2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 25-29, 2020, Las Vegas, NV, USA (Virtual)

A Mixed-Integer Model Predictive Control Approach to Motion Cueing in Immersive Wheelchair Simulator

Le Anh Dao[1], Alessio Prini[2], Matteo Malosio[3], Angelo Davalli[4], Marco Sacco[5]

Abstract— **To allow wheelchair (electronic or manual) users to practice driving in different safe, repeatable and controlled scenarios, the use of simulator as a training tool is considered here. In this context, the capabilities of providing high fidelity motions for users of the simulator is highlighted as one of the most important aspects for the effectiveness of the tool. For this purpose, the motion cueing algorithm (MCA) is studied in our work to regenerate wheelchair motion cues by transforming motions of the real or simulated wheelchair into the simulator motion. The studied algorithm is developed based on Model Predictive Control (MPC) approach to efficiently optimize the motions of the platform. The overall problem is formulated using mixed-integer quadratic programming (MIQP) which involves not only the vestibular model, strict constraints of the platform but also the perception threshold in the optimization cost function. In the end, the performance assessment of the system using different control techniques is analyzed, showing the effectiveness of the proposed approach in the simulation environment.**

I. INTRODUCTION

Training in the laboratory is an effective and necessary step of enabling people after an accident to practice and familiarize themselves with the relevant skills before returning to work and social life in a controlled and safe environment, and learning to use the wheelchair is one of them. In fact, operating a wheelchair may not be an easy task since it requires a good combination of different skills, such as vision, strength, endurance, orientation and perception of the surrounding space [1]. For this reason, people forced on a wheelchair after an accident often experience difficulties when using it, which may lead to the practice of wrong behaviors, unsafety for oneself and others, and in the medium/long term could also lead to isolation and low quality of life [1]. Within this context, it is suggested to have a period of intense training in laboratory environment for wheelchair users employing an immersive wheelchair simulator as a possible training tool which provides the users with different sensory stimuli (visual, auditory, haptic and vestibular) in a controlled way; detailed discussion on the role of different sensory stimuli in the context can be found in [2]. Among them, the capability of emulating the physical behavior of the wheelchair by generating proper vestibular feedback to the patient is crucial and is the focus of this paper. Relating to this aspect, a so called Motion Cueing Algorithm (MCA) is being realized to move the patient in such a way that reproduces the motions and the physical interaction with the physical world considering the limitation of admissible motions of the platform.

Various approaches for MCA have been investigated widely. Among them, Model Predictive Control (MPC) stands out as a particularly useful methodology with capabilities of considering the information about the platform workspace, human vestibular system model as well as their constraints in the design of MCA. Thanks to the rapid development of computational power, the application of this technique is not limited in the process industry where slow dynamics is dominant but is also adopted by a fast dynamics system. In fact, in recent years the application of MPC technique in MCA has been flourishing such as in [3], [4]. However, similar works for wheelchair simulator are still poorly investigated, despite the fact that some typical movements of wheelchairs have a different characteristic compared to the other vehicles. In particular, the possibility to execute rotations with a very small radius of curvature about the vertical axis is one of them.

In these works, and also in many others of the same context, the perception threshold concept is widely discussed, however there is still a lack of a tool to explicitly involve these factors in the cost function terms of MPC designed for MCA which hinders the controller to exploit these flexibilities effectively. In fact, it is not necessary to track exactly the references of felt forces and felt rotation rates; staying inside an acceptance level around them (denoted as comfort band) is enough for having good motion cueing. A few works including Augusto et al. presented the perception thresholds of roll and pitch rates as hard constraints, which are then used to simulate lateral and longitudinal accelerations, respectively. This consideration aligns with the work in [5] which claims that the rate of tilt (i.e. roll and pitch rates) must be below the perception thresholds. However, Berger et al. discussed that the rate of tilt can be above the threshold without significantly affecting the driver's performance, provided the visual acceleration correlates with the effective body acceleration [6]. This controversy raises an idea that it would be convenient to let the system violate the comfort bands at a cost (no cost imposed if staying inside the band), in the interest of the entire system. Then, how hard the constraints of the comfort bands will be decided easily by modifying the weights associated to the stay-outside-band cost.

To this point, the innovative contribution is the exploitation

*This work was supported by the Istituto Nazionale per l'Assicurazione Contro Gli Infortuni sul Lavoro of Italy [PDT2/1 - Rientr@ project]

[1]Le Anh Dao, [2]Alessio Prini, [3]Matteo Malosio, [5]Marco Sacco are with Istituto di Sistemi e Tecnologie Industriali per il Manifatturiero Avanzato Consiglio Nazionale delle Ricerche, via Alfonso Corti 12, 20133 Milano, Italia {leanh.dao, alessio.prini, matteo.malosio, marco.sacco}@stiima.cnr.it

[4]Angelo Davalli is with Centro Protesi Inail, Via Rabuina 14, 40054 Vigorso di Budrio, Bologna, Italy a.davalli@inail.it

of the aforementioned thresholds directly in the control design as a way to improve the performance of MCA and allow the system to exploit the potential of staying outside the comfort bands. In this paper, the application of MPC approach for MCA is also adopted; in particular, to this aim, the approach described in [7] and [8] to employ the mixed logical dynamic framework is utilized to treat continuous and discrete variables (i.e., if-else conditions) introduced by the presentation of the perception thresholds; which then turns the formulated optimization problem into a Mixed-Integer Quadratic Programming (MIQP) one. To the best of our knowledge, no other work has made these considerations in the context of MCA before.

