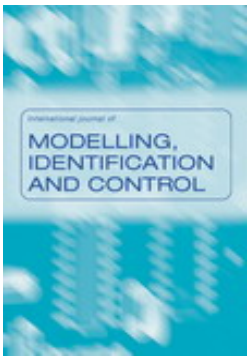
INDERSCIENCE PUBLISHERS
Linking academia, business and industry through research



International Journal of Modelling, Identification and Control

ISSN online: 1746-6180 - ISSN print: 1746-6172
https://www.inderscience.com/ijmic

6DoF Stewart motion platform control using switchable model predictive control

Jiangwei Zhao, Zhengjia Xu, Dongsu Wu, Yingrui Cao, Jinpeng Xie

DOI: 10.1504/IJMIC.2025.10074856

Article History:

Received:	11 November 2024
Last revised:	20 August 2025
Accepted:	15 October 2025
Published online:	24 December 2025

6DoF Stewart motion platform control using switchable model predictive control

Jiangwei Zhao

School of Information Engineering,
Pingdingshan University,
Pingdingshan, Henan, China
Email: 6507@pdsu.edu.cn

Zhengjia Xu*

Department of Engineering,
Durham University,
Durham, UK
Email: xu_zhengjia1@163.com
*Corresponding author

Dongsu Wu

College of Civil Aviation,
Nanjing University of Aeronautics and Astronautics,
Nanjing, Jiangsu, China
Email: tissle@nuaa.edu.cn

Yingrui Cao and Jinpeng Xie

School of Information Engineering,
Pingdingshan University,
Pingdingshan, Henan, China
Email: 1091939954@qq.com
Email: maomao200703@163.com

Abstract: Due to its high rigidity, manoeuvrability, and strength-to-weight ratio, the 6-DoF Stewart platform is widely used in flight simulators for replicating pilot motion cues. However, upset prevention and recovery training (UPRT) involves rapid angular changes that exceed motor tolerance, and classical washout filter (CWF)-based motion cueing algorithms (MCAs) struggle to meet high-accuracy and fast-response requirements. This study develops a model predictive control (MPC)-based MCA to manage nonlinearities and workspace limitations in hexapod simulators. To address control uncertainties from constraint extraction (COTC), a switchable MPC (S-MPC) architecture is proposed for adaptive response. Simulations show that within the operating envelope, MPC-MCA achieves high tracking accuracy, while outside it, the S-MPC mechanism provides optimal switching control. Under horizontal stall UPRT conditions, the proposed S-MPC-MCA improves motion tracking performance by 42.34% and 65.30% over MPC-MCA and CWF-MCA, respectively, based on the average absolute scale (AAS) criterion.

Keywords: model adaptive architecture; motion cueing algorithm; model predictive control; Stewart motion platform control.

Reference to this paper should be made as follows: Zhao, J., Xu, Z., Wu, D., Cao, Y. and Xie, J. (2025) '6DoF Stewart motion platform control using switchable model predictive control', *Int. J. Modelling, Identification and Control*, Vol. 46, No. 2, pp.100–110 .

Biographical notes: Jiangwei Zhao is a Lecturer at the School of Aircraft Control and Information Engineering at Pingdingshan University. He received his PhD in Engineering from Nanjing University of Aeronautics and Astronautics. His research interests include modelling and control of aircraft, modelling of parallel robots, and motion control of flight simulators.

Zhengjia Xu is a Lecturer in Mechatronics Engineering at Aston University, specialising in autonomous systems, resilient GNSS, network analysis and other navigation aspects. He earned his PhD in Aerospace from Cranfield University and previously served as PDRA in Durham University, and Research Fellow in PNT within its Centre for Autonomous and Cyber-Physical Systems. His prior industry roles at Drone Defence Services Ltd and ASH Wireless involved passive radar, SDR, and embedded IoT system development.

Dongsu Wu is currently an Associate Professor and Master's Supervisor at the School of Civil Aviation, Nanjing University of Aeronautics and Astronautics. He obtained his Doctoral degree from Nanjing University of Aeronautics and Astronautics, served as a visiting scholar sponsored by the China Scholarship Council (CSC), and studied at the University of Houston in the USA. His research areas mainly include flight simulation and flight safety, virtual reality human-computer interaction technology, robotics and nonlinear control, and computer vision technology.

Yingrui Cao is a current student majoring in Aircraft Control and Information Engineering at the School of Information Engineering, Pingdingshan University. His research areas include robot modelling and control, aircraft modelling and control, flight dynamics simulation technology, and intelligent aircraft cluster collaborative control.

Jinpeng Xie is a Lecturer in the School of Information Engineering at Pingdingshan University. He obtained his Master's degree from Zhengzhou University. His research areas include communication and network technology, integrated circuits and microelectronics, artificial intelligence and information processing, as well as intelligent terminals and new displays.

1 Introduction

A flight simulator is a sophisticated device engineered to replicate the flight conditions of an aircraft, providing an authentic emulation of the aircraft's behaviour and interaction with its environment. Given its superior mechanical performance including high stiffness, exceptional manoeuvrability and a high strength-to-weight ratio, the six-degree-of-freedom (DoF) Stewart platform has found widespread application in flight-simulator systems (see Figure 1).

In the early development of motion cueing algorithms (MCAs), Schmidt and Conrad (1970) introduced a framework based on the classical washout filter (CWF) for Stewart platform-based aircraft motion simulation, incorporating translational, rotational, and tilt-coordination channels. Subsequent research applied particle swarm optimisation (PSO) techniques to enhance the performance of the CWF-based MCA, thereby achieving more efficient utilisation of the available workspace (Asadi et al., 2016b). More recently, adaptive MCA frameworks have been proposed, in which the parameters of the CWF-based MCA are continuously tuned to maximise fidelity with respect to reference motion signals (Parrish et al., 1975).

Furthermore, fuzzy logic control theory has been applied to the MCA domain to formulate the adaptive MCA problem, with the human vestibular system incorporated to better replicate realistic motion perception (Zadeh, 1988; Hwang et al., 2009; Muyeen and Al-Durra, 2013; Asadi et al., 2015a, 2019, 2022; Baghaee et al., 2018; Qazani et al., 2020c, 2021c). In parallel, the linear quadratic regulator (LQR) has emerged as another widely adopted approach, integrating the human vestibular model into the optimal MCA formulation to reproduce motion perception more effectively (Asadi et al., 2016a; Qazani et al., 2020a, 2020d, 2021a; Cleij et al., 2020; Kang et al., 2022), owing to its capability of systematically weighting control and performance objectives. Nevertheless, these advantages remain confined to the parameter space, and the absence of adaptiveness in systematic parameter values can introduce substantial errors due to modelling uncertainties.

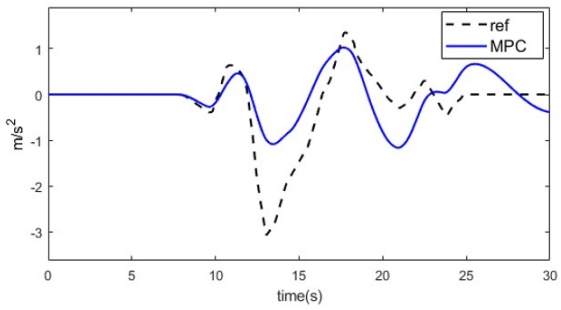
To further enhance MCA performance in terms of minimising tracking error, increasing flexibility, and improving fidelity in the systematic parameter domain, model predictive control (MPC)-based formulations have been investigated (Dagdelen et al., 2009; Asadi et al., 2015b, 2023; Khusro et al., 2020; van der Ploeg et al., 2020; Qazani et al., 2021b). These approaches integrate human vestibular system dynamics and flight simulation manoeuvring models into the MCA framework, thereby establishing optimal control solutions that explicitly account for both performance objectives and system constraints. A comparative study on practical platforms confirmed that MPC-based MCAs deliver significant performance improvements over conventional CWF-based methods, particularly under operational constraints (Cleij et al., 2019). Furthermore, to address leg-length constraints in Stewart platforms, a linear MPC-MCA with consideration of terminal conditions (COTC) was proposed to mitigate physical limitations (Qazani et al., 2020a).

