Control Engineering Practice 17 (2009) 170–184

Contents lists available at ScienceDirect

Control Engineering Practice

journal homepage: www.elsevier.com/locate/conengprac

Optimal motion cueing for 5-DOF motion simulations via a 3-DOF motion simulator

Yang-Hung Chang [a], Chung-Shu Liao [b], Wei-Hua Chieng [a,*]

[a] Department of Mechanical Engineering, National Chiao-Tung University, 1001 Ta Hsueh Road, Hsinchu 30010, Taiwan
[b] R&D Department, Internet Motion Navigator Inc., No. 118-1, Danuan Road, Tucheng City, Taipei County 236, Taiwan

ARTICLE INFO

Article history:
Received 20 May 2006
Accepted 30 May 2008
Available online 14 July 2008

Keywords:
Motion cueing
Motion simulator
Online optimization
DOF
Euler angles

ABSTRACT

This study presents a novel motion-cueing strategy, which is applied to a motion simulator with three rotational degrees of freedom (DOF) to perform the roll, pitch, yaw, surge, and sway motions by using an online optimization algorithm. The weighting functions are adaptively tuned in each step, and the optimal Euler angles are obtained analytically. This motion-cueing algorithm is efficient since it requires no recursive search on the optimal solution. Experiments demonstrating the validity of the 5-DOF motion simulation are also included.

1. Introduction

The goal of a motion simulator is to give a realistic impression of a pilot's motion, as in an aircraft or a racing car. Unfortunately, this goal is not easy to achieve because simulators are limited by their workspace features and actuator capabilities such as maximum torque and velocity. Engineers have begun to solve this unrealism in simulators by developing a motion cue strategy, known as "washout filtering". The aim of washout filtering is to transform trajectories generated by a dynamic virtual reality (VR) model incorporating very large displacements into driving system commands that can give realistic motion cues to a pilot within the simulator's limited workspace.

Washout involves separating the motion cues into high- (onset) and low- (sustained) frequency components, enabling the cues to be managed and displayed within the physical confines of a given platform system. The washout must provide a high-pass filtering system, which may be linear or nonlinear, to limit the simulator cab excursions. Nonlinear designs include adaptive filters and other optimal control techniques, which are applied against various criteria.

Many schemes have been presented. Schmidt and Bjorn (1970) analyzed the motion drive signals for piloted flight simulators. Conrad and Schmidt (1971) proposed techniques for calculating motion drive signals. Sinacori (1973) proposed a practical approach to motion simulation. Bowles, Parrish, and Dieudonne (1975) presented the coordinated adaptive washout for motion simulators. Sivan, Ish-shalom, and Huang (1982) introduced an optimal control approach to the design of moving flight simulators. Ariel and Sivan (1984) addressed the false cue reduction in moving flight simulators. Reid and Nahon (1986) developed a flight simulator driving algorithm. Nahon and Reid (1990) developed simulator motion drive algorithms. Reid, Nahon, and Kirdeikis (1992) developed the adaptive simulator motion software with supervisory control. Idan and Sahar (1996) presented the robust controller for a dynamic six degree of freedom (DOF) flight simulator. Pouliot, Gosselin, and Nahon (1998) analyzed the motion simulation capabilities of 3-DOF flight simulators. Moshe and Nahon (1999) analyzed the offline comparison of classical and robust flight simulator motion control. Martin (2000) considered the whole body motion of motion cueing. Liao, Huang, and Chieng (2004) proposed another novel washout filter algorithm for a 6-DOF motion simulator. Chang, Chieng, Liao, and Jeng (2006) developed a master switching technique for electronic cam control with special reference to multi-axis coordinated trajectory following.

The theory and development of the optimal algorithm for a 6-DOF flight simulator have recently been discussed by Wu and Cardullo (1997) and Telban and Cardullo (2005). Their approach incorporates a mathematical model of the human vestibular system, constraining the pilot sensation error between the simulated aircraft and motion platform dynamics. The problem is to determine a transfer function matrix that relates the desired

* Corresponding author. Fax: +886 3 5720634.
E-mail addresses: jack.me92g@nctu.edu.tw (Y.-H. Chang), brian20040806@hotmail.com (C.-S. Liao), whc@cc.nctu.edu.tw (W.-H. Chieng).

0967-0661/$ - see front matter © 2008 Elsevier Ltd. All rights reserved.
doi:10.1016/j.conengprac.2008.05.016

