

A novel Motion Cueing Algorithm based on real-time optimization and periodic invariant sets

Martin Soyer, Sorin Olaru, Zhou Fang

▶ **To cite this version:**

Martin Soyer, Sorin Olaru, Zhou Fang. A novel Motion Cueing Algorithm based on real-time optimization and periodic invariant sets. 4th IEEE Conference on Control Technology and Applications (CCTA 2020), Aug 2020, Montréal (virtual), Canada. pp.19-24, 10.1109/CCTA41146.2020.9206343 . hal-02899834

HAL Id: hal-02899834

https://centralesupelec.hal.science/hal-02899834

Submitted on 14 Jan 2021

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

A novel Motion Cueing Algorithm based on real-time optimization and periodic invariant sets

Martin Soyer[1,2], Sorin Olaru[2], Zhou Fang[1]

***Abstract*— This paper deals with control design of Motion Cueing Algorithms for driving simulation. The development of driving-assistance systems and the gradual move towards autonomous driving led automobile manufacturers to focus on high performance driving simulation in order to validate novel functionalities and driving confort before production. Driving simulators are currently constrained environments because of the workspace size and the actuators resistance. As part of the software operating the platform, a control block has to manage the position of a cabin by guaranteeing realistic acceleration feelings to a driver. In this purpose, the controller is usually design within Model Predictive Control (MPC) framework. However large prediction horizons and constrained tracking problems implies heavy computational burden. In this paper, a novel MPC-based motion cueing algorithm is proposed considering periodic invariant sets as a key concept to decrease the complexity of the real-time optimization.**

I. INTRODUCTION

This structure of such a virtual driving simulator is illustrated in Fig 1 on Renault's platform, ULTIMATE which is a Symmetric Large Displacement Systems with a less than 1500 kg payload [1]. The control of the underlying mechanical system represents the ultimate goal of the present control application while the performance is being evaluated in terms of the driver stimuli.

In the cabin, a driver handles, with practically all the real world degrees of freedom, a virtual vehicle. This virtual vehicle is receiving drivers' maneuver signals and simulates its movements within a software block that delivers the acceleration profile that need to be rendered to the driver. This acceleration profile represents a reference signal for the high level control system which in turn is in charge of providing references to the lower level control blocks:

- position references to rails control systems
- tilt angles references for the hexapod control.

This cascaded control process is depicted in a schematic way in Fig 2.

This paper will focus on the control block that provides positions in this constrained workspace (compared to a real situation in a outdoor environment) in order to maximize the acceleration rendering of the driver and consequently to avoid motion sickness [2]. This system is known in the literature

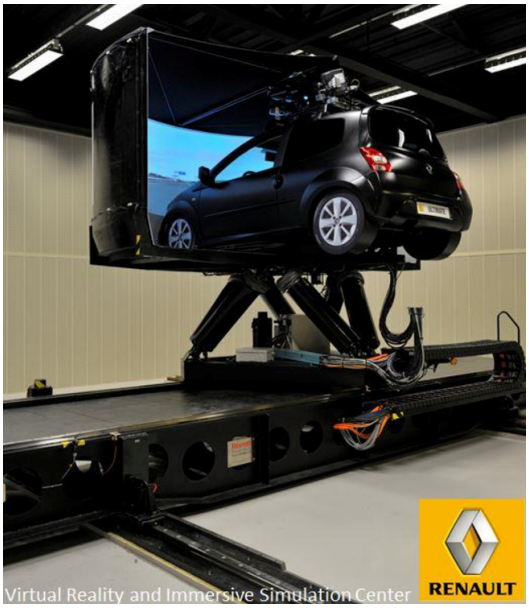


Fig. 1. Renault's ULTIMATE Driving Simulator

as Motion Cueing Algorithm (MCA) and it represents a research interest for industrial and academics for decades. Historically, the first motion cueing algorithms were filter-based and are still used in industry due to their Real-Time implementation attractiveness, the main principle is to isolate high frequency movements by reproducing their accelerations and to compensate the gravity feelings using the tilt with low frequency accelerations. The main drawback of a such approach is the relative conservatism due to the constraints avoidance design philosophy. In other words those kind of algorithms fail to exploit the whole workspace. Moreover, filter properties leads to backlash effect [2] and cause motion sickness when braking. Since the last decades, the literature is focused on optimization-based algorithms and particularly on Model Predictive Control (MPC) which was used in MCA for