To address the aforementioned challenges, this study proposes a novel switchable model predictive control (S-MPC)-based MCA, which demonstrates promising performance in simulating complex flight scenarios, as illustrated in Figure 2(b). Within the simulator's operational boundaries, the S-MPC-MCA with COTC is capable of delivering more realistic motion responses with faster system dynamics. Beyond these boundaries, the S-MPC-MCA without COTC achieves improved tracking accuracy by approximating nonlinear dynamics through linearisation. In addition, an innovative switching mechanism is introduced to ensure smooth transitions between modes, thereby enhancing overall system stability. The performance of the proposed S-MPC framework is evaluated by examining terminal states (Kwon and Pearson, 1977, 1978), implementing an infinite output prediction horizon (Kouvaritakis and Cannon, 2016), and optimising the terminal weighting matrix (TWM) (Kwon et al., 1983; Kwon and Byun, 1989).

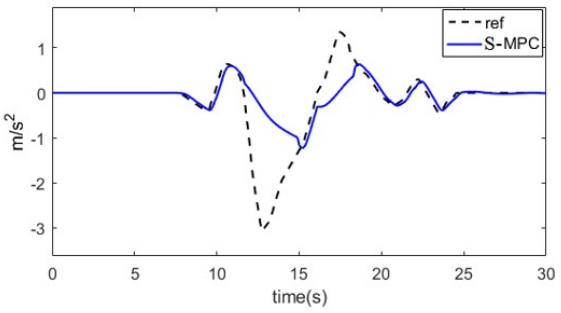
Figure 1 Stewart flight simulator motion platform (see online version for colours)

The remainder of this paper is organised as follows. Section 2 presents the modelling of the MPC system, including the human vestibular motion perception model and the kinematic model of the aircraft simulator. The proposed S-MPC-MCA, both with and without consideration of terminal conditions (COTC), is detailed in Section 3. Section 4 describes the simulation setup and discusses the results, highlighting the superiority of the S-MPC-MCA compared with conventional CWF-MCA and MPC-MCA approaches. Finally, Section 5 summarises the main findings and provides concluding remarks.

Figure 2 Longitudinal tracking performance of MPC-MCA and S-MPC-MCA, (a) MPC-MCA longitudinal acceleration tracking trajectory (b) S-MPC-MCA longitudinal acceleration tracking trajectory (see online version for colours)



(a)



(b)

2 Synthetic Stewart flight simulator modelling

This section presents the synthetic modelling of the Stewart flight simulator platform and its integration with the human vestibular system.

2.1 Human vestibular system model

The vestibular system is the primary sensory organ responsible for perceiving inertial cues such as gravity and motion, and consists of the semicircular canals and the otolith organs.

The semicircular canals, which act as angular velocity sensors, are fluid-filled structures aligned approximately along the three orthogonal axes of the human head. They are primarily responsible for detecting rotational motion, thereby enabling the perception of angular velocity. The transfer function of the linear semicircular canal model is expressed in equation (1).

$$\frac{\hat{\omega}}{\omega} = \frac{T_L T_a s^2}{(T_L s+1)(T_a s+1)(T_S s+1)} \tag{1}$$

where ω denotes the input angular velocity signal in the vestibular model, $\hat{\omega}$ represents the perceived angular velocity, and T_L and T_a are the model parameters associated with the semicircular canals.

The otolith organs are responsible for perceiving translational motion, and their behaviour can be represented by the transfer function given in equation (2).

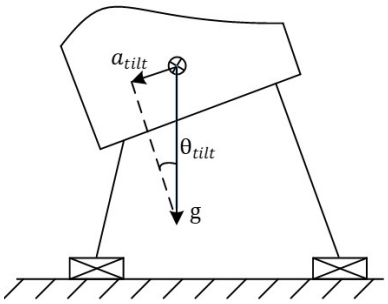
$$\frac{\hat{a}}{a} = \frac{\Gamma_a s+1}{\Gamma_L s+1} \cdot \frac{K}{\Gamma_s s+1} \tag{2}$$

where a denotes the input translational acceleration signal in the vestibular model, $\hat{a}$ represents the translational acceleration perceived by the human body, and Γ_a, Γ_L, and K are the otolith model parameters.

Given the limited motion workspace of simulator platforms, sustained acceleration cues are estimated using the tilt coordination effect, which exploits the lateral and longitudinal components of gravitational acceleration (Grant et al., 1984). In this approach, the low-frequency component

of the acceleration signal is converted into the tilt angle of the simulator platform, as illustrated in Figure 3.

Figure 3 Schematic diagram illustrating the coordination transfer between horizontal and vertical tilt in the 3simulator platform



Accordingly, the expression for computing the translational acceleration in tilt coordination as a function of the tilt angle is derived as follows:

$$\theta_{\text{tilt}} = \arcsin\left(\frac{a_{\text{tilt}}}{g}\right) \approx \frac{a_{\text{tilt}}}{g} \tag{3}$$

where θ_{tilt} denotes the tilt coordination angle, a_{tilt} is the translational acceleration component induced by tilt coordination, and g is the gravitational acceleration.

The corresponding transfer function of tilt coordination is given by:

$$\frac{\hat{a}_t}{\omega_{\text{tilt}}} = \frac{g \cdot K\left(\Gamma_a s + 1\right)}{s\left(\Gamma_L s + 1\right)\left(\Gamma_s s + 1\right)} \tag{4}$$

where ω_{tilt} denotes the input angular velocity of the platform tilt applied to the vestibular model, and $\hat{a}_t$ represents the translational acceleration perceived by the human body.

Consequently, the overall vestibular system can be modelled by combining the state-space representations of the semicircular canals, the otolith organs, and the tilt coordination mechanism:

$$x_P = \mathbf{A}_p x_p + \mathbf{B}_p u_p \tag{5}$$

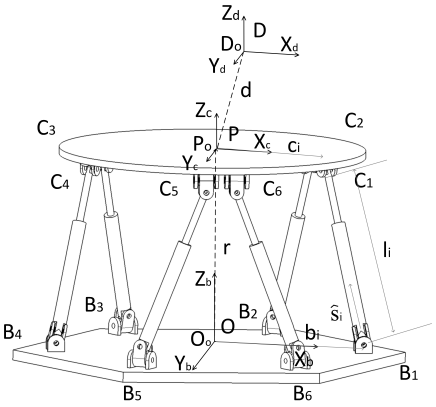
$$y_p = \mathbf{C}_p x_p \tag{6}$$

where $x_p = [x_{oth}\ x_{tilt}\ x_{acc}]^T \in \mathbb{R}^{21\times 1}$ denotes the overall state vector of the human vestibular model, with x_{oth}, x_{tilt}, and x_{acc} representing the state vectors of the otolith organ, tilt coordination, and semicircular canal models, respectively. The input vector is defined as $u_p = [a_p\ \ \omega_{P,tilt}\ \ \omega_{P,rot}\ ]^T \in \mathbb{R}^{8\times 1}$, where a_p, $\omega_{P,tilt}$, and $\omega_{P,rot}$ denote the platform translational acceleration, tilt coordination angular velocity, and rotational angular velocity, respectively. The output vector is given by $y_p = [\ \hat{a}_P\ \ \hat{a}_{P,tilt}\ \ \hat{\omega}_{P,rot}]^T \in \mathbb{R}^{8\times 1}$, where $\hat{a}_P$, $\hat{a}_{P,tilt}$, and $\hat{\omega}_{P,rot}$ correspond to the perceived platform translational acceleration, the translational acceleration generated by tilt coordination, and the perceived rotational angular velocity, respectively. The state-space model matrices are denoted as $\mathbf{A}_p \in \mathbb{R}^{21\times 21}$, $\mathbf{B}_p \in \mathbb{R}^{21\times 8}$, and $\mathbf{C}_p \in \mathbb{R}^{8\times 21}$ for the integrated vestibular system.