Nomenclature

$\mathbf{x}^*$ local minimizer governed by KKT conditions
$\boldsymbol{\mu}^*$ multiplier vector of inequality constraints
$\mathbf{g}$ inequality constraints vector
T sampling time
g acceleration due to gravity
$f_\bullet$ cost function of $\bullet$
$E_\bullet$ error vector of $\bullet$
$Q_\bullet$ weighting matrix of $\bullet$
$\omega_\bullet$ angular velocity along $\bullet$-axis
$\hat{\omega}_\bullet$ reference angular velocity along $\bullet$-axis
$\omega_{\bullet,\,bound}$ maximum angular velocity along $\bullet$-axis
$\omega_{\bullet,\,indiff}$ indifference threshold for angular velocity along $\bullet$-axis
$\omega_{\bullet,\,k+1}$ desired angular velocity along $\bullet$-axis at step k+1
$\hat{\omega}_{\bullet,k+1,highpass}$ reference angular velocity along $\bullet$-axis after high-pass filtering at step k+1
$\theta_{\bullet,\,k}$ angle of $\bullet$ at step k
$\theta_{\bullet,\,k+1}$ angle of $\bullet$ at step k+1
$\hat{\theta}_{\bullet,k+1,highpass}$ reference angle of $\bullet$ calculated from $\hat{\omega}_{\bullet,k+1,highpass}$ at step k+1
$a_\bullet$ linear acceleration along $\bullet$-axis
$\hat{a}_\bullet$ reference linear acceleration along $\bullet$-axis
$a_{\bullet,\,k+1}$ desired linear acceleration along $\bullet$-axis at step k+1
$\hat{a}_{\bullet,k+1}$ reference linear acceleration along $\bullet$-axis at step k+1
$\hat{a}_{\bullet,k+1,lowpass}$ reference linear acceleration after low-pass filtering along $\bullet$-axis at step k+1
$a_{\bullet,\,k+1,\,highpass}$ desired high-frequency linear acceleration along $\bullet$-axis at step k+1
$\hat{a}_{\bullet,k+1,highpass}$ reference linear acceleration after high-pass filtering along $\bullet$-axis at step k+1
$\hat{\alpha}_{\bullet,k+1,highpass}$ reference angular acceleration along $\bullet$-axis at step k+1
$\ell_{\bullet,k}$ equivalent moment arm of $\bullet$
$||\bullet||_2$ norm of $\bullet$
$sign(\bullet)$ sign of $\bullet$
α, β, γ Euler angles of *XYZ* fixed coordinate
α', β', γ' Euler angles of *ZYZ* coordinate
θ_1, θ_2, θ_3 output angle of motors 1, 2, and 3
${}^G_C\mathbf{T}$ transformation from global coordinate to cockpit coordinate
${}^G_C\mathbf{R}_{XYZ}$ rotation matrix from global to cockpit coordinate by *XYZ* fixed angles
${}^G_C\mathbf{R}_{Z'Y'Z'}$ rotation matrix from global to cockpit coordinate by *ZYZ* Euler angles

simulator motion input to the aircraft input, such that a cost function constraining the pilot sensation error (between simulator and aircraft) is minimized.

However, these studies concern motion simulators with full spatial DOF, i.e. 6-DOF; the problem of real-time optimal motion-cueing techniques for simulating specific VR motion in a motion simulator with limited DOF has rarely been considered.

This study proposes a novel algorithm that evaluates the real-time optimal motion-cueing strategy for a motion simulator solely with the rotational DOF (yaw, pitch and roll). The algorithm optimizes the additional linear onset cues, provided the attitude and sustained cues remain. This algorithm comprises a linear classical washout filter (CLWF), a yawing washout filter (YWF), an adaptive washout filter (AWF) and a real-time optimal motion-cueing algorithm (ROMA). The present algorithm individually transforms the high-frequency and the low-frequency linear motions into the output angles of a 3-DOF simulator. These output angles are incorporated into the cockpit attitude control to perform the 5-DOF motion. The ROMA first defines a quadratic cost function to be minimized. This cost function corresponding to the performance index of 5-DOF motion is then decoupled into three Euler angles associated with the 3-DOF simulator. The restrictions of workspace and actuator capabilities are presented as inequality constraints of the motion performance optimization problem. Since the cost function has a quadratic (convex) form, KKT conditions may be introduced to locate the global optimum. Prior to the motion optimization ROMA, the YWF is applied to prevent the simulator cab excursion from exceeding the workspace. After the motion optimization ROMA, the AWF is applied if necessary to reset the simulator position gradually. All washout motions are performed in the insensible acceleration or rate to the pilot. The remaining Euler angles of the 3-DOF simulator, i.e. pitch and roll, should simultaneously account for the cockpit angular motion and the residual tilt presenting the linear motion. The bounds of the pitch and roll angles are formulated implicitly, and are calculated during each sampling time. The motor commands are obtained by substituting the desired Euler angles into the inverse kinematics model of the 3-DOF simulator.

2. Human perception and the performance index of motion cueing

The motion cues within the motion simulation can be categorized into six components, namely the translations and rotations along *x*, *y*, and *z* axes, respectively. These rotation angles are known as the Euler angles (yaw, pitch, and roll). In flight, any aircraft rotates about its center of gravity, a point that is the average location of the mass of the aircraft. It can be defined as a three-dimensional coordinate system through the center of gravity, with each axis perpendicular to the other two axes. The orientation of the aircraft can then be defined as the DOF of the parts of the aircraft along these principal axes (*x*, *y*, and *z* axes), as shown in Fig. 1.

