Hierarchical mixed-integer model predictive control for real-time large-area MR-HIFU hyperthermia

van Wordragen, J.

Award date:
2019

Department of Mechanical Engineering

Control Systems Technology Group

Hierarchical mixed-integer model predictive control for real-time large-area MR-HIFU hyperthermia

J. van Wordragen

CST2019.074

Master's thesis
M.Sc. Mechanical Engineering

Supervisors:
Ir. D.A. Deenen
Dr. Ir. Bram de Jager
Prof. Dr. Ir. W.P.M.H. Heemels

September 23, 2019

ACKNOWLEDGEMENTS

This thesis marks the end of my time as a student at the Eindhoven University of Technology. I would like to thank my direct supervisor Daniel Deenen for the support throughout the whole project, and the fun trips to Cologne where we did a good job in last-minute debugging to achieve the best experimental results possible. Special thanks to Lukas Sebeke for his helping hand during these experiments. In addition, I would like to thank my supervisors Bram de Jager and Maurice Heemels for sharing their knowledge and insights during our bi-weekly meetings. Furthermore, a big thanks to all fellow students in the DCT lab for the necessary distractions from serious work, and countless games of table soccer.

Jesper van Wordragen,
September 2019

Hierarchical mixed-integer model predictive control for real-time large-area MR-HIFU hyperthermia

Jesper van Wordragen
Control Systems Technology Group
Department of Mechanical Engineering
Eindhoven University of Technology
Eindhoven, The Netherlands
j.v.wordragen@student.tue.nl

***Abstract*—In this project, a novel model predictive control (MPC) scheme is proposed to make MPC real-time feasible for large-area magnetic-resonance-guided high-intensity focused ultrasound (MR-HIFU) hyperthermia (HT) treatments. HT treatments involve the local heating of tumors to enhance the cancer-killing potential of radio- and chemotherapies. For the treatment of larger tumorous regions in MR-HIFU HT, mechanical movement of the heating actuator is required due to the limited range in which heat can be generated from a fixed actuator location. Due to system restrictions, the actuator locations are fixed to a finite amount of a-priori determined coordinates. Hence, including the actuator movement into the MPC problem leads to the formulation of a computationally heavy mixed-integer programming (MIP) problem. To achieve computation times that are feasible for real-time implementation, a hierarchical model predictive control (HMPC) structure is proposed, in which the original MIP problem is decomposed into two control layers. A high-level MPC algorithm using a low-complexity prediction model is used to approximate the original MIP problem, which serves the main goal of determining the actuator trajectory. The resulting trajectory is then provided to a low-level MPC which refines the optimal (continuous) inputs based on a high-complexity model. Additional contributions are the investigation and comparison of appropriate reduction methods for the derivation of the high-level prediction model, optimization of actuator locations to optimally treat irregularly shaped target areas, and a stability analysis of HMPC applied to hyperthermia. Both simulation studies and experiments are conducted to validate the performance of the HMPC controller.**

***Index Terms*—Healthcare, large-area hyperthermia, hierarchical MPC, cancer treatment, high-intensity focused ultrasound.**

I. Introduction

In cancer treatment, hyperthermia is used as an adjuvant technique to improve treatment success. Mild local/regional hyperthermia involves the heating of tumorous tissue to temperatures of $39 - 45$ [°C] for a duration of 90 minutes. A significant increase in effectiveness of primary methods such as chemo- and radiotherapy is shown when used in combination with hyperthermia, see, for example, [1–3]. Besides, the use of temperature-sensitive liposomes for heat-mediated drug release is enabled, resulting in more efficient delivery of anticancer drugs to specifically targeted regions, see [4, 5]. If not heated, healthy tissue is not affected by the treatment, which makes it highly appealing for clinical application to enhance treatment effectiveness and improve quality of life for the patients in therapy.

Hyperthermia treatments come in multiple forms in which different types of energy are used to apply heat, including microwave, radiofrequency, and ultrasound. In this work, the main focus will be on the magnetic-resonance-guided high-intensity focused ultrasound (MR-HIFU) technique, which is a completely noninvasive procedure in which sound waves are used to deposit energy in the patient's body with a spatial accuracy in the millimiter range. The thermal response of the treated area is monitored via volumetric temperature maps that are obtained using an MRI scanner, see [6]. This enables the use of closed-loop control to realize and maintain the desired temperature distribution.

In Figure 1, a schematic representation of the HIFU heating is depicted. For a fixed transducer position, the focal point can be moved over the focal plane by changing the amplitudes and phases of the individual driving signals. Modulation of these settings to move the focal spot is referred to as electronic beam steering.

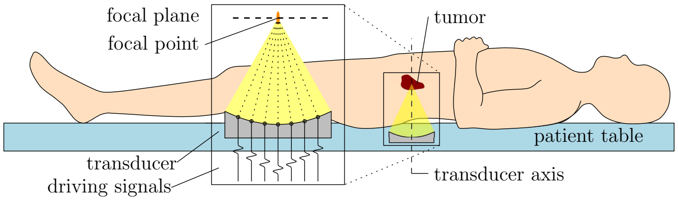


Fig. 1: Schematic representation of HIFU heating. Image reproduced from [7].

Current clinical implementations typically use a mostly predetermined, ad-hoc, sonication plan that is possibly extended with simple feedback control. Examples are [8, 9] and [10], in which binary strategies to scale the sonication power and time, and proportional-integral-derivative (PID) control are used, respectively. Using these types of feedback control, the future thermal behavior and the input constraints (resulting from actuator limitations) are not taken into account, which negatively impacts treatment quality.

More recent works propose a more advanced type of control, namely model predictive control (MPC), see [7, 11].

In MPC, a mathematical model is used to make predictions about the system's future behavior, and control inputs are calculated by solving an optimization that is subjected to a cost function and constraints, see [12, 13]. Closed-loop control is achieved by sequentially solving the MPC problem but only applying the first input of the predicted sequence, after which the MPC problem is solved again using new measurement data. Therefore, MPC control enables to take future thermal behavior of the patient's body into account and is capable of dealing with input constraints. Hence, MPC has the potential to provide superior performance for hyperthermia treatments. In particular, in [7], advanced MPC control is presented using offset-free control to remove the steady-state offset resulting from plant-model mismatch.

New challenges arise when treating larger tumors, as this requires heating of a larger target area. Unfortunately, the lateral movement of the focal spot using electronic beam steering is limited in range (≤ 16 [mm] in diameter), impeding the treatment of larger areas. To enable the treatment of such large target areas, mechanical movement of the HIFU transducer is needed, see [14]. In [14], a binary feedback algorithm is used in which the mechanical location of the transducer typically follows some predefined trajectory or is chosen ad-hoc, resulting in suboptimal trajectories and hence reduced treatment quality.

In [15], the mechanical movement of the transducer is included in the MPC problem. Due to system limitations, which will be discussed later on, the position of the transducer is fixed to a finite number of predetermined, discrete, locations. This introduces an integer part into the MPC problem in which now simultaneous optimization occurs over both the continuous input sequence, and an integer sequence that describes the transducer path. The resulting optimization can then be formulated as a mixed-integer programming problem (MIP). In [15], it was shown that using mixed-integer MPC (MI-MPC) had great potential for large-area MR-HIFU, since the results were promising in terms of performance. Unfortunately, due to the large state dimension used in the prediction model combined with the integer problem introduced by including the transducer movement, real-time computational feasibility of the MI-MPC formulation has proven difficult.

The main contribution of this paper is the formulation of a hierarchical MPC (HMPC) controller suited for large-area MR-HIFU treatments to reduce the high computational complexity of the MI-MPC problem. This is done by decomposing the control into two levels. A high-level MPC controller uses a lower-complexity prediction model to determine the cell-to-cell transducer trajectory and communicates this path to a low-level controller. At the low-level, a second MPC algorithm is used to determine the optimal inputs for the provided transducer path using a higher-complexity prediction model. By doing so, the low-level controller solves a computationally fast quadratic programming problem (QP), since the integer variables are provided by the high level.

Additional contributions are the formulation of the reduced order prediction models and enabling treatment of irregularly shaped tumors. The high-level prediction models are obtained through model reduction. Multiple methods are compared by means of simulation. Moreover, in previous works only circular treatment areas are considered. However, as tumors may be irregularly shaped in reality, the placement of the transducer positions for arbitrarily shaped target areas is discussed in this paper. Ultimately, the goal of this work is to provide a significant step forward in fulfilling the potential of MPC control in large-area hyperthermia treatments.

The sequel of this paper is organized as follows. In Section II a description of the system is given with a derivation of a mathematical model to capture the thermal behavior. Next, the MI-MPC problem based on the work in [15] is discussed in Section III, followed up by Section IV in which the proposed hierarchical controller is elaborated on. Several reduced-order prediction models are discussed in Section V followed by Section VI in which the optimization of transducer locations for arbitrarily shaped target areas is treated. Afterwards, a stability analysis is included in which closed-loop stability and performance of the HMPC approach is discussed in Section VII. Both simulation- and experimental results are shown to validate the performance of the proposed controller in Section VIII and Section IX, respectively. Finally, the key achievements are summarized and suggestions for future work are included in Section X.

II. System description

This section introduces the hyperthermia-treatment setup, and discusses the modeling procedure to obtain a state space model of the system's thermal dynamics. In addition, the observer model that is used for state estimation is discussed, and the goal of HT treatments is elaborated on.

A. *MR-HIFU setup*

The controller designs in this paper are specifically suited for implementation on the Profound Sonalleve MR-HIFU platform, which is a commercial system that is already being clinically used, see [16]. The Sonalleve system offers incision-free alternatives to traditional surgical treatments of uterine fibroids, and enables incision-free, radiation-free palliative treatment of pain associated with bone metastases. The setup is a combination of an MRI scanner used for thermometry and a dedicated patient table in which the HIFU transducer is integrated. Figure 2 shows an image of the setup. The HIFU applicator uses a phased-array transducer to generate the ultrasound waves. Electronic beam steering is used to laterally move the focal spot over the focal plane. For this particular system, the focal spot can be chosen anywhere on the focal plane within an 16 millimeter diameter circle centered around the transducer axis. The resulting power deposition is relatively distributed in the beam's axial direction compared to its narrow profile in the radial direction. This motivates the choice to only consider temperature control in the two-dimensional focal plane.

To acquire accurate MRT data, baseline images are required for each transducer position. That is, due to the change in magnetic field when the heating actuator changes position, a baseline image is required corresponding to each position. This limits the possible locations from which sonication is allowed to an a-priori defined set of discrete positions. Using the baseline images, the MRI scanner obtains a new temperature map ever 3.2 seconds. Motivated by this, the computational limit for the MPC optimization problem is set to 3 seconds such that during the acquisition of a temperature map, a new MPC solution can be obtained. Also, HIFU-heating is disabled during movement of the transducer, since reliability of the MRT data is reduced due to the changing magnetic field.

Fig. 2: Sonalleve MR-HIFU system

B. Thermal process modeling

For the subset in the focal plane $\Omega \subset \mathbb{R}^2$ that describes the patient domain, the prediction model, as used in the model predictive controller, captures the thermal behavior of the body in the presence of HIFU heating. In previous works, see e.g. [7, 15], Pennes' bioheat equation as described in [17] is successfully used to obtain an accurate prediction model. The tissue's temperature evolution is modeled as

$$\rho c \frac{\partial T(r,t)}{\partial t} = k\nabla^2 T(r,t) + Q(r,t) - w_b c_b (T(r,t) - T_b), \quad (1)$$

in which $T : \Omega \times \mathbb{R}_{\geq 0} \to \mathbb{R}$ is the temperature profile, such that $T(r,t)$ denotes the temperature at location $r = [r_x, r_y]^\top \in \Omega$ at time instant t, and $Q : \Omega \times \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ describes the power deposition density. The parameters $\rho, c, k, w_b, c_b \in \mathbb{R}_{\geq 0}$ are the tissue- and blood parameters being the mass density, specific heat capacity and thermal heat conductivity of the tissue, and the blood perfusion coefficient of the tissue and specific heat capacity of blood, respectively. For this work, all parameters are assumed to be constant and homogeneous over the entire patient domain.

The power deposition density is modeled as in [7], and is given by

$$Q(r,t) = F(r,t)P(t), \quad (2)$$

with $F : \Omega \times \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ the acoustic deposition intensity profile and $P : \mathbb{R}_{\geq 0} \to \mathbb{R}_{\geq 0}$ the sonication power. The acoustic deposition intensity profile F is modeled as a radially symmetric two-dimensional Gaussian distribution centered around the focus location $r_f : \mathbb{R}_{\geq 0} \to \Omega$

$$F(r,t) = \frac{\alpha}{2\pi\sigma_f^2} \exp\Big(- \frac{\|r - r_f(t)\|^2}{2\sigma_f^2} \Big), \quad (3)$$

with $r \in \Omega$, $t \in \mathbb{R}_{\geq 0}$, standard deviation $\sigma_f = 2.4$ [mm] and scaling factor $\alpha \in \mathbb{R}_{\geq 0}$.

C. State space model

Using (1)-(3), a discrete-time state space realization of the thermal model can be obtained. The model is discretized spatially using the central difference scheme and in time using forward Euler discretization. This results in a prediction model that is suited for control in the form

$$x_{k+1} = Ax_k + Bu_k, \quad (4a)$$

$$y_k = x_k + v_k, \quad (4b)$$

with $x_k \in \mathbb{R}^n$ the state vector representing the voxel temperature elevation with respect to the body's mean temperature of 37 [°C] at instant $k \in \mathbb{N}$, which connects to real time through $t_k = kT_s$, with sampling interval T_s. The input-to-state matrix $B \in \mathbb{R}^{n \times m}$ captures the effect of applying input $u_k \in \mathbb{R}^m$ to the system at instant k. Within a single sampling interval, the focus location r_f can be rapidly scanned over multiple points. These points are referred to as sonication points and are determined a-priori, such that u_k describes the average power applied per sonication point. The possible sonication points are captured in the set $\mathcal{S} = \{s_1, s_2, \dots, s_m\}$, with $s_i = [r_x \ r_y]^\top \in \mathbb{R}^2$, $i = 1, 2, \dots, m$, the location of a single sonication point. The locations of sonication points are not chosen arbitrarily, but are limited to be within a 16 millimeter diameter of a transducer location $\tau_i = [r_x \ r_y]^\top \in \mathbb{R}^2$, $i = 1, 2, \dots, n_t$, contained in the set $\mathcal{T} = \{\tau_1, \tau_2, \dots, \tau_{n_t}\}$. Therefore, the points at which power can be deposited at instant k depend on the current transducer location and are a subset of $\mathcal{S}$. More specific, the input vector u_k is constructed as

$$u_k = \begin{bmatrix} u_k^1 \\ \vdots \\ u_k^{n_t} \end{bmatrix} \in \mathbb{R}^m, \quad (5)$$

where $u_k^i \in \mathbb{R}^{n_s}$, for $i = 1, 2, \dots, n_t$, and $n_s = m/n_t$ being the number of sonication points associated to a single transducer location. Each vector u_k^i now captures the inputs at sonication points s_q, with $q = (i-1)n_s + 1, \dots, in_s$, which are associated to transducer location τ_i. Hence, given that the transducer is at one of the locations in $\mathcal{T}$ at instant k, only the vector u_k^i that corresponds to the current location can contain nonzero inputs, whereas all the other inputs are zero. In addition, in case the transducer is moving at instant k, all entries of u_k are zero since heating is disabled during movement of the actuator.

The output $y_k \in \mathbb{R}^n$ contains the measured temperatures corresponding to the voxel locations $v_i = [r_x \ r_y]^\top \in \Omega$,

$i = 1, 2, \ldots, n_v$, in the set $\mathcal{V} = \{v_1, v_2, \ldots, v_{n_v}\}$. By choosing the spatial discretization to specifically match the measurement grid, a full state measurement is enabled, and it follows that $n = n_v$. In all simulations and experiments contained in this work, the measurement grid consists of $n = 1296$ voxel locations corresponding to a 36×36 grid with a uniform spacing of 2.25 [mm] between two neighboring voxels.

The measurements are corrupted by noise represented by $v_k \in \mathbb{R}^n$. To obtain improved temperature estimates $\hat{x}_k$ with respect to noisy measurement data, a model-based observer is used. The state estimator is designed as a Luenberger observer following

$$\hat{x}_k = \hat{y}_k^- + L(y_k - \hat{y}_k^-), \tag{6}$$

where

$$\hat{y}_k^- = A\hat{x}_{k-1} + Bu_{k-1} \tag{7}$$

denotes the model-based measurement estimate at instant k before applying the correction using output injection to obtain the current state estimate.

D. Goal of hyperthermia treatments

In the sequel, the region of interest in which an elevated temperature is desired is denoted as $\mathcal{R} \subset \Omega$. Hence, the subset $\mathcal{V}^{\mathcal{R}} = \mathcal{V} \cap \mathcal{R}$ contains the voxel locations that are located inside the region of interest. The goal of the MR-HIFU HT treatments is now to elevate the temperature of the voxels in $\mathcal{V}^{\mathcal{R}}$ to match the reference profile $x^r \in \mathbb{R}^{n_{\mathcal{R}}}$, with $n_{\mathcal{R}}$ the number of voxels in $\mathcal{V}^{\mathcal{R}}$. The desired temperature range in the entire patient domain Ω is specified by an upper and lower temperature bound which are represented by $\bar{x} \in \mathbb{R}^n$ and $\underline{x} \in \mathbb{R}^n$, respectively. Within the desired temperature range, preference is given to the temperature distribution described by x^r. The numeric values for the temperature bounds and setpoint are based on the work in [7], where a flat temperature distribution of five degrees elevation is set as the reference. The lower bound features an elevated plateau of four degrees on $\mathcal{R}$ and is zero on $\Omega \setminus \mathcal{R}$. The upper bound is defined as an elevated plateau of six degrees on the sub-space consisting of the ROI with an outwards offset of 7 millimeters, and is set to three degrees outside of this area.

III. Mixed-integer MPC

In this section, the MI-MPC problem is discussed based on the work presented in [15], which aims to achieve the aforementioned treatment goal.

A. Optimization variables

In the MI-MPC formulation, simultaneous optimization occurs over the states of the system, the inputs to be applied and the transducer trajectory. The state vector $x_k \in \mathbb{R}^n$ contains the temperatures at instant k corresponding to the discrete locations in $\mathcal{V}$. The input vector $u_k \in \mathbb{R}^m$ contains the power deposition at the sonication points in $\mathcal{S}$, at instant k. The position of the transducer at instant k is captured via the vector

$$\delta_k^\top = [\delta_k^1, \delta_k^2, \ldots, \delta_k^{n_t}] \in \{0,1\}^{n_t},$$

in which δ_k^i equals 1 if the transducer is at position τ_i at instant k, or when it is traveling towards position τ_i, and is zero otherwise. To penalize violations of the temperature bounds $\bar{x}$ and $\underline{x}$, the slack variables $\bar{\epsilon}_k = \|\max(x_k - \bar{x}, \mathbf{0})\|_\infty$ and $\underline{\epsilon}_k = \|\max(\underline{x} - x_k, \mathbf{0})\|_\infty$ are introduced, in which the maximum operator is used component-wise, and $\|v\|_\infty := \max_i |v_i|$ with v_i being an element of v. The optimization variables for the model-predictive controller are then contained in $\zeta_k^\top = [U_k^\top \; X_k^\top \; \Delta_k^\top \; \underline{\varepsilon}_k^\top \; \bar{\varepsilon}_k^\top]$, in which

$$U_k = \begin{bmatrix} u_{0|k} \\ u_{1|k} \\ \vdots \\ u_{N-1|k} \end{bmatrix}, X_k = \begin{bmatrix} x_{0|k} \\ x_{1|k} \\ \vdots \\ x_{N|k} \end{bmatrix}, \Delta_k = \begin{bmatrix} \delta_{0|k} \\ \delta_{1|k} \\ \vdots \\ \delta_{N-1|k} \end{bmatrix},$$

$$\underline{\varepsilon}_k = \begin{bmatrix} \underline{\epsilon}_{0|k} \\ \underline{\epsilon}_{1|k} \\ \vdots \\ \underline{\epsilon}_{N|k} \end{bmatrix}, \bar{\varepsilon}_k = \begin{bmatrix} \bar{\epsilon}_{0|k} \\ \bar{\epsilon}_{1|k} \\ \vdots \\ \bar{\epsilon}_{N|k} \end{bmatrix},$$

such that $U_k \in \mathbb{R}^{Nm}$, $X_k \in \mathbb{R}^{(N+1)n}$, $\Delta_k \in \{0,1\}^{Nn_t}$ and $\underline{\varepsilon}_k, \bar{\varepsilon}_k \in \mathbb{R}_{\geq 0}^{N+1}$, with N the prediction horizon of the controller. The subscript $i|k$ denotes the predicted value of the respective variable at future instant $k+i$, based on the knowledge available at instant k.