2.2 Stewart motion platform model

The geometric configuration and reference inertial frames of the 6-DoF Stewart platform are illustrated in Figure 5. In this figure, the coordinate origin and the platform's fixed base centroid (point O_0) are defined, where O_0 remains stationary and does not follow the motion of the platform. The platform frame is attached to the platform's centre of mass (point P_0 in Figure 4) and moves together with the platform. The operator frame is fixed to the operator's head and undergoes the corresponding motions. The coordinate origin of this frame coincides with the operator's eye point, and the associated coordinate system is denoted as D_0.

Figure 4 Geometric configuration and reference coordinate frames of the simulator platform.



The inverse kinematic relations of the motion platform are considered to constrain the actuator velocity and length, incorporating the kinematic formulations proposed by Harib and Srinivasan (2003). To determine the leg length, a closed-loop equation is derived for each mounting point $\boldsymbol{c}_i$ associated with its corresponding leg l_i as:

$$\boldsymbol{c}_i = \boldsymbol{r} + \boldsymbol{R}_P^O \cdot \boldsymbol{c}_i^P \tag{7}$$

where $\boldsymbol{r}$ denotes the position vector of O_P on the motion platform expressed in the inertial frame. $\boldsymbol{c}_i$ and $\boldsymbol{c}_i^P$ represent the position vectors of the connection point between the platform and the i^{th} leg, expressed in the inertial frame and the platform coordinate frame, respectively.

$\boldsymbol{R}_O^P$ denotes the rotation matrix that transforms vectors from the platform-fixed coordinate frame to the inertial frame:

$$\boldsymbol{R}_P^O = \begin{bmatrix} \boldsymbol{c\psi c\phi} - \boldsymbol{c\theta s\phi s\psi} & -\boldsymbol{s\psi c\phi} - \boldsymbol{c\theta s\phi c\psi} & \boldsymbol{s\theta s\phi} \\ \boldsymbol{c\psi s\phi} + \boldsymbol{c\theta c\phi s\psi} & -\boldsymbol{s\psi s\phi} + \boldsymbol{c\theta c\phi c\psi} & -\boldsymbol{s\theta c\phi} \\ \boldsymbol{s\psi s\theta} & \boldsymbol{c\psi s\theta} & \boldsymbol{c\theta} \end{bmatrix}$$

where $\boldsymbol{c}$ and $\boldsymbol{s}$ denote the cosine and sine functions, respectively, and ϕ, θ, and ψ represent the roll, pitch, and yaw angles.

The closed-loop equation for the i^{th} leg vector l_i can then be expressed as:

$$\boldsymbol{L}_i = l_i \cdot \hat{\boldsymbol{s}}_i = \boldsymbol{c}_i - \boldsymbol{b}_i = \boldsymbol{r} + \boldsymbol{R}_P^O \cdot \boldsymbol{c}_i^P - \boldsymbol{b}_i \tag{8}$$

where $\boldsymbol{L}_i$ denotes the vector of the i^{th} leg, l_i is its length, and $\hat{\boldsymbol{s}}_i$ represents the corresponding unit direction vector. $\boldsymbol{b}_i$ denotes the position vector of the connection point between the base and the i^{th} leg in the inertial frame. Based on these definitions, the closed-loop leg equation introduced in equation (9) can be reformulated in terms of leg vectors.

Furthermore, the rate of change of the leg length $\dot{l}_i$ can be derived by differentiating the closed-loop relation in equation (9). This yields the following expression:

$$\dot{l}_i = \frac{d}{dt}\left(\sqrt{\boldsymbol{L}_i \cdot \boldsymbol{L}_i}\right) \tag{9}$$

The rate of change of the i^{th} leg length, $\dot{l}_i$, is a nonlinear function of two factors: the platform linear velocity $\boldsymbol{v}_p$ and the angular velocity $\boldsymbol{\omega}_p$. Accordingly, $\dot{l}_i$ can be expressed as:

$$\dot{l}_i = h(\boldsymbol{x}_l) \tag{10}$$

where $\boldsymbol{x}_l = \begin{bmatrix}\boldsymbol{v}_p\, \boldsymbol{\omega}_p\end{bmatrix}^T \in \mathbb{R}^{6\times 1}$ denotes the platform motion state vector, consisting of the linear velocity $\boldsymbol{v}_p$ and angular velocity $\boldsymbol{\omega}_p$.

For analytical tractability, the nonlinear relation in equation (11) is linearised about the origin of the coordinate system O_0, yielding:

$$\dot{l}_i = \boldsymbol{A}_i \boldsymbol{v}_p + \boldsymbol{B}_i \boldsymbol{\omega}_p \tag{11}$$

2.3 Integrated system model

The integrated system model consists of the vestibular system model with tilt coordination and the kinematic model of the simulator motion platform. The state-space representation of the integrated system, expressed in the simulator's inertial frame, is given by:

$$\dot{\mathbf{x}}_t = \begin{cases} \dot{r}_p = v_p \\ \dot{v}_p = a_p \\ \dot{\beta}_{p,\text{rot}} = \omega_{p,\text{rot}} \\ \dot{\beta}_{p,\text{tilt}} = \omega_{p,\text{tilt}} \\ \dot{x}_p = \mathbf{A}_p x_p + \mathbf{B}_p \left[a_P^D; \omega_{P,\text{tilt}}^D; \omega_{P,\text{rot}}^D\right] \\ \dot{l}_i = \mathbf{A}_i v_p + \mathbf{B}_i \omega_p \end{cases} \tag{12}$$

$$a_P^D = a_p + \boldsymbol{v}_p \times \left(\boldsymbol{R}_0^P \boldsymbol{q}^P\right) + \boldsymbol{\omega}_p \times \boldsymbol{\omega}_p \times \left(\boldsymbol{R}_0^P \boldsymbol{q}^P\right) \tag{13}$$

$$\omega_{P,\text{rot}}^D = \boldsymbol{C}_D^0 \omega_{p,\text{rot}} \tag{14}$$

$$\omega_{P,\text{tilt}}^D = \boldsymbol{C}_D^0 \omega_{p,\text{tilt}} \tag{15}$$

$$\omega_p = \omega_{p,\text{rot}} + \omega_{p,\text{tilt}} \tag{16}$$

where $\boldsymbol{r}_p$, $\boldsymbol{v}_p$, $\boldsymbol{a}_p$, $\boldsymbol{\beta}_p$, and $\boldsymbol{\omega}_p$ denote the displacement vector, translational velocity vector, translational acceleration vector, angular displacement vector, and angular velocity vector of the simulator motion platform in the platform frame, respectively. Similarly, $\boldsymbol{a}_P^D$, $\boldsymbol{\omega}_{P,\text{tilt}}^D$, and $\boldsymbol{\omega}_{P,\text{rot}}^D$ denote the translational acceleration vector, tilt coordination angular velocity, and rotational angular velocity vector at the operator's eyepoint in the operator frame, respectively.