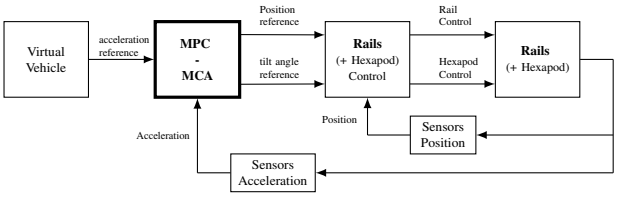


Fig. 2. Block Diagram of MCA process (considered elements in this paper are in bold)

[1,2] Renault Technocentre, AD Simulation & VR center, Guyancourt,Université Paris-Saclay, CNRS, CentraleSupélec, Laboratoire des signaux et systèmes, 91190, Gif-sur-Yvette, France martin.soyer@renault.com

[1] Renault Technocentre, AD Simulation & VR center, Guyancourt, France zhou.fang@renault.com

[2] Université Paris-Saclay, CNRS, CentraleSupélec, Laboratoire des signaux et systèmes, 91190, Gif-sur-Yvette, France sorin.olaru@centralesupelec.fr

the first time in 2004 [3]. MPC-MCA provides a framework that takes the bounds of the workspace and the limitations of actuators into account in the computation of control actions [4], [5], [6], [7], [8].

Recursive feasibility guarantee is a main issue for the class of real-time optimization-based control [9], [10]. Our design builds on an alleviated formulation of MPC problem from a computational point of view and imposes, at each iteration, that the next state remains into the Maximal Controllable Set [11] which is the largest controlled invariant admissible sets with respect to states and inputs constraints. The main benefit of this formulation besides complexity is the consideration of a large domain of attraction to avoid conservativeness. This property can be practically exploited for systems of small dimensions, but the constraints of the Maximal Controllable set can easily make the optimization problem difficult to handle starting from 3 dimensions to higher ones [12]. This issue needs to be considered for the system controlled by MCA, especially if the vestibular system is considered in the model [13]. The main theoretical contribution of this work on MCA is the simplification of the maximal controllable set in the constraints imposed for the real-time optimization by means of a periodic controlled invariant [14] set which provides a much simpler domain of attraction by maintaining recursive feasibility of the whole process.

First we focus on the system modelling and describe the simplifying assumptions, two models will be considered in the following, in 2 and 3 dimensions. Subsequently, we discuss on performance criteria for controller design that prevent from motion sickness and ensure the best acceleration feelings. Finally, An optimization-based MCA will be proposed using the periodic invariance properties in order to lighten the optimization problem by keeping the recursive feasibility.

Notation : A *Polytope* P in the *H-Representation* is a set $P = \{x \in \mathbb{R}^n \ | Fx \leq g\}$ where $F \in \mathbb{R}^{q \times n}$ and $g \in \mathbb{R}^q$. A *Polyhedron* denotes a bounded polytope. A *Polyhedron* P in *V-Representation* is a set $P = \{\sum_i^n \lambda_i v_i / \forall i \ \lambda_i \geq 0\}$ where $\{v_1, ..., v_n\}$ are vertex of the polyhedron. $\|.\|_Q^2$ will denotes the squared weighted 2-norm such that $\forall x \in \mathbb{R}^n \ \|x\|_Q^2 = x^T Q x$ where Q is a positive definite $n \times n$ matrix.

II. Description of the system and simplified assumptions

This paper considers two models of the MCA in order to represent the main dynamical block with control input which aims to provide the closest acceleration feelings to the driver, as close as possible to the real driving situation.

In the following we will concentrate on the linear movements of rails as its action is directly related to the perceived acceleration while hexapod tends to redistribute weight acceleration in between axes (tilt coordination).

In both models considered next we employ simplifying assumptions that rails respond ideally with no inertia[1], thus, the acceleration state computed by MCA is the same with the one perceived by the driver.