B. Constraints

Mathematical constraints are now formulated to include the physical-system constraints into the optimization problem. Firstly, the system dynamics are captured in equality constraints

$$x_{0|k} = \hat{x}_k \tag{8a}$$

$$x_{i+1|k} = Ax_{i|k} + Bu_{i|k} \quad i \in [0, N], \tag{8b}$$

with $i, k \in \mathbb{N}$ and $\hat{x}_k$ the state estimate at instant k. Inequality constraints are introduced to capture the temperature bounds as soft constraints of the form

$$\underline{x} - \mathbf{1}\underline{\epsilon}_{i|k} \leq x_{i|k} \leq \bar{x} + \mathbf{1}\bar{\epsilon}_{i|k} \qquad i \in [0, N], \tag{9a}$$

$$\underline{\epsilon}_{i|k} \geq 0 \qquad i \in [0, N], \tag{9b}$$

$$\bar{\epsilon}_{i|k} \geq 0 \qquad i \in [0, N], \tag{9c}$$

with $\mathbf{1}$ the all-ones vector of suitable dimension. To ensure positive inputs only that are below the maximal input $u_{max} \in \mathbb{R}_{>0}$, the inequality constraints

$$\mathbf{0} \leq u_{i|k} \leq \mathbf{1}u_{max} \quad i \in [0, N-1], \tag{10}$$

with $\mathbf{0}$ the all-zeros vector of suitable dimension, are included. Since a single transducer is used, only one of the entries in the δ vector can be a one at each instant. This constraint is formulated as

$$\mathbf{1}^\top \delta_{i|k} \leq 1 \quad i \in [0, N-1]. \tag{11}$$

Using the structure of the input vector (5), the inputs are coupled to the transducer location via the inequality constraints

$$Z u_{i|k} \leq \delta_{i|k} P_{max} \quad i \in [0, N-1], \tag{12}$$

in which P_{max} is the maximal total power input and

$$Z = I \otimes \mathbf{1}_{n_s}^\top \in \mathbb{R}^{n_t \times m},$$

with $\mathbf{1}_{n_s}$ the all-ones vector of dimension n_s, and $\otimes$ denoting the Kronecker product, such that the product $Z u_{i|k}$ contains the sums of the inputs per cell. Finally, the last constraints implement the inability to apply heat to the system when the actuator is traveling. To do so, a movement matrix $\bar{M}$ is specified that corresponds to the configuration of transducer positions in $\mathcal{T}$, such that $\bar{M}(i,j)$ specifies the movement time that it takes to move the heating actuator from position τ_i to position τ_j. Since the state-space model for the MPC problem is in discrete time, the movement times in $\bar{M}$ are integer values representing the number of samples that the corresponding move of the transducer takes. Then, using the movement matrix, additional matrices $M_d \in \{0,1\}^{n_t \times n_t}$ referred to as delay matrices are constructed according to

$$M_d(i,j) = \begin{cases} 1, \text{ if } \bar{M}(i,j) < d \\ 0, \text{ else} \end{cases} \qquad d = 1, 2, \ldots, d_{max}, \tag{13}$$

with d_{max} the maximal travel time in samples that occurs in $\bar{M}$. Consequently, $M_d(i,j) = 1$ corresponds to a movement time of less than d samples from location τ_i to τ_j. The constraints that prevent inputs from being nonzero while the transducer is moving are now formulated as

$$Z u_{i|k} \leq M_d \delta_{i-d|k} P_{max} \quad i \in [0, N-1], d \in [1, d_{max}], \tag{14}$$

in which the history of the transducer path now plays a role in constraining the current and future inputs. To clarify the constraints (14), the following example shows how the δ vectors that describe the position of the transducer come in and possibly allow for control.

Example 1. *A three-cell example system is shown in Figure 3, from which the movement matrix*

$$\bar{M} = \begin{bmatrix} 0 & 1 & 2 \\ 1 & 0 & 3 \\ 2 & 3 & 0 \end{bmatrix} \tag{15}$$

follows. The maximal delay is three sample times, resulting in the delay matrices

$$M_1 = I_3 \quad M_2 = \begin{bmatrix} 1 & 1 & 0 \\ 1 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad M_3 = \begin{bmatrix} 1 & 1 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \end{bmatrix}. \tag{16}$$

In case $u_{0|k}$ is now desired to be nonzero at the inputs related to the first location τ_1, it can be derived from constraints (12) *and* (14) *that this is only possible if*

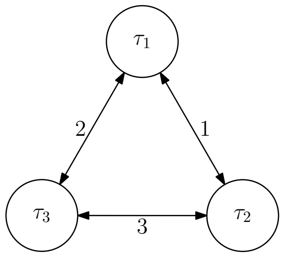


Fig. 3: Three-cell configuration with indicated movement times.

$$\delta_{0|k} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad \delta_{-1|k} = \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix} \quad \delta_{-2|k} \in \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix} \right\}$$

and

$$\delta_{-3|k} \in \left\{ \begin{bmatrix} 1 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 1 \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \right\}.$$

The same analysis can be done for the other transducer positions. This illustrates how the history of the transducer trajectory is now included in the input constraints to capture the travel times of the heating actuator.

Throughout the sequel of this report, only symmetric movement matrices are considered because during experiments it was observed that the movement times are not direction dependent. This effectively means that the transducer moves equally fast from τ_i to τ_j as the other way around.

C. Cost function

In the region of interest, a reference temperature profile is defined as $x^r \in \mathbb{R}^{n_\mathcal{R}}$, with $n_\mathcal{R}$ the number of voxels located inside the ROI. The performance variables, being the states that correspond to the voxels in $\mathcal{V}^\mathcal{R}$, are obtained via the linear mapping $x^\mathcal{R} = Hx \in \mathbb{R}^{n_\mathcal{R}}$, where $H \in \{0,1\}^{n_\mathcal{R} \times n}$ is a matrix with exactly one 1 in each row. Hence, following the way the reference profile and temperature bounds are specified, see Section II-D, it holds that $H\underline{x} \leq x^r \leq H\bar{x}$. The control objective is now formulated as the constrained optimization problem in the form of a mixed-integer-problem

$$\min_{\zeta_k} \sum_{i=0}^{N} (x^\mathcal{R}_{i|k} - x^r)^\top Q (x^\mathcal{R}_{i|k} - x^r) + f^\top \epsilon_{i|k} \tag{17a}$$

$$\text{subject to (8) – (14),} \tag{17b}$$

with $Q \in \mathbb{R}^{n_\mathcal{R} \times n_\mathcal{R}}$ and $f \in \mathbb{R}^2$ the weighting matrices for the states in the region of interest and the slack variables $\epsilon_{i|k} = [\underline{\epsilon}_{i|k}, \bar{\epsilon}_{i|k}]^\top$, respectively. In closed-loop control, problem (17) is solved at each instant k from which the input u_k is chosen as $u_k = u_{0|k}$ using the receding-horizon principle.

The state dimension in (17) is typically large (order 10^3) and both real- and integer variables are present, which make the optimization of problem (17) computationally demanding. In the next section, the computational complexity of (17) is alleviated by using hierarchical MPC or reduced-order MPC, in which the approximation of the dynamics (4a) plays a key role.

IV. Hierarchical model predictive control

In this section, the framework of the hierarchical controller is discussed. A short literature review is included in which background information on hierarchical control is presented. Afterwards, the control structure is discussed and each control layer is elaborated on.

A. Background on hierarchical MPC

In literature, several applications are found which show the working principle and benefits of hierarchical MPC. Generally, the decomposition of the control problem can be based on a difference in functionality, difference in time-sale, or based on a spatial decomposition, see [18]. A survey on multi-agent MPC is found in [19] in which the hierarchical control structure is discussed.

A wide variety of applications in which the principle of hierarchical control is shown are found in literature. These include, control of redundant refrigeration circuits [20], generating and tracking trajectories in highly automated vehicles [21], control for the fuel cell hybrid electric vehicles [22], matching uncertain wind generation with PEV charging demand in a microgrid [23] and in building energy management systems [24]. All applications show the efficacy of the hierarchical control in combination with the MPC framework.

Each work shows how a complex control problem is decomposed into sub-problems of lower complexity. Suitability of decomposition type depends strongly on the application. Therefore, following the examples in literature, it is believed that the MR-HIFU problem is well suited for HMPC control since a functional decomposition in control layers lies at hand, but that a case-specific approach is needed.

An alternative to hierarchical MPC to reduce computational complexity of the heavy MIP is reduced-order MPC (RO-MPC) in which the MPC problem is not decomposed, but a single MPC layer is used that makes use of reduced order dynamics, see, for example, [25, 26]. Interestingly, the high-level of the HMPC setup proposed in this work involves a RO-MPC problem, allowing for direct comparison between HMPC and RO-MPC.

B. HMPC control structure

Inspired by the examples in literature, a functional decomposition is proposed for the MR-HIFU problem as depicted in Figure 4. The decomposition in functionality is the following. At the high level, the main goal is to determine the cell-to-cell trajectory of the heating actuator, whereas the focus of the low-level is to compute the optimal heating inputs for the provided transducer path.

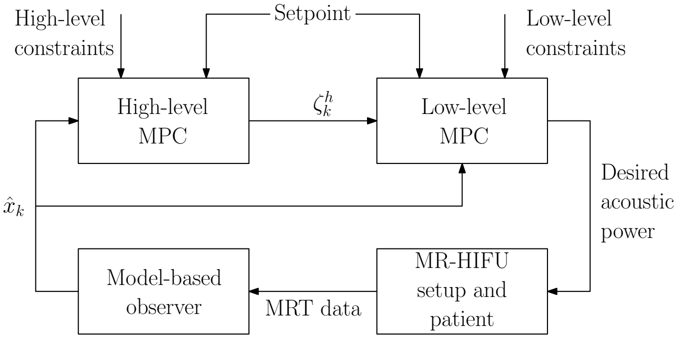


Fig. 4: Closed-loop HMPC control for MR-HIFU.

At the high level, an MPC problem is formulated that is subjected to constraints and an objective function, in which simultaneous optimization occurs over the continuous heating inputs and the integer variables describing the transducer path. Hence, the high-level optimization vector ζ_k^h contains the predicted inputs, state predictions and transducer path. This leads to the formulation of a mixed-integer optimization problem similar to the original problem (17). However, real-time feasibility is achieved by using reduced-order dynamics instead of using (4a) for state predictions, which yields the formulation of a RO-MPC problem at the high level.

The low-level MPC problem receives ζ_k^h and uses the predicted transducer path to fix Δ_k in the original problem (17). Hence, since the integer part is removed, a QP problem remains. As a QP requires much less computational power, this allows for the use of the full-order model (4a) for state predictions to compute the desired acoustic power.

Using the MR-HIFU setup, the desired acoustic power is applied to the patient. Each instant, new MRT data is obtained from which a new state estimate $\hat{x}_k$ is computed using the observer as in (6). The next two subsections are dedicated to elaborate on the details of both the high- and low-level MPC problems, respectively.

C. High-level MPC

In the high-level controller, the computationally expensive optimization (17) is approximated by a simplified problem. An approximation of (4a) is captured in a model of order $n_h < n$, which follows the dynamics

$$x_{k+1}^h = A^h x_k^h + B^h u_k^h, \tag{18}$$

with $x^h \in \mathbb{R}^{n_h}$ and $u^h \in \mathbb{R}^{m_h}$ the state vector and input vector of the reduced system, respectively. The matrices $A^h \in \mathbb{R}^{n_h \times n_h}$ and $B^h \in \mathbb{R}^{n_h \times m_h}$ describe the dynamics of the reduced model. In Section V, multiple reduced order models are discussed, which all allow for a linear transformation between the high- and low-level state of the form $x^h = Tx$, with $T \in \mathbb{R}^{n_h \times n}$ the transformation matrix. The full state vector is approximated by $\tilde{x} = T^\dagger x^h$, with $T^\dagger$ the pseudo-inverse of transformation matrix T. An approximation of the performance variables is then computed as $\tilde{x}^{\mathcal{R}} = H\tilde{x}$.

The slack variables $\epsilon_{i|k}$, $i = 1, \ldots, N$, are excluded in the high-level MPC problem. The reason is that in (17) the slack variables help in preventing temperature peaks at single voxel resolution. The reduced-order models discussed in this work do not have the ability to predict the behavior of single voxel temperatures and therefore implementing slack variables cannot serve this exact purpose. However, in [15] it was observed that for increased input powers (around 200 Watts), larger regions of the patient domain are predicted to exceed the upper temperature limits when the weight on the slack variables is set to zero. Hence, in case the maximal available power exceeds the usual limit of 100 Watts it does make sense to include the slack variables, but then applied to the approximated state prediction $\tilde{x}$. Since these cases are not considered in this project, the slack variables are excluded from the high-level optimization problem for simplicity.

The optimization variables of the reduced problem are contained in the vector ${\zeta^h_k}^\top = [{U^h_k}^\top {X^h_k}^\top \Delta_k^\top]$, with $U^h_k \in \mathbb{R}^{Nm_h}$ and $X^h_k \in \mathbb{R}^{(N+1)n_h}$. The vector Δ_k remains unchanged with respect to the original problem (17). Using the reduced order dynamics, the approximated optimization problem is formulated as the mixed-integer problem

$$\min_{\zeta^h_k} \sum_{i=0}^{N} (\tilde{x}^{\mathcal{R}}_{i|k} - x^r)^\top Q (\tilde{x}^{\mathcal{R}}_{i|k} - x^r) \tag{19a}$$

$$\text{s.t.} \quad x^h_{0|k} = T\hat{x}_k \tag{19b}$$

$$x^h_{i+1|k} = A^h x^h_{i|k} + B^h u^h_{i|k} \quad i \in [0, N] \tag{19c}$$

$$Z^h u^h_{i|k} \le M_d \delta_{i-d|k} P_{max} \quad i \in [0, N-1], d \in [1, d_{max}] \tag{19d}$$

$$\mathbf{1}^\top \delta_{i|k} \le 1 \quad i \in [0, N-1] \tag{19e}$$

$$\mathbf{0} \le u^h_{i|k} \le \mathbf{1} u^h_{max} \quad i \in [0, N-1], \tag{19f}$$

with u^h_{max} the maximal individual input of the high-level model, and

$$Z^h = I \otimes \mathbf{1}^\top_{n_{\bar{s}}}, \tag{20}$$

with $n_{\bar{s}} = m_h/n_t$. Note that in case $m_h = m$, it holds that $Z = Z^h$. The upper limit is not necessarily equal to u_{max}, since the input space can be formulated differently for the reduced model. In case RO-MPC is used, all control actions are extracted from solving (19). For situations in which $m_h \neq m$, it is assumed that a linear mapping exists such that $u_k = T_u u^h_k$, with $T_u \in \mathbb{R}^{m \times m_h}$ the mapping matrix that depends on the type of model reduction. For hierarchical control, the solution of (19) is passed through to the low level. A reduction in computational complexity is expected for significantly reduced high-level prediction models.

D. Low-level MPC

The low-level controller is similar to the original MIP controller (17). However, the transducer positions are now no longer optimization variables, but are provided by the high-level controller. This yields the optimization problem in the form of a QP

$$\min_{\zeta_k} \sum_{i=0}^{N} (x^{\mathcal{R}}_{i|k} - x^r)^\top Q (x^{\mathcal{R}}_{i|k} - x^r) + f^\top \epsilon_{i|k} \tag{21a}$$

$$\text{s.t.} \quad x_{0|k} = \hat{x}_k \tag{21b}$$

$$x_{i+1|k} = A x_{i|k} + B u_{i|k} \quad i \in [0, N] \tag{21c}$$

$$Z u_{i|k} \le \bar{\delta}_{i|k} P_{max} \quad i \in [0, N-1] \tag{21d}$$

$$\mathbf{0} \le u_{i|k} \le \mathbf{1} u_{max} \quad i \in [0, N-1] \tag{21e}$$

$$\underline{x} - \mathbf{1}\underline{\epsilon}_{i|k} \le x_{i|k} \le \bar{x} + \mathbf{1}\bar{\epsilon}_{i|k} \quad i \in [0, N] \tag{21f}$$

$$\underline{\epsilon}_{i|k} \ge 0 \quad i \in [0, N] \tag{21g}$$

$$\bar{\epsilon}_{i|k} \ge 0 \quad i \in [0, N], \tag{21h}$$

with $\bar{\delta}_i$ the parameters that describe the transducer path which are defined according to

$$\bar{\delta}_{i|k} = \begin{cases} \delta_{i|k}, & \text{if } \mathbf{1}^\top u^h_{i|k} > 0, \\ \mathbf{0}, & \text{else.} \end{cases} \tag{22}$$

Hence, the high-level input is used to determine whether the transducer is moving at future instant i, and the predicted inputs $u_{i|k}$ are constrained to be zero accordingly. Both the high- and the low-level MPC problem are solved with the Gurobi solver, see [27].

Remark. *Updating $\bar{\delta}_{i|k}$ following* (22) *is an easy to implement, pragmatic, approach to include the transducer path into the optimization problem. A more formal way would be to include constraints* (19d) *into the low-level problem as well, but then acting on $u_{i|k}$. However, since no occurrences are observed in which $u^h_{i|k}$ is predicted to be all zero at instances where a nonzero input would be allowable, no further action is taken.*

V. Reduced order prediction models

In this section, the computation of the reduced-order prediction models is discussed. Three different approaches are considered with different properties. A short discussion is included in which pros and cons of the methods are compared.

A. Balanced truncation

The first option that is considered to derive a high-level model of reduced order that captures the dynamics with sufficient detail is by balanced truncation, see [28, 29]. The core idea of this reduction is to apply a balancing transformation to the state such that the corresponding transformed reachability and observability Gramians are equal and diagonal. A reduced-order state space model can be obtained from the balanced system by truncation, which removes states that are the most difficult to reach and to observe. The truncation order is user-defined and is typically chosen on the basis of tuning. Advantages of the balancing type reduction are that the reduced system possesses the same properties as the balanced system with respect to reachability, observability and stability, and that an error bound between the output of the reduced system and the original system is known.

For continuous-time systems, the infinite-horizon reachability and observability gramians are computed by solving the Lyapunov equations

$$A_c\mathcal{P} + \mathcal{P}A_c^\top + B_cB_c^\top = 0 \tag{23}$$

$$A_c^\top\mathcal{Q} + \mathcal{Q}A_c + C^\top C = 0, \tag{24}$$

in which $\mathcal{P}, \mathcal{Q} \in \mathbb{R}^{n\times n}$ are the reachability and observability Gramian respectively, and $A_c \in \mathbb{R}^{n\times n}$ and $B_c \in \mathbb{R}^{n\times m}$ are the continuous-time system matrices.

To avoid numerical trouble in calculating the balancing transformation, a minimal realization of the system is calculated first. That is, only states that are both reachable and observable are retained, while all other states are removed from the system. This results in a system with the same transfer function as the original system, but with a minimal number of states. In order to calculate a minimal realization, the reachable and observable set of states $\mathcal{RO}$ is calculated according to

$$\mathcal{RO} = \{x \in \mathbb{R}^n \mid x \in \text{Im}(\mathcal{PQ})\}, \tag{25}$$

with Im$(\cdot)$ denoting the image of a matrix, which follows from the properties that Im$(\mathcal{P})$ denotes the reachable set of states and ker$(\mathcal{Q})$ denotes the unobservable set of states, with ker$(\cdot)$ being the kernel of a matrix. Consequently, the set of states in $\mathcal{RO}^\perp$, which is the complement of $\mathcal{RO}$, are to be removed from the system description. To do so, the transformation matrix

$$T = \begin{bmatrix}\text{Im}(\mathcal{RO}) & \text{Im}(\mathcal{RO}^\perp)\end{bmatrix}^\top \tag{26}$$

is defined.

Using the transformation matrix (26), the system matrices are transformed according to

$$\bar{A} = TA_cT^{-1} \qquad \bar{B} = TB_c \qquad \bar{C} = CT^{-1}. \tag{27}$$

The minimal realization is now obtained by writing the transformed system matrices in (27) as

$$\bar{A} = \begin{bmatrix}A_{11} & A_{12}\\ A_{21} & A_{22}\end{bmatrix} \in \mathbb{R}^{n\times n}, \quad \bar{B} = \begin{bmatrix}B_1\\ B_2\end{bmatrix} \in \mathbb{R}^{n\times m}, \tag{28}$$

$$\bar{C} = \begin{bmatrix}C_1 & C_2\end{bmatrix} \in \mathbb{R}^{n\times n}, \tag{29}$$

from which the minimal system matrices

$$A_m = A_{11} \in \mathbb{R}^{r_m\times r_m}, \qquad B_m = B_1 \in \mathbb{R}^{r_m\times m}, \tag{30}$$

$$C_m = C_1 \in \mathbb{R}^{n\times r_m}, \tag{31}$$

follow, with r_m the number of columns in Im$(\mathcal{RO})$.

Remark. *Calculation of the minimal representation of the system may remove states that are reachable, but due to numerical tolerances are interpreted as unreachable. The same holds for observable states. The set $\mathcal{RO}$ has r_m columns which equals the rank of $\mathcal{PQ}$. This rank equals the number of singular values which are smaller than a certain tolerance. Therefore, the minimal representation is here only a numerically minimal representation. Theoretical aspects on reachability and observability of the system are not investigated any further.*

Using the minimal system matrices (A_m, B_m, C_m), the corresponding reachability- and observability Gramians $(\mathcal{P}_m, \mathcal{Q}_m)$ are computed by solving (23) and (24), respectively. A balancing transformation matrix can now be computed, see [29], which obtains balanced Gramians with decreasing singular values on the diagonal. First the Cholesky reachability Gramian $\mathcal{P}_m$ is computed as

$$\mathcal{P}_m = S^\top S. \tag{32}$$

A singular value decomposition is now performed on the product $S^\top\mathcal{Q}_mS$ as

$$S^\top\mathcal{Q}_mS = \Gamma\Sigma\Theta^\top, \tag{33}$$

from which the balancing transformation matrix is calculated as

$$T_b = \Sigma^{1/4}\Theta S^{-1}. \tag{34}$$

The reduced model is now obtained by computing the balanced representation of the minimal system as

$$A_b = T_bA_mT_b^{-1}, \quad B_b = T_bB_m, \quad C_b = C_mT_b^{-1}, \tag{35}$$

and truncating the matrices in (35) to the user-defined order n_h, after which the discrete time model is obtained using forward Euler discretization.

An error bound on the output of the reduced-order system is known and is given by

$$||G - G^h||_{H_\infty} \leq 2(\sigma_{n_{h+1}} + \ldots + \sigma_{r_m}), \tag{36}$$

in which G and G^h are the transfer functions of the minimal- and the reduced system, and $\sigma_{n_h+1}, \ldots, \sigma_{r_m}$ are the truncated singular value. The error bound provides useful information on the accuracy of the reduced model and can be used for initial estimates on the desired reduction order.

Remark. *The presented balanced truncation procedure does not incorporate any input constraints in the construction of the reduced model. Constraints* (11)*, which limit heating to a single transducer location, allow to interpret the state space model as a switched system in which the input-to-state matrix is mode dependent. Each transducer position is then coupled to a mode-dependent $B_{q(t)}$ matrix, in which $q(t)$ specifies the mode at time t. In [30], balanced truncation for linear switched systems of the form*

$$\dot{x}(t) = A_{q(t)}x(t) + B_{q(t)}u(t) \tag{37}$$

$$y(t) = C_{q(t)}x(t), \tag{38}$$

is discussed, in which all system matrices are mode dependent. Following the procedure in [30] for the simplified case in which only the B matrix is switched, the resulting observability and reachability Gramians that are used for model reduction turn out to be equivalent to the ones obtained by solving the Lyapunov equations (23)-(24)*, respectively. Therefore, neglecting the switched nature of the system is validated.*

B. POD basis reduction

A second model-reduction technique that is considered is a data-based approach. The reduction technique is based on a proper orthogonal decomposition (POD) of a data set, see [29]. The POD basis problem aims to find an orthonormal basis that can best describe a given data set. For the data matrix $W = [w_1, w_2, \ldots, w_{n_w}] \in \mathbb{R}^{n \times n_w}$, the set of orthonormal basis functions $\Phi = \{\phi_1, \phi_2, \ldots, \phi_r\}$, with $\phi_i \in \mathbb{R}^n$, $i = 1, \ldots, r$, with $r < n_w$ should be such that the approximation error computed by the sum

$$\sum_{j=1}^{n_w} \|w_j - \sum_{k=1}^{r} \langle w_j, \phi_k \rangle \phi_k\|_2, \tag{39}$$

in which $\langle \cdot, \cdot \rangle$ denotes the inner product, is minimal.