The discretised state-space representation of the MPC system can be expressed as:

$$\boldsymbol{x}_m(k+1) = \boldsymbol{A}_m \boldsymbol{x}_m(k) + \boldsymbol{B}_m \boldsymbol{u}(k) \tag{17}$$

$$\boldsymbol{y}(k) = \boldsymbol{C}_m \boldsymbol{x}_m(k) \tag{18}$$

$\boldsymbol{x}_m = \begin{bmatrix}\boldsymbol{r}_p\ \boldsymbol{v}_p\ \boldsymbol{a}_p\ \boldsymbol{\beta}_{p,\text{rot}}\ \boldsymbol{\beta}_{p,\text{tilt}}\ \boldsymbol{x}_p\ l_i\end{bmatrix}^T$ and $\boldsymbol{y}$ denote the state and output vectors of the MPC model, respectively. $\boldsymbol{A}_m$, $\boldsymbol{B}_m$, and $\boldsymbol{C}_m$ are the corresponding system matrices, while $\boldsymbol{u}(k)$ represents the input vector, defined as $\begin{bmatrix}\boldsymbol{a}_p\ \boldsymbol{\omega}_{p,\text{rot}}\ \boldsymbol{\omega}_{p,\text{tilt}}\end{bmatrix}^T$.

Accordingly, the iterative state-space representation of equations (17)–(18) can be written as:

$$\boldsymbol{x}(k+1) = \boldsymbol{A}\boldsymbol{x}(k) + \boldsymbol{B}\Delta\boldsymbol{u}(k) \tag{19}$$

$$\boldsymbol{y}(k) = \boldsymbol{C}\boldsymbol{x}(k) \tag{20}$$

In the above equations, the control input increment is defined as $\Delta\boldsymbol{u}(k) = \boldsymbol{u}(k) - \boldsymbol{u}(k-1)$, and the augmented state vector is given by $\boldsymbol{x}(k) = \begin{bmatrix}\boldsymbol{x}_m(k) - \boldsymbol{x}_m(k-1)\ \ \boldsymbol{y}(k)\end{bmatrix}^T$. The corresponding system matrices $\boldsymbol{A}$, $\boldsymbol{B}$, and $\boldsymbol{C}$ are defined as:

$$\boldsymbol{A} = \begin{bmatrix}\boldsymbol{A}_m & \boldsymbol{0} \\ \boldsymbol{C}_m\boldsymbol{A}_m & \boldsymbol{I}\end{bmatrix}, \quad \boldsymbol{B} = \begin{bmatrix}\boldsymbol{B}_m \\ \boldsymbol{C}_m\boldsymbol{B}_m\end{bmatrix}, \quad \boldsymbol{C} = \begin{bmatrix}\boldsymbol{0}\ \boldsymbol{I}\end{bmatrix} \tag{21}$$

3 S-MPC-MCA

The proposed S-MPC-MCA is developed to enhance the dynamic simulation capability of aircraft simulators by improving motion cue fidelity, reducing tracking errors, and ensuring stable performance, particularly under complex and rapidly changing flight conditions. Unlike conventional MPC-based MCAs, where insufficient terminal conditions often cause unexpected fluctuations of the simulator platform, the S-MPC-MCA incorporates a switching mechanism to ensure stable and realistic motion cues.

The overall architecture of the S-MPC-MCA consists of five main components:

- an MPC-based MCA with COTC
- an MPC-based MCA without COTC
- an adaptive weight regulator (AWR)
- a supervisory controller (SC)
- a switch mixer (SM).

3.1 S-MPC-MCA system architecture

The system architecture of the S-MPC-MCA is illustrated in Figure 5. At its core, the SC continuously evaluates the feasibility of the two MPC-based controllers and orchestrates the switching between them. Specifically, when the MPC with COTC fails to generate a feasible solution,

the SC initiates a switch to the MPC without COTC to maintain continuity of control. Conversely, once the lateral or longitudinal acceleration tracking errors are reduced below a predefined threshold, the SC re-engages the MPC with COTC to ensure higher-fidelity motion cueing.

Direct switching, however, may result in abrupt state changes that induce noticeable motion artefacts and even destabilise the simulator platform. To address this, a time-varying weighting mechanism is embedded within the SC. This mechanism gradually adjusts control weights during transitions, thereby smoothing the switching process, suppressing undesirable transients, and improving both system stability and pilot perception of motion cues.

3.2 Controller design for MPC-MCA with COTC

The MPC-MCA with COTC is formulated with a cost function $J(k)$ designed to minimise motion sensation errors while faithfully reproducing the perceptual dynamics experienced by aircraft pilots. To achieve this, the reference trajectory $\boldsymbol{x}_{\text{ref}}$ is explicitly incorporated into the objective function, ensuring that predicted platform states accurately follow the desired motion cues.

Here, $\boldsymbol{x}(t+n|t)$ denotes the predicted state at the n^{th} prediction step, with sampling period T_s. Based on this formulation, the optimisation problem for the MPC-MCA with COTC can be expressed as:

$$\begin{aligned}
&\min_{u(i),x(k)\in U} J(k) \\
&= \sum_{i=1}^{N_p-1} \left(\boldsymbol{x}_{\text{ref}}(k+i \mid k) - \boldsymbol{x}(k+i \mid k)\right)^T \\
&\times \boldsymbol{Q}\left(\boldsymbol{x}_{\text{ref}}(k+i \mid k) - \boldsymbol{x}(k+i \mid k)\right) \\
&+ \sum_{i=1}^{N_c-1} \boldsymbol{u}^T(k+i \mid k)\boldsymbol{S}\boldsymbol{u}(k+i \mid k) \\
&+ \sum_{i=1}^{N_c-1} \Delta\boldsymbol{u}^T(k+i \mid k)\boldsymbol{R}\Delta\boldsymbol{u}(k+i \mid k) \\
&+ \left(\boldsymbol{x}_{\text{ref}}(k+N_p \mid k) - \boldsymbol{x}(k+N_p \mid k)\right)^T \\
&\times \boldsymbol{Q}_{N_p}\left(\boldsymbol{x}_{\text{ref}}(k+N_p \mid k) - \boldsymbol{x}(k+N_p \mid k)\right)
\end{aligned} \tag{22}$$

$$\begin{aligned}
&\boldsymbol{x}(k+i \mid k) \in E, \Delta\boldsymbol{u} \in \Delta U, \boldsymbol{u}(k+i \mid k) \in U, \\
&i = 1, \ldots, N_p
\end{aligned} \tag{23}$$

$$\boldsymbol{x}_a^p(k+N_p \mid k) + \boldsymbol{x}_a^{p,\text{tilt}}(k+N_p \mid k) = \boldsymbol{x}_a^{a,\text{ref}}(k) \tag{24}$$

$$\boldsymbol{x}(k+N_p \mid k) = 0 \tag{25}$$

In the above equations, N_p and N_c denote the prediction horizon and the control horizon, respectively. $\boldsymbol{R}$, $\boldsymbol{S}$, and $\boldsymbol{Q}$ are diagonal weighting matrices (DWMs) associated with the input rate, input, and output, respectively. The constraints are defined as $\boldsymbol{x}(k+i \mid k) \in E$, $\Delta\boldsymbol{u} \in \Delta U$, and $\boldsymbol{u}(k+i \mid k) \in U$, which correspond to the admissible state set, input rate limits, and input bounds.

Furthermore, $\boldsymbol{x}_a^p(k+N_p \mid k)$ and $\boldsymbol{x}_a^{p,\text{tilt}}(k+N_p \mid k)$ represent the predicted translational acceleration and tilt coordination acceleration of the vestibular system, while $\boldsymbol{x}_a^{a,\text{ref}}(k)$ denotes the acceleration perceived by the aircraft pilot's vestibular system at the initial instant of the prediction.