The role of MCA is to preserve as faithfully as possible the unconstrained acceleration of a vehicle within the constrained environment. The system has the virtual vehicle acceleration as input and provides the constrained position to the rail controller, thus globally it can be assimilated a chain of integrators subjected to constraints on position (workspace boundaries), on speed, acceleration and jerk (actuators limitations). Parameters of the ULTIMATE Driving Simulator are summarized in Table I.

TABLE I
PARAMETERS OF ULTIMATE

Parameter	Notation	Limit
Position	p_k	2.6 m
Speed	v_k	3 m/s
Acceleration	a_k	5 m/s^2
Jerk	j_k	600 m/s^3
Sampling Time	T_s	8ms

A. State-Space model with acceleration as a state component

The first model used in the numerical study is a triple integrator with a 3-dimensional state $x_k = \begin{bmatrix} p_k & v_k & a_k \end{bmatrix}^T$ and $u_k = j_k$:

$$\begin{cases} x_{k+1} &= \begin{bmatrix} 1 & T_s & \frac{1}{2}T_s^2 \\ 0 & 1 & T_s \\ 0 & 0 & 1 \end{bmatrix} x_k + B = \begin{bmatrix} \frac{1}{6}T_s^3 \\ \frac{1}{2}T_s^2 \\ T_s \end{bmatrix} u_k \\ y_k &= \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} x_k \end{cases}$$

subject to constraints :

$$\begin{bmatrix} -2.6 \\ -3 \\ -5 \end{bmatrix} \leq x_k \leq \begin{bmatrix} 2.6 \\ 3 \\ 5 \end{bmatrix} \qquad -600 \leq u_k \leq 600 \tag{1}$$

In this model, jerk is weakly constrained whereas state-space is highly compelling. This constrained LTI model is generally accepted by motion cueing community and used in modern MPC-based MCA [4].

B. State-Space model with acceleration as a control action

The second model is a constrained double integrator with a 2-dimensional state $x_k = \begin{bmatrix} p_k & v_k \end{bmatrix}^T$ and $u_k = a_k$:

$$x_{k+1} = Ax_k + Bu_k$$

subject to constraints:

$$\begin{bmatrix} -2.6 \\ -3 \end{bmatrix} \leq x_k \leq \begin{bmatrix} 2.6 \\ 3 \end{bmatrix} \qquad -5 \leq u_k \leq 5 \tag{2}$$

where

$$A = \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} \text{ and } B = \begin{bmatrix} \frac{1}{2}T_s^2 \\ T_s \end{bmatrix}$$

The advantage of this model is to have a 2 dimensions state-space, however the restrictive constraints are moved toward the input signal.

[1]The model being discrete then one can consider the simplifications to be related to the intersample behavior which is more relevant for the given perception objective.

III. MCA Performance criteria for control design

In this paper we aim to design a tracking MPC-based MCA for different linear models with constraints, therefore the criteria to be optimized at each iteration has an influence on tracking performance. Despite the relative classical design framework, the challenges come from the hard constraints and their implications, from the fact that the reference trajectory may violate the boundaries of the feasible domain during a turn or emergency braking and the real time compilation demands (less than 8ms as sampling rate). We recall the main role of MCA which is the restitution of this acceleration and particularly keeping the global shape of the acceleration profile to guarantee comfort from a feelings perspective which is actually non-standard feature due the fact that future trajectories are uncertain.

However, tracking an acceleration in a constrained environment may imply a drift toward the boundaries of the feasible domain in the state space and then let a significant part of the controllable set unused. A solution to avoid the drift is to regulate the cabin to the center of the workspace with an unperceived change in the acceleration profile. Similar, a centering of the control signal toward the center of feasible interval for actuators will allow an improved controllability as response to virtual vehicle movements (Figure 2).

As the jerk is weakly constrained in the model (1) it may evolve rapidly and damage actuators. As a consequence, a weight for the control action has to be considered in the MPC cost function. Motion sickness may appear when the acceleration feeling is delayed compared to the expectation of the driver from visual stimuli [1]. Practically this delay is partly due to computation of control actions, and consequently even for LTI model (1) to the solving of optimization problems. The choice of a quadratic criteria can be advantageous in view of complexity of (convex) programming.