The basis to minimize (39) can be obtained by computing the singular value decomposition of the data matrix W as

$$W = \Gamma \Sigma \Theta^\top, \tag{40}$$

with $\Gamma = [\gamma_1, \ldots, \gamma_n] \in \mathbb{R}^{n \times n}$, $\Theta = [\theta_1, \ldots, \theta_{n_w}] \in \mathbb{R}^{n_w \times n_w}$ and

$$\Sigma = \begin{bmatrix} \bar{\Sigma} & 0 \\ 0 & 0 \end{bmatrix},$$

with $\bar{\Sigma} = \operatorname{diag}(\sigma_1, \ldots, \sigma_n)$ and n the rank of matrix W. Setting $\phi_i = \gamma_i$, $i = 1, 2, \ldots, r$, is then a POD basis, i.e. (39) is minimal.

A reduced-order model can now be constructed by using time series data of a simulation to fill the matrix W. The singular value decomposition is then performed to pick relevant directions. For a truncation order n_h, the projection matrix to obtain the high-level state is then defined as $T = [\gamma_1, \gamma_2, \ldots, \gamma_{n_h}]^\top \in \mathbb{R}^{n_h \times n}$, such that $x^h = Tx$ and $\tilde{x} = T^\top x^h$. The reduced-order system matrices are obtained by applying Galerkin projection of the original model on the orthonormal basis T, which leads to the system matrices

$$A^h = T^\top A T \qquad\qquad B^h = T^\top B, \tag{41}$$

for the high-level prediction model. Appendix A contains a proof regarding the stability of the reduced order prediction model.

No formal bound on the prediction error is obtained. However, a quantification of the approximation error with respect to the data matrix is given by the largest truncated singular value

$$\|W - \tilde{W}\|_2 = \sigma_{r_b+1}, \tag{42}$$

with $\tilde{W} = T^\top T W$ the approximated data matrix. Hence, the largest neglected singular value gives an indication of the model accuracy.

C. Heuristic approach

As a third reduced order model, a heuristic approach is considered. In [14], control decisions are made on the basis of average cell temperatures. Motivated by this, a simplified model is constructed in which the state vector contains average temperatures over larger areas. Taking just the average cell temperatures comes with the disadvantages that temperatures of voxels that are not located in a cell, are not monitored. To this extent, the patient domain is divided into subspaces using a Voronoi diagram. In general, a Voronoi diagram divides a space such that the resulting Voronoi cells R_j associated with site $P_j \in \mathbb{R}^p$, being a point in the p-dimensional space, satisfy the property

$$R_j = \{r \in \mathbb{R}^p \mid d(r, P_j) \leq d(r, P_i) \quad \forall i \neq j\}, \tag{43}$$

in which $d : \mathbb{R}^p \times \mathbb{R}^p \to \mathbb{R}_{\geq 0}$ specifies a distance function, see [31]. For this project, a convenient choice of sites to divide the patient domain are the locations of transducers. This leads to the definition of the Voronoi cells in the patient domain as

$$R_j = \{r \in \Omega \mid d(r, \tau_j) \leq d(r, \tau_i) \quad \forall i \neq j\}, \tag{44}$$

with the distance function now working on the subspaces $d : \Omega \times \mathcal{T} \to \mathbb{R}_{\geq 0}$ and $i, j \in \{1, 2, \ldots, n_t\}$. Hence, in total the number of Voronoi cells is equal to the number of transducer positions in $\mathcal{T}$.

In order to split the region of interest from the subspace outside the region of interest, a refinement is made according to

$$I_j = R_j \cap \mathcal{R} \qquad\qquad j \in \{1, 2, \ldots, n_t\}, \tag{45a}$$

$$O_j = R_j \setminus \mathcal{R} \qquad\qquad j \in \{1, 2, \ldots, n_t^o\}, \tag{45b}$$

in which it is possible that $n_t^o < n_t$ in case one or more of the Voronoi cells is located completely inside the region of interest.

An illustrative example of the partition of the patient domain into cells using a Voronoi diagram is provided in Figure 5. Each cell is enclosed by the Voronoi edges and the boundaries of the patient domain and region of interest. For this example, the reduced state is of dimension ten.

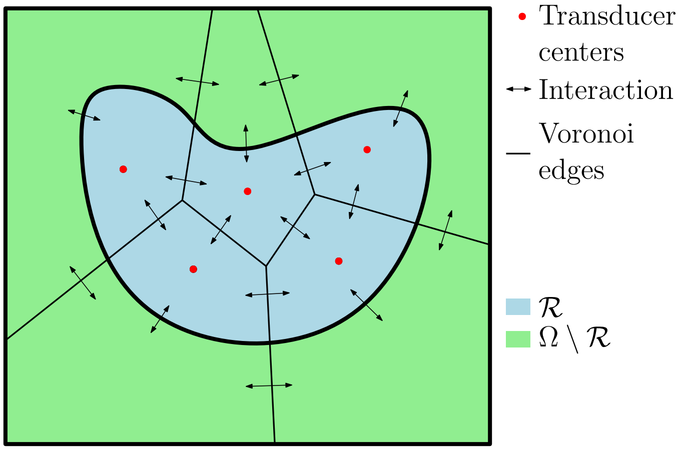


Fig. 5: Voronoi partition.

Each cell in (45) is associated with a set of voxel locations that are contained within the boundaries of the respective cell. The indexes of the voxels in each set are captured in

$$\mathcal{I}_j^I = \{i \in \{1,\dots,n\} \mid v_i \in \mathcal{V} \cap I_j\},\ j \in \{1,\dots,n_t\}, \tag{46a}$$

$$\mathcal{I}_j^O = \{i \in \{1,\dots,n\} \mid v_i \in \mathcal{V} \cap O_j\},\ j \in \{1,\dots,n_t^o\}, \tag{46b}$$

such that the average temperature of each cell is calculated as

$$x_j^I = \frac{1}{N_j^I} \sum_{i \in \mathcal{I}_j^I} x^i, \qquad j \in \{1,\dots,n_t\}, \tag{47a}$$

$$x_j^O = \frac{1}{N_j^O} \sum_{i \in \mathcal{I}_j^O} x^i, \qquad j \in \{1,\dots,n_t^o\}, \tag{47b}$$

with x^i the i-th element of the full-state vector x, and N_j^I and N_j^O the number of elements in the sets $\mathcal{I}_j^I$ and $\mathcal{I}_j^O$, respectively. Consequently, the reduced state vector is constructed as

$$x_k^h = [x_{1,k}^I, \dots, x_{n_t,k}^I, x_{1,k}^O, \dots, x_{n_t^o,k}^O]^\top, \tag{48}$$

which is obtained from the full temperature profile via the linear mapping $x_k^h = Tx_k$ that follows from (47).

Using the transformation matrix, the reduced input-to-state matrix is defined as $B^h = TB$. To define the A matrix for the high-level, physical modeling is used. To do so, the energy balance

$$\rho V c \frac{dT}{dt} = \dot{Q}_{in} + \dot{Q}_{out}, \tag{49}$$

with density ρ in [kg/m^3], volume V in [m^3], specific heat capacity c in [J/kg·K], temperature T in [K], time t in [s] and energy flow $\dot{Q}$ in [J/s], is used. For each cell, the energy inflow $\dot{Q}_{in}$ and outflow $\dot{Q}_{out}$ are considered to be only the applied heat by the transducer and heat conduction between neighboring cells, respectively. The energy flow due to heat conduction is modeled using Fourier's law

$$\dot{Q}_{out} = -kA_q \nabla T, \tag{50}$$

with conductivity k in [W/m·K], area A_q over which heat transfer occurs in [m^2] and the temperature gradient ∇T in [K/m]. Both the volume V in (49) and the area A_q in (50) depend on the out of plane direction z. Since this direction is not considered in the prediction model, the z-dependency is removed by normalizing both

$$V = A_{cell} \Delta z \text{ and } A_q = L \Delta z,$$

with A_{cell} the area of the cell perpendicular to the z direction, and L the length over which heat transfer occurs, in the z direction which yields

$$\rho A_{cell} c \frac{dT}{dt} = -kL\nabla T. \tag{51a}$$

Using these relations, the temperature evolution as a result of heat conduction can be found for each cell by summing over all interaction terms. The temperature gradient between two interacting cells is calculated as

$$\nabla T = \frac{T_i(k) - T_j(k)}{\Delta r_{i,j}}, \tag{52}$$

with $\Delta r_{i,j}$ the Euclidean distance between the center points, which are calculated as

$$M_j^I = \frac{1}{N_j^I} \sum_{i \in \mathcal{I}_j^I} v_i \qquad \forall j \in \{1,\dots,n_t\}, \tag{53a}$$

$$M_j^O = \frac{1}{N_j^O} \sum_{i \in \mathcal{I}_j^O} v_i \qquad \forall j \in \{1,\dots,n_t^o\}, \tag{53b}$$

corresponding to the cells (45). For each cell, the temperature update can now be calculated using

$$T_c(k+1) = T_c(k) - \frac{k}{\rho A_{cell,c} c} \sum_{i \in \mathcal{N}_c} L_{c,i} \frac{T_c(k) - T_i(k)}{\Delta r_{c,i}} T_s + B_c^h u_k, \tag{54}$$

with $A_{cell,c}$ in [m^2] the area of cell c, $L_{c,i}$ in [m] the length of the touching edge between cell c and i, $\mathcal{N}_c$ the set of cells indexes that have a touching edge with cell c, T_s the sampling time and B_c^h the row of the input matrix that corresponds to cell c.

Besides the reduction in state dimension, the Voronoi partition allows a reduction in input space as well without losing accuracy with respect to (54). All inputs that only effect a single element of the reduced state vector can be grouped together. Specifying $\bar{u}_i \in \{0,1\}^m$, $i = 1,2,\dots,m$, with a single 1 at the i-th position and $\bar{x}_j^h \in \{0,1\}^{n_h}$, $j = 1,2,\dots,n_h$, with a single 1 at the j-th position, all i-th inputs for which it holds that

$$\langle B^h \bar{u}_i, \bar{x}_j^h \rangle = 1, \tag{55}$$

for a certain j, can then be grouped together without losing model accuracy. This allows a reduction in optimization variables which further reduces computational times.

D. Discussion

Out of the three presented methods, two methods allow for a user-defined reduction order being the balanced truncation and the POD-based model. The Voronoi model does not have this possibility in the way the reduction is modeled now, which can be problematic in case the model turns out to be insufficiently accurate. For the other two models, this can be solved by increasing the reduction order until the desired accuracy is achieved.

In terms of computational times, the Voronoi model is expected to outperform the other two models for similar reduction orders based on the sparsity of the system matrices. For both the balanced truncation and the POD-based models, the matrix A^h is typically a full matrix caused by the full transformation matrix, whereas the Voronoi method yields a sparse matrix. Therefore, higher prediction horizons are possible within computational constraints when using the Voronoi model.

Comparing the balanced truncation to the POD-based method, it is expected that the POD model is more case specific which is beneficial for the accuracy of the model. That is, since the POD model is based on simulation data that is obtained from a case study in which the reduced model is planned to be used in the HMPC controller, the vectors that span the transformation matrix are more specific. For the balanced truncation this is not the case since the transformation follows solely from the matrices of the original system. Hence, in case an accurate data set is available that is related to the desired case, the POD model is expected to be more accurate than the balanced truncation for equal reduction orders.

VI. Transducer location optimization

In this section, two methods are discussed for the calculation of the transducer positions. Both methods rely on an optimization procedure, but differ in their objectives. Using the calculated transducer positions, the input-to-state matrix can be defined as is needed for state predictions in the MPC controller.

A. Method I: utility-based optimization

The first optimization method is based on optimizing the utility resulting from placing a transducer center at a certain location. For the particular case of determining the transducer locations, a sensible choice of utility would be a quantitative measure of the ability to apply heat to the system. Therefore, the utility is defined as the negative sum of distances from the transducer centers to each voxel in the set $\mathcal{V}^{\mathcal{R}}$. This definition is such that a maximal utility is obtained when the sum of distances is minimal. Using distance as a utility measure is a common approach in literature for solving facility location problems, see e.g. [32], in which facilities are to be placed such that clients are optimally served. In the hyperthermia context, the transducer could be interpreted as the facility which supplies the region of interest with heat. Hence, the voxels in the region of interest can be interpreted as clients.

The optimal placement of the transducers is now computed by solving the optimization problem

$$\min_{\mathcal{T}} \Big(\sum_{v_i \in \mathcal{V}^{\mathcal{R}}} w_i \min_{\tau_j \in \mathcal{T}} d(\tau_j, v_i) \Big) \tag{56a}$$

$$\text{s.t.} \quad \tau_j \in \Omega \quad \forall j \in [1, n_t], \tag{56b}$$

with $d : \mathbb{R}^2 \times \mathcal{V}^{\mathcal{R}} \to \mathbb{R}_{\geq 0}$ chosen to define the Euclidean distance and $w_i \in \mathbb{R}_{\geq 1}$, $i = 1, \ldots, n_v$, constants, which allow additional weighting for voxels v_i, $i = 1, \ldots, n_v$. Since most heat flux typically occurs at the edges of the region of interest due to the temperature gradient, it could be beneficial to increase the weight on the voxels at the edge of the ROI which results in transducer locations that are more positioned towards the edge of the ROI and therefore allowing more heat generation here to counteract the heat flux. In addition, in case of a known source of local heat loss, such as a blood vessel, these areas could be weighted extra as well to counteract the heat loss. However, determining the weights a-priori based on this information is not straightforward.

Due to the obvious non-convex nature of the optimization problem (56), the genetic algorithm provided by Matlab is used to solve the problem which enhances the likelihood of finding the global optimum, see [33]. Note that constraint (56b) is assumed to be inactive, but is included to limit the search space. This assumption is reasonable since the patient domain is typically two to three times larger than the region of interest in dimensions. Therefore, placing the transducers at the edge of the patient domain would result in a low utility.

A useful extension to optimization problem (56) would be to include a minimal utility constraint, which gives the opportunity to search for a minimal amount of transducer positions to comply with this constraint. The minimal amount of transducer locations is then the minimal amount of locations for which the optimization problem, in which the utility constraint is included, is feasible. The implementation as it is now also allows to find a minimal amount of transducer locations by iteratively increasing the number of locations and monitoring the utility until the lower bound is met. Therefore, the additional constraint is not implemented.

The advantages of this method are the ease of implementation and applicability to arbitrary sets of voxels in the region of interest $\mathcal{V}^{\mathcal{R}}$. A main disadvantage is that the system's thermal behavior is in no way included into the optimization, potentially leading to suboptimal placing of the transducer centers. Therefore, a second method is considered which does include some thermal behavior.

B. Method II: including thermal behavior

An approximation of the steady-state performance is made to represent average behavior of the system by allowing simultaneous heating at all transducer-locations. For a fixed input-to-state matrix this problem can be formulated as a quadratic programming problem (QP). Since the input-to-state matrix B is dependent on the location of the transducers, the transducer location problem is formulated as the clustered optimization problem

$$\min_{\mathcal{T}} \Big(\min_{\zeta^{qp}} \; (x^{\mathcal{R}} - x^r)^\top Q (x^{\mathcal{R}} - x^r) + f^\top \epsilon \Big) \tag{57a}$$

$$s.t. \quad x = Ax + B_f(\mathcal{T})u \tag{57b}$$

$$x \leq \bar{x} + \mathbf{1}\epsilon \tag{57c}$$

$$\mathbf{0} \leq u \leq \mathbf{1}u_{max} \tag{57d}$$

$$\mathbf{1}^\top u \leq P_{max} \tag{57e}$$

$$\epsilon \geq 0 \tag{57f}$$

$$\tau_j \in \Omega \quad \forall j \in [1, n_t], \tag{57g}$$

in which $x \in \mathbb{R}^n$, $u \in \mathbb{R}^m$ and $\epsilon \in \mathbb{R}_{\geq 0}$ are the optimization variables of the QP problem, being the steady-state temperature profile, heating input and slack variable, respectively, and are contained in the vector $\zeta^{qp} = [u^\top \; x^\top \; \epsilon]^\top$. Using the function $B_f : \mathcal{T} \to \mathbb{R}^{n \times m}$, the input-to-state matrix for the current configuration of transducer locations is obtained. The reference temperature map in the region of interest x^r and the maximum temperature map $\bar{x}$ are defined similar to the MPC problem as discussed in section IV. Equality constraint (57b) ensures that the obtained solution is a steady-state temperature profile. The input constraints are similar to the MPC problem (21), but of reduced complexity since the input is no longer constrained to a single transducer location. Note that no slack variable is included to penalize violations of the lower temperature bound as this typically does not occur in steady steady.

The usage of a genetic algorithm turned out to be computationally intractable for solving (57). Therefore, to make solving (57) computationally tractable, the solution is obtained by first solving (56) and providing the outcome as an initial guess to a gradient-based solver. Convexity of the problem is not investigated, so no conclusions are drawn with respect to global optimality. However, it can be concluded that the configuration of transducer locations as obtained from (57) will be no worse than the locations obtained from (56) with respect to the performance criterion (57a).

C. Comparison

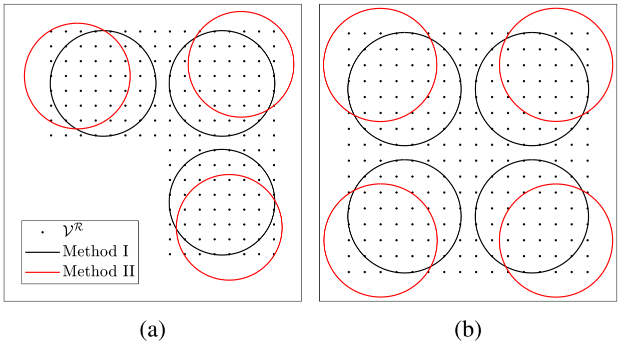
To compare the two methods, the transducer locations in $\mathcal{T}$ are optimized for two different regions of interest, resulting in different sets of voxels $\mathcal{V}^{\mathcal{R}}$. The first ROI is chosen as a corner profile whereas the second is a square region of interest. In Figure 6, the resulting transducer locations that follow from both optimization methods are plotted. The circles represent the range in which focal spots can be generated by electronic beam steering, corresponding to the transducer locations in $\mathcal{T}$. In both the first and the second method all voxels are equally weighted, i.e. $w_i = 1$, $i = 1, \ldots, n_v$, $Q = I$ and $f = 0$.

The first method results in a homogeneously distributed set of transducer locations over the region of interest, while the second method finds the optimal transducer locations to lie closer to the edges of the ROI. These results stress the importance of heating the edges, due to the largest heat flux by diffusion there, and in case of using the first method, an additional weight on the critical edges can be used to enhance steady-state performance. However, method I comes with the disadvantage that the additional weights will always be an a-priori guess, while method II automatically incorporates an approximation of the system's thermal behavior to optimize the locations.

A quantitative performance comparison is made in which for several chosen transducer paths, the low-level MPC optimization problem (21) is solved. To find a periodic solution, the equality constraint

$$x_{N_\sigma} = x_0, \tag{58}$$

with N_σ the length of the transducer path, is added. This way, the result of the low-level optimization provides the optimal temperature profiles in steady-state when periodically following the given transducer path. The transducer path $\sigma = \{\sigma_0, \sigma_1, \ldots, \sigma_{N_\sigma}\}$ is defined such that a zero corresponds to no heating allowed, whereas $\sigma_i = t$ implies that heating is allowed at transducer location τ_t. Performance is quantified by calculating the average stage cost over the length of N_σ time instants. In Table I and Table II, the average stage costs are shown for method I and II, for both the corner case and the square case, respectively. Relative difference in performance is expressed by $r = -100 \times (\text{Cost II - Cost I})/\text{Cost I}$. Numbering of the transducer locations is clockwise, starting with the upper right corner.



(a)

(b)

Fig. 6: Transducer locations for two different sets $\mathcal{V}^{\mathcal{R}}$. Plot (a) shows a corner profile with three transducer locations, whereas (b) shows a square region of interest with four locations. The circles indicate the region in which heat can be applied.

TABLE I: Performance comparison of transducer configurations I and II for the corner case in Figure 6a.

σ	Cost I	Cost II	r [%]
$\{1, 0, 2, 0, 3, 0\}$	6.89	1.84	73.3
$\{1, 1, 0, 2, 2, 0, 3, 3, 0\}$	7.01	1.90	72.9
$\{1, 0, 0, 2, 0, 0, 3, 0, 0\}$	7.14	1.97	72.4
$\{1, 1, 0, 0, 2, 2, 0, 0, 3, 3, 0, 0\}$	7.29	2.06	71.8

TABLE II: Performance comparison of transducer configurations I and II for the square case in Figure 6b.

σ	Cost I	Cost II	r [%]
$\{1, 0, 2, 0, 3, 0, 4, 0\}$	8.64	0.91	89.5
$\{1, 1, 0, 2, 2, 0, 3, 3, 0, 4, 4, 0\}$	8.87	1.00	88.7
$\{1, 0, 2, 0, 4, 0, 3, 0\}$	8.64	1.97	72.4
$\{1, 1, 0, 2, 2, 0, 4, 4, 0, 3, 0\}$	8.87	1.00	88.7

A clear increase in performance is observed for method II. This indicates a significant potential benefit of placing the transducers according to method II in addition to the fact that no a-priori heuristic weighting of the voxels is needed. However, robustness against model inhomogeneity is decreased when using the second method. In case local spots of increased perfusion are present inside the ROI that are not included in the model, utility plays an important role in counteracting this disturbance. Therefore, one may still decide on using method I for transducer placement depending on the application.

As a validation of the simplified optimization problem (57), the QP costs are computed for the transducer locations as obtained through both optimization methods. The numeric values are

$$\begin{aligned} &\text{corner case:} \quad C_{qp}^{I} = 6.71 \qquad && C_{qp}^{II} = 1.73 \\ &\text{square case:} \quad C_{qp}^{I} = 8.32 \qquad && C_{qp}^{II} = 0.76, \end{aligned}$$

which shows a slightly optimistic estimate of the steady-state cost that are computed for different sequences, but turns out to be a good indicator. The slightly lower cost is as expected, since in problem (57) the heating is not limited to a single location at a time, which enhances performance and therefore results in a lower cost. Conclusively it can be said that the reduced problem is well suited for optimization of transducer locations, such that steady-state performance is enhanced.