It should be noted that $\boldsymbol{Q}_{N_p}$ is the terminal weighting matrix for the outputs under terminal conditions. Based on the Riccati equation, it can be computed as:

$$\begin{aligned}
\boldsymbol{Q}_{N_p} = {} & \boldsymbol{A}^T\boldsymbol{Q}_{N_p}\boldsymbol{A} - \boldsymbol{A}\boldsymbol{Q}_{N_p}\boldsymbol{B}\left(\boldsymbol{B}^T\boldsymbol{Q}_{N_p}\boldsymbol{B} + \boldsymbol{R}\right)^{-1} \\
& \times \boldsymbol{B}^T\boldsymbol{Q}_{N_p}\boldsymbol{A} + \boldsymbol{Q}
\end{aligned} \tag{26}$$

The acceleration perceived by the simulator pilot's vestibular system is primarily composed of the translational acceleration generated by the platform's linear motion and the acceleration induced by the platform's tilting mechanism. Equation (25) ensures that the simulator pilot's vestibular system experiences the same acceleration $\boldsymbol{x}_a^{a,\text{ref}}$ as that of the aircraft pilot's vestibular system. To broaden the feasible range of optimised control solutions, constraints are imposed only at the terminal stage of the cost function, where the terminal state is denoted as $\boldsymbol{x}(k+N_p \mid k)$. A further distinction between the MPC-MCA without COTC and its counterpart with COTC lies in the terminal state condition: in the latter, the endpoint constraint is set to zero, as specified in equation (26).

3.3 Controller design for MPC-MCA without COTC

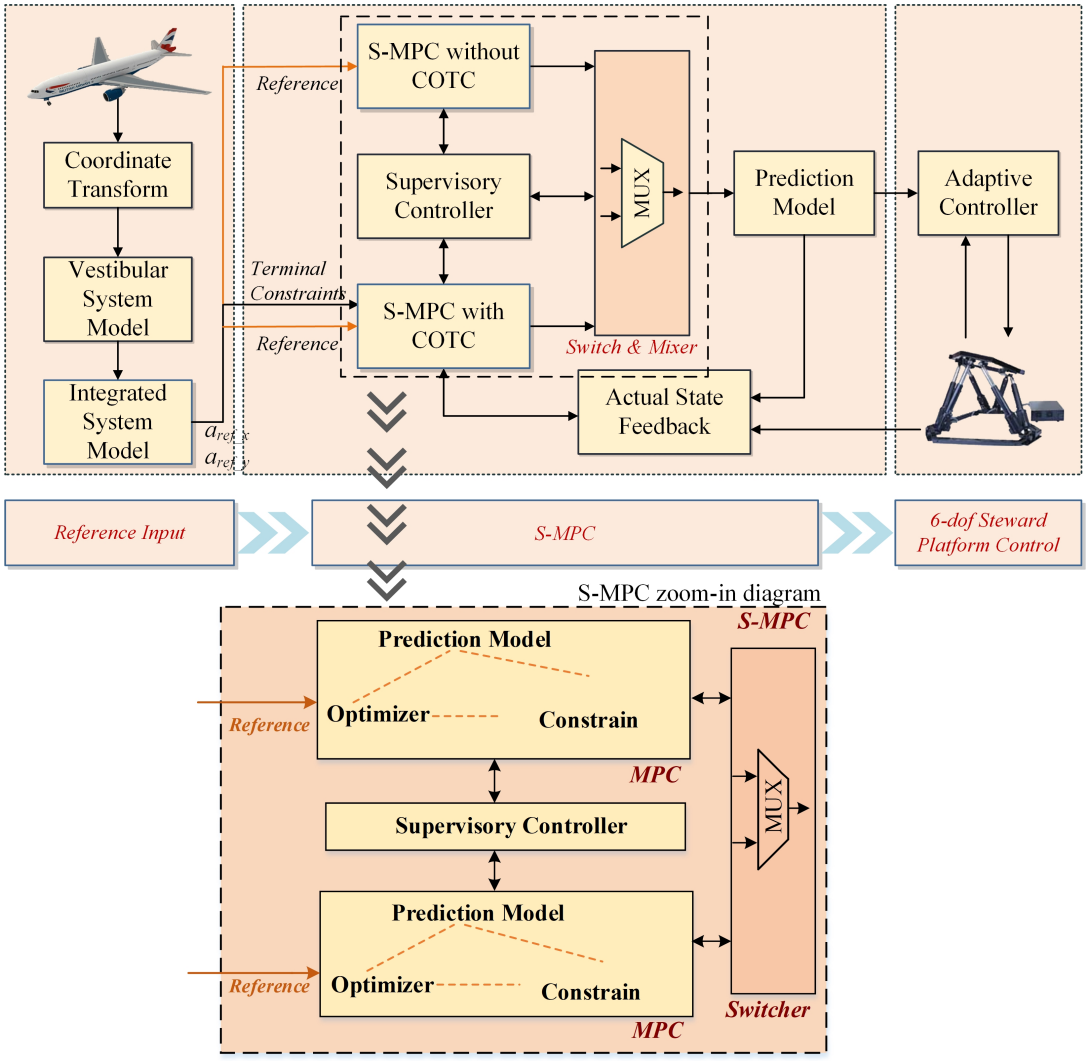
The presence of terminal state constraints in the MPC-MCA with COTC may render the optimisation problem infeasible. To ensure solvability, the optimisation problem is reformulated by removing these terminal constraints.

Two main distinctions exist between the MPC-MCA with and without COTC. First, the version without COTC eliminates the terminal state constraint. Second, its TWMs are set equal to the original weighting matrices $\boldsymbol{Q}$ defined in equation (23) (Dagdelen et al., 2009). With these modifications, the optimisation problem for the MPC-MCA without COTC is formulated as:

$$\begin{aligned}
&\min_{u(i),x(k)\in U} J(k) \\
&= \sum_{i=1}^{N_p-1} \left(\boldsymbol{x}_{\text{ref}}(k+i \mid k) - \boldsymbol{x}(k+i \mid k)\right)^T \\
&\times \boldsymbol{Q}\left(\boldsymbol{x}_{\text{ref}}(k+i \mid k) - \boldsymbol{x}(k+i \mid k)\right) \\
&+ \sum_{i=1}^{N_c-1} \boldsymbol{u}^T(k+i \mid k)\boldsymbol{S}\boldsymbol{u}(k+i \mid k) \\
&+ \sum_{i=1}^{N_c-1} \Delta\boldsymbol{u}^T(k+i \mid k)\boldsymbol{R}\Delta\boldsymbol{u}(k+i \mid k)
\end{aligned} \tag{27}$$

In the above equation, $\boldsymbol{R}$, $\boldsymbol{S}$, and $\boldsymbol{Q}$ denote the DWMs specific to the input rate, input, and output.

Figure 5 S-MPC integrated MCA system diagram (see online version for colours)



3.4 Stability analysis

Previous studies (Cleij et al., 2019; Rengifo et al., 2018) developed a feedback control law for linear systems to address the minimum-energy regulator problem with fixed terminal constraints. This formulation was later extended to linear time-varying systems using a receding-horizon approach, which has been demonstrated to guarantee asymptotic stability. Building on these results, the present study adopts the same methodology to evaluate the stability of the MPC-based MCA. In addition, for the case without COTC, stability can still be ensured by selecting a sufficiently long prediction horizon, such as the infinite output prediction horizon (Rengifo et al., 2018). Based on these considerations, the feedback control law can be expressed as:

$$u(t) = -\boldsymbol{R}^{-1}\boldsymbol{B}^{p}\boldsymbol{P}^{-1}(t+T_s, t+N_pT_s)\boldsymbol{A}x(t) \tag{28}$$

The Riccati equation is employed to compute $\boldsymbol{P}^{-1}$, which serves as the solution for the associated optimal control problem and provides the basis for the feedback gain calculation:

$$\begin{aligned}
&\boldsymbol{P}(t+k \mid t) \\
&= \boldsymbol{A}^{-1}\boldsymbol{P}(t+k+1 \mid t)\boldsymbol{A}^{-1T} \\
&- \boldsymbol{A}^{-1}\boldsymbol{P}(t+k+1 \mid t)\boldsymbol{A}^{-1T} - \boldsymbol{A}^{-1T}\boldsymbol{C}^{T}\boldsymbol{E}^{T} \\
&\times \left[\boldsymbol{I} + \boldsymbol{E}\boldsymbol{C}\boldsymbol{A}^{-1}\boldsymbol{P}(t+k+1 \mid t)\boldsymbol{C}^{T}\boldsymbol{E}^{T}\right]^{-1} \\
&\times \boldsymbol{E}\boldsymbol{C}\boldsymbol{A}^{-1}\boldsymbol{P}(t+k+1 \mid t)\boldsymbol{A}^{-1T} \\
&+ \boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^{T}
\end{aligned} \tag{29}$$

The matrix $\boldsymbol{P}$ is obtained from equation (29) together with the inverse sum spanning from N_p to the specified time step. In addition, $\boldsymbol{P}(t+T_s, t+N_pT_s)$ is constrained to zero as an additional condition imposed on the system.