The rest of the paper is organized as follows. The system settings, brief control architecture and general view on MCA are introduced in Section II, then Section III discusses on the description and modelling of the vestibular model, MPC cost function terms and constraints. After that, the simulation results are presented and discussed in Section IV, before closing the paper with several concluding remarks reported in Section V.

II. SYSTEM DESCRIPTION

This section is devoted to the wheelchair simulator system description together with its control architecture; A brief discussion on MCA based on MPC approach is also presented in this section.

A. The system setting

The wheelchair simulator and its coordinate system are depicted in Fig. 1. It is composed of two main parts: a mechatronic platform and a virtual reality system (VR) equipped with a physics engine. The mechatronic platform is moved in space by a parallel Gough-Stewart platform. The users can interact with the VR and move within it in two ways: either through a joystick, for an electronic wheelchair, or through a pair of haptic wheels sensorized and actuated by a torque motor, in order to emulate the behavior of a manual wheelchair. The training content in the VR simulates movements of the users in both indoor and outdoor environments.

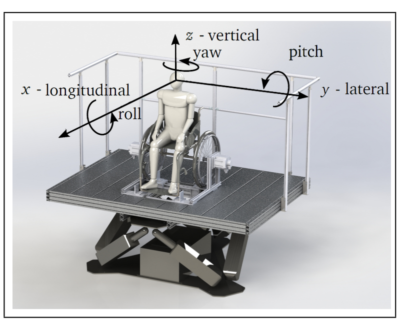

Fig. 1. The system settings and its coordinate system

Focusing only on the control architecture of the platform, a two-level control scheme is realized namely high (HL) and low-level (LL) controller. HL is responsible for posing setpoints calculation to render a realistic motion considering the PKM DOF physical limits. The LL control is responsible for the motion control of the mechatronic platform and wheels, and ensures that the system correctly reproduces the trajectory defined by the HL. However, in the scope of this paper, only the works related to HL is discussed. For the sake of simplicity, LL is assumed to work perfectly in this paper.

More in details about the development of MCA based on MPC approach (i.e., high level controller - HL) is shown in Fig. 2. In summary, the following four main steps in order are performed:

- HL receives as inputs the reference rotation rates and translational accelerations.
- The human's vestibular model transforms the received signals to the felt rotation rates and felt specific forces which then will be used as setpoints for the MPC block.
- MPC block generates position and rotation angles of the platform in such a way that helps the user experience the same feeling as in a real wheelchair; or technically speaking, to track correctly the MPC setpoints.
- Commanding the generated position and rotation angles to the LL.

Notice that through this document, the terms "the references" or "the references of felt forces and felt rotation rate" or "the MPC setpoints" imply the same thing. Interested readers may refer to [3], [4], [7] and references therein for the development of MPC.

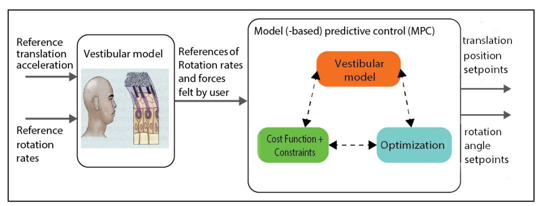

Fig. 2. Scheme of Motion Cueing Algorithm based on MPC approach

III. MPC SYSTEM MODEL AND OPTIMIZATION PROBLEM

This section starts with the detailed development of the vestibular model with and without the presence of perception thresholds. Then, relating to formulating the MPC optimization problem, this section focuses on the construction of different objectives as well as the constraints spanning over all the time instants inside the prediction(H_p) and control (H_c) horizons ($H_c \leq H_p$) . In the end, the optimization problem is formulated as a MIQP one which can be solved by using ILOG's CPLEX 12.8 (an efficient solver based on branch-and-cut algorithm).

A. Nomenclature

The main parameters and decision variables used in this section are described in Table I:

TABLE I

MAIN NOMENCLATURE

k	Discrete time step based on the sampling time
i	Mutually perpendicular coordinate axes (x, y or z)
$\omega_i, \hat{\omega}_i$	Rotation rate and felt rotation rate along i-axis
ϕ, θ, Φ	Platform: Roll angle (x-axis), pitch angle (y-axis), yaw angle (z-axis)
$\dot{\phi}, \dot{\theta}, \dot{\Phi}$	Roll rate, pitch rate and yaw rate
$\hat{\dot{\phi}}, \hat{\dot{\theta}}, \hat{\dot{\Phi}}$	Felt roll rate, felt pitch rate, felt yaw rate
$f_i, \hat{f}_i$	specific force and felt specific force, along i-axis
a_i	Translational acceleration along i-axis of the platform
g	Standard gravity
$\gamma_{S,i}$	Perception thresholds for rotation rate along i-axis
$\gamma_{O,i}$	Perception thresholds for specific force along i-axis
p_i, v_i	Platform position and velocity along i-axis
$r_{\dot{\phi}}, r_{\dot{\theta}}, r_{\dot{\Phi}}$	References for felt roll, pitch and yaw rates
$r_{f_x}, r_{f_y}, r_{f_z}$	References for felt specific forces