VII. Stabilizing HMPC for hyperthermia

So far, stability properties of hierarchical MPC are not addressed. The goal of this section is to provide a stability analysis of HMPC applied to the hyperthermia case. As a first step, the proposed controller consisting of the two MPC problems (19) and (21) is altered to a setup in which the two MPC problems are defined as tracking problems, since tracking MPC allows for an insightful stability analysis. First, the approach of the stability analysis is discussed, after which the separate parts are elaborated on in detail.

A. Stability approach

In order to draw conclusions on the stability of the HMPC controller, the stability analysis is split up into multiple steps. Firstly, stability of MPC is analysed for the simplified case in which the high-level dynamics are assumed to capture the true dynamics of the system. To define an appropriate reference signal for the tracking-MPC problem, an optimization procedure is presented, from which a periodic reference signal is generated that is suited for periodic tracking as in, for example, [34]. Afterwards, stability towards this trajectory is shown.

As a second step, the interconnection between high- and low-level is analyzed. Since the high-level dynamics do not capture the true behavior of the system, there exists a mismatch of the form

$$x_{k+1}^h = Tx_{k+1} \neq A^h x_k^h + B^h u_k^h. \tag{59}$$

This mismatch is included in the analysis by considering a bounded disturbance that acts on the high-level dynamics. A link can now be made to tube-based robust MPC, see [12], from which conclusions regarding the stability of the tracking HMPC are drawn.

B. Generation of periodic reference trajectory

For the generation of reference trajectory as used in the tracking MPC is calculated by solving an optimization problem similar to the high-level problem (19), but now over a horizon of N_σ samples and with additional constraints to ensure periodicity of the solution. This leads to the mixed-integer optimization problem

$$\begin{aligned} \min_{\zeta^\sigma} \sum_{i=0}^{N_\sigma} (\tilde{s}_i^{\mathcal{R}} - x^r)^\top Q (\tilde{s}_i^{\mathcal{R}} - x^r) & \qquad \text{(60a)} \\ \text{s.t.} \quad s_{i+1} = As_i + Bu_i^s \quad i \in [0, N_\sigma - 1] & \qquad \text{(60b)} \\ s_0 = s_{N_\sigma} & \qquad \text{(60c)} \\ Zu_i^s \leq M_d \delta_{i-d}^s P_{max} \quad i \in [0, N_\sigma - 1], & \\ d \in [1, d_{max}] & \qquad \text{(60d)} \\ Zu_0^s \leq M_d \delta_{N_\sigma - d}^s P_{max} \quad d \in [1, d_{max}] & \qquad \text{(60e)} \\ \mathbf{1}_{ns}^\top \delta_i^s \leq 1 \quad i \in [0, N_\sigma - 1] & \qquad \text{(60f)} \\ \mathbf{0} \leq u_i^s \leq \mathbf{1} u_{max}^h \quad i \in [0, N_\sigma - 1], & \qquad \text{(60g)} \end{aligned}$$

with the optimization variables now defined as $\zeta^\sigma = [\mathbf{s}, \mathbf{u}^s, \boldsymbol{\Delta}^s]$ consisting of

$$\begin{aligned} \mathbf{s} &= [s_0, s_1, \ldots, s_{N_\sigma}] \in \mathbb{R}^{n_h \times (N_\sigma + 1)}, & \text{(61a)} \\ \mathbf{u}^s &= [u_0^s, u_1^s, \ldots, u_{N_\sigma - 1}^s] \in \mathbb{R}^{m_h \times N_\sigma}, & \text{(61b)} \\ \boldsymbol{\Delta}^s &= [\delta_0^s, \delta_1^s, \ldots, \delta_{N_\sigma - 1}^s] \in \mathbb{R}^{n_t \times N_\sigma}, & \text{(61c)} \end{aligned}$$

being the reduced-order temperature profiles, input- and delta-sequence of the reference signal, respectively. The full-order states in the region of interest are approximated as $\tilde{s}_i^{\mathcal{R}} = HT^\dagger s_i$. Compared to problem (19), the constraints in (60e) are added, which ensure that the input sequence is periodically admissible, i.e. the input sequence $\bar{U} = \mathbf{1}^\top \otimes \mathbf{u}^s$ is admissible for all sizes of $\mathbf{1}$, by making sure that the first input is admissible at the first instant after executing one period. In addition, constraint (60c) ensures periodicity of the obtained temperature profile.

As an illustrative example, problem (60) is solved for the three-cell corner case as depicted in Figure 6a. The transducers are placed according to the utility-based optimization method. The weighting matrix Q is defined as $Q = T^{\dagger^\top} \bar{Q} T$, with $\bar{Q} \in \mathbb{R}^{n \times n}$ a diagonal matrix in which the weight on the ROI voxels is a hundred times larger than for the other voxels. By doing so, the Q matrix is known to be positive definite and the ROI voxels are dominant in the penalty matrix, which is in line with the control objective. The reduced-order model that is used in this example is obtained using a balanced truncation wit a reduction order of $n_h = 45$.

For various lengths of the periodic reference signal the optimization problem (60) is solved, resulting in the optimal switching sequences σ and average costs as stated in Table III. A performance measure J_{avg} is calculated as the average cost over the first N_σ instants according to

$$J_{avg} = \frac{1}{N_\sigma} \sum_{i=0}^{N_\sigma - 1} (\tilde{s}_i^{\mathcal{R}} - x^r)^\top Q (\tilde{s}_i^{\mathcal{R}} - x^r), \tag{62}$$

with $\tilde{s}_i^{\mathcal{R}}$, $i = 0, 1, \ldots, N_\sigma - 1$, now the optimal sequence that follows from optimization of (60). This allows for comparison between the various sequence lengths. The reason that the last term $\tilde{s}_{N_\sigma}^{\mathcal{R}}$ is excluded from the average cost is that the variable s_{N_σ} is only included in the optimization to ensure periodicity of the solution by using constraint (60c). Hence, including the last term would give a biased view on the performance since the initial state would be weighed twice. Uniform movement times are implemented of one sample time between each location. A minimal horizon of six samples is chosen since this is the minimal sequence length such that each location can be visited. Out of the tested prediction horizons, the horizons of six and twelve samples turn out to be optimal. It turns out that the sequence in which each location is visited once for the duration of one sample is optimal. Note that an increased sequence length does not necessarily imply a lower average cost. For example, from Table III it becomes clear that sequence lengths that are not a multiple of six samples are suboptimal since the optimal path that is described by the the sequence of six and twelve samples cannot be exactly followed.

In Figure 7 the average temperature profile $\tilde{X}_{ss}$ corresponding to the twelve-sample reference sequence, which has the lowest average cost, is shown together with the voxel locations in the set $\mathcal{R}$. An elevated temperature plateau of approximately five degrees is achieved over almost the whole region of interest, with slightly increased temperatures at the corners.

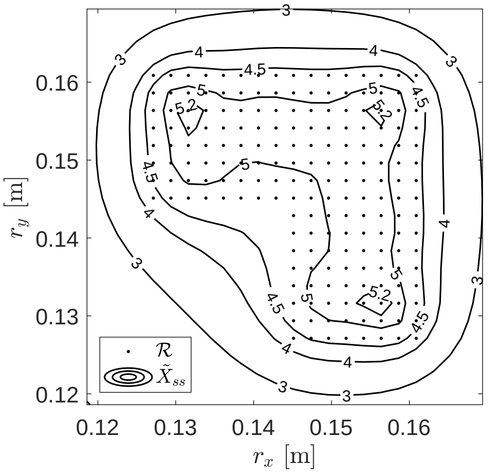


Fig. 7: Average temperature profile of the optimal reference sequence.

TABLE III: Optimal switching sequences and costs for the three-cell case.

N_σ	σ	J_{avg}
6	$\{0,1,0,3,0,2\}$	56.40
7	$\{0,1,1,0,3,0,2\}$	56.44
8	$\{0,3,0,1,0,3,0,2\}$	56.45
9	$\{0,3,0,1,1,0,3,0,2\}$	56.47
10	$\{0,3,0,2,0,1,0,3,0,2\}$	56.45
11	$\{0,3,3,0,2,0,1,0,3,0,2\}$	56.47
12	$\{0,1,0,3,0,2,0,1,0,3,0,2\}$	56.40
13	$\{0,1,0,3,3,0,2,0,1,0,3,0,2\}$	56.42
14	$\{0,1,0,3,0,2,0,3,0,1,0,2,0,3\}$	56.43
15	$\{0,1,0,3,0,2,2,0,3,0,1,0,2,0,3\}$	56.44

C. Stability of periodic tracking MPC

The solution of (60) is now used in a tracking MPC similar to [34], in which the high-level dynamics are assumed to capture the true behavior of the system. To periodically track the sequence $\mathbf{s}$ in (61a) using MPC, at each instant k a reference signal $\mathbf{r}_k = [r_k, r_{k+1}, \dots, r_{k+N}] \in \mathbb{R}^{n_h \times (N+1)}$ is defined where $r_i = s_t$, with

$$t = i - \left\lfloor \frac{i}{N_\sigma} \right\rfloor N_\sigma, \tag{63}$$

where $\lfloor \cdot \rfloor$ denotes the floor function. The input sequence $\mathbf{u}_k^r = [u_k^r, u_{k+1}^r, \dots, u_{k+N-1}^r] \in \mathbb{R}^{m_h \times N}$ and delta sequence $\mathbf{\Delta}_k^r = [\delta_k^r, \delta_{k+1}^r, \dots, \delta_{k+N-1}^r] \in \mathbb{R}^{n_t \times N}$ that correspond to the reference signal are generated similar to $\mathbf{r}_k$, but based on the sequences $\mathbf{u}^s$ and $\mathbf{\Delta}^s$, respectively.

The MPC cost function is now defined as

$$J(\zeta^h, k) = e_{N|k}^{h^\top} P e_{N|k}^h + \sum_{i=0}^{N-1} e_{i|k}^{h^\top} Q e_{i|k}^h, \tag{64}$$

with $Q, P \in \mathbb{R}^{n_h \times n_h}$ the stage-penalty and terminal-penalty matrices, respectively, and $e_{i|k}^h = x_{i|k}^h - r_{k+i} \in R^{n_h}$ the predicted error with respect to the reference signal. The cost function depends on the optimization parameters in ζ^h and is time-varying because of the time-dependent reference trajectory. Using the periodic input sequence, the error dynamics are derived as

$$\begin{aligned} e_{k+1}^h &= x_{k+1}^h - r_{k+1} && \text{(65a)} \\ &= A^h x_k^h + B^h u_k^h - (A^h r_k + B^h u_k^r) && \text{(65b)} \\ &= A^h e_k^h + B^h \Delta u_k, && \text{(65c)} \end{aligned}$$

with $\Delta u_k^h = u_k^h - u_k^r$ the difference in input between the reference sequence and current input. Since A^h is known to be Schur, it can be concluded that for $\Delta u_k^h = 0,\ k \in \mathbb{N}$, the error converges to zero.

A value function is now defined as

$$V(E_k^{h^*}) = J(\zeta_k^{h^*}, k), \tag{66}$$

with $\zeta_k^{h^*} = [U_k^{h^*\top} X_k^{h^*\top} \Delta_k^{*\top}]$ the optimization vector at instant k, consisting of the optimal parameters

$$U_k^{h^*} = \begin{bmatrix} u_{0|k}^{h^*} \\ u_{1|k}^{h^*} \\ \vdots \\ u_{N-1|k}^{h^*} \end{bmatrix}, \quad X_k^{h^*} = \begin{bmatrix} x_{0|k}^{h^*} \\ x_{1|k}^{h^*} \\ \vdots \\ x_{N|k}^{h^*} \end{bmatrix}, \quad \Delta_k^* = \begin{bmatrix} \delta_{0|k}^* \\ \delta_{1|k}^* \\ \vdots \\ \delta_{N-1|k}^* \end{bmatrix},$$

such that (64) is minimal and complies to the high-level MPC constraints (19c) - (19f), and $E_k^{h^*} \in \mathbb{R}^{(N+1)n_h}$ the vector containing the errors $e_{i|k}^{h^*} = x_{i|k}^{h^*} - r_{k+i},\ i = 0, 1, \dots, N$, stacked on top of each other.

Stability of the tracking MPC controller is now proven by showing that the value function (66) satisfies the Lyapunov properties

$$V(\mathbf{0}) = 0, \tag{67a}$$

$$V(E^h_k) > 0 \qquad \forall E^h_k \in \mathbb{R}^{(N+1)n_h} \neq \mathbf{0}, \tag{67b}$$

$$V(E^h_{k+1}) - V(E^h_k) < 0 \qquad \forall E^h_k \in \mathbb{R}^{(N+1)n_h} \neq \mathbf{0}. \tag{67c}$$

The first two properties (67a)-(67b) are ensured due to the matrices Q and P being positive definite. The last property (67c) can be proven by considering the inequality

$$V(\tilde{E}^{h^*}_{k+1}) - V(E^{h^*}_k) < 0, \tag{68}$$

in which $\tilde{E}^{h^*}_{k+1}$ is the error sequence that corresponds to the shifted optimization vector $\tilde{\zeta}^{h^*}_{k+1}$, which is the vector $\zeta^{h^*}_k$, but shifted one step and added with one step of the periodic sequence, see e.g. [35]. Hence, $\tilde{\zeta}^{h^*}_{k+1}$ is such that the input- and delta sequence are $\tilde{U}^{h^*}_{k+1} = [u^{h^*}_{1|k}, \dots, u^{h^*}_{N-1|k}, u^r_{k+N}]$ and $\tilde{\Delta}^*_{k+1} = [\delta^*_{1|k}, \dots, \delta^*_{N-1|k}, \delta^r_{k+N}]$, respectively. From equation (64) it can be seen that

$$\begin{aligned} V(\tilde{E}^{h^*}_{k+1}) = V(E^{h^*}_k) &- e^{h^*\top}_{0|k} Q e^{h^*}_{0|k} - e^{h^*\top}_{N|k} P e^{h^*}_{N|k} \\ &+ e^{h^*\top}_{N|k} Q e^{h^*}_{N|k} + e^{h^*\top}_{N|k} A^{h^\top} P A^h e^{h^*}_{N|k}, \end{aligned} \tag{69}$$

From which it can be concluded that

$$V(\tilde{E}^{h^*}_{k+1}) - V(E^{h^*}_k) \leq -e^{h^*\top}_{0|k} Q e^{h^*}_{0|k} < 0 \quad \forall e^{h^*}_{0|k} \in \mathbb{R}^{n_h} \neq \mathbf{0} \tag{70}$$

if

$$e^{h^*\top}_{N|k} [A^{h^\top} P A^h - P + Q] e^{h^*}_{N|k} \leq 0, \tag{71}$$

which holds if the matrix inequality

$$A^{h^\top} P A^h - P \leq -Q \tag{72}$$

is satisfied. Using optimality, it is known that $V(E^{h^*}_{k+1}) \leq V(\tilde{E}^{h^*}_{k+1})$. Hence, it is concluded that stability of the tracking MPC for the simplified case in which the system follows the reduced dynamics can be guaranteed by construction of P.

This analysis assumes that the shifted input sequence $\tilde{U}^{h^*}_{k+1}$ is always admissible at $k+1$. To comply with this assumption and ensure recursive feasibility, additional constraints should be added on the delta variables such that the input sequence $[U^{h^*}_k, u^r_{k+N}, \dots, u^r_{k+N+h}]$ is admissible, with h the number of travel instances after the prediction horizon until the first nonzero input occurs in the reference sequence. In case of only zero travel times between the transducer locations it is obvious that no additional constraints are necessary since $h = 0$, always. For uniform movement times of p samples between each location, the additional constraints can be specified as

$$\delta_{N-i|k} = \delta^r_{k+N-i} \ \forall i \in [1, q], \tag{73}$$

with $q = h - p$. For non-uniform movement times, the additional constraints become more complex since multiple trajectories could be allowed. For brevity reasons, these cases are not considered in this paper.

D. Stabilizing HMPC

In the previous subsections, the reduced-order dynamics are assumed to capture the true behavior of the system. However, as previously mentioned, simplifying the dynamical model of the system is inherently linked to the existence of prediction errors, i.e., $x^h_{k+1} \neq A^h x^h_k + B^h u_k$. To incorporate this uncertainty, consider the disturbed system

$$x^h_{k+1} = A^h x^h_k + B^h u^h_k + w_k, \tag{74}$$

in which $w_k \in \mathcal{W} \subset \mathbb{R}^{n_h}$ is the disturbance at instant k with $\mathcal{W}$ assumed to be a compact, convex, subset containing the origin. The size of the set $\mathcal{W}$ depends on the accuracy of the reduced model, i.e., a more accurate model shrinks the size of the set $\mathcal{W}$. For example, the error bound (36) shows how the accuracy of the reduced model obtained by balanced truncation depends on both the reduction order and the input. This illustrates that both the order of the reduced model and the inputs influence the boundaries of the set $\mathcal{W}$. The exact relation between the error bound (36) and the set $\mathcal{W}$ is not investigated. In addition, for the other reduction types no formal output bound is found, which makes it difficult to formally quantify the bounds on the disturbance w_k.

The difference between the nominal high-level dynamics

$$\bar{x}^h_{k+1} = A^h \bar{x}^h_k + B^h \bar{u}^h_k \tag{75}$$

and (74) is captured in the error vector $e_k = x^h_k - \bar{x}^h_k \in \mathbb{R}^{n_h}$. For the case in which the input on the nominal system and the disturbed system are the same, i.e., $u^h_k = \bar{u}^h_k$, the error between the real and the nominal system follows the dynamics

$$e_{k+1} = A^h e_k + w_k. \tag{76}$$

A compact set $\mathcal{E}_k$ is now defined as

$$\mathcal{E}_k = \bigoplus_{j=0}^{k-1} (A^h)^j \mathcal{W} = \mathcal{W} \oplus A^h \mathcal{W} \oplus \ldots \oplus (A^h)^{k-1} \mathcal{W}, \tag{77}$$

with $\oplus$ the Minkowski sum, such that from (76) it follows that $e_k \in \mathcal{E}_k$ if $e_0 = 0$. Since A^h is known to be Schur for all three reduction methods discussed in Section V, the set

$$\mathcal{E}_\infty = \bigoplus_{j=0}^{\infty} (A^h)^j \mathcal{W} \tag{78}$$

exists and is forward invariant for the error dynamics (76), i.e., $e \in \mathcal{E}_\infty$ implies that $Ae + w \in \mathcal{E}_\infty$ for all $w \in \mathcal{W}$, see [36]. Hence, the high-level state is known to be contained in the tube $x^h_k \in \{\bar{x}^h_k\} \oplus \mathcal{E}_\infty$ for all instances k, provided that $e_0 \in \mathcal{E}_\infty$. In tube-based MPC, see [12, 37], a typical approach to shrink the set (78) is to use feedback control on the error between the nominal state and the actual state. The input to the disturbed system (74) is then defined as

$$u^h_k = \bar{u}^h_k + K e_k, \tag{79}$$

with $K \in \mathbb{R}^{m_h \times n_h}$ the feedback gain matrix, and $\bar{u}^h_k$ the MPC input that follows from the stabilizing controller applied to the nominal high-level system (75). This changes the error dynamics (76) to

$$e_{k+1} = (A + BK)e + w_k, \tag{80}$$

and consequently the set (78) changes to

$$\mathcal{E}_\infty = \bigoplus_{j=0}^{\infty} (A^h + BK)^j \mathcal{W}, \tag{81}$$

which can be expected to be smaller than the set (78), see [12]. Hence, since the nominal high-level state is known to converge to the reference under inputs $\bar{u}_k^h$, the disturbed system converges to $x_k^h \in \{r_k\} \oplus \mathcal{E}_\infty$ as $k \to \infty$. From this result, the full-order state is known to converge to the time-varying set $x_k \in \mathcal{X}_\infty(k)$ defined as

$$\mathcal{X}_\infty(k) = \{x_k \in \mathbb{R}^n \mid Tx_k \in \{r_k\} \oplus \mathcal{E}_\infty\}, \tag{82}$$

as $k \to \infty$, from which stability to a bounded set around the reference trajectory is concluded.

Due to the present input constraints, the applicability of a feedback gain (79) to shrink the outer-bounding tube is limited. However, the HMPC framework allows the use of low-level control to track the high-level state predictions, which effectively shrinks the tube. One possible choice of the low-level MPC controller is

$$\min_{\zeta_k} \sum_{i=0}^{N} (x_{i|k} - T^\dagger \bar{x}_{i|k}^h)^\top \bar{Q} (x_{i|k} - T^\dagger \bar{x}_{i|k}^h) \tag{83a}$$

$$\text{subject to (21b)} - \text{(21e)}, \tag{83b}$$

with $\bar{Q} \in \mathbb{R}^{n \times n}$ the weighting matrix, in which the full-order states predicted by the nominal controller are tracked. Note that the nominal state trajectory may be generated at the initial time, or generated sequentially using standard MPC for the nominal system.

Remark. *The cost function* (83a) *is not necessarily the best choice with respect to reducing the disturbance w_k. Implementation of the low-level stage cost $(Tx_{i|k} - \bar{x}_{i|k}^h)^\top (Tx_{i|k} - \bar{x}_{i|k}^h)$, which effectively minimizes w_k over the prediction horizon, has proven difficult caused by bad scaling of the matrix $T^\top T$. Nevertheless, it is assumed that* (83a) *contributes in reducing the uncertainty of the nominal model.*

E. Performance of stabilizing HMPC

Using the generated example of a reference trajectory for the three-cell case, see Figure 7, a closed-loop simulation with the stabilizing HMPC controller implemented is performed. The weighting matrix Q is defined equal to the matrix used in computation of the reference sequence. From the defined Q matrix, the stabilizing terminal weighting matrix is computed such that the Lyapunov equation (72) is satisfied with equality. A prediction horizon of $N = 5$ is used for both control layers and a reduced order model of dimension $n_h = 45$ is used that follows from a balanced truncation.