It should be emphasised that the prediction horizon N_p must be sufficiently long to satisfy the requirements of the feedback control law in equation (30). The closed-loop system remains stable provided that the system in equation (15) is both controllable and observable. Specifically, if the pair (A, B) is controllable and (A, C) is observable, then, following Theorems 1 and 2 of (Kwon and Pearson, 1978), the stability of the system in equation (15) under terminal state constraints can be formally guaranteed.

To further assess the robustness of the proposed control framework against lateral stall error injections, three representative scenarios are considered: violent turbulence, wind shear-induced stall, and lateral stall with details to be discussed in the next section.

3.5 Time complexity analysis

The computational complexity of the proposed algorithm is primarily determined by matrix multiplications and additions. These operations typically exhibit a complexity ranging from $O(n^2)$ to $O(n^3)$, depending on matrix dimensions. However, since the sizes of the matrices involved are fixed and independent of the prediction horizon N_p, the associated computational cost can be regarded as constant.

The dominant factor in the overall complexity arises from the for-loop in the MPC algorithm, whose number of iterations is directly proportional to N_p. This loop is responsible for constructing the prediction model and solving the optimisation problem, and since each iteration involves a constant number of fixed-size matrix operations, the resulting complexity scales linearly with N_p.

Therefore, the overall time complexity of the algorithm can be expressed as $O(N_p)$, indicating that the computational cost grows linearly with the prediction horizon while remaining independent of the fixed matrix dimensions.

4 Simulation and results

4.1 Simulation setup

Both the S-MPC-MCA with and without COTC were implemented in MATLAB to assess controller performance. A virtual sensor was positioned at the pilot's eyepoint to capture reference acceleration and angular velocity signals, which were subsequently processed through the vestibular system model. The simulation data structure comprised lateral and longitudinal acceleration components, which served as equality constraints, together with angular velocity signals and corresponding platform state variables required for solving the optimisation problem.

Table 1 6-DoF Stewart simulator configurations

	Excursion	*Vel.*	*Acc.*
x	±1.7 m	±1.5 m/s	±10 m/s^2
y	±1.7 m	±1.5 m/s	±10 m/s^2
z	[2.2, 3.8] m	±1 m/s	±7 m/s^2
row	±25 °	±30°/s	±200°/s^2
pitch	±25 °	±30°/s	±200°/s^2
yaw	±30 °	±30°/s	±200°/s^2
$l_{1\ldots6}$	[2.5, 4.5] m		

Comprehensive simulations were performed under two representative flight scenarios – aircraft turbulence and horizontal wing stall – to examine the robustness and fidelity of the proposed approach. The results indicate that the S-MPC-MCA achieves a 25% improvement in error reduction compared with the conventional MPC-MCA, yielding an error rate of 0.0000125 versus 0.00001 for MPC-MCA.

The simulator parameters employed for performance evaluation are summarised in Table 1.

4.2 S-MPC performance under bumpy flight scenario

Bumpy flight conditions represent a typical challenge for flight simulators, as they induce rapid variations in lateral acceleration over short time intervals. Upset prevention and recovery training (UPRT) requires simulators to accurately reproduce these abrupt acceleration changes, which demands a MCA with both rapid response capability and high-fidelity somatosensory simulation.

Figure 6(a) illustrates the tracking performance of different MCAs under such conditions. To quantitatively compare performance, the normalised average absolute difference (NAAD) and average absolute scale (AAS) metrics were employed, with reference tracking evaluated across S-MPC, MPC-based MCA, and CWF. For fairness, each algorithm was tuned to maximise reference tracking accuracy while ensuring actuator positions remained within physical constraints.

The results show that the S-MPC-MCA achieves consistently lower NAAD and AAS values compared with MPC-MCA and CWF, confirming its superior tracking performance. Notably, during the intervals 10–13 s and 16–18 s, the S-MPC-MCA automatically switched controllers in response to physical platform limitations, thereby maintaining higher motion fidelity. Figure 6(b) further supports these findings, highlighting that S-MPC outperforms both benchmarks, whereas CWF consistently demonstrates inferior tracking performance due to its limited adaptability.

Figures 6(c)–6(f) provide additional insights into angular velocity tracking. Although all algorithms remained bounded within the range of -0.5 to 0.5 deg/s due to platform constraints, the S-MPC-MCA achieved the closest approximation to reference trajectories, combining accuracy with adaptability. While CWF exhibited performance comparable to the MPC-MCA in certain conditions, it lacked the responsiveness and robustness observed in the S-MPC-MCA.

4.3 S-MPC performance under horizontal stall scenario

Horizontal stall of aircraft wings is another critical stall phenomenon, characterised by abrupt and severe changes in lateral acceleration. As with the bumpy scenario, accurate replication of these dynamics is essential for UPRT applications. Figure 6(a) presents the tracking performance of the S-MPC-MCA under horizontal stall conditions, verifying its ability to deliver realistic motion cues with enhanced fidelity compared to conventional MPC and CWF approaches.

Figure 6 Performance analysis of UPRT simulation results in the bumpy scenario, (a) lateral acceleration (b) longitudinal acceleration (c) vertical acceleration (d) roll rate, (e) pitch rate(f) yaw rate (see online version for colours)