B. Vestibular system description and modeling

Located in the inner ear, the vestibular system is composed of semicircular canals and otoliths which are sensitive to angular accelerations, and to linear accelerations and gravity, i.e. specific force, respectively. The construction of the vestibular system models with and without the perception thresholds are presented in the following:

1) Vestibular model without perception thresholds: In this section, the description of the vestibular model is based on the work of [3]. For semicircular channel, the following transfer function from rotation rate to felt rotation rate is considered as

$$H_{S,i} = \frac{\hat{\omega}_i}{\omega_i} = \frac{T_{S,l,i} T_{S,a,i} s^2}{(T_{S,l,i}s+1)(T_{S,s,i}s+1)(T_{S,a,i}s+1)} \quad (1)$$

Similarly, the otoliths transfer function that relates the felt specific force to the specific force stimulus along i-axis is

$$H_{O,i} = \frac{\hat{f}_i}{f_i} = \frac{K(T_{O,a,i}s+1)}{(T_{O,l,i}s+1)(T_{O,s,i}s+1)} \quad (2)$$

Referring to [3], the following coefficients of the Semicircular and Otoliths channels model are chosen and presented in Table II.

TABLE II

THE COEFFICIENTS OF THE SEMICIRCULAR AND OTOLITHS CHANNELS

	$T_{S,l,i}$	$T_{S,s,i}$	$T_{S,a,i}$	$T_{O,l,i}$	$T_{O,s,i}$	$T_{O,a,i}$	K
x	6.1	0.1	30	5.33	0.66	13.2	0.4
y	5.3	0.1	30	5.33	0.66	13.2	0.4
z	10.2	0.1	30	5.33	0.66	13.2	0.4

To avoid pushing the platform to its limitation due to the sustained components of the acceleration, tilt coordination is used to simulate the specific forces by tilting the motion platform. Based on [3], which considers the use of small angles approximation, the relation between specific force and acceleration is presented as

$$\begin{bmatrix} f_x \\ f_y \\ f_z \end{bmatrix} = \begin{bmatrix} a_x + gsin(\theta) \\ a_y - gcos(\theta)sin(\Phi) \\ a_z - gcos(\theta)cos(\Phi) \end{bmatrix} \approx \begin{bmatrix} a_x + g\theta \\ a_y - g\Phi \\ a_z - g \end{bmatrix} \quad (3)$$

To impose properly the constraints in the MPC, it is convenient to present the positions and velocities of the platform in the output of system model. This can be done by using simple "integral system" of inputs $\begin{bmatrix} a_x, & a_y, & a_z \end{bmatrix}^T$ (i.e., translational accelerations of the platform along x, y and z axis) to generate outputs $\begin{bmatrix} p_x, & v_x, & p_y, & v_y, & p_z, & v_z \end{bmatrix}^T$(i.e., position and velocities of platform).

Due to space limitation, the transformation from these formulations to the presentation in the form of state-space model will be left out; interested readers are referred to [3] for more detailed description. In the end, the final form of the model is presented as

$$\sum_V = \{A_V, B_V, C_V, D_V\} \quad (4)$$

where input and output vectors are:
$u_V(k) = \begin{bmatrix} a_x(k) & a_y(k) & a_z(k) & \dot{\phi}(k) & \dot{\theta}(k) & \dot{\Phi}(k) \end{bmatrix}^T \in \mathbb{R}^6$
$y_V(k+1) = \begin{bmatrix} y_{Tr}(k+1)^T & y_C(k+1)^T \end{bmatrix}^T \in \mathbb{R}^{15}$
$(y_{Tr} = \begin{bmatrix} \hat{\dot{\phi}} & \hat{\dot{\theta}} & \hat{\dot{\Phi}} & \hat{f}_x & \hat{f}_y & \hat{f}_z \end{bmatrix}^T)$
$(y_C = \begin{bmatrix} \phi & \theta & \Phi & p_x & v_x & p_y & v_y & p_z & v_z \end{bmatrix}^T)$

2) Vestibular model with perception thresholds: Depending on the measuring methods, the experimental setup and the experimental conduction, various thresholds can be found in the literature; in this paper, we will select a set of the perception threshold from the works with the presence of visualization in the simulator setting. The values of the perceived threshold are listed in Table III:

TABLE III

THE CONSIDERED PERCEPTION THRESHOLDS

	x	y	z
$\gamma_{S,i}[deg/s]$	6.3 (in [9])	3.6 (in [5])	9 (in [10])
$\gamma_{O,i}[m/s^2]$	0.17 (in [5])	0.17 (in [5])	0.28 (in [5])

Let
$r(k) = \begin{bmatrix} r_{\dot{\phi}}(k) & r_{\dot{\theta}}(k) & r_{\dot{\Phi}}(k) & r_{f_x}(k) & r_{f_y}(k) & r_{f_z}(k) \end{bmatrix}^T$
be the vector of reference trajectories for corresponding outputs $y_{Tr}(k)$
$y_{Tr}(k) = \begin{bmatrix} \hat{\dot{\phi}}(k) & \hat{\dot{\theta}}(k) & \hat{\dot{\Phi}}(k) & \hat{f}_x(k) & \hat{f}_y(k) & \hat{f}_z(k) \end{bmatrix}^T$
and
$\gamma(k) = \begin{bmatrix} \gamma_{S,x} & \gamma_{S,y} & \gamma_{S,z} & \gamma_{O,x} & \gamma_{O,y} & \gamma_{O,z} \end{bmatrix}^T$ the constant vector of the perception thresholds independently to the time instant k; $r_n(k), \gamma_n(k), y_n(k)$ are the n-th elements of