Figure 8a shows the evolution of the cost function of the nominal system $\bar{V}_k$ that is calculated as

$$\bar{V}_k = \bar{e}_{N|k}^{h^*\top} P \bar{e}_{N|k}^{h^*} + \sum_{i=0}^{N-1} \bar{e}_{i|k}^{h^*\top} Q \bar{e}_{N|k}^{h^*}, \tag{84}$$

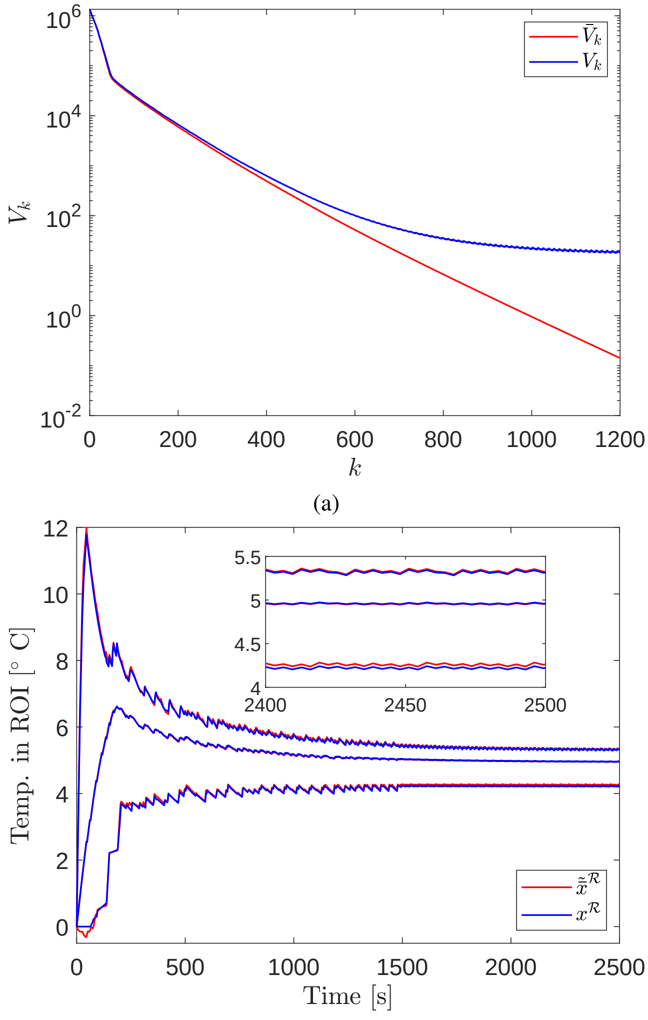
together with the actual cost function V_k that is obtained by applying the low-level inputs to the system, and is calculated as

$$\begin{aligned} V_k = {} & (Tx_{N|k}^* - r_{k+N})^\top P (Tx_{N|k}^* - r_{k+N}) \\ & + \sum_{i=0}^{N-1} (Tx_{i|k}^* - r_{k+i}) Q (Tx_{i|k}^* - r_{k+i}), \end{aligned} \tag{85}$$

in which $\bar{e}_{i|k}^{h^*}$ and $x_{i|k}^*$, $i = 1, \ldots, N$, are the optimized variables that follow from the high- and low-level MPC problems, respectively. It is observed that for the nominal system the value function indeed decreases every instant, whereas the real value function converges towards a constant level. This is in line with our expectations, since the error e_k is known to converge to a bounded set. This makes that the cost function does not necessarily converge to zero, but is upper-bounded depending on the size of $\mathcal{E}_\infty$. The exact properties of $\mathcal{E}_\infty$ and the reduction in size by introducing the low-level MPC (83) are not investigated. However, the tube-based like analysis clearly shows stability towards a bounded set around the reference, which is in line with the analysis of the stabilizing HMPC controller.

In Figure 8b, the evolution of the minimal, average and maximal temperature of the nominal system $\bar{\tilde{x}}^{\mathcal{R}} = T^\dagger \bar{x}^h$ are shown together with $x^{\mathcal{R}}$. The difference between the temperature measures is small, which implies that the sequence of predicted states by the reduced model are well traceable for the low-level MPC. In the first fifty seconds of the simulation, a negative temperature is observed in the temperature profile $\bar{\tilde{x}}^{\mathcal{R}}$, which is physically not possible. Although this may seem potentially problematic, the input behavior is captured correctly by the model since a positive input yields a positive increase in temperature at the locations at which the input is applied. If this was not the case, i.e., a positive input would result in a decreased temperature at the locations of the inputs, issues could occur regarding the stability of the reduced model. Hence, the negative temperatures are an undesirable effect induced by approximating the dynamics, but are not problematic for control purposes.

The temperature evolution shows high, undesirable, temperature peaks up to twelve degrees during the heat-up phase. After approximately 2000 seconds, a steady-state solution is reached which is close to the reference trajectory. To investigate the effect of including the terminal weighting matrix P which satisfies (72), the same simulation is performed but now with $P = Q$. Figure 9 shows the evolution of the temperatures for the two cases in which P is determined using (72) or $P = Q$. A clear difference in heat-up behavior is observed with a much more favorable temperature evolution for the case in which $P = Q$.



(a)

(b)

Fig. 8: Evolution of the value function for both the nominal system and real system (a), and temperature evolution of the nominal and real system (b).

In Appendix C, a more detailed analysis is included on the cause of the undesirable behavior. It turns out that the terminal penalty dominates the cost function when P is calculated to satisfy (72). Two methods are discussed which increase the prediction horizon in order to reduce the relative weight of the terminal term in the total cost function. Although increasing the prediction horizon indeed reduces the temperature peaks significantly, it turns out that computational feasibility becomes an issue for real-time implementation. Hence, it is concluded that the presented HMPC controller with stability guarantee is not well suited for real-life implementation.

F. Discussion

In addition to the undesirable behavior of the proposed HMPC controller with stability guarantee, a second disadvantage is that the steady-state behavior is potentially suboptimal since the optimization problem (60), from which the reference trajectory follows, uses the the reduced-order model to describe the system dynamics. Therefore, the obtained reference trajectory is not necessarily the optimal trajectory, but is an approximation only. The originally proposed controller consisting of the MPC problems (19) and (21) does not face this potential suboptimality as the low-level MPC problem (21), that uses the full-order model (4a) for state predictions, aims to control the states in the ROI to the reference profile x^r, instead of tracking the (potentially suboptimal) high-level state predictions. Hence, the originally proposed controller has the potential to obtain better steady-state performance compared to the HMPC controller with stability guarantee.

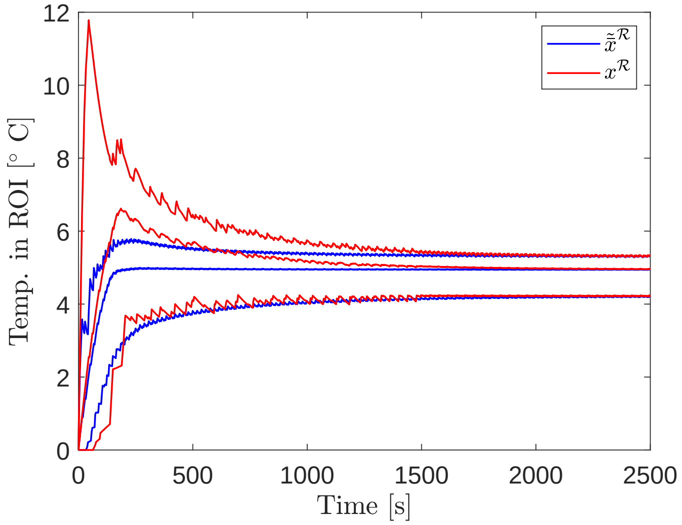


Fig. 9: Comparison of the temperature evolution following from HMPC control with different terminal penalties.

The case in which $P = Q$ shows more desirable behavior, but stability can not be guaranteed by the presented approach. In literature, see [38], a method to guarantee stability is found for which a terminal penalty or constraint is not needed, but stability is guaranteed for a sufficiently large prediction horizon. Due to the complexity of the presented approach, the computation of the prediction horizon which guarantees stability is not established. Comparing the tracking controller with $P = Q$ to the originally proposed controller, the main difference is again that in the originally proposed controller, the full-order model is used to optimize performance instead of tracking the potentially sub-optimal high-level state predictions.

Following this discussion, the HMPC controller as originally proposed in Section IV will be used in all following simulations and experiments.

VIII. Simulation Study

In this section, simulation studies are performed to assess the performance of the HMPC controller as proposed in Section IV. Firstly, a method is discussed to select appropriate reduction orders for the high-level prediction models. Afterwards, case studies are discussed and the comparison to reduced-order MPC and the current clinical implementation based on [8] is made.

A. Order selection for reduced order model

In order to select an appropriate order for the reduced model in the high-level controller, simulations have been performed in which the HMPC optimization problem, consisting of problems (19) and (21), is solved in parallel to the full MIP problem (17). For closed-loop control the inputs of the full MIP problem are used, whereas the HMPC results are stored for comparison. Figure 10 schematically shows this control structure. This approach allows to evaluate the suboptimality that is induced when using hierarchical MPC, by comparing the cost function values of the two approaches. For a fixed horizon, it follows from optimality that at instance k, $J_k^{MIP} \leq J_k^{HMPC}$ in which J_k^{MIP} denotes the cost associated to problem (17) and J_k^{HMPC} the low-level cost of the HMPC controller associated to (21). Sub-optimal behavior is induced at instances at which the predicted transducer paths of the controllers do not coincide. In case both controllers predict the same path, the two costs are theoretically the same.

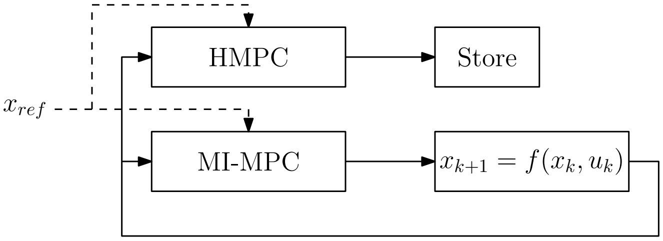


Fig. 10: Parallel control structure

An example study to select appropriate orders for the high-level prediction models is performed for the 5-cell case as depicted in Figure 11. A circular region of interest is chosen with a radius of 16 millimeters. The transducers are placed axisymmetrically over the region of interest with per cell a fixed sonication-point pattern consisting of 13 individual points. The movement matrix for the five-cell case is defined to be

$$\bar{M} = \begin{bmatrix} 0 & 2 & 3 & 3 & 2 \\ 2 & 0 & 2 & 2 & 2 \\ 3 & 2 & 0 & 2 & 2 \\ 3 & 2 & 2 & 0 & 2 \\ 2 & 2 & 2 & 2 & 0 \end{bmatrix}, \tag{86}$$

from which it follows that the maximal movement time is up to three times the sample time. Simulation parameters are set to the values as stated in Table IV.

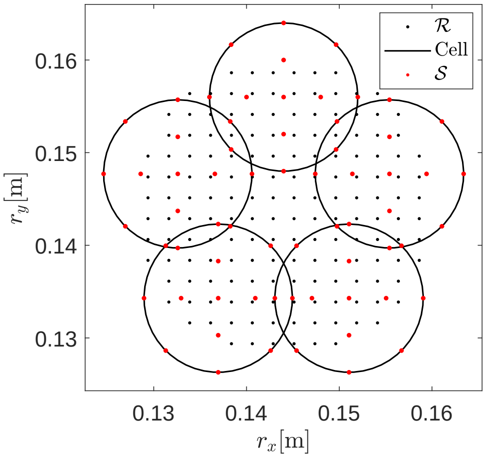


Fig. 11: 5-cell test case

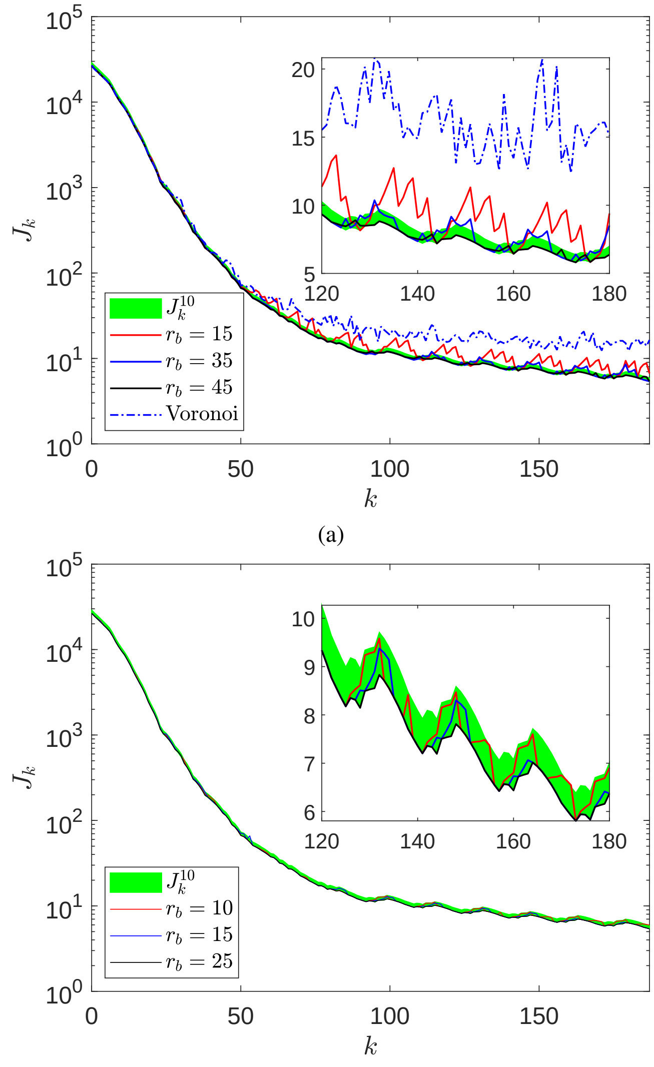
TABLE IV: Simulation parameters

	Q	f	T_{ref}	u_{max}	P_{max}	N	T_s
value	I	$\mathbf{0}$	5	15	100	8	3.2

All three reduced models as discussed in Section V are analyzed with different reduction orders. Figure 12 shows the cost at each simulation instant for different reduction models and orders. A performance interval with respect to the MI-MPC solution of ten percent J_k^{10} is plotted as well which indicates the bound $J_k^{MIP} \leq J_k^{10} \leq 1.1 J_k^{MIP}$. Hence, when the HMPC cost is within this region, it shows that the predicted transducer path results in a low-level cost within ten percent of the optimal cost at instant k. From Figure 12a it is observed that out of the three simulated reduction orders for the balanced truncation model, only $r_b = 45$ satisfies the performance criterion to be within J_k^{10} at all instances k. Comparing this to the POD basis model as shown in Figure 12b, it is observed that a reduction order of just $r_b = 10$ satisfies the performance criterion.

The Voronoi model shows significant suboptimal behavior J_k^{10} criterion, see Figure 12a. Contrary to the other two reduction models, the accuracy of the Voronoi model, as it is defined now, can not be improved by increasing the reduction order. In order to increase the accuracy of the Voronoi model, an optimization over the physical parameters in the A^h matrix could be performed, or the model should be changed by increasing the number of sites in the Voronoi model to allow a higher model order which potentially increases the accuracy of the model. However, since the other two models show promising results, the mentioned options to increase the performance of the Voronoi model are not further investigated.

Computational times for the balanced truncation models and POD models are compared to each other in Table V, while Table VI shows the computational times of the MI-MPC solution and HMPC using the Voronoi model. The mean and maximal computation times are shown together with the number of times N_v that the threshold of three seconds that follows from the system restrictions, see Section II-A, is trespassed. The total number of simulation steps is equal to 188 for this example. From Tables V it is concluded that for a prediction horizon of eight samples, the balanced truncation is not well suited to be used for HMPC control, since the loss in performance is expected to be above ten percent for reduction orders which are computationally feasible. On the other hand, the performance loss when using a POD model



(a)

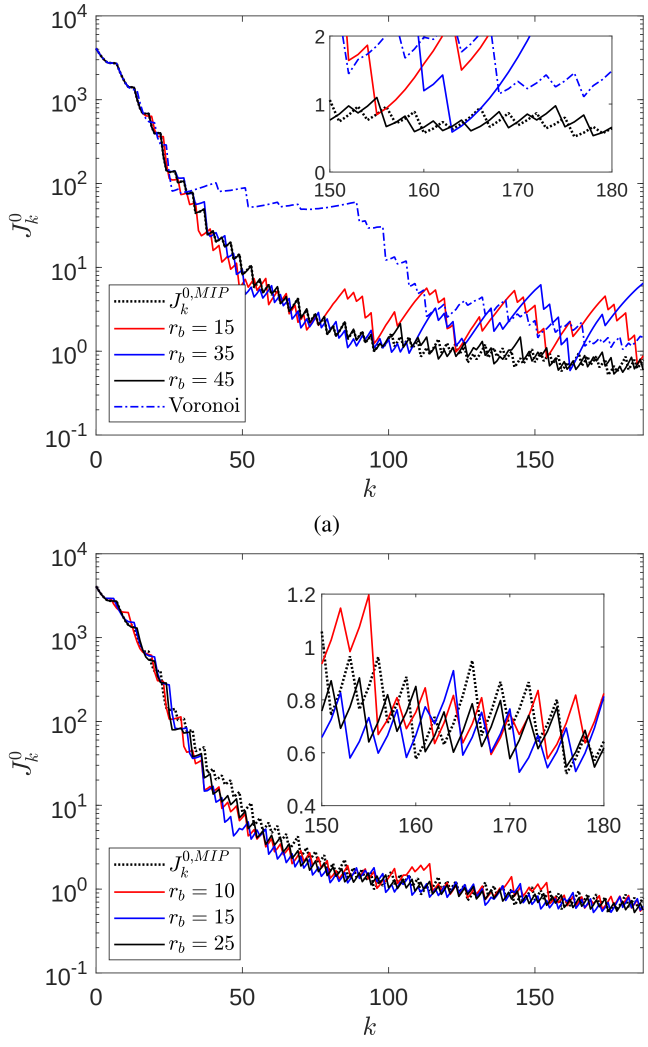
(b)

Fig. 12: Performance comparison between HMPC and full-MIP control for the five-cell case with (a): balanced truncation models and the Voronoi model, and (b): POD-basis models.

with a reduction order of fifteen is expected to be acceptable and computational feasibility is achieved. Table VI shows that the MI-MPC solution is indeed not suited for real-time implementation. In addition, the low computation times of the Voronoi solution show the potential of low-order sparse prediction models. Therefore, a recommendation for future work is to investigate other forms of sparse reduced models that allow for an increased accuracy.

To evaluate how the J_k^{10} criterion relates to closed-loop performance, simulations have been performed with the HMPC controller incorporated in the control loop instead of the MI-MPC controller. The closed-loop performance is measured using the metric J_k^0, which corresponds to the current stage cost.

$$J_k^0 = (x_k^{\mathcal{R}} - x^r)^\top Q (x_k^{\mathcal{R}} - x^r) + f^\top \epsilon_k. \tag{87}$$



(a)

(b)

Fig. 13: Closed-loop performance comparison between HMPC control and full-MIP control for the five-cell case with (a): balanced truncation models and the Voronoi model, and (b): POD-basis models.

TABLE V: Computation times in seconds for different reduced models

	Balanced			POD		
	mean	max	N_v	mean	max	N_v
$r_b = 10$	-	-	-	0.67	1.21	0
$r_b = 15$	0.83	2.40	0	0.99	2.06	0
$r_b = 25$	-	-	-	2.30	4.69	15
$r_b = 35$	3.79	9.14	160	-	-	-
$r_b = 45$	6.64	14.74	188	-	-	-

TABLE VI: Computation times in seconds of MI-MPC solution and HMPC using Voronoi model

	mean	max	N_v
MI-MPC	6.51	29.62	188
Voronoi	0.31	0.78	0

Figure 13 shows the evolution of the performance metric J_k^0

for each closed loop simulation. For the balanced truncation it is observed that for reduction orders $r_b = 15$ and $r_b = 35$, for which the J_k^{10} criterion was violated, the loss in steady-state performance is indeed significant. It was observed that for these reduction orders, the transducer stays in one position for a couple of samples in steady-state, whereas for increased reduction orders and the MI-MPC solution the transducer stays typically one sample at a single location. This difference in transducer path causes the suboptimal behavior. For reduction order $r_b = 45$, which satisfies the performance criterion, it is observed that steady-state performance is comparable to the full MIP case. The Voronoi solution shows severe suboptimal behavior in both heat-up and steady state, which is as expected based on the results related to the J_k^{10} criterion. Therefore, the current definition of the Voronoi model is indeed not suited for HMPC control.

For the POD case, in which all three analyzed reduction orders satisfy the performance criterion, it is observed that for all three cases the steady-state performance is comparable to the MIP solution. In both the balanced truncation case and the POD case, it is observed that during the heat-up phase, the HMPC controller outperforms the full MIP controller. Although this seems counter intuitive, one should realize that the full MIP solution is not necessarily the optimal trajectory over the complete simulation. Hence, when deviating from the MIP solution, it can occur that at a next time instant, a lower value is obtained using a different controller. This effect is expected to reduce with an increased prediction horizon since optimality of the overall behavior is known to increase with growing prediction horizon.

The combination of the results in Figure 12 and Figure 13 leads to the conclusion that the J_k^{10} performance criterion is well suited as a means to decide which reduction order to select for which reduction type. The computation times are then inspected to determine whether the desired reduction order is feasible for real-time implementation.

B. Case study I: 5-cell circular ROI

For the 5-cell configuration as depicted in Figure 11, further analysis of the behavior of the control loop is provided. To mimic real-life experiments, output noise is added to the simulation, which is modeled as uncorrelated zero-mean Gaussian white noise with standard deviation $\sigma_v = \sqrt{0.2}$ [°C]. All results that are shown originate from a simulation in which a POD basis model is used at the high level with a reduction order of $r_b = 15$ and a prediction horizon $N = 8$. Figure 14 shows the temperature evolution in the region of interest and the input power over time. In total five lines are drawn which represent the temperatures (from low to high) $T_{min}, T_{10}, T_{avg}, T_{90}$ and T_{max}. The T_{10} and T_{90} temperature measures refer to the highest and lowest tenth percentile within the examined area, respectively, and subscripts min, avg and max denote the minimal, average and maximal temperature, respectively. It can be seen that the average temperature neatly converges towards the setpoint of five degrees. During heat-up the maximal available power is used after which the input power decreases when the temperature comes close to the setpoint. Short temperature peaks are observed above six degrees. Introducing a ceiling cost could possibly prevent this behavior, however, since the temperature peak is so short and the violation is not severe, no further action is taken. In steady state, the transducer typically stays at a position for no longer than one sample, see Figure 14b. This is in line with the optimal switching sequences as found for the three cell-case in Section VII-B, in which similar behavior is observed.