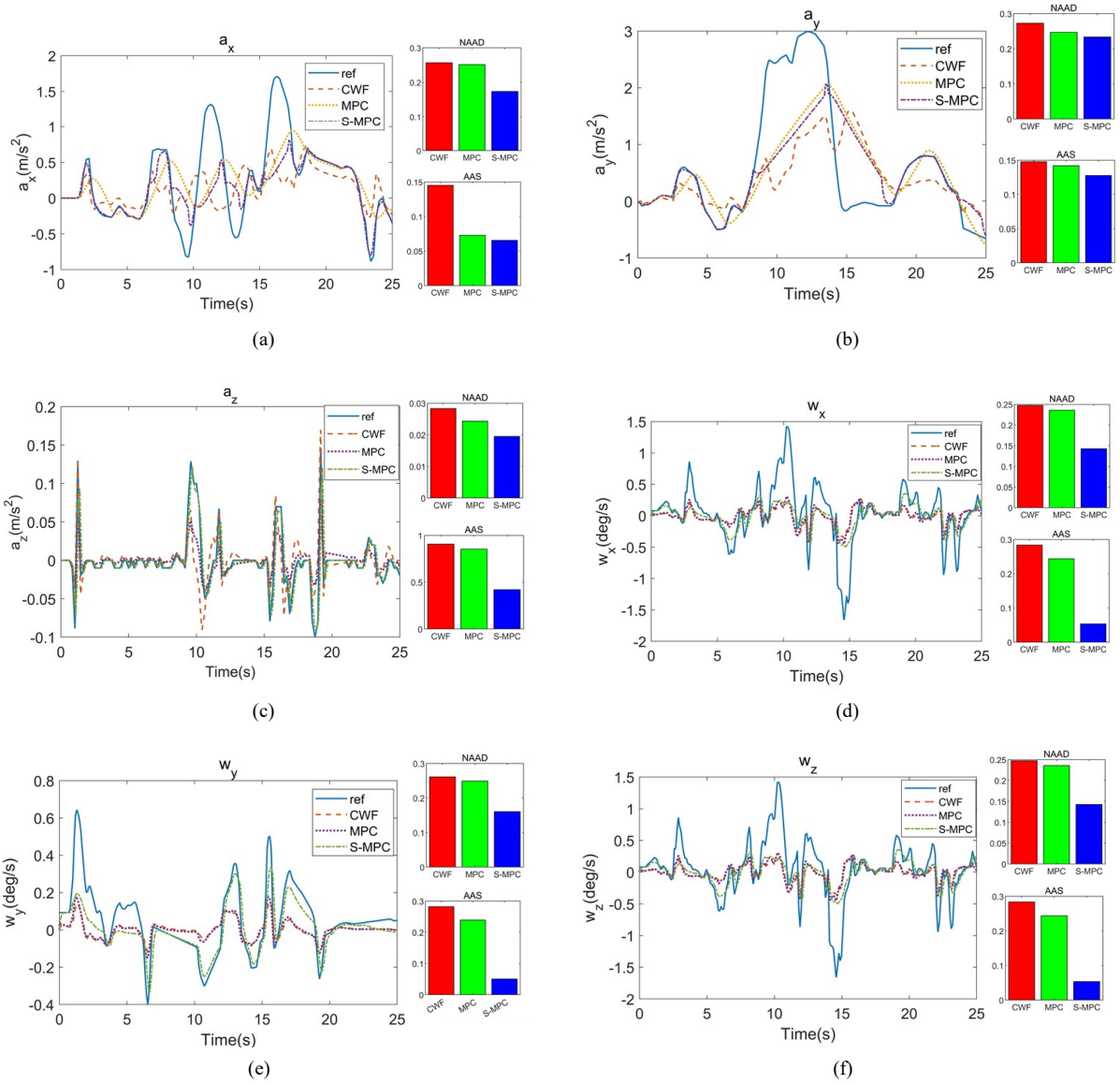


Figure 7 Performance analysis of lateral acceleration changes over different MCAs in UPRT under horizontal stall scenario (see online version for colours)

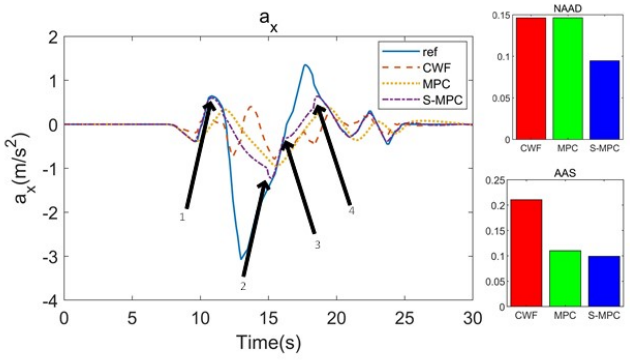
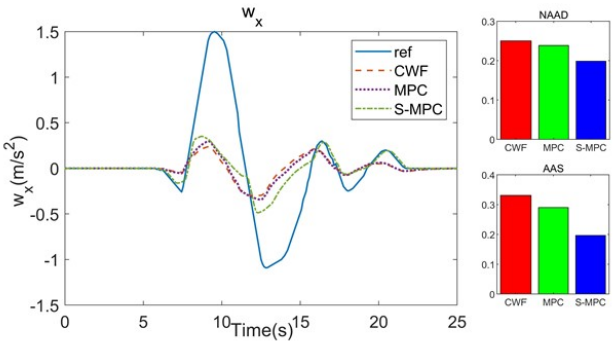


Figure 8 Performance analysis of roll rate changes over different MCAs in UPRT under horizontal stall scenario (see online version for colours)



As illustrated in Figures 7 and 8, the S-MPC-based MCA demonstrates a substantial improvement in tracking performance compared with both the CWF- and MPC-based MCAs, particularly in terms of its rapid convergence to reference trajectories. The transitions between stages 1–2 and 3–4 are effectively managed by employing the MPC without COTC, thereby ensuring accurate approximation tracking at the edge of the simulator's performance envelope. For the remaining simulation phases, the MPC with COTC is applied to achieve the best possible approximation tracking within the simulator's operational envelope, while maintaining high motion fidelity.

During the horizontal stall scenario, the AAS indices obtained for the S-MPC, MPC, and CWF algorithms are 0.196, 0.279, and 0.324, respectively. These results highlight the superior stability and precision of the S-MPC approach. Quantitatively, the S-MPC algorithm outperforms the MPC- and CWF-based approaches by 42.34% and 65.30%, respectively, thereby validating its robustness under demanding UPRT conditions.

5 Conclusions

This study proposed a novel S-MPC-MCA, which integrates the dynamic response of the human vestibular system with the kinematic model of a six-DoF Stewart platform. Within the simulator's operating envelope, accurate and rapid tracking of high-dynamic manoeuvres is achieved through the MPC-MCA with COTC. Beyond the operational envelope, optimal tracking is ensured by switching to the MPC-MCA without COTC, thereby preserving fidelity under conditions that would otherwise compromise feasibility.

The proposed algorithm was evaluated under representative UPRT scenarios, with performance benchmarked against CWF- and MPC-based MCAs. Results confirm that while MPC-based MCA consistently outperforms CWF by reducing motion perception errors, its performance remains constrained by terminal state feasibility issues. By contrast, the S-MPC approach dynamically switches between controllers in real time, achieving superior convergence, higher stability, and greater adaptability. Overall, the S-MPC-MCA demonstrates excellent simulation performance under both nominal and overloaded UPRT conditions, offering a promising solution for next-generation high-fidelity flight simulators.

References

Asadi, H., Bellmann, T., Mohamed, S. et al. (2022) 'Adaptive motion cueing algorithm using optimized fuzzy control system for motion simulators', *IEEE Transactions on Intelligent Vehicles*, Vol. 8, No. 1, pp.390–403, DOI: 10.1109/TIV.2022.3147862.

Asadi, H., Bellmann, T., Qazani, M.R.C. et al. (2023) 'A novel decoupled model predictive control-based motion cueing algorithm for driving simulators', *IEEE Transactions on Vehicular Technology*, Vol. 72, No. 6, pp.7024–7034, DOI: 10.1109/TVT.2023.3237317.

Asadi, H., Lim, C.P., Mohamed, S., Nahavandi, D. and Nahavandi, S. (2019) 'Increasing motion fidelity in driving simulators using a fuzzy-based washout filter', *IEEE Transactions on Intelligent Vehicles*, Vol. 4, No. 2, pp.298–308, DOI: 10.1109/TIV.2019.2904388.

Asadi, H., Mohamed, S. and Nahavandi, S. (2015) 'Incorporating human perception with the motion washout filter using fuzzy logic control', *IEEE/ASME Transactions on Mechatronics*, Vol. 20, No. 6, pp.3276–3284, DOI: 10.1109/TMECH.2015.2405934.

Asadi, H., Mohamed, S., Zadeh, D.R. and Nahavandi, S. (2015) 'Optimisation of nonlinear motion cueing algorithm based on genetic algorithm', *Vehicle System Dynamics*, Vol. 53, No. 4, pp.525–545, DOI: 10.1080/00423114.2014.1003948.

Asadi, H., Mohamed, S., Lim, C.P. et al. (2016a) 'Robust optimal motion cueing algorithm based on the linear quadratic regulator method and a genetic algorithm', *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, Vol. 47, No. 2, pp.238–254, DOI: 10.1109/TSMC.2016.2523906.

Asadi, H., Nahavandi, S., Lim, C.P. et al. (2016b) 'A particle swarm optimization-based washout filter for improving simulator motion fidelity', *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Budapest, Hungary, pp.1963–1968, DOI: 10.1109/SMC.2016.7844527.