$r(k), \gamma(k), y(k)$ respectively. Then if we denote $C_n^e(k)$ as error (or effective error) in tracking the reference $r_n(k)$ of the related term, the following statements hold:
$C_n^e =$

$$\begin{cases} y_n(k) - r_n(k) & \text{if } y_n(k) - r_n(k) \geq \gamma_n(k) \\ 0 & \text{if } \gamma_n(k) > |y_n(k) - r_n(k)| \\ -(y_n(k) - r_n(k)) & \text{if } y_n(k) - r_n(k) \leq -\gamma_n(k) \end{cases} \quad (5)$$

The presentation can be rewritten as:

$$C_n^k = \delta_n(k)\,(y_n(k) - r_n(k)) \quad (6)$$

with

$$\begin{cases} \delta_n(k) = 1 & \text{if } y_n^T(k) - r_n(k) \geq \gamma_n(k) \\ \delta_n(k) = 0 & \text{if } \gamma_n(k) > |y_n^T(k) - r_n(k)| \\ \delta_n(k) = -1 & \text{if } y_n^T(k) - r_n(k) \leq -\gamma_n(k) \end{cases} \quad (7)$$

In order to formulate our optimization problem into an MIQP one, here we employ the same technique discussed in [7], [8] to convert a logical statement of a given form into linear mixed-integer constraints. As an example, given a function $h(k)$, it states that the statement $h(k) >= 0 \iff \delta_{std} = 1$ is true if and only if

$$\begin{cases} -m_{std}\,\delta std \leq h(k) - m_{std} \\ -(M_{std} + \epsilon)\,\delta_{std} \leq -h(k) - \epsilon \end{cases} \quad (8)$$

In [7], [8], it also states that the product between logical and continuous variables $y_{std} = \delta_{std}\,h(k)$ is equivalent to a set of inequalities as follows:

$$\begin{cases} y_{std} \leq M_{std}\,\delta_{std} \\ y_{std} \geq m_{std}\,\delta_{std} \\ y_{std} \leq h(k) - m_{std}\,(1 - \delta_{std}) \\ y_{std} \geq h(k) - M_{std}\,(1 - \delta_{std}) \end{cases} \quad (9)$$

where h is a function upper and lower bounded by M_{std} and m_{std}, respectively, δ_{std} is a logical variable, y_{std} is a real variable and ϵ is a small tolerance to transform a constraint of the form $s < 0$ into $s \leq 0$ which is suitable for the functionality of the common MIQP solver to solve. In order to bring our problem into the standard formulation described above, the integer variable $\delta_n(k)$ can be decomposed into the sum of two binaries variables $\delta_{1,n}(k)$ and $\delta_{2,n}(k)$. Then (7) becomes:

$$\begin{cases} \delta_{1,n}(k) = 1 \iff y_n(k) - r_n(k) \geq -\gamma_n(k) \\ \delta_{2,n}(k) = 1 \iff y_n(k) - r_n(k) \geq \gamma_n(k) \\ \delta_n(k) = \delta_{1,n}(k) + \delta_{2,n}(k) - 1 \\ \delta_{1,n}, \delta_{2,n} = \{0, 1\} \end{cases} \quad (10)$$

Denoting $z_{1,n}(k) = \delta_{1,n}(k)\ y_{Tr,n}(k)$ and $z_{2,n}(k) = \delta_{2,n}(k)\ y_{Tr,n}(k)$, the statement is equivalent to a set of constraints (according to the standard statement) relating to $z_{q,n}(k)$ and $\delta_{q,n}(k)$ with $q = \{1, 2\}$

$$\begin{cases} -(m_n - r_n(k) + \gamma_n)\,\delta_{q,n}(k) \leq y_{Tr,n}(k) - m_n \\ -(M_n - r_n(k) + \gamma_n + \epsilon)\,\delta_{q,n}(k) \leq \\ -y_{Tr,n}(k) + r_n(k) - \gamma_n - \epsilon \\ z_{q,n}(k) \leq M_n\,\delta q, n(k) \\ z_{q,n}(k) \geq m_n\,\delta q, n(k) \\ z_{q,n}(k) \leq y_{Tr,n}(k) - m_n\,(1 - \delta_{q,n}(k)) \\ z_{q,n}(k) \geq y_{Tr,n}(k) - M_n\,(1 - \delta_{q,n}(k)) \end{cases} \quad (11)$$

where m_n and M_n are the minimum and maximum values of $y_{Tr,n}(k)$. The effective error $C_n^e(k)$ in (6) is rewritten as

$$C_n^e(k) = (\delta_{1,n}(k) + \delta_{2,n}(k) - 1)(y_{Tr,n}(k) - r_n(k)) = z_{1,n}(k) + z_{2,n}(k) - y_{Tr,n}(k) - \delta_{1,n}(k)\,r_n(k) - \delta_{2,n}(k)\,r_n(k) + r_n(k) \quad (12)$$