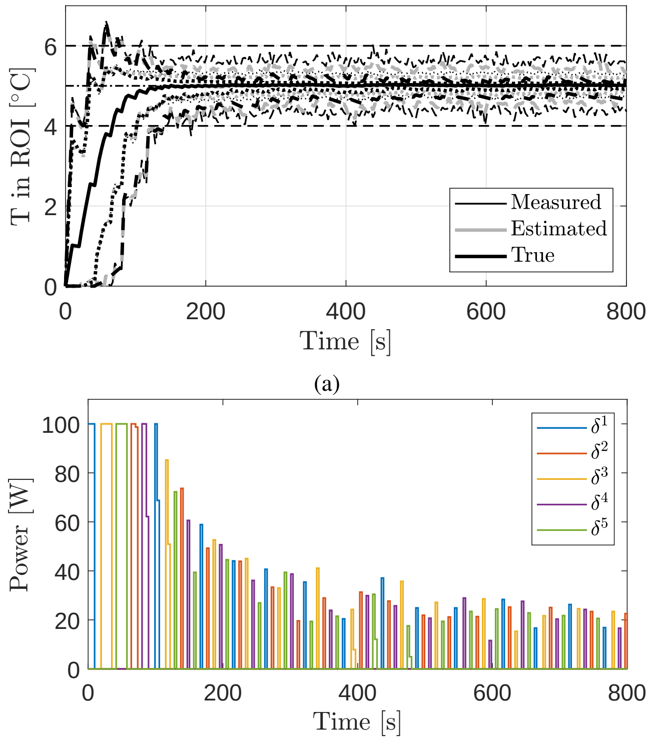
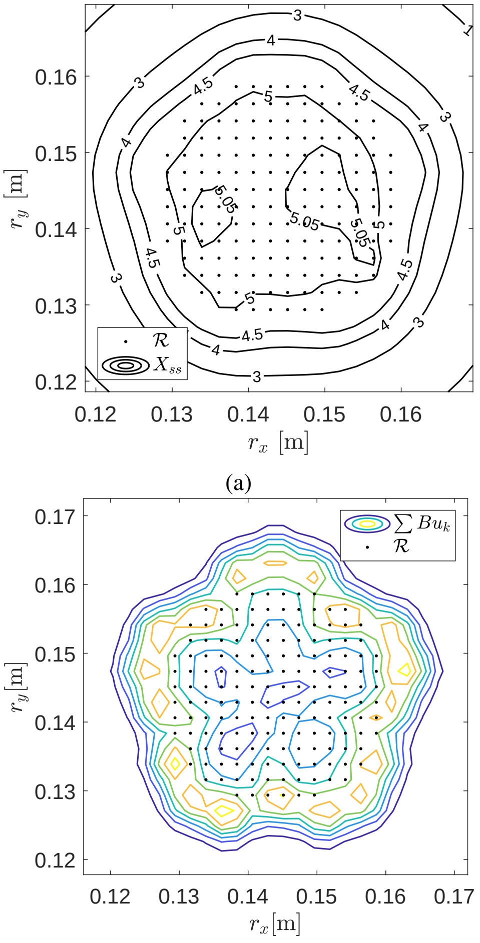


Fig. 14: Temperature evolution in the region of interest (a) and power inputs (b) for the five-cell test case.

Since Figure 14a gives a limited view on the spatial temperature distribution, an extra image is provided, which shows the average temperature profile in steady-state, see Figure 15a. The profile X_{ss} is taken as the average of the last twenty simulation steps corresponding to the time interval of 736-800 seconds. A rather flat temperature plateau is observed over almost the whole ROI. Outside the region of interest, the temperature gradient is steep. Figure 15b shows the summation of the Bu_k term over the whole time-span of the simulation. It is observed that most heat is applied at the outer sonication points. This is as expected since from the temperature gradient in Figure 15a it is observed that at these regions most diffusion occurs being proportional to $\nabla^2 T(r,t)$, see (1).

C. Case study II: 3-cell corner profile

A second simulation study is performed using the 3-cell configurations as depicted in Figure 6a. The sonication points are spread over the cells equal to the five-cell case with thirteen



(a)

(b)

Fig. 15: Spatial distributions for the five-cell case of the average temperature profile in steady-state (a) and sum of Bu_k terms (b).

sonication points per cell. Uniform movement times of three samples are defined, from which the movement matrix

$$\bar{M} = \begin{bmatrix} 0 & 3 & 3 \\ 3 & 0 & 3 \\ 3 & 3 & 0 \end{bmatrix} \qquad (88)$$

follows. To obtain a well-performing high-level prediction model, the same performance test procedure as for the five-cell case is executed. Simulations are performed in which the transducer locations are placed according to method II in which the system's thermal behavior is included in the placing of the cells, see Section VI-B. A prediction horizon of $N = 9$ is used. The results are captured in Table VII with an extra column J_v, which indicates the number of times that the J_k^{10} criterion is violated. The number of simulation steps for this example is 251.

For brevity reasons, the equivalent of Figures 12 and 13 for the three-cell case can be found in Appendix B, but the key observations are discussed now. For all tested combinations of reduction orders and model types, it is observed that the J_k^{10} criterion is violated. This could indicate performance loss in closed-loop control. However, the closed-loop simulations with HMPC implemented show that only one of the tested cases induces a significant performance loss in the J_k^0 metric, being the balanced reduction case with order $r_b = 15$. For all other cases, comparable performance to the full MIP controller is obtained. Hence, violating the J_k^{10} criterion does not necessarily imply a noticeable loss in closed-loop performance. Therefore, the J_k^{10} should only be interpreted as a first-step indicator for potential performance loss. As a validation, incorporating the HMPC controller in the loop turns out useful to determine the desired reduced-model order.

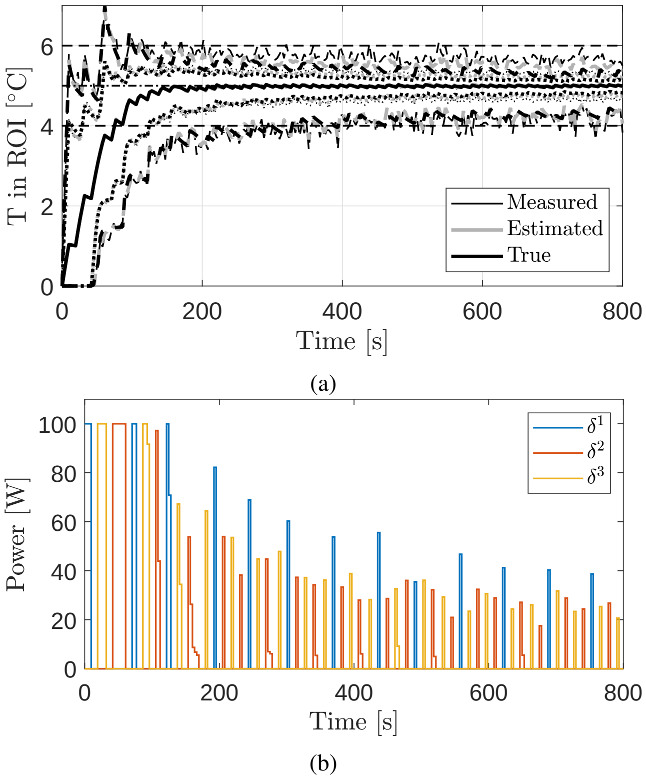
In terms of computation times, all tested combinations are computationally feasible except for a single iteration using the balanced truncation model for order $r_b = 35$. Following this analysis, it turns out that a reduction order of $r_b = 25$ is appropriate for real-time implementation for both reduction methods. Comparing Table V to Table VII shows a clear reduction in computational times for the three-cell case. Even for a larger prediction horizon the computation times are smaller for the same reduction order. This illustrates the effect that the number of integer variables has on the computational complexity of the optimization problem.

TABLE VII: Computation times in seconds for different reduced models

	Balanced				POD			
	mean	max	N_v	J_v	mean	max	N_v	J_v
MIP	3.31	12.12	251	-	3.31	12.12	251	-
$r_b = 10$	-	-	-	-	0.27	0.5	0	10
$r_b = 15$	0.32	0.75	0	77	0.43	0.93	0	12
$r_b = 25$	0.79	1.79	0	22	1.06	2.11	0	13
$r_b = 35$	1.48	3.03	1	22	-	-	-	-

Figure 16 shows the temperature evolution and power input over time in which HMPC is used with the high-level model being the POD model for order $r_b = 25$. Again, a short temperature peak above six degrees is observed during heat-up. Furthermore, the temperature neatly evolves towards the setpoint of five-degrees and the transducer typically stays one sample in a cell to apply heat. In steady state, it is observed that cell one is least visited, but more power is applied here during each visit compared to the others. A possible explanation is that the first position corresponding to δ^1, the upper right cell, needs less heating since it is located at a single corner in the region of interest, whereas the other two cells are located in the neighborhood of two corners. From the temperature gradient shown in Figure 17a it is observed that indeed at the positions corresponding to δ^2 and δ^3, more heat-diffusion occurs and therefore need more visits of the transducer compared to the first location. The sum of Bu_k terms as shown in Figure 17b confirm that the corners of $\mathcal{R}$ are indeed critical.

To investigate the effect of transducer locations on the performance of the HMPC controller, the same simulation is performed for the corner case, but now with the cells located using the utility-based optimization as discussed in Section VI-A. The POD basis for the reduced model now originates

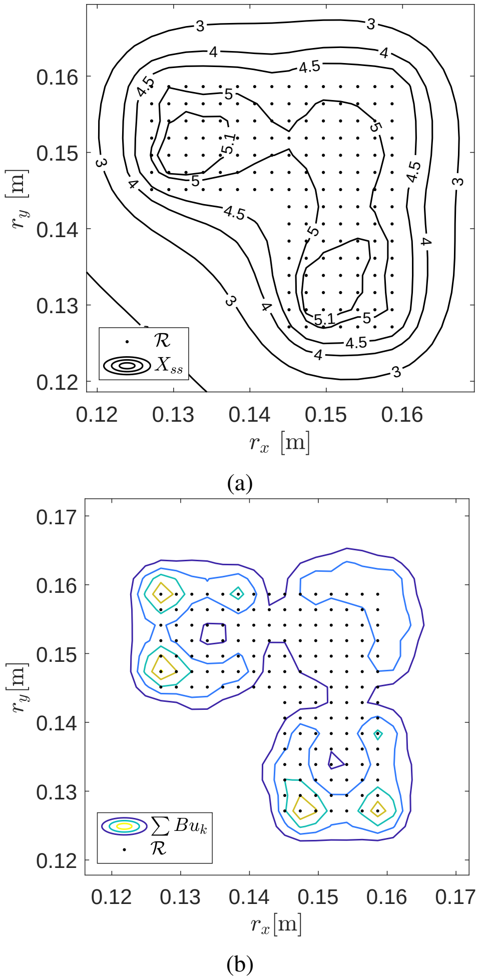


(a)

(b)

Fig. 16: Temperature evolution in the region of interest (a) and power inputs (b) for the three-cell test case.

from a simulation using the unreduced MIP controller with correct transducer locations. The difference in performance is illustrated in Figure 18, which shows the evolution of J_k^0 for both cases. In steady state, a significant performance advantage is observed for the case in which cells are located using method II. This is as expected based on the results in Section VI-C. During heat-up, it is observed that the first method outperforms the second method. A possible explanation is that the first method enables more heat generation inside the region of interest since all sonication points are within the boundaries of the ROI. This can be beneficial in heating-up the target area quickly and thus minimizing the MPC cost. However, as treatments typically take around 90 minutes of which just a short period is needed for heat-up, the steady-state behavior is most important.



(a)

(b)

Fig. 17: Spatial distributions for the three-cell case of the average temperature profile in steady-state (a) and sum of Bu_k terms (b).

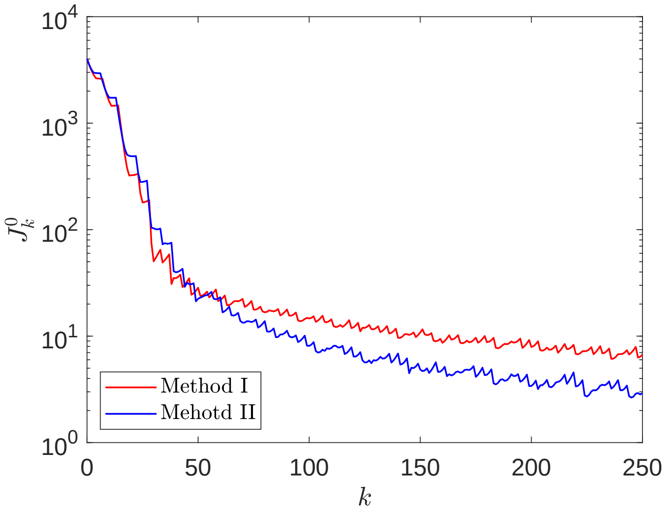


Fig. 18: Evolution of J_k^0 for different cell configurations.

D. Reduced-order MPC versus HMPC

To illustrate the difference in performance between reduced-order MPC and HMPC, the three-cell case with cells placed according to method II is used again. The performance of the two controllers is again compared using the J_k^0 metric. Additionally, to evaluate the approximation error of the reduced model, the approximated cost $\tilde{J}_k^0$ is calculated according to (87), but the estimated temperatures $\tilde{x}_k^{\mathcal{R}} = HT^{\dagger}x_k^h$ are used instead of the true temperatures $x_k^{\mathcal{R}}$. Figure 19a shows the two costs $J_k^{0,rom}$ and $\tilde{J}_k^0$ corresponding to the actual RO-MPC performance and the approximated performance, respectively, together with the closed-loop HMPC performance. The same reduction model is used for both simulations, being the $r_b = 25$ POD model. A significant loss in performance is observed for the RO-MPC controller with respect to the HMPC setup. In addition, a significant difference between the approximated performance and actual performance is observed. Therefore, it is concluded that the reduced model is not accurate enough to be used for closed-loop control.

Increasing the model order to $r_b = 55$ gives the results shown in Figure 19b. The approximated- and real performance of the RO-MPC controller now closely match each other and the performance is comparable to the HMPC controller which uses a model order of $r_b = 25$. Table VIII shows the average calculation times for both controllers. A decomposition is made in high-level calculation times and low-level computation times to show that the low-level step is computationally very fast. The computation times of the RO-MPC controller show that real-life implementation is not possible. The combination of Figure 19 and Table VIII shows that extending RO-MPC to HMPC yields a considerable performance improvement with negligible increase in computation times. Therefore, to obtain performance comparable to the full MIP controller in real-life, in which computational constraints are present, the HMPC control structure is found to be the best option.

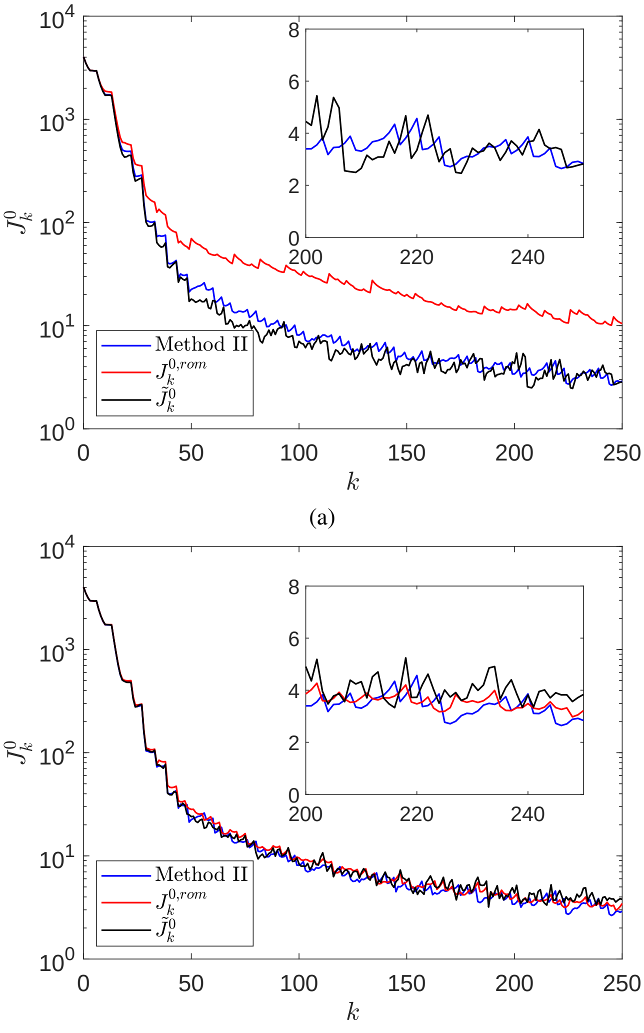
TABLE VIII: Computation times in seconds for HMPC and RO-MPC.

	high-level	low-level	total
HMPC $r_b = 25$	0.99	0.07	1.06
RO-MPC $r_b = 55$	8.16	-	8.16

E. Comparison to binary feedback control

To compare the presented HMPC algorithm to the binary feedback control as is currently used in clinics, the algorithm presented in [8] is used as a guideline to design a controller with comparable performance as the current clinical implementation. The heating algorithm works as depicted in Figure 20.

In each cell, three sub-trajectories are defined as circular shapes of diameters $d_s \in \{4, 10, 16\}$ centered around the transducer axis. The sonication points are distributed axisymmetrically over circles with diameters d_s with an increasing amount of points $n_s \in \{5, 15, 20\}$ per circle for an increasing trajectory diameter. Each circle of sonication points now corresponds to the sub-trajectory with equal diameter. Note that the number of sonication points per cell is 40 compared to 13 for HMPC. The increased number of sonication points has more potential for better performance, however, as a larger number of sonication points means an increased amount of optimization variables in the MPC problem, this is undesirable for the HMPC approach. In the binary feedback control this plays no role, since the input powers are not controlled individually by means of an online optimization procedure, but follow a predetermined protocol. Therefore, if the performance of the binary feedback is worse than HMPC, it is really due to the algorithm, since the input selection actually has better potential.



(a)

(b)

Fig. 19: Performance comparison between reduced-order MPC and HMPC, for (a) $r_b = 25$ and (b) $r_b = 55$.

During heat-up, a predetermined sonication plan is used in which the maximal power is applied to the coldest cell for three samples long after which the average cell temperature is checked. This heat-up plan continues until the average cell temperature is above 4.5 degrees elevation. Next, the algorithm as depicted in Figure 20 is used. To heat up a cell, the average temperatures of the sub-trajectories are monitored, which are calculated as the average temperature of all voxels contained within the boundary of the respective sub-trajectory. Starting from the innermost trajectory, the temperature of the first sub-trajectory is checked and heating is applied if the average temperature is below the threshold of $\bar{T}$ degrees elevation. The sonication power that is used for each sub-trajectory is equally distributed over the corresponding sonication points, and the power is tuned by means of simulation to match the goal of a five degree temperature elevation in the region of interest. If the average temperature of the sub-trajectory is above $\bar{T}$ degrees elevation, but the average cell-temperature is below this threshold, the successive sub-trajectory is selected being the next outward trajectory. This continues until the threshold is met for the average cell temperature, after which the transducer moves to the coldest cell and repeats this heating protocol. Note that it could occur that the heating of a sub-trajectory is followed up by the heating of an inward sub-trajectory. However, as heat diffuses outwards, this is unlikely to happen and will only occur in the presence of measurement noise.

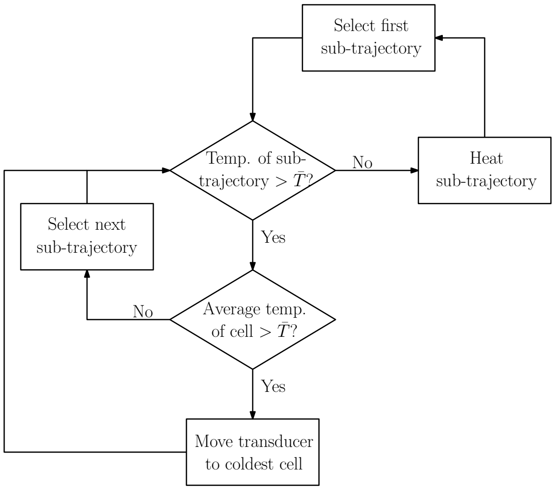

Select first sub-trajectory
Temp. of sub-trajectory > $\bar{T}$?
No
Heat sub-trajectory
Yes
Select next sub-trajectory
No
Average temp. of cell > $\bar{T}$?
Yes
Move transducer to coldest cell

Fig. 20: Schematic illustration of the binary feedback algorithm.

The binary feedback control is simulated in closed loop for the 5-cell and 3-cell case studies presented in Sections VIII-B and VIII-C, respectively, and compared to the HMPC approach. For the 3-cell case, the cells are located according to optimization method II as presented in Section VI-B, which includes the thermal behavior of the system. The sonication powers per trajectory are set to $P_s \in \{5, 10, 15\}$ [W] for the five-cell case and to $P_s \in \{10, 15, 20\}$ [W] for the three-cell case, in which the input power increases for an increasing trajectory diameter, and the temperature threshold is set to $\bar{T} = 4.7$ degrees for both cases. These numeric values are determined via tuning, such that for both cases an average temperature of five degrees elevation is reached in steady-state. The maximal power is set to $P_{max} = 100$ [W], which corresponds to the maximal power in the other simulation studies. In Figure 21, the results of applying the binary feedback algorithm are shown for both cases, which include the temperature evolution of the minimal, maximal and average temperature in Figures 21a and 21b, the steady-state temperature profile X_{ss} in Figures 21c and 21d, and the spatial distribution of the summation of Bu_k over the entire duration of the simulation in Figures 21e and 21f. Additionally, Figures 21a and 21b also show the HMPC results to allow for comparison between the two controllers, and X_{ss} is calculated as the average temperature profile over the last twenty simulation steps corresponding to the time interval 736-800 [s].

The evolution of the minimal, maximal and average temperatures show that a much tighter temperature range is achieved when using HMPC control compared to the binary feedback algorithm. From the spatial profiles X_{ss} it is observed that compared to the HMPC controller, see Figures 15a and 17a, the temperature profiles are less flat, which is not desirable as a homogeneous profile is preferred. Comparing the spatial distributions of $\sum Bu_k$ to the results obtained when using HMPC, as shown in Figures 15b and 17b, it is observed that most heat generation now occurs inside the ROI, whereas the HMPC controller mostly heats the edges of the region of interest. This also explains the less flat temperature profiles since, keeping in mind that there is no out of plane diffusion, heating the edge of the ROI will achieve a flatter temperature profile compared to heating the center.