Baghaee, H.R., Mirsalim, M. and Gharehpetian, G.B. (2018) 'Performance improvement of multi-DER microgrid for small- and large-signal disturbances and nonlinear loads: novel complementary control loop and fuzzy controller in a hierarchical droop-based control scheme', *IEEE Systems Journal*, Vol. 12, No. 1, pp.444–451, DOI: 10.1109/JSYST.2016.2580617.

Cleij, D., Pool, D.M., Mulder, M. et al. (2020) 'Optimizing an optimization-based MCA using perceived motion incongruence models', *Proceedings of the 19th Driving Simulation and Virtual Reality Conference*, Antibes, France, pp.9–11.

Cleij, D., Venrooij, J., Pretto, P. et al. (2019) 'Comparison between filter- and optimization-based motion cueing algorithms for driving simulation', *Transportation Research Part F: Traffic Psychology and Behaviour*, Vol. 61, pp.53–68, DOI: 10.1016/j.trf.2017.04.005.

Dagdelen, M., Reymond, G., Kemeny, A., Bordier, M. and Maïzi, N. (2009) 'Model-based predictive motion cueing strategy for vehicle driving simulators', *Control Engineering Practice*, Vol. 17, No. 9, pp.995–1003, DOI: 10.1016/j.conengprac.2009.03.002.

Grant, J., Best, W. and Lonigro, R. (1984) 'Governing equations of motion for the otolith organs and their response to a step change in velocity of the skull', *Journal of Biomechanical Engineering*, Vol. 106, No. 4, pp.302–308, DOI: 10.1115/1.3138498.

Harib, K. and Srinivasan, K. (2003) 'Kinematic and dynamic analysis of Stewart platform-based machine tool structures', *Robotica*, Vol. 21, No. 5, pp.541–554, DOI: 10.1017/S0263574703005046.

Hwang, T-S., Yeh, S-K., Lin, J-R. and Su, W-P. (2009) 'Adaptive motion washout filter design by using self-tuning fuzzy control', *Proceedings of the IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Singapore, pp.811–815, DOI: 10.1109/AIM.2009.5229912.

Kang, J., Gui, Z., Liu, Y. et al. (2022) 'LDCT image quality improvement algorithm based on optimal wavelet basis and MCA', *Signal, Image and Video Processing*, Vol. 16, No. 8, pp.2303–2311, DOI: 10.1007/s11760-022-02196-1.

Khusro, Y.R., Zheng, Y., Grottoli, M. et al. (2020) 'MPC-based motion-cueing algorithm for a 6-DOF driving simulator with actuator constraints', *Vehicles*, Vol. 2, No. 4, pp.625–647, DOI: 10.3390/vehicles2040036.

Kouvaritakis, B. and Cannon, M. (2016) *Model Predictive Control*, Springer International Publishing, Cham, Switzerland.

Kwon, W.H. and Byun, D.G. (1989) 'Receding horizon tracking control as a predictive control and its stability properties', *International Journal of Control*, Vol. 50, No. 5, pp.1807–1824, DOI: 10.23919/ACC.1988.4790065.

Kwon, W. and Pearson, A. (1977) 'A modified quadratic cost problem and feedback stabilization of a linear system', *IEEE Transactions on Automatic Control*, Vol. 22, No. 5, pp.838–842, DOI: 10.1109/TAC.1977.1101619.

Kwon, W. and Pearson, A. (1978) 'On feedback stabilization of time-varying discrete linear systems', *IEEE Transactions on Automatic Control*, Vol. 23, No. 3, pp.479–481, DOI: 10.1109/TAC.1978.1101749.

Kwon, W.H., Bruckstein, A.M. and Kailath, T. (1983) 'Stabilizing state-feedback design via the moving horizon method', *International Journal of Control*, Vol. 37, No. 3, pp.631–643, DOI: 10.1080/00207178308932998.

Muyeen, S.M. and Al-Durra, A. (2013) 'Modeling and control strategies of fuzzy logic controlled inverter system for grid inter connected variable speed wind generator', *IEEE Systems Journal*, Vol. 7, No. 4, pp.817–824, DOI: 10.1109/JSYST.2013.2239893.

Parrish, R.V., Dieudonne, J.E. and Bowles, R.L. (1975) 'Coordinated adaptive washout for motion simulators', *Journal of Aircraft*, Vol. 12, No. 1, pp.44–50, DOI: 10.2514/3.59800.

Qazani, M.R.C., Asadi, H. and Nahavandi, S. (2020a) 'A motion cueing algorithm based on model predictive control using terminal conditions in urban driving scenario', *IEEE Systems Journal*, Vol. 15, No. 1, pp.445–453, DOI: 10.1109/JSYST.2020.2994154.

Qazani, M.R.C., Asadi, H., Bellmann, T. et al. (2020b) 'Adaptive washout filter based on fuzzy logic for a motion simulation platform with consideration of joints' limitations', *IEEE Transactions on Vehicular Technology*, Vol. 69, No. 11, pp.12547–12558, DOI: 10.1109/TVT.2020.3023478.

Qazani, M.R.C., Asadi, H., Bellmann, T. et al. (2020c) 'Adaptive washout filter based on fuzzy logic for a motion simulation platform with consideration of joints' limitations', *IEEE Transactions on Vehicular Technology*, Vol. 69, No. 11, pp.12547–12558, DOI: 10.1109/TVT.2020.3023478.

Qazani, M.R.C., Jalali, S.M.J., Asadi, H. et al. (2020d) 'Optimising control and prediction horizons of a model predictive control-based motion cueing algorithm using butterfly optimization algorithm', *Proceedings of the IEEE Congress on Evolutionary Computation (CEC)*, Glasgow, UK, pp.1–8, DOI: 10.1109/CEC48606.2020.9185842.

Qazani, M.R.C., Asadi, H. and Nahavandi, S. (2021a) 'An optimal motion cueing algorithm using the inverse kinematic solution of the hexapod simulation platform', *IEEE Transactions on Intelligent Vehicles*, Vol. 7, No. 1, pp.73–82, DOI: 10.1109/TIV.2021.3068286.

Qazani, M.R.C., Asadi, H., Mohamed, S. et al. (2021b) 'A time-varying weight MPC-based motion cueing algorithm for motion simulation platform', *IEEE Transactions on Intelligent Transportation Systems*, Vol. 23, No. 8, pp.11767–11778, DOI: 10.1109/TITS.2021.3106970.

Qazani, M.R.C., Asadi, H., Rostami, M. et al. (2021c) 'Adaptive motion cueing algorithm based on fuzzy logic using online dexterity and direction monitoring', *IEEE Systems Journal*, Vol. 16, No. 2, pp.1945–1953, DOI: 10.1109/JSYST.2021.3059285.

Rengifo, C., Chardonnet, J.R., Paillot, D. et al. (2018) 'Solving the constrained problem in model predictive control based motion cueing algorithm with a neural network approach', *Proceedings of the Driving Simulation Conference Europe VR*, Antibes, France, pp.63–69.

Schmidt, S.F. and Conrad, B. (1970) *Motion Drive Signals for Piloted Flight Simulators*, Technical Report NASA CR-1601, NASA Ames Research Center, Moffett Field, CA, USA.

van der Ploeg, J.R., Cleij, D., Pool, D.M. et al. (2020) 'Sensitivity analysis of an MPC-based motion cueing algorithm for a curve driving scenario', *Proceedings of the Driving Simulation Conference Europe VR*, Antibes, France.

Zadeh, L.A. (1988) 'Fuzzy logic', *Computer*, Vol. 21, No. 4, pp.83–93, DOI: 10.1109/2.53.

Zhuangkun, W. and Weisi, G. (2024) 'Control layer security: exploiting unobservable cooperative states of autonomous systems for secret key generation', *IEEE Transactions on Mobile Computing*, pp.9989-10000, DOI: 10.1109/TMC.2024.3369754.