Repeating the above procedure for each of output of $y_{Tr,n}$, the compact form of these inequalities is:

$$E_{1,n}^B(k) u_n^B(k) \leq E_{2,n}^B y_{Tr,n}(k) + E_{3,n}^B(k) \quad (13)$$

where $u_n^B = \begin{bmatrix} z_{1,n}(k) & z_{2,n}(k) & \delta_{1,n}(k) & \delta_{2,n}(k) \end{bmatrix}^T \in \mathbb{R}^2 \times \{0,1\}^2$, and $E_{1,n}^B(k)$, $E_{2,n}^B$, $E_{3,n}^B(k)$ are suitable matrices derived from the corresponding inequalities in (11); the matrices $E_{1,n}^B(k)$ and $E_{3,n}^B(k)$ are generally time-varying due to the time varying references $r_n(k)$.
By combining these constraints for all the outputs, the final form can be written as

$$E_1^B(k)\,u^B(k) \leq E_2^B y_{Tr}(k) + E_3^B(k) \quad (14)$$

where
$u^B(k) = \begin{bmatrix} u_1^B(k)^T & u_2^B(k)^T & ... & u_6^B(k)^T \end{bmatrix}^T$;
$E_1^B(k) = \text{diag}\{E_{q,n}^B(k)\}$, $q = 1, 2, ..., 6$;
$E_2^B = \begin{bmatrix} (E_{2,1}^B)^T & (E_{2,2}^B)^T & ... & (E_{2,6}^B)^T \end{bmatrix}^T$;
$E_3^B(k) = \begin{bmatrix} (E_{3,1}^B)^T & (E_{3,2}^B)^T & ... & (E_{3,6}^B)^T \end{bmatrix}^T$;
and diag{} refers to the block diagonal matrix.
Finally, the new system model to be used in MPC formulation is constructed by adding $u^B(k)$ (i.e., set of auxiliary variables $z_{1,n}(k)$, $z_{2,n}(k)$ and logical variables $\delta_{1,n}(k)$, $\delta_{2,n}(k)$) into the original system model in (4) as follows:

$$\sum_{\bar{V}} = \{A_{\bar{V}}, B_{\bar{V}}, C_{\bar{V}}, D_{\bar{V}}\} \quad (15)$$

where
$A_{\bar{V}} = A_V$, $B_{\bar{V}} = \begin{bmatrix} B_V & | & 0 \end{bmatrix}$,
$C_{\bar{V}} = C_V$, $D_{\bar{V}} = \begin{bmatrix} D_V & | & 0 \end{bmatrix}$,
$u_{\bar{V}}(k) = \begin{bmatrix} u_V(k)^T & u^B(k+1)^T \end{bmatrix}^T \in \mathbb{R}^{6+12} \times \{0,1\}^{12}$,
$y_{\bar{V}}(k+1) = y_V(k+1) \in \mathbb{R}^{15}$

C. MPC cost function terms

The MPC cost function includes several terms, which address different objectives.

- w_T, J_T are the weights and cost terms that account for tracking the desired perception trajectory of $\hat{\dot{\phi}}$, $\hat{\dot{\theta}}$, $\hat{\dot{\Phi}}$, $\hat{f}_x$ $\hat{f}_y$ and $\hat{f}_z$ (i.e., y_{Tr} in (4)).
- w_R, J_R are the weights and cost terms to move the platform back to the neutral position, which are equivalent to tracking zero positions and velocities for $\phi, \theta, \Phi, p_x, v_x, p_y, v_y, p_z, v_z$, (i.e., y_C in (4)).
- w_S, J_S are the weights and cost terms for the smoothness of decision variables which is preferred for the operation of the platform's actuators. These terms are applied for only physical decision variables: a_x, a_y, a_z, $\dot{\phi}$, $\dot{\theta}$, $\dot{\Phi}$ (i.e., u_V in (4))

In the end, these considered terms are constructed spanning the whole prediction horizon and should be expressed, as a function of the decision variables ($D\,u_{\bar{V}}$ as in (15)) in the form of:

$$argmin_{u_{\bar{V}}} J = u_{\bar{V}}^T\, A\, u_{\bar{V}} + B\, u_{\bar{V}} + C, \text{ subject to: } D\, u_{\bar{V}} \leq b \tag{16}$$

Detailed description of the mentioned terms is explained as follows:

1) Reference tracking performances: This cost function term focuses on minimizing the effective error in tracking problems of felt rotations and felt specific forces. The formulation of the effective error are presented in (6), (12). Notice that $C_n^e(k)$ is a mixed-integer linear function which is composed of both continuous, integer decision variables therein. The minimization of the error between the sensation signals in the platform and in the wheelchair is the utmost goal of this work. The cost function term together with their weights are formulated as

$$J_T = \sum_{j=1}^{H_p} \sum_{n=1}^{6} w_{T,n}\, C_n^e(k+j|k) \tag{17}$$

Notice that, at time instant k, the actual references at time $k+j$ ($j = 1, 2, ..., H_p$) are not available but need to be predicted. The prediction of these references are denoted as $\hat{r}_n(k+j|k)$ which, in the perfect prediction case, equal to the actual ones (i.e., $\hat{r}_n(k+j|k) = r_n(k+j)$).