To quantitatively compare the two controllers, the stage cost related to the current temperature profile J_k^0, see (87), is taken as a performance measure again. Figure 22 shows the evolution of J_k^0 for both controllers and both cases. The results show that the HMPC performs a factor ten to a hundred times better than the binary feedback control when using J_k^0 to measure performance. Therefore, it is concluded that the proposed HMPC strategy significantly outperforms the control strategy that is currently used for large-area hyperthermia treatments. In addition, the tuning of the parameters for the binary feedback controller is a cumbersome process based on trial and error. The tuning of the HMPC controller is more intuitive as it only involves the definition of the Q matrix. Hence, besides achieving better performance, the HMPC comes with the additional benefit that no case-specific tuning of parameters is needed.

IX. Experiments

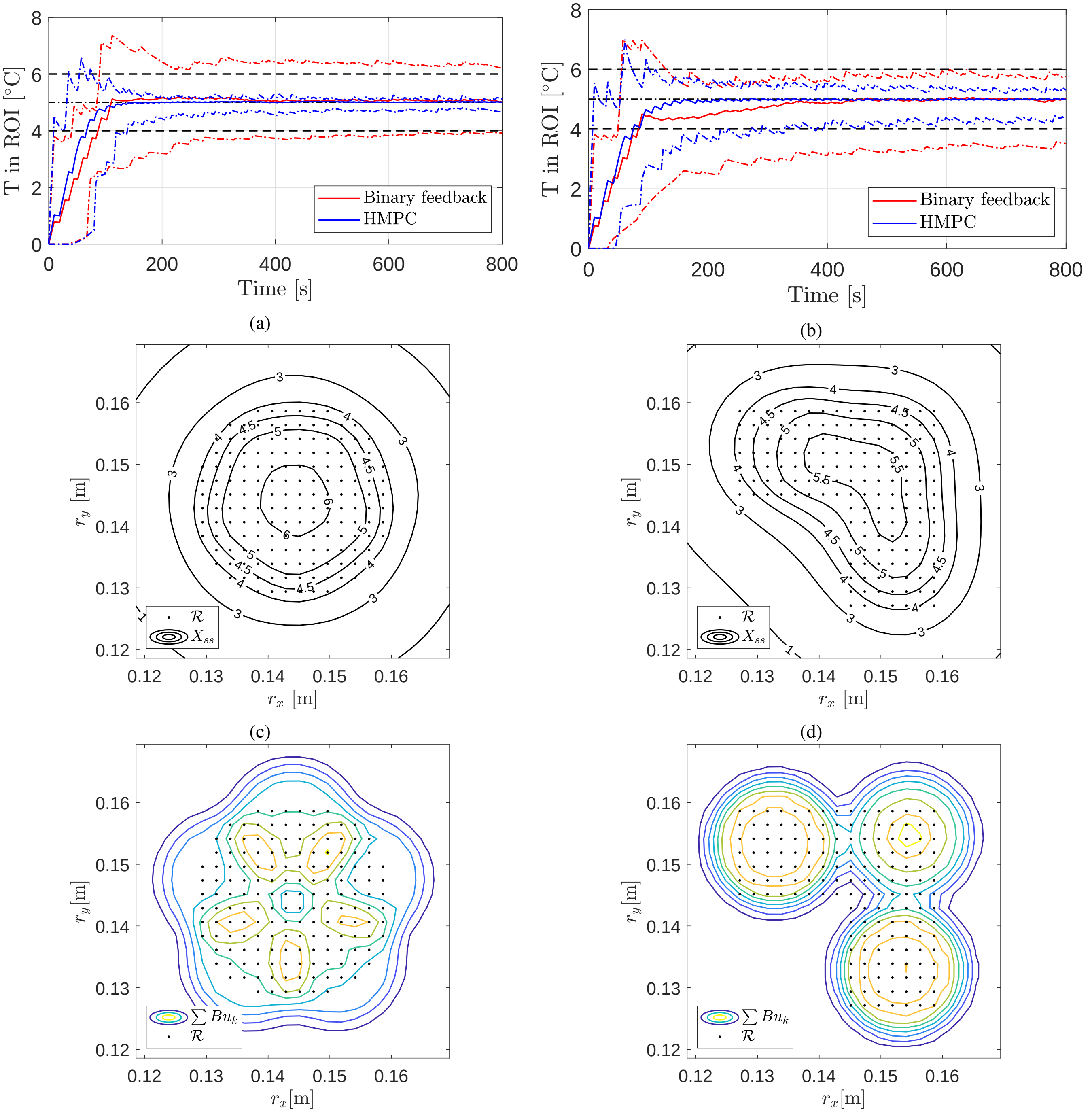
Experiments are conducted at the Uniklinik Köln which is a university hospital located in Cologne, Germany. The experimental setup consists of the Profound Sonalleve MR-HIFU therapy platform and a phantom is used to execute the in-vitro experiments. The phantom used for experiments is a polymethyl metacrylate (PMMA) container filled with a polymer mixture.

Due to solver times of the controller, a small adaptation is needed for correct implementation. So far, the solution at instant k is assumed to be instantaneously available from which the current input $u_{0|k}$ follows. Since in practice the computational delay to obtain the current input is up to three seconds, a one-step-ahead approach is used. That is, instead of calculating the current input, the first input and transducer position are now fixed as

$$u_{0|k} = u_{1|k-1} \tag{89a}$$

$$\delta_{0|k} = \delta_{1|k-1}, \tag{89b}$$

and the input to be applied at $k + 1$ is extracted from the solution at instant k. The observer step, which provides the controller with the estimated state, requires negligible computation times (<0.05 [s]). Computation of the control inputs is done using Matlab, while Phyton is used for communication



(a)

(b)

(c)

(d)

(e)

(f)

Fig. 21: Results obtained by applying the binary feedback algorithm for the five-cell (left column) case and three-cell case (right column). Evolution of the temperatures T_{min}, T_{avg}, and T_{max} over time in (a) and (b) for both HMPC and binary feedback, spatial distribution of X_{ss} in (c) and (d), and $\sum Bu_k$ in (e) and (f)

with the MR-HIFU setup. A visual representation of the information flow is provided in Figure 23.

From simulations, non-physical transducer paths are observed. That is, following the constraints (14) in the MPC problem, the individual $\delta_{i|k}$ vectors are not constrained to be physically possible, but only to construct input constraints that incorporate the effect of mechanical displacement of the transducer by inhibiting nonzero inputs in a treatment cell when the transducer is not ready at the corresponding location. Hence, the movement constraints are well suited for simulation, but the elements of Δ_k do not directly describe the desired transducer path. To correctly communicate the

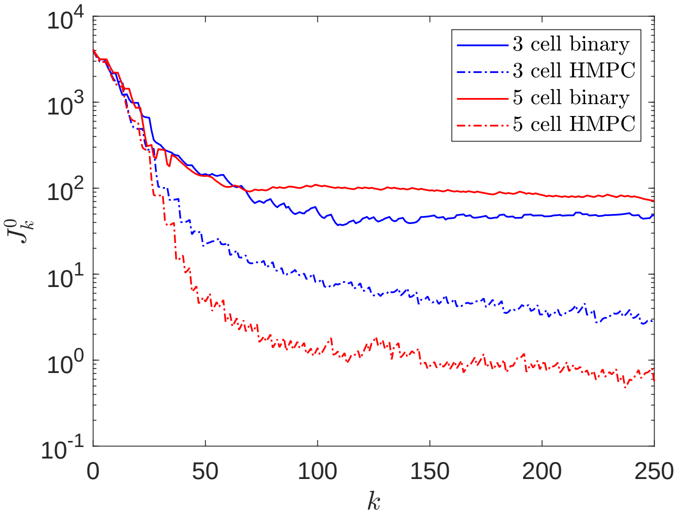


Fig. 22: Performance comparison between binary feedback control and HMPC based on J_k^0 for both the three-cell and five-cell case.

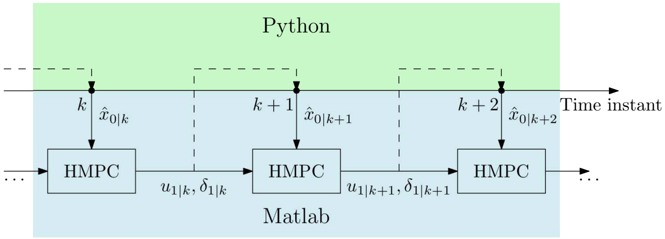


Fig. 23: Information flow during experiments. The arrows connecting the HMPC blocks contain the information related to constraints (89)

transducer position to the system, the elements of Δ_k are updated according to

$$\delta_{i|k} = \begin{cases} \delta_{i|k} & \text{if } u_{i|k} \neq 0 \\ \delta_{i+j_{min}|k} & \text{else} \end{cases}, \tag{90}$$

where

$$j_{min} = \min\{j \in \mathbb{N} \mid u_{i+j|k} \neq 0\}$$

and $\mathbb{N}$ is the set of natural numbers. Note that only the current vector $\delta_{0|k}$ is communicated to the system, but updating the whole sequence allows for easier to interpret data and could function in a fall-back strategy at instances where a failure occurs in the MPC algorithm. Implementing both adaptations (89) and (90) makes the HMPC controller suited for real-time implementation. Several experiments are conducted using the five-cell and three-cell cases as discussed in the simulation study.

A. 5-cell circular ROI

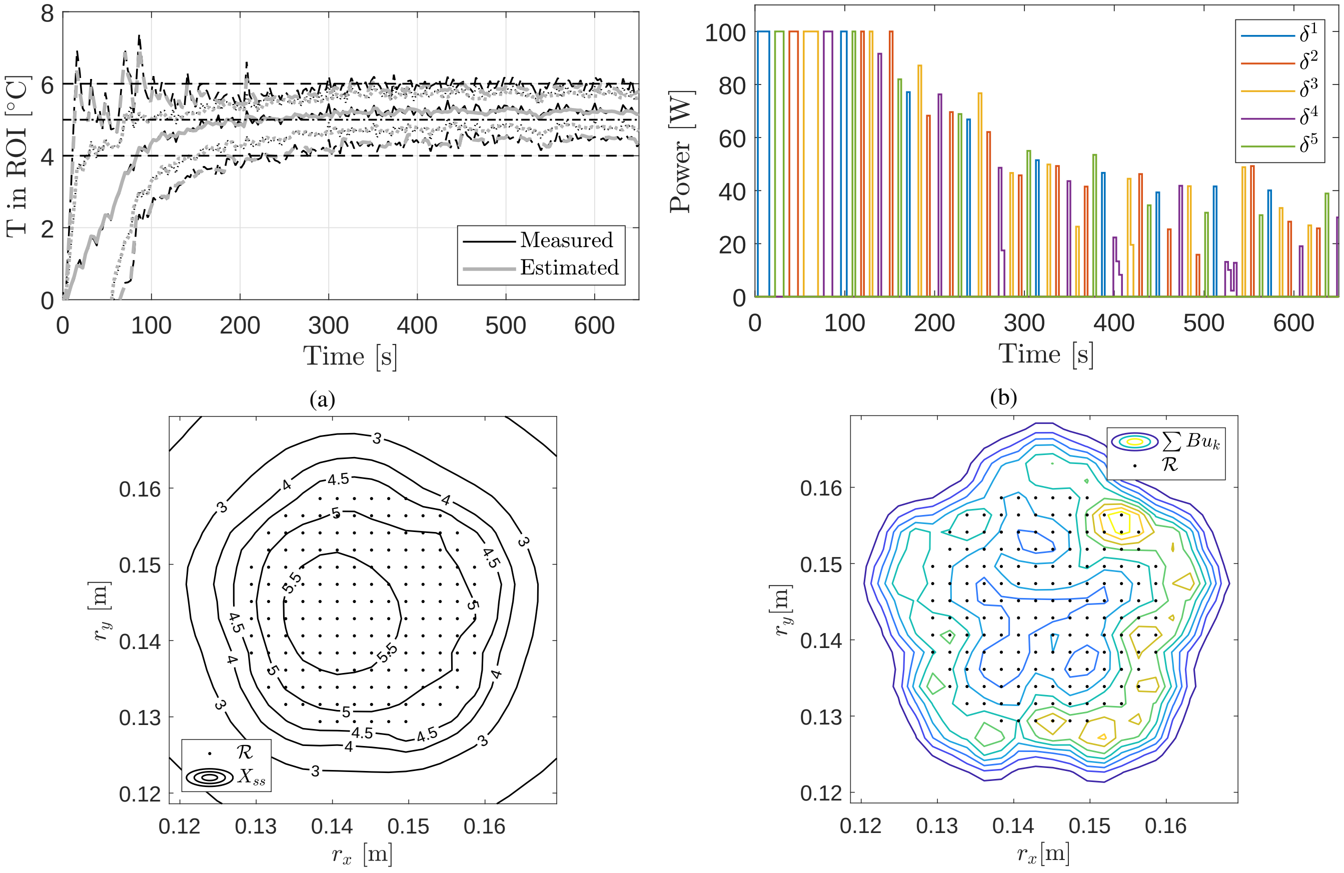
The first experiment is the five cell case with a circular ROI as depicted in Figure 11. A slightly different movement matrix then the one used in simulations is obtained via measurements on the real system and is defined as

$$\bar{M} = \begin{bmatrix} 0 & 3 & 3 & 3 & 2 \\ 3 & 0 & 2 & 3 & 2 \\ 3 & 2 & 0 & 2 & 3 \\ 3 & 3 & 2 & 0 & 3 \\ 2 & 2 & 3 & 3 & 0 \end{bmatrix}. \tag{91}$$

The HMPC controller uses a POD model of order $r_b = 20$ with a prediction horizon of $N = 7$. Experimental results are shown in Figure 24. Both the temperature evolution over time and the sum of inputs show similar behavior as in the simulations. During the heat-up phase maximal power is used, after which the inputs decrease on average when the temperature setpoint is reached. The average temperatures are slightly above the setpoint which could indicate a form of model mismatch. From the spatial profiles of X_{ss} and the sum of Bu_k it is observed that the maximal temperature is located on the left side of the ROI, while most heat is sonicated on the right side of the region of interest. A possible explanation for this behavior is found in a mismatch between the baseline images that are used for thermometry. During some experiments, clear jumps in the temperature profile were observed when the baseline image changed. Although no clear temperature jumps are observed in this particular experiment, it is believed that a mismatch still causes the off-centered temperature peak. In particular, Figure 24b shows that at location corresponding to δ^4 at $t \approx 400$ [s] and $t \approx 540$ [s], remarkably low inputs are generated compared to the other inputs. Therefore, the baseline image of location four seems to be not in line with the other images. Besides the off-centered temperature peak, the temperature profile is less flat compared to simulations. Potential causes are unforeseen out of plane interaction effects and a mismatch of the thermal parameters of the model compared to the real phantom values. Therefore, further model identification is advised to further improve performance. This is left as a recommendation for future work as it is beyond the scope of this project.

B. 3-cell corner profile

The second experiment is the three-cell case with the transducer positions placed using the utility-based optimization. Due to the limited number of experiments, the other cell configuration that follows from the optimization procedure in Section VI-B is not tested unfortunately. The HMPC controller uses a POD model of order $r_b = 20$ with a prediction horizon of $N = 9$ and the movement matrix is equal to (88). In Figure 25 the experimental results of the second experiment are shown. The temperature evolution shows a neat convergence towards the setpoint but from the lower temperature it is seen that steady-state is not yet reached. Therefore, the experiment time of approximately ten minutes turns out to be a bit short. The input over time shows similar behavior as the simulation result in Figure 16b in which the location corresponding to δ^1 is least visited. Although the configuration of the cells is slightly different in this experiment, the same reasoning as for the simulation study holds to explain this behavior. The



(a)

(b)

(c)

(d)

Fig. 24: Experimental results for the five-cell case. Evolution of the temperatures T_{min}, T_{10}, T_{avg}, T_{90} and T_{max} over time (a), input power over time (b) and spatial distributions of X_{ss} and $\sum Bu_k$ in (c) and (d), respectively.

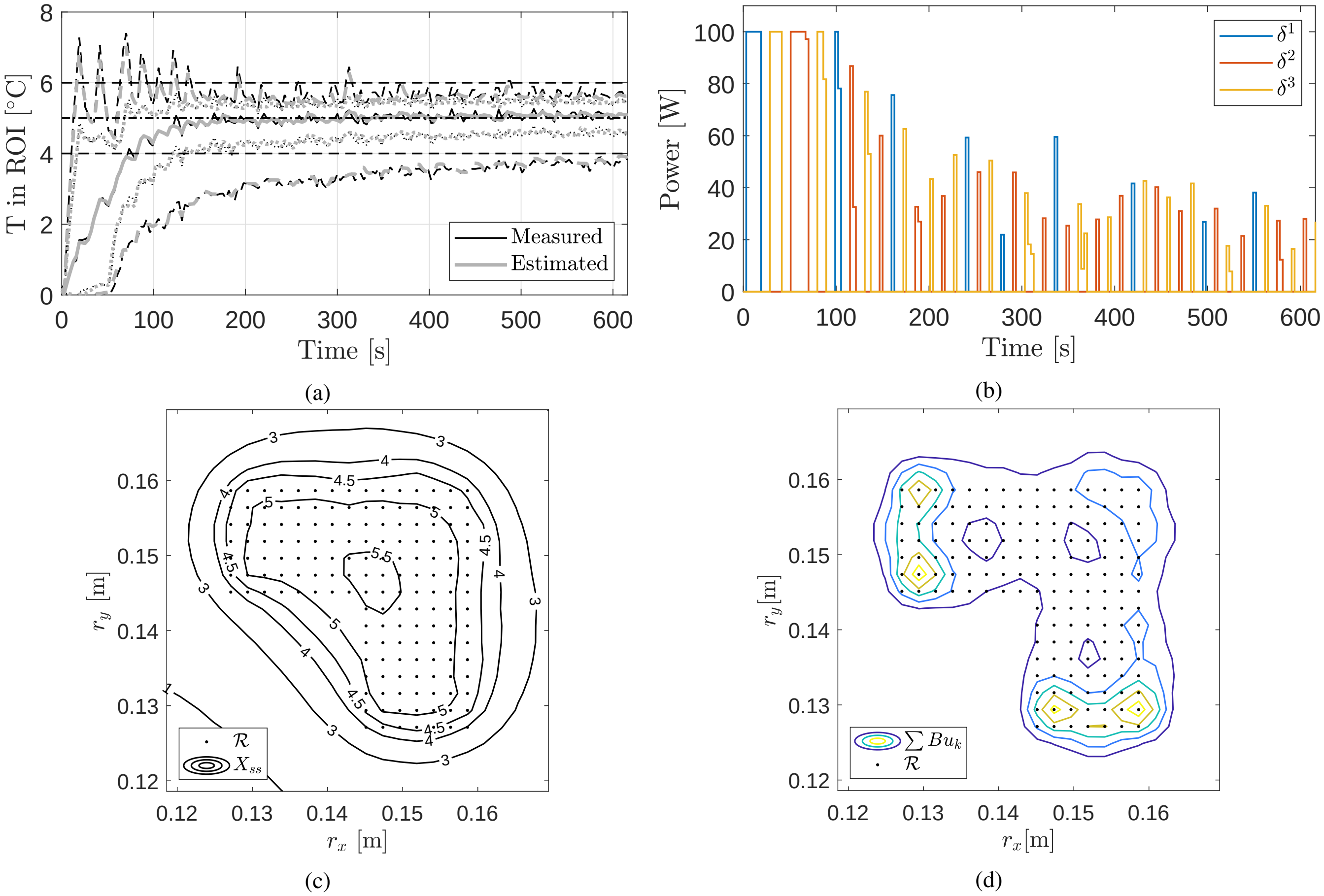
spatial distributions of the temperature profile and input terms show symmetry over the diagonal of the ROI, which indicates that for this experiment the baseline images are well aligned. A less flat temperature distribution is obtained compared to simulations as is also the case for the five-cell configuration.

From the experimental data it is observed that the temperature peaks above six degrees are now a bit more violent compared to simulation results. Two possible explanations are thought of:

- By fixing the first input in each optimization sequence, a form of suboptimal control is introduced since the most recent state knowledge is not used to calculate this input. Due to an existing mismatch between model and real-life, it could occur that temperature peaks exceed the maximal predicted temperatures in the previous time step.
- A second explanation is found in a mismatch of the input model. Besides under- or over estimating the parameters of the input-profile, another cause could be the misalignment between the sonication points and monitored temperature locations. Since the sonication points are not constrained to be at voxel-center coordinates, the peak values of the temperature increase caused by applied heat are not seen in the state vector. Therefore, at a next iteration, a temperature increase could occur at spots which is not caused by applying heat, but originates from a previous sonication.

X. Conclusions and Recommendations

The main contribution of this paper is a novel hierarchical control structure in which two MPC controllers operate together to determine optimal control actions in large-area MR-HIFU treatments. Three different reduced-order prediction models are analyzed out of which the data-based POD models show the best performance. A simulation study shows that compared to the previously presented full blown mixed-integer MPC controller in [15], similar performance is obtained, but the HMPC controller shows more potential for real-time implementation based on a significant reduction in computational times. In Table IX, the computational times are shown for two test cases consisting of five and three transducer cells, for which both controllers obtain similar performance. In addition, a significant increase in performance is observed when using HMPC compared to a binary feedback algorithm as in [8]. Experiments are conducted which confirm the applicability of the presented controller on a real-life setup.



(a)

(b)

(c)

(d)

Fig. 25: Experimental results for the three-cell case. Evolution of the temperatures T_{min}, T_{10}, T_{avg}, T_{90} and T_{max} over time (a), input power over time (b) and spatial distributions of X_{ss} and $\sum Bu_k$ in (c) and (d), respectively.

TABLE IX: Computation times in seconds for MI-MPC and HMPC, for which closed-loop performance is comparable.

	5-cell		3-cell	
	mean	max	mean	max
MI-MPC	6.51	29.62	3.31	12.12
HMPC	0.99	2.06	1.06	2.11

A second contribution is making treatments of irregularly shaped tumors possible. In previous works, only circular target areas are considered with the cells located on a hexagonal grid. To enable treatments of irregular shapes, two methods are presented to determine the transducer locations for arbitrarily shaped tumors. The first method uses the maximization of a utility function, which results in a (somewhat) homogeneous distribution of the transducer locations over the region of interest. A second method includes a simplified representation of the system's thermal behavior in which input constraints related to the movement of the heating actuator are neglected, and the objective is to optimize the steady-state temperature profile. Typically, it is observed that the second method places the transducer locations at the outer edge of the region of interest as required to counteract heat loss due to steep temperature gradients. Out of the two options, the second method performs better in closed-loop simulation.