Tuning the MPC prediction horizon is a sensitive issue, technically, during the operation of the simulator with a driver, acceleration trajectory is not known in advance as long as it depends on the effective driver's behavior (the weight on the pedal, braking, turn, reflexes,...). In our control problem, the tracking trajectory is updated in real-time upon a software simulation thus leading to a possible abrupt change of the parameters particularly $\mathbf{y^{ref}}$ in the MPC cost function $J(\mathbf{x}, \mathbf{u}, \mathbf{y^{ref}})$. Moreover, manufacturers can face often in practice one of these situations :

- A predefined scenario in order to validate a specific functionality, the driver is guided by instructions of a supervisor, signaling and traffic laws, then its trajectory can be planned in advance.
- An autonomous driving scenario in order to study the driver's behavior, in that case the trajectory of the vehicle is well known in advance.

Therefore, acceleration data can be estimated by computer simulations of an autonomous vehicle or by an average trajectory of real simulations with other non-optimal MCA such as filter-based ones.

Criteria for model (1) will be written as follows by reducing the information to the essential components :

$$J_k^1 = \sum_{i=1}^{N_h} \|y_{k+i+1}^{ref} - Cx_{k+i+1}\|_{q_y}^2 + \|x_{k+i}\|_{Q_x}^2 + \|u_{k+i}\|_R^2 \tag{3}$$

While the first and the third components are classical MPC ingredients, the second aims to introduce a weighting to the return of the body to the center of the symmetric feasible domain.

Criteria for model (2) can be written in a more compact way by excluding prediction on the jerk :

$$J_k^2 = \sum_{i=1}^{N_h} \|u_k^{ref} - u_k\|_R^2 + \|x_{k+i}\|_{Q_x}^2 \tag{4}$$

Here the tracking objective takes the form of tracking the input signal but relying it to state contribution.

With these basic elements for modelling performance criteria, the design of MPC-based MCAs for the two models (1) and (2) can be formulated according to a classical tracking MPC approach.

IV. A receding optimization-based solution for tracking

In this section we will provide tools to design a MPC-based MCA with periodic invariance properties in order to accelerate computation of control action by enforcing the recursive feasibility. We use for the presentation of the main definitions a linear discrete-time dynamical model :

$$\left\{ \begin{array}{rcl} x_{k+1} & = & Ax_k + Bu_k \\ y_k & = & Cx_k \end{array} \right. \tag{5}$$

with the origin as fixed point and subject to states and control constraints :

$$\left\{ \begin{array}{l} x_k \in \mathbb{X},\ \forall k \in \mathbb{N} \\ u_k \in \mathbb{U},\ \forall k \in \mathbb{N} \end{array} \right. \tag{6}$$

where $\mathbb{X} \subset \mathbb{R}^n$ and $\mathbb{U} \subset \mathbb{R}^m$ are compact convex set containing the origin in their respective interior. The system is assumed to be controllable and the output state y_k measurable. These structures with respect to the MCA model aims to show the extended framework of the proposed design approach.

A. Preliminary notions

In MPC theory, recursive feasibility guarantee can be obtained considering invariant sets with respect to states and inputs constraints. In order to introduce periodic invariance, we first recall different notions of invariance and controlled invariance.

Definition IV.1 (Positive invariance)**.** *A set Ω is said to be positively invariant for the autonomous system $x_{k+1} = f(x_k)$ if $x_{k+1} \in \Omega$, $\forall x_k \in \Omega$.*

Definition IV.2 (Maximal Controllable set)**.** *Given a proper controlled invariant set $\Omega \subset \mathbb{X}$ we define the* Maximal controllable set to Ω*, denoted $\mathcal{C}(\Omega)$, to be the collection of initial states $x_0 \in \mathbb{X}$ for which there exists a finite admissible control sequence that brings x_0 to Ω.*

Definition IV.3 (N-Steps Controllable set to Ω)**.** *Given a controlled invariant set $\Omega \subset \mathbb{X}$ and $N \in \mathbb{N}^*$, a set C_N is said to be N-Step controllable set to Ω if Ω can be reached in at most N steps.*

For models (1) and (2) with linear structures, given a stabilizing feedback law $u_k = -Kx_k$, the iterative procedure in [15] can be used to construct the maximal positively invariant admissible polyhedral set Ω_∞ which is, in fact, the largest positively invariant set that respect constraints and the linear control structure.