2) Return-to-zero cost function term: As staying in the neutral position gives the platform a possibility to move freely in any direction, it is suggested to bring them to this position whenever possible. Tracking of these variables is performed to ensure the washout of the motion platform as usually mentioned in the related field articles.

$$J_R = \sum_{j=1}^{H_p} \sum_{n=1}^{9} w_{R,n}\, (y_{C,n}(k+j|k) - 0)^2 \tag{18}$$

3) Smoothness of decision variables: Abrupt changes in the decision variables may damage the actuators and adversely affect the platform motion. A soft constraints is here included in the cost function to smooth the profile of decision variables. This consideration is applied only for the decision variables u_V in (4) which directly associate with the physical terms in our system, while the extended ones in u^B due to the introduction of the perception thresholds are not necessarily smoothed.

$$J_S = \sum_{j=0}^{H_c-1} \sum_{n=1}^{6} w_{S,n}\, (\Delta u_{V,n}(k+j|k))^2 \tag{19}$$

where $\Delta u_{V,n}(k+j|k) = u_{V,n}(k+j|k) - u_{V,n}(k+j-1|k)$

D. MPC constraints

To ensure the computed commands produce the feasible movements for the platform, the following constraints are considered in the optimization problem:

1) Constraints on decision variables and their variations: Except for the auxiliary decision variable, the rest of the control inputs are bounded, according to the catalogue, as presented in Table. IV.

TABLE IV
CONSTRAINTS ON DECISION VARIABLES AND THEIR VARIATIONS

	a_x	a_y	a_z	$\dot{\phi}$	$\dot{\theta}$	$\dot{\Phi}$
u	±5.9	±5.9	-4.9: 6.9	±30	±30	±40
Δu	±3	±3	±3	±15	±15	±20
Unit	m/s^2	m/s^2	m/s^2	deg/s	deg/s	deg/s

2) Constraints for platform movements: In this category, the constraints related to Euler angles and translation movement are considered as in Table. V.

TABLE V
CONSTRAINTS ON PLATFORM MOVEMENTS

$\phi[deg]$	$\theta[deg]$	$\Phi[deg]$	$p_x[m]$	$v_x[m/s]$
±19.6	±19	±23.3	±0.24	±0.51
$p_y[m]$	$v_y[m/s]$	$p_z[m]$	$v_z[m/s]$	
±0.23	±0.51	±0.19	±0.30	

3) Constraints from the introduction of auxiliary variables: Besides the constraints mentioned above, due to the presentation of auxiliary decision variables, the designed MPC optimization problem also includes a set of additive constraints presented in (14) for time instant k and their spanning at other time instants inside the prediction horizon; the presentation of these constraints are as follows:

$$E_1^B(k)\, u^B(k+j|k) \leq E_2^B y_{Tr}(k+j|k) + E_3^B(k+j|k)$$
$$\forall j = 1, 2, ...H_p \tag{20}$$

From the cost functions discussed in (18), (19), (20) and the constraints in this sub-section, it is clear to see the possibility to reformulate them into a generic Quadratic Programming problem as in (16) since the all outputs, the constraints and the cost function terms are either quadratic or linear functions of the decision variables or their variations.

IV. SIMULATION RESULTS

This section is dedicated to investigate the performances of designed MPC for MCA in simulation environment. The prediction horizon and control horizon are set to 15 steps which account for 3-second ahead prediction considering the sampling time of high-level control is 0.2 seconds. A simulation period of 15 seconds is adopted. The weights are selected following the one in [3] together with the trial-and-error approach considering the fact that tracking the references is the utmost goal of the work. The following weights are adopted in this paper:

TABLE VI
WEIGHTS ON TRACKING THE REFERENCES, RETURN-TO-ZERO AND SMOOTHING THE DECISION VARIABLES TERMS

	$\hat{\dot{\phi}}$	$\hat{\dot{\theta}}$	$\hat{\dot{\Phi}}$	$\hat{f}_x$	$\hat{f}_y$	$\hat{f}_z$
Weights	$w_{T,1}$	$w_{T,2}$	$w_{T,3}$	$w_{T,4}$	$w_{T,5}$	$w_{T,6}$
Value	20	20	20	20	20	80
	ϕ	θ	Φ	p_x	v_x	p_y
Weights	$w_{R,1}$	$w_{R,2}$	$w_{R,3}$	$w_{R,4}$	$w_{R,5}$	$w_{R,6}$
Value	1	1	1	10	15	10
	v_y	p_z	v_z			
Weights	$w_{R,7}$	$w_{R,8}$	$w_{R,9}$			
Value	15	5	9			
	a_x	a_y	a_z	$\dot{\phi}$	$\dot{\theta}$	$\dot{\Phi}$
Weights	$w_{S,1}$	$w_{S,2}$	$w_{S,3}$	$w_{S,4}$	$w_{S,5}$	$w_{S,6}$
Value	10	10	10	0.5	0.5	0.1

Two simulations are realized as follows:

- Simulation-S1. Performance in tracking the desired perception trajectories;
- Simulation-S2. Impact of the references' prediction error in the tracking performance;

Two cases of the actual wheelchair motion profiles are adopted (named C1, C2). While C1 represents the case that the user accelerates then decelerates the wheelchair along the longitudinal axis in a straight, flat road; C2 simulates the case when the user executes an 80-degree rotation in 3 seconds around the vertical axis then accelerates the wheelchair along the longitudinal axis. Their trajectories are shown in Fig. 3.

