) application in simulation and experiments of a real hyperthermia treatment.

REFERENCES

- [1] D. Q. Mayne, "Model predictive control: Recent developments and future promise," *Automatica*, vol. 50, no. 12, pp. 2967–2986, 2014.
- [2] A. Bemporad, F. Borrelli, and M. Morari, "Model predictive control based on linear programming - the explicit solution," *IEEE Trans. on Automatic Control*, vol. 47, no. 12, pp. 1974–1985, 2002.
- [3] S. Hovland, K. Willcox, and J. T. Gravdahl, "MPC for Large-Scale Systems via Model Reduction and Multiparametric Quadratic Programming," in *Conf. Decision and Control*, 2006, pp. 3418–3423.
- [4] W. Heemels, B. Genuit, and L. Lu, "Approximation of explicit model predictive control using regular piecewise affine functions: an input-to-state stability approach," *IET Control Theory & Applications*, vol. 6, no. 8, pp. 1015–1028, 2012.
- [5] J. Jerez, E. Kerrigan, and G. Constantinides, "A condensed and sparse QP formulation for predictive control," in *Conf. Decision and Control and European Control Conf.*, 2011, pp. 5217–5222.
- [6] J. Rawlings, D. Mayne, and M. Diehl, *Model Predictive Control: Theory, Computation, and Design*, 2nd ed. Nob Hill, 2019.
- [7] S. Paulraj and P. Sumathi, "A Comparative Study of Redundant Constraints Identification Methods in Linear Programming Problems," *Mathematical Problems in Engineering*, pp. 1–16, 2010.
- [8] M. Jost and M. Mönnigmann, "Accelerating model predictive control by online constraint removal," in *Conf. Decision and Control*, vol. 35, 2013, pp. 5764–5769.
- [9] M. Jost, G. Pannocchia, and M. Mönnigmann, "Online constraint removal: Accelerating MPC with a Lyapunov function," *Automatica*, vol. 57, pp. 164–169, 2015.
- [10] S. Nouwens, B. de Jager, M. Paulides, and W. Heemels, "Constraint-adaptive MPC for large-scale systems: Satisfying state constraints without imposing them," in *IFAC Conf. on Nonlinear Model Predictive Control*, vol. 54, 2021, pp. 232–237.
- [11] D. Deenen, B. Maljaars, L. Sebeke, B. de Jager, E. Heijman, H. Grull, and W. Heemels, "Offset-Free Model Predictive Temperature Control for Ultrasound-Based Hyperthermia Cancer Treatments," *IEEE Trans. on Control Systems Technology*, pp. 1–15, 2020.
- [12] R. Hendrikk, S. Curto, B. De Jager, E. Maljaars, G. Van Rhoon, M. Paulides, and W. Heemels, "POD-Based Recursive Temperature Estimation for MR-Guided RF Hyperthermia Cancer Treatment: A Pilot Study," in *Conf. Decision and Control*, 2018, pp. 5201–5208.
- [13] J. van der Zee, "Heating the patient: A promising approach?" *Annals of Oncology*, vol. 13, no. 8, pp. 1173–1184, 2002.
- [14] D. Hensley, R. Orendorff, E. Yu, C. Danielson, V. Salgaonkar, and C. Diederich, "Model predictive control for treating cancer with ultrasonic heating," in *Am. Control Conf.*, 2015, pp. 220–225.
- [15] S. Boyd and L. Vandenberghe, *Convex optimization*, 7th ed. Cambridge: Cambridge University Press, 2009.
- [16] A. Halder, "Smallest Ellipsoid Containing p-Sum of Ellipsoids with Application to Reachability Analysis," *arXiv*, Jun 2018.
- [17] P. S. Yarmolenko, E. J. Moon, C. Landon, A. Manzoor, D. W. Hochman, B. L. Vigiante, and M. W. Dewhurst, "Thresholds for thermal damage to normal tissues: An update," *International Journal of Hyperthermia*, pp. 320–343, 2011.
- [18] G. Bitsoris, "Positively invariant polyhedral sets of discrete-time linear systems," *International Journal of Control*, pp. 1713–1726, 1988.