Then the maximal controllable set to Ω_∞ can be infinitely determined due to the class of polyhedral sets but for LTI systems finite approximations can be constructed using for example the reachable set procedure in [11]. A lightened formulation of MPC with guarantees of recursive feasibility can be derived from the use of a N-Steps Controllable set C_N as approximation for the maximal controllable set :

$$\begin{aligned} \underset{(u_k,\dots,u_{k+N_h-1})}{\text{minimize}} \quad & J_k^i(x_{k+1},\dots,x_{k+p},u_k,\dots,u_{k+p-1}) \\ \text{subject to} \quad & \\ x_{k+1+j} = Ax_{k+j} + Bu_{k+j}, \quad & \forall j \in [\![1, N_h]\!] \\ u_{k+j} \in \mathbb{U}, \quad & \forall j \in [\![0, N_h - 1]\!] \\ x_{k+1} \in C_N & \end{aligned} \tag{7}$$

B. Periodic invariance

The main methodological contribution is the using of periodic invariance property in order to simplify the maximal controllable set C_N by conserving an invariance property to guarantee the recursive feasibility of MPC.

Definition IV.4 (Periodic invariance [16])**.** *Given $p \in \mathbb{N}^*$. A set $C \in \mathbb{X}$ containing the origin is said to be p-invariant if for all initial state $x_k \in C$ there exists a control sequence $(u_k, ..., u_{k+p-1}) \in \mathbb{U}^p$ such that $x_{k+p} \in C$ and $(x_{k+1}, ..., x_{k+p-1}) \in \mathbb{X}$.*

In other words a state leaving a p-invariant set returns in at most p steps. It can be shown that any full dimensions set containing the origin contained in C_N is controlled periodic invariant with respect to $(\mathbb{X}, \mathbb{U})$. In the following, a controlled p-invariant inner approximation of C_N will be denoted $\widetilde{C}_N$.

In order to exemplify the concept let us consider two sets $\widetilde{C}_N^1$ and $\widetilde{C}_N^2$ for models (1) and (2) that are respectively p_1-invariant and p_2-invariant with $p_1 = 15$ and $p_2 = 19$, these sets are depicted on Fig.3.

Considering controlled periodic invariant sets, several finite-time optimization problems denoted by $\mathcal{P}^i(p, x_k)$ can be rewritten for the same model i but with a different optimization cost and optimization structure ($i \in \{1, 2\}$):

$$\begin{aligned} \underset{(u_k,\dots,u_{k+p-1})}{\text{minimize}} \quad & J_k^i(x_{k+1},\dots,x_{k+p},u_k,\dots,u_{k+p-1}) \\ \text{subject to} \quad & \\ x_{k+1+j} = Ax_{k+j} + Bu_{k+j}, \quad & \forall j \in [\![1, p]\!] \\ x_{k+j} \in \mathbb{X}, \quad & \forall j \in [\![1, p-1]\!] \\ u_{k+j} \in \mathbb{U}, \quad & \forall j \in [\![0, p-1]\!] \\ x_{k+p} \in \widetilde{C}_N & \end{aligned} \tag{8}$$

These optimization problems will replace the classic MPC formulation in order to decrease based on switching the complexity both off-line and on-line.

Practically, complexity of $\mathbb{X}$ is low (a square for a 2D state space and a cube for a 3D state space) and $\widetilde{C}_N$ one's is of the same order, this formulation is consequently less complex than (7).

Consideration of simpler periodic invariant sets implies a loss of recursive feasibility from a classical MPC framework point of view, to compensate for this loss, Algorithm 1 is described next:

- If the current state is in $\widetilde{C}_N$, then the optimization problem (8) is solved for a horizon p.
- Else the state left $\widetilde{C}_N$, and the problem is solved with a decreasing horizon until the state returns in $\widetilde{C}_N$.