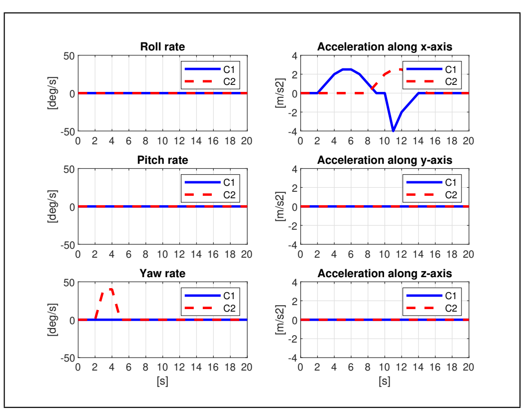

Fig. 3. The considered actual wheelchair motion profiles

To evaluate the simulation results, two measured criteria are considered. The first criterion is MAE [%] (Mean Absolute Error Percentage). As suggested by name, the mean value of distances between the felt motions with respect to the comfort bands is firstly calculated. The computed value is then divided by the maximum value of the considered term to obtain the percentage value. The maximum values of a_x, a_y, a_z, $\dot{\phi}$, $\dot{\theta}$, $\dot{\Phi}$ are 4 $[m/s^2]$, 4 $[m/s^2]$, 4 $[m/s^2]$, 30 $[deg/s]$, 30 $[deg/s]$ and 45 $[deg/s]$, respectively. The second criterion is RC [%] (Rate of Confliction) which measures the percentage of time when the felt motions are outside of the comfort bands. On the other hand, Root Mean Square Error percentage (RMSE[%]), which is based on the standard criterion RMSE, is used to measure the reference prediction errors in S2.

A. *Reference tracking performances in C1 and C2 (Simulation - S1)*

The objective of Simulation-S1 is to test the performances of our above designed control system (denoted as HL1) in C1, C2. The same test is performed using a benchmark controller based also on the MPC approach (denoted as HL2). In all the simulations in Simulation-S1, it is assumed that the prediction of the references is perfect. To simplify the process of developing the benchmark controller, HL2 is designed the same as HL1 with some modifications as follows:

- HL2 set very high weights for the felt tilt rate tracking (i.e., $w_{T,1}$ and $w_{T,2}$) with respect to all the other weights so that the comfort bands for these two terms are numerically equivalent to the hard constraints, while the weights related to tracking the references of the felt yaw rate and the felt specific forces are set to zero so that no comfort bands for these factors are considered.
- Adding into the cost function of HL2 the standard quadratic cost function terms which penalize the tracking errors of the felt yaw rate and the felt specific forces with respect to their references. Hence, HL2 tries to track exactly the references of these terms instead of only staying inside the comfort bands.

A certain outcome of the first modification is that the quality for tracking tilt rate references of HL2 is always at least equal or better than HL1 as HL2 treats the tilt rate as the hard constraints so that the felt tilt rate always stays inside the comfort bands, but it is not the case for HL1. Indeed, observing Fig. 4, Fig. 5 and Table. VII, HL2 provides slightly better tracking performance for roll rate in C1 (MAE and RC of 1.5 [%] and 13.3[%] with HL1 and MAE and RC of 0 [%] and 0[%] with HL2), but by sacrificing the performance of this term, HL1 outperforms HL2 significantly for felt force tracking performance (MAE and RC of 0.8 [%] and 8[%] with HL1 and MAE and RC of 4.9 [%] and 46.7[%] with HL2). On the other hand, HL1 outperforms HL2 in all aspects in C2.

B. *Impact of prediction error of references in the tracking performance (Simulation - S2)*

This simulation aims to test the impact of the prediction error in the overall performance of our system, with only HL1 being used in this case. For the sake of simplicity, the