The combination of the two discussed contributions is seen as a significant step forward in reaching the full potential of MPC control in hyperthermia treatments, ultimately enhancing treatment quality and success. However, the presented work does leave room for discussion and subsequent improvement.

As a first recommendation, the accuracy of the prediction models could be further increased by performing an elaborate model identification. From experiments, it was observed that unexpected temperature peaks can occur of which the cause is most likely a plant-model mismatch. Moreover, the accuracy of the reduced-order models depends on the accuracy of the full-order model. Hence, the potential benefits of a better full-order model are twofold since both the high- and low-level predictions become more accurate.

Secondly, robustness against system in-homogeneity is not addressed in this work, which could be problematic for real-life implementation in which a perfectly homogeneous plant is unlikely. An interesting approach to include some robustness is to extend the presented controller to an offset-free MPC controller as is done in, for example, [7]. Using a disturbance estimator, the steady-state offset resulting from plant-model mismatch can be removed, and could also be used in addition to the recommended model identification. The applicability of the offset-free algorithm in a hierarchical setting is therefore seen as an interesting research topic to further investigate.

Appendix A
Stability of POD-reduced models

Stability of the POD-based models can be proven by realizing that the system matrices that results from discretization of the bioheat equation (1) are symmetric and Schur, see [39]. Now, following the min-max theorem, it is known that the eigenvalues of A satisfy

$$\lambda_{min}(A) = \min_{y, \|y\|=1} y^\top A y \tag{92}$$

$$\lambda_{max}(A) = \max_{y, \|y\|=1} y^\top A y. \tag{93}$$

Similarly, the eigenvalues of A^h satisfy

$$\lambda_{min}(A^h) = \min_{y, \|y\|=1} y^\top T^\top A T y \tag{94}$$

$$\lambda_{max}(A^h) = \max_{y, \|y\|=1} y^\top T^\top A T y. \tag{95}$$

Realizing now that

$$\{y \mid y \in \text{Im}(T), \|y\| = 1\} \subseteq \{y \mid \|y\| = 1\}, \tag{96}$$

it can be concluded that the eigenvalues of A^h must satisfy

$$\lambda_{min}(A) \leq \lambda_i(A^h) \leq \lambda_{max} \quad i \in [1, n_h], \tag{97}$$

which implies stability of the reduced system.

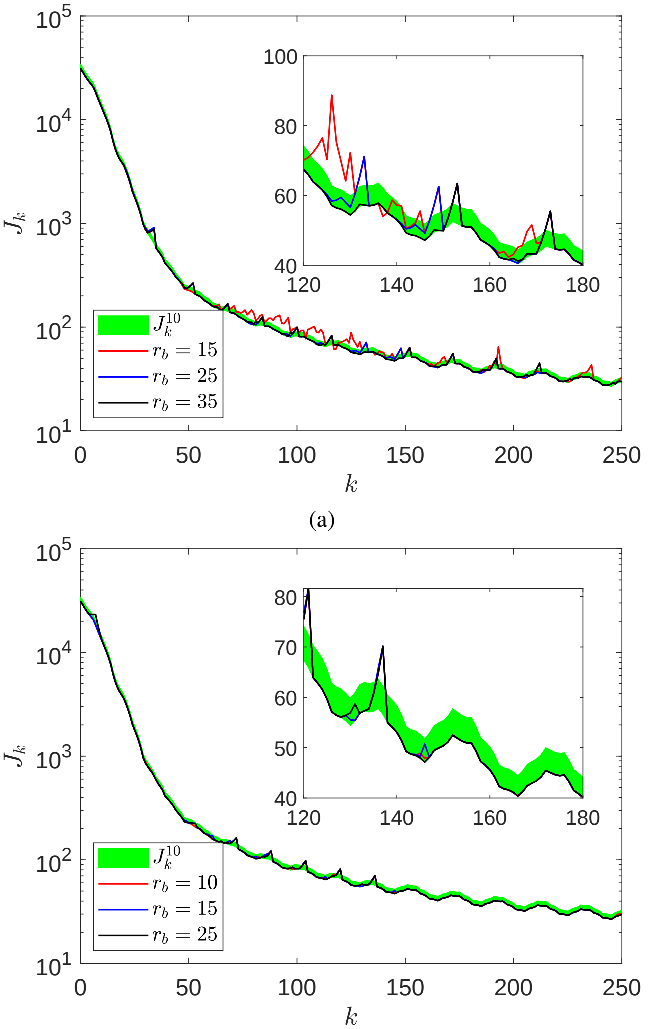
Appendix B
Performance of corner case

Figures 26 and 27 shows the evolution of the cost function for the cases in which HMPC is run parallel to MI-MPC and for closed-loop HMPC, respectively. Subfigures 26a and 27a show results for reduced-models that are derived using balanced truncation, whereas the results in subfigures 26b and 27b are obtained using POD-based models.

Appendix C
Reducing the effect of the terminal weight

Figure 28 shows the nominal value function for both cases in which the terminal penalty is set to satisfy the Lyapunov equation (72) and to $P = Q$, respectively. In addition, the terminal penalty in the cost function $e_{N|k}^\top P e_{N|k}$ is shown together with $\tilde{e}_{N|k}^\top P \tilde{e}_{N|k}$ which corresponds to the terminal error for the case in which $P = Q$, but then weighted using P. The difference in the initial value of V_k shows that the P matrix dominates the cost function since including P results in an increase of approximately a factor 10^2. Moreover, the difference between the terminal term and V_k is small which also indicates the dominance of the terminal weight. Hence, optimization when using the term $e_{N|k}^\top P e_{N|k}$ indeed reduces the terminal weight, however the temperature evolution is affected in an undesirable way.

To reduce the undesirable effects of implementing the terminal weighting matrix, two options are discussed.

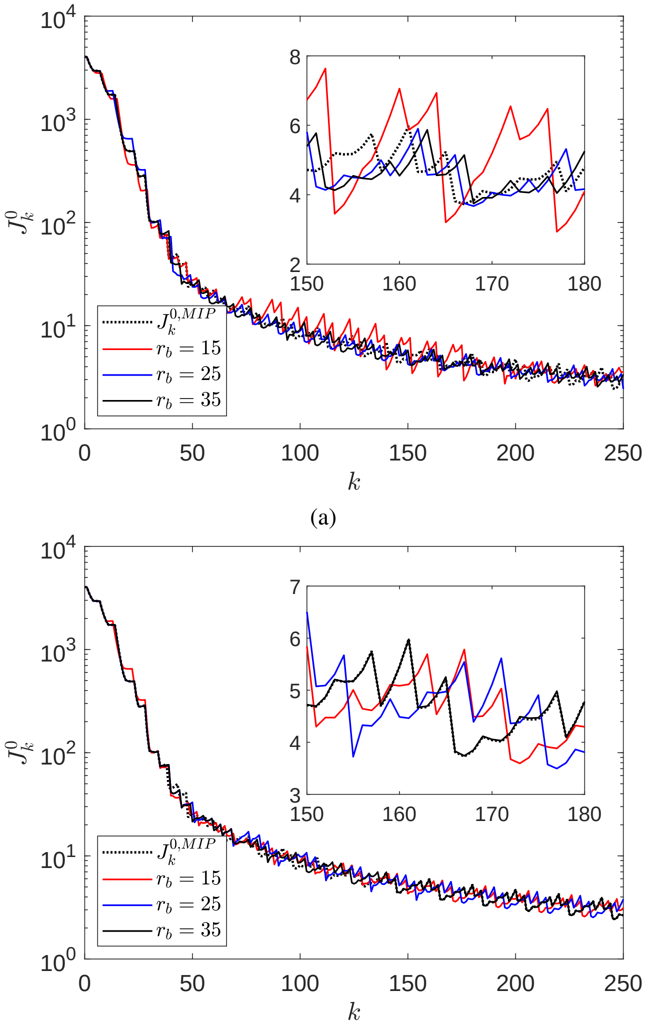


(a)

(b)

Fig. 26: Performance comparison between HMPC and full-MIP control for the three-cell case with (a): balanced truncation models and the Voronoi model, and (b): POD-basis models.

The first option aims at reducing the influence of the terminal penalty in (64) by increasing the prediction horizon N. This decreases the relative weight of the terminal penalty with respect to the sum of stage costs in the cost function. In order to make long horizon predictions computationally tractable, a move blocking approach has been adopted as presented in, for example, [40]. Move blocking enables a reduction in optimization variables by fixing the input to be constant over a span of several time steps. Instead of solving for the input sequence $U_k = [u_{0|k}, u_{1|k}, \ldots, u_{N-1|k}] \in R^{n_u \times N}$, the reduced problem solves for $\hat{U}_k = [\hat{u}_{0|k}, \hat{u}_{1|k}, \ldots, \hat{u}_{M-1|k}] \in R^{n_u \times M}$, with $M < N$, such that $U_k = (T_u \otimes I_{n_u})\hat{U}_k$, with $T_u \in \{0,1\}^{N \times M}$ a so-called blocking matrix and n_u the dimension of a single input vector. The blocking matrix is specified such that each row contains exactly one nonzero element being a one. An



(a)

(b)

Fig. 27: Closed-loop performance comparison between HMPC and full-MIP control for the three-cell case with (a): balanced truncation models and the Voronoi model, and (b): POD-basis models.

example of a blocking matrix:

$$T_u = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 1 \end{bmatrix},$$

which corresponds to a prediction horizon of 5 steps, while only three inputs are calculated.

A second option comes from the idea to decrease the relative weight of P with respect to the matrix Q, which in turn decreases the effect of the terminal term in the total optimization cost. From (72) it can be seen that a smaller A^h matrix leads to a smaller P matrix, which decreases the relative weight of P over Q. Since A^h is a stable matrix, it is obvious to see that for a larger sampling time, A^h decreases.

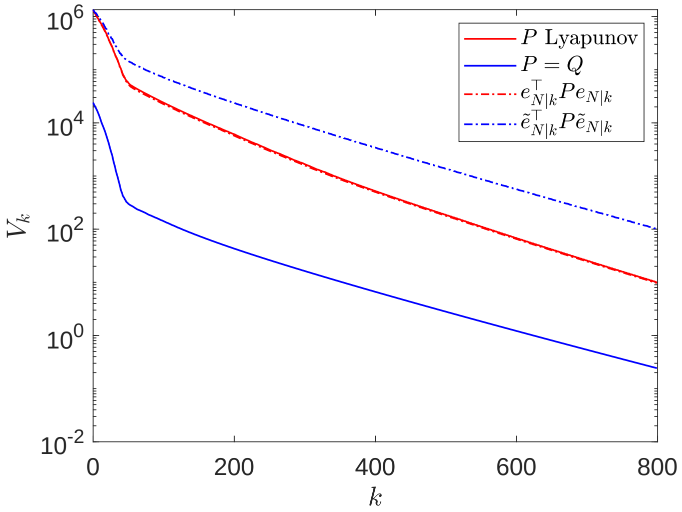


Fig. 28: Evolution of value function for different terminal weights for the single cell case.

Hence, by making state predictions with a larger sample time in between the individual instances, the prediction horizon in seconds effectively grows. The potential benefit of this technique is limited by the maximal sampling time for which the discrete time system is still stable. Using the forward Euler discretization technique, the maximal sample time T_s to obtain a stable discrete time system is known to be $T_s \leq -2/\underline{\lambda}$, in which $\underline{\lambda}$ is the minimal eigenvalue of the continuous time A matrix.

To investigate the effect of an increased prediction horizon, simulations are performed on a simplified case with a single transducer location. Since this removes the computationally heavy integer part from the optimization, state predictions are now performed using the full order model, i.e., $A^h = A$ and $B^h = B$. Figure 29 shows the evolution of the maximal temperature for the single-cell case in which the prediction horizon is varied over the values $N \in \{5, 10, 20, 60\}$. For the horizon $N = 20$, the blocking matrix

$$T_u = \begin{bmatrix} I_5 & 0 \\ 0 & I_3 \otimes \mathbf{1}_5 \end{bmatrix} \tag{98}$$

is used and for the $N = 60$ case the same blocking matrix is used with an increased sample time by a factor three. From the figure it becomes clear that increasing the prediction horizon indeed suppresses the violent temperature peaks. However, even with a prediction horizon of $N = 60$ samples, temperature peaks above six degrees are observed. In Table X, the average and maximal computational time for the simulation results in Figure 29 are stated. A prediction horizon of $N = 10$ already exceeds real-time feasible computation times. Therefore, it is concluded that limiting the effect of the terminal weight to an acceptable level by increasing the prediction horizon can not be done in a computationally feasible way using the discussed methods.

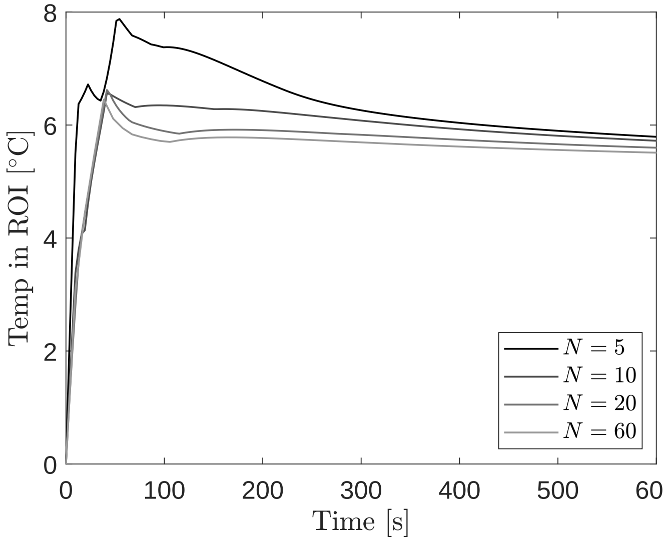


Fig. 29: Evolution of maximal temperatures for increasing prediction horizons.

TABLE X: Computational times in seconds for the simulations in Figure 29.

	$N = 5$	$N = 10$	$N = 20$	$N = 60$
mean	0.863	4.995	24.816	22.919
max	1.296	6.780	28.010	26.991

References

[1] R.D. Issels, L.H. Lindner, J. Verweij, et al. Effect of neoadjuvant chemotherapy plus regional hyperthermia on long-term outcomes among patients with localized high-risk soft tissue sarcoma. *JAMA Oncol*, 4:483–492, 2018.

[2] M. Mallory, E. Gogineni, G. C. Jones, et al. Therapeutic hyperthermia: The old, the new, and the upcoming. *Crit. Rev. Oncol. Hematol.*, 97:56–64, 2016.

[3] N. R. Datta, S. Gómez Ordóñez, U. S. Gaipl, et al. Local hyperthermia combined with radiotherapy and-/or chemotherapy: Recent advances and promises for the future. *Cancer Treat. Rev.*, 41(9):742–753, 2015.

[4] M. de Smet, N. M. Hijnen, S. Langereis, et al. Magnetic resonance guided high-intensity focused ultrasound mediated hyperthermia improves the intratumoral distribution of temperature-sensitive liposomal doxorubicin. *Investigative Radiology*, 48(6):395–405, 2013.

[5] R.M. Staruch, R. Chopra, and K. Hynynen. Localised drug release using MRI-controlled focused ultrasound hyperthermia. *Int. J. Hyperth.*, 27(2):156–171, 2011.

[6] E. Maloney and J.H. Hwang. Emerging HIFU applications in cancer therapy. *Int. J. Hyperth.*, 31(3):302–309, 2015.

[7] D.A. Deenen, E. Maljaars, L. Sebeke, et al. Offset-Free Model Predictive Control for Enhancing MR-HIFU Hyperthermia in Cancer Treatment. *IFAC Conf. Nonlinear Model Predict. Control*, 2018.

[8] J. K. Enholm, M. O. Köhler, B. Quesson, et al. Improved volumetric MR-HIFU ablation by robust binary feedback control. *IEEE Trans. Biomed. Eng.*, 57(1):103–113, 2010.

[9] A. Partanen, P.S. Yarmolenko, A. Viitala, et al. Mild hyperthermia with magnetic resonance-guided high-intensity focused ultrasound for applications in drug delivery. *Int. J. Hyperth.*, 28(4):320–336, 2012.

[10] C. Mougenot, B. Quesson, B. D. de Senneville, et al. Three-dimensional spatial and temporal temperature control with MR thermometry-guided focused ultrasound (MRgHIFU). *Magn. Reson. Med.*, 61(3):603–614, 2009.

[11] L. Sebeke, X. Luo, E. Heijman, et al. Prediction-based controller for MR-HIFU mediated hyperthermia. *Eur. Symp. Focus. Ultrasound Ther.*, page 23, 2017.

[12] J. B. Rawlings, D. Q. Mayne, and M. M. Diehl. *Model Predictive Control: Theory, Computation, and Design*. Nob Hill Publishing, LLC, Madison, 2nd edition, 2017.

[13] D. Q. Mayne. Model predictive control: Recent developments and future promise. *Automatica*, 50(12):2967–2986, 2014.

[14] M. Tillander, S. Hokland, J. Koskela, et al. High intensity focused ultrasound induced *in vivo* large volume hyperthermia under 3D MRI temperature control. *Med. Phys.*, 43(3):1539–1549, 2016.

[15] K. S. Mohan. *Mixed-integer optimization for control of large-area MR-HIFU hyperthermia treatment*. Msc graduation project, TU/e, Eindhoven, 2018.

[16] Profound web page. https://profoundmedical.com/sonalleve/. Accessed: 30-8-2019.

[17] H. Pennes. Analysis of Tissue and Arterial Blood Temperatures in the Resting Human Forearm. *J. Appl. Physiol.*, 1(2):93–122, 1948.

[18] P. Tatjewski. Advanced control and on-line process optimization in multilayer structures. 32:71–85, 2008.

[19] R. R. Negenborn, B. de Schutter, and J. Hellendoorn. Multi-Agent Model Predictive Control: A Survey. Technical report, Delft University of Technology, Delft, 2004.

[20] E. Luchini, A. Schirrer, and M. Kozek. A hierarchical MPC for multi-objective mixed-integer optimisation applied to redundant refrigeration circuits. *IFAC-PapersOnLine*, 50(1):9058–9064, 2017.

[21] G. P. Reddy. Hierarchical Model Predictive Control for Trajectory Generation and Tracking in Highly Automated Vehicles. *Msc Thesis TU Delft*, 2016.

[22] S. Liu, Y. Bin, Y. Li, et al. Hierarchical Model Predictive Control for the Fuel Cell Hybrid Electric Vehicles. *2018 37th Chinese Control Conf.*, pages 3599–3605, 2018.

[23] P. Kou, Y. Feng, D. Liang, et al. A model predictive control approach for matching uncertain wind generation with PEV charging demand in a microgrid. *Int. J. Electr. Power Energy Syst.*, pages 488–499, 2019.

[24] A. Abreu, R. Bourdais, and H. Guéguen. Inter-Layer Interactions in Hierarchical MPC for Building Energy Management Systems. *IFAC-PapersOnLine*, 50(1):12027–12032, 2017.

[25] J. Hasenauer, M. Löhning, M. Khammash, et al. Dynamical optimization using reduced order models: A method

to guarantee performance. *Journal of Process Control*, 22:1490–1501, 2012.

[26] M. Löhning, M. Reble, J. Hasenauer, et al. Model predictive control using reduced order models: Guaranteed stability for constrained linear systems. *Journal of Process Control*, 24:1647–1659, 2014.

[27] Gurobi solver. https://www.gurobi.com/products/gurobi-optimizer/. Accessed: 17-9-2019.

[28] P. Benner, M. Ohlberger, A. Cohen, et al. *Model Reduction and Approximation*. Society for Industrial and Applied Mathematics, Philadelphia, PA, 2017.

[29] S. Weiland. *Model reduction [lecture slides]*. University of Technology Eindhoven, 2017.

[30] I. P. Duff, S. Grundel, and P. Benner. New Gramians for Linear Switched Systems: Reachability, Observability, and Model Reduction, 2018.

[31] F. Aurenhammer and R. Klein. *Voronoi Diagrams [lecture notes]*. Fern Universität Hagen. source: http://www.pi6.fernuni-hagen.de/downloads/publ/tr198.pdf.

[32] R.F.Z. Masoud Hekmatfar. *Facility Location*. Physica-Verlag, 2009.

[33] Genetic algorithm matlab. https://nl.mathworks.com/help/gads/ga.html. Accessed: 09-9-2019.

[34] D. Limon, T. Alamo, D. Muñoz de la Peña, et al. MPC for tracking periodic reference signals. *IFAC Proceedings Volumes*, 45(17):490 – 495, 2012. 4th IFAC Conference on Nonlinear Model Predictive Control.

[35] M. Lazar. *Model Predictive Control [lecture slides]*. University of Technology Eindhoven, 2018.

[36] I. Kolmanovsky and E. G. Gilbert. Theory and computation of disturbance invariant sets for discrete-time linear systems. *Math. Probl. Eng.*, 4(4):317–367, 1998.

[37] D.Q. Mayne, M.M. Seron, and S.V. Raković. Robust control of constrained linear systems with bounded disturbances. *Automatica*, 41:219–224, 2005.

[38] J. A. Primbs and V. Nevistić. Feasibility and stability of constrained finite receding horizon control. *Automatica*, 36(7):965 – 971, 2000.

[39] R.W.M. Hendrikx. *Reduced-model-based temperature estimation and predictive control for MR-guided RF hyperthermia*. Msc graduation project, University of Technology Eindhoven, Eindhoven, 2017.

[40] R. Cagienard, P. Grieder, E.C. Kerrigan, et al. Move blocking strategies in receding horizon control. *IEEE CDC*, 43(2):2023 – 2028, 2004.

TU/e Technische Universiteit
Eindhoven
University of Technology

Declaration concerning the TU/e Code of Scientific Conduct for the Master's thesis

I have read the TU/e Code of Scientific Conduct[i].

I hereby declare that my Master's thesis has been carried out in accordance with the rules of the TU/e Code of Scientific Conduct

Date

03 - 09 - 2019

Name

J. v. Wordragen

ID-number

0855388

Signature

Submit the signed declaration to the student administration of your department.

[i] See: http://www.tue.nl/en/university/about-the-university/integrity/scientific-integrity/
The Netherlands Code of Conduct for Academic Practice of the VSNU can be found here also.
More information about scientific integrity is published on the websites of TU/e and VSNU