Algorithm 1 Tracking with p-invariant MPC

Initialization ($x_0 \in \mathcal{B} \subset \mathbb{X}$)
$k = 1$
$j = 1$ % index counting the steps of a periodic window
repeat for each k
  if $x_k \in \widetilde{C}_N$ **then**
    j=p
    Solve $\mathcal{P}^i(p, x_k) \rightarrow (u_k^1, ..., u_{k+p-1}^1))^*$
  else
    j=j-1
    Solve $\mathcal{P}^i(j, x_k) \rightarrow (u_k^1, ..., u_{k+j-1}^1))^*$
  end if
  $x_{k+1} = Ax_k + Bu_k^1$
  $k = k + 1$
until $k = N_{simu}$

Proposition IV.1. *The iterative procedure in Algorithm 1 is recursively feasible if the initial state is in $\widetilde{C}_N$.*

Proof. Assume the procedure is feasible at step k.

- If $x_k \in \widetilde{C}_N$ which is controlled p-invariant then it exists $(u_k, ..., u_{k+p-1}) \in \mathbb{U}^p$ such that $(x_{k+1}, ..., x_{k+p-1}) \in \mathbb{X}$ and $x_{k+p} \in \widetilde{C}_N$.
- If $x_k \notin \widetilde{C}_N$, there exists $j \in [\![1, p-1]\!]$ such that $x_{k-j} \in \widetilde{C}_N$. Consequently, there exists a control sequence $(u_{k-j}, ..., u_{k-j+p-1}) \in \mathbb{U}^p$ such that $(x_{k-j+1}, ..., x_k, ..., x_{k-j+p-1}) \in \mathbb{X}$ and $x_{k-j+p} \in \widetilde{C}_N$. As a consequence of these two properties there always exists a control sequence that leads states toward $\widetilde{C}_N$.

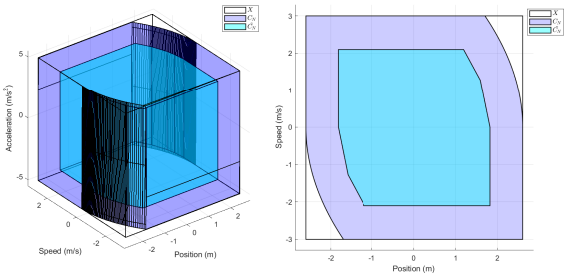


Fig. 3. Left: 3D State space representation for model (1), C_N and $\widetilde{C_N}$, Right : 2D State space representation for model (2), C_N and $\widetilde{C_N}$

The receding horizon optimization is in charge of selecting the horizon p at each time step in order to enforce this property.

□

V. Simulation Results

The study case elaborated in this simulation considers the acceleration reference associated to a signal perceived during a slalom. Here, results from simulations of MPC-MCA from (7) are compared with results from application of Algorithm 1 to models (1) and (2). The periodic invariant set is obtained by contracting an inner approximation of the maximal controlable set with a periodic index p_1.The chosen prediction horizon is similar to invariance periods : $N_h = p_1$.

Weightings in the criteria are $q_y = 100$, $Q_x = diag(100, 1, 50)$ and $R = 1$ for model (1), $Q_x = diag(100, 1)$ and $R = 50$ for model (2).

Acceleration is rendered with timely reaction and respecting the profile although the constraints are activated. In particular the shape of the reference profile is respected (fig. 4) and it is to highlight that the periodic invariant MPC applied to model (2) reaches saturations in acceleration an is consequently closer to the reference. However responses of periodic invariant MCAs are affected by switching of the cost function which affects the acceleration. All three algorithms are conservative as they don't use the whole workspace in position (fig. 4) altough the 2D system manages the workspace in a better way, positioning the virtual car closer to the center of symmetry.

In fig. 5 the jerk is presented as a control action for classical MPC (7) and Algorithm 1 applied to model 1. We can observe a more aggressive behavior of control action when state trajectory leaves $\widetilde{C}_N^1$ as the system tends to return within this latter set.

From a computational point of view, Table II shows the theoretical decrease of the number of constraints with respect to optimization problems to solve during procedures.