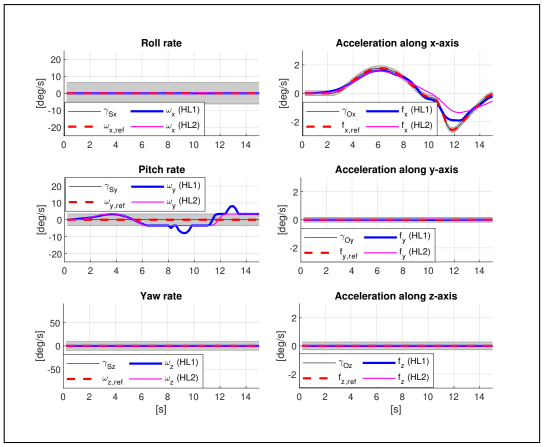

Fig. 4. The tracking performance of perceived simulation signals - C1 case

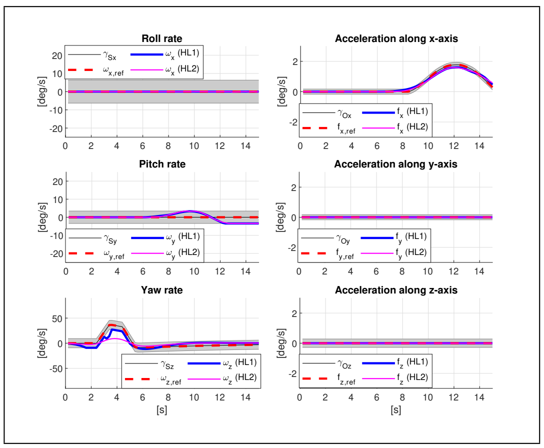

Fig. 5. The tracking performance of perceived simulation signals - C2 case

TABLE VII

TRACKING PERFORMANCE IN C1 AND C2

		MAE [%]			RC [%]		
		Roll	Yaw	Force	Roll	Yaw	Force
C1	HL1	1.5	0	0.8	13.3	0	8
	HL2	0	0	4.9	0.0	0.0	46.7
C2	HL1	0	0.7	0	0	2.7	0
	HL2	0	4.3	0.1	0	14.7	13.3

development of a reference profile prediction is not included in this paper. However, in the simulation environment, some levels of prediction errors are emulated by artificially adding a white Gaussian noise to the actual reference profiles. The SNR (i.e., signal-noise ratio) of the additive noise is set equal to 7, 15 in order to model, on average, the 20%, 8%, respectively, based on RMSE [%].

In the end, we can observe from the obtained results reported in Table VIII and Table VII, that the quality of references prediction has a high impact on performances of tracking felt yaw rate and specific force. However, the same behavior is not realized for the felt roll rate with small differences in all cases. On average, the lower prediction error of the references leads to a higher performance for our system.

TABLE VIII

IMPACT OF THE REFERENCE PREDICTION ERRORS

		MAE [%]			RC [%]		
		Roll	Yaw	Force	Roll	Yaw	Force
C1	SNR = 7	2.1	0	1.0	17.3	0	22.7
	SNR = 15	1.8	0	0.9	16.0	0	13.3
C2	SNR = 7	0.1	3.4	0.2	1.3	29.3	14.7
	SNR = 15	0.1	1.3	0.0	1.1	10.7	4

V. CONCLUSIONS

This paper proposes an innovative approach on modelling and optimization for Motion Cueing Algorithm of an immersive wheelchair simulator. The vestibular model, the perception thresholds, the platform's constraints and various tracking problems are merged together into an optimization problem of MPC. In particular, an efficient tool to model and formulate the mixed continuous and discrete behaviors of the perception thresholds is introduced using MIQP, opening a way to fully exploit the potential of this factor. Simulation results validated the efficiency of the designed approach; the studied context is, but not strictly limited to, the wheelchair simulator. In the end, investigation related to the quality of the references prediction is studied, showing a strong impact of the prediction quality on the tracking reference performances of our system. Hence, a clear direction for future research is towards the development of the references prediction. Moreover, the study of adapting the control system to the driving style or the level of impairment of patients would be another interesting topic for future work together with the study of fast MPC techniques to make our controller easily develop in practice.

REFERENCES

[1] J. Trotter, “Wheelchair Users Problems With Community Living,” Canadian Family Physician, vol. 31, p. 1493, 1985.

[2] S. Arlatia, D. Spoladore, S. Mottura, A. Zangiacomi, G. Ferrignoa, R. Sacchetti and M. Sacco, “Analysis for the design of a novel integrated framework for the return to work of wheelchair users,” WORK: A Journal of Prevention, Assessment & Rehabilitation, vol. 61, no. 4, 2018.

[3] B. D. C. Augusto and R. J. L. Loureiro, ”Motion Cueing in the Chalmers Driving Simulator: A Model Predictive Control Approach,” Chalmers University of Technology, Göteborg, Sweden, 2009

[4] A. Beghi, M. Bruschetta and F. Maran, ”A real time implementation of MPC based Motion Cueing strategy for driving simulators”, 51st IEEE Conference on Decision and Control, 2012, Maui, Hawaii, USA

[5] L. Reid and M. A. Nahon, ”Flight simulation motion-base drive algorithms: part 1,” Developing and testing equations. Tech. rep. University of Toronto, 1985

[6] D.R.Berger, J. Schulte-Pelkum, and H. H. Bülthoff, “Simulating believable forward accelerations on a Stewart motion platform”, Max Planck Institute for Biological Cybernetics Technical Report, 2007

[7] A. Parisio, E. Rikos and L. Glielmo, ”A Model Predictive Control Approach to Microgrid Operation Optimization”, IEEE Transactions on Control System Technology, Vol. 22, NO. 5, September 2014

[8] L. A. Dao, A. Dehghani-Pilehvarani, A. Markou and L. Ferrarini, ”A hierarchical distributed predictive control approach for microgrids energy management”, Elsevier, vol. 48, 2019.

[9] A. Nesti, C. Masone, M. Barnett-Cowan, P. R. Giorano, H. H. Bülthoff, and P. Pretto, “Roll rate thresholds and perceived realism in driving simulation”. In: Proceedings of the Driving Simulation Conference, Driving Simulation Association (Sept. 2012). Paris, France.

[10] A. Gundry, “Thresholds to roll motion in a flight simulator,” Journal of Aircraft 14.7 (1977). DOI: 10.2514/3.58832.