Fig. 8 represents computation time of optimization problem at each iteration as a function of time and confirms the decrease of computational burden. In this situation, comparison between models (1) and (2) is not obvious because p_2 is slightly higher than p_1 on one hand and the state trajectory remains in $\widetilde{C}_N$ in the 2-dimensions case so the optimization

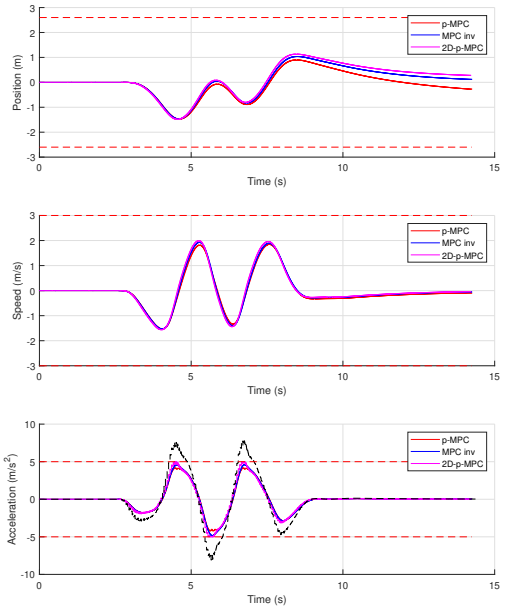


Fig. 4. Top : Positions in function of time, Middle : Speeds in function of time, Bottom : Accelerations in function of time (reference signal dashed)

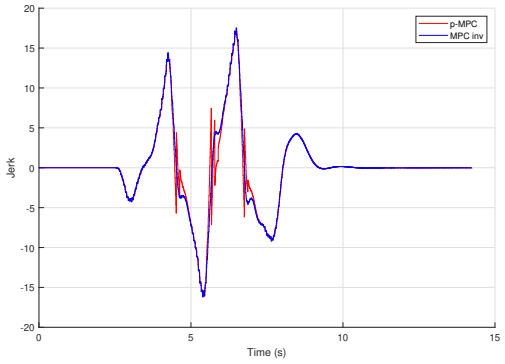


Fig. 5. Jerk in function of time for model (1)

is always performed in the worst case as it is shown on Figures 6 and 7.

VI. Conclusion

This paper presented a novel MCA that alleviates computational costs by guaranteeing recursive feasibility properties and by having similar acceleration rendering performance. The low complexity periodic invariant sets are representing the theoretical notions that replaces the costly maximal control invariant sets of the terminal constraints and long predictions horizons. Future works should focus on application of such strategies on more complex models including computational delays and nonlinearities in order to control hexapod and improve the acceleration feelings of drivers.

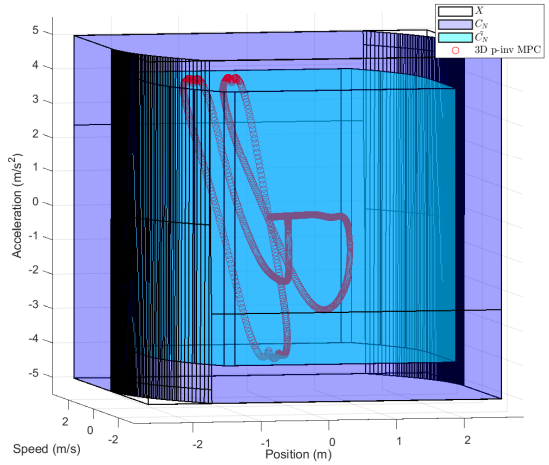


Fig. 6. 3D State space trajectory for p-invariant algorithm for model (1)

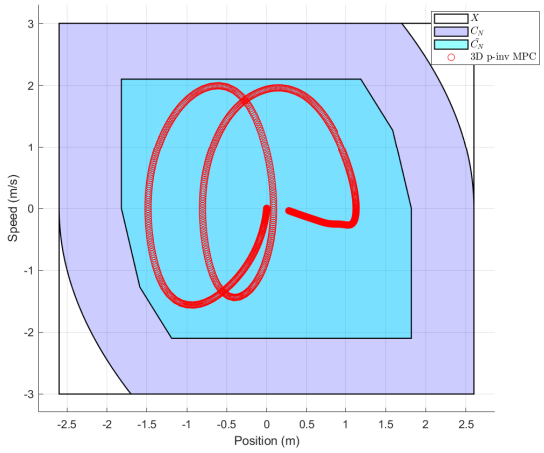


Fig. 7. 2D State space trajectory for p-invariant algorithm for model (2)

TABLE II

COMPLEXITY OF OPTIMIZATION PROBLEM

MPC Formulation	Number of constraints
Model (1) Invariant MPC	489
Model (1) p-invariant MPC	170 (worst case)
Model (2) p-invariant MPC	152 (worst case)

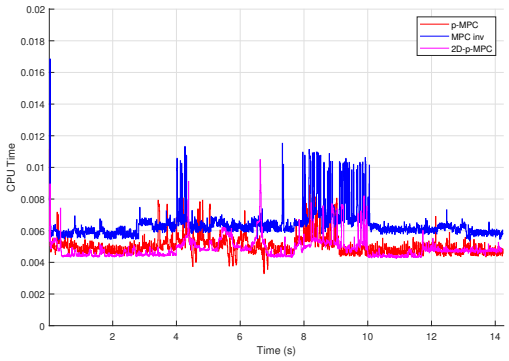


Fig. 8. Computation time

REFERENCES

[1] D. L. Fisher, M. Rizzo, J. Caird, and J. D. Lee, *Handbook of driving simulation for engineering, medicine, and psychology*. CRC Press, 2011.

[2] Z. Fang and A. Kemeny, "Review and prospects of Renault's MPC based motion cueing algorithm for driving simulator," in *Proceedings of the Driving Simulation Conference Europe, Paris, France*, 2014, pp. 4–5.

[3] M. Dagdelen, G. Reymond, A. Kemeny, M. Bordier, and N. Maïki, "MPC based motion cueing algorithm: Development and application to the ULTIMATE driving simulator," in *Conférence simulation de conduite*, 2004, pp. 221–233.

[4] Z. Fang and A. Kemeny, "Motion cueing algorithms for a real-time automobile driving simulator," in *Driving Simulation Conference, Paris, France, Sept*, 2012, pp. 6–7.

[5] ——, "Explicit MPC motion cueing algorithm for real-time driving simulator," in *Proceedings of The 7th International Power Electronics and Motion Control Conference*, vol. 2. IEEE, 2012, pp. 874–878.

[6] ——, "An efficient model predictive control-based motion cueing algorithm for the driving simulator," *Simulation*, vol. 92, no. 11, pp. 1025–1033, 2016.

[7] Z. Fang, M. Tsushima, E. Kitahara, N. Machida, D. Wautier, and A. Kemeny, "Motion cueing algorithm for high performance driving simulator using yaw table," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 15 965 – 15 970, 2017, 20th IFAC World Congress.

[8] S. Munir, M. Hovd, Z. Fang, S. Olaru, and A. Kemeny, "Complexity reduction in motion cueing algorithm for the ultimate driving simulator," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 10 729–10 734, 2017.

[9] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

[10] J. Löfberg, "Oops! I cannot do it again: Testing for recursive feasibility in MPC," *Automatica*, vol. 48, no. 3, pp. 550–555, 2012.

[11] H. Nguyen, P.-O. Gutman, S. Olaru, M. Hovd, and F. Colledani, "Improved vertex control for time-varying and uncertain linear discrete-time systems with control and state constraints," in *Proceedings of the 2011 American Control Conference*. IEEE, 2011, pp. 4386–4391.

[12] C. E. T. Dorea and J. Hennet, "(A, B)-invariant polyhedral sets of linear discrete-time systems," *Journal of Optimization Theory and Applications*, vol. 103, no. 3, pp. 521–542, 1999.

[13] J. A. Houck, R. J. Telban, and F. M. Cardullo, "Motion cueing algorithm development: Human-centered linear and nonlinear approaches," 2005.

[14] M. Soyer, S. Olaru, K. Ampountolas, S. Scialanga, and Z. Fang, "Periodic set invariance as a tool for constrained reference tracking," in *21st IFAC World Congress*. IFAC, 2020.

[15] E. G. Gilbert and I. Kolmanovsky, "Discrete-time reference governors for systems with state and control constraints and disturbance inputs," in *Proceedings of 1995 34th IEEE Conference on Decision and Control*, vol. 2. IEEE, 1995, pp. 1189–1194.

[16] Y. I. Lee and B. Kouvaritakis, "Constrained robust model predictive control based on periodic invariance," *Automatica*, vol. 42, no. 12, pp. 2175–2181, 2006.