Motion simulation attempts to provide task-critical motion and force information (i.e. "cues") and any required components of the stressor-induced workload increment that would be present in flight or other vehicles (Martin, 2000). In practice, a motion simulator focuses most strongly on "linear acceleration" and "angular velocity" (Reid & Nahon, 1986). Defining an error of motion can help to construct a particular performance index for

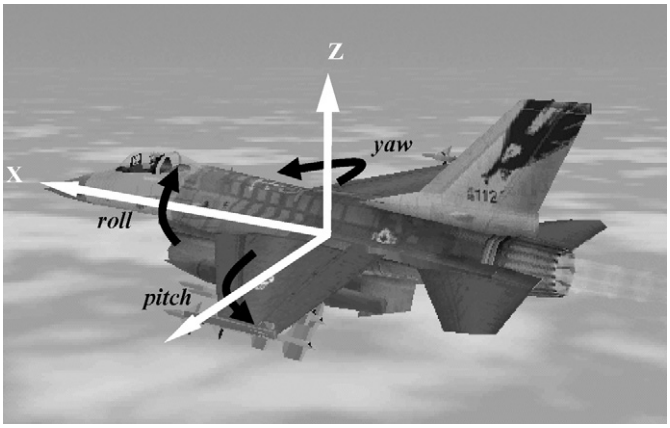

Fig. 1. Coordinates on an aircraft.

real-time processing of motion control. The error of a 6-DOF motion is defined as a vector of two norms of differences between the actual and required angular velocity/linear acceleration, as follows:

$$E_{overall} = \begin{bmatrix} ||\omega_x - \hat{\omega}_x||_2 \\ ||\omega_y - \hat{\omega}_y||_2 \\ ||\omega_z - \hat{\omega}_z||_2 \\ ||a_x - \hat{a}_x||_2 \\ ||a_y - \hat{a}_y||_2 \\ ||a_z - \hat{a}_z||_2 \end{bmatrix} \quad (1)$$

Motion cueing attempts to minimize the above error vector. In this study, a motion simulator with three rotational DOF is concerned. This motion simulator is not naturally capable of performing any linear motion. To present the cues of linear acceleration to a pilot, the cockpit needs to be offset from the pivot of the simulator mechanism, and the rotational motion cue must be sacrificed. Moreover, only the linear accelerations greater than the indifference threshold (Reid & Nahon, 1986) have to be transformed into angles.

Fig. 2 shows a particular 3-DOF motion simulator. This motion simulator employs three rotational DOF. Since the 3-DOF motion simulator is pivoted by a ball joint, and the cockpit is supported on top of the pivot, the heave motion, i.e. linear motion along the z-axis, must be left behind during the error minimization. This study is concerned with the following five motion cues:

$$E_{total} = \begin{bmatrix} ||\omega_x - \hat{\omega}_x||_2 \\ ||\omega_y - \hat{\omega}_y||_2 \\ ||\omega_z - \hat{\omega}_z||_2 \\ ||a_x - \hat{a}_x||_2 \\ ||a_y - \hat{a}_y||_2 \end{bmatrix} \quad (2)$$

Since the surge and sway motion can only be induced by the pitch and roll motion of the motion simulator, the overall minimization

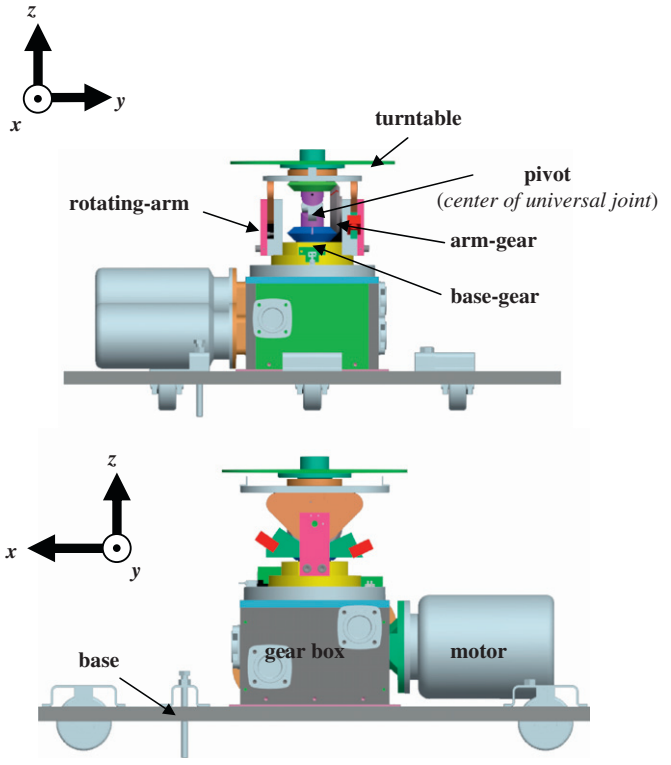

Fig. 2. Motion simulator mechanism with three rotational DOF.

problems can be divided into the following three sub-problems: yaw, longitudinal, and lateral:

$$E_{yaw} = [||\omega_z - \hat{\omega}_z||_2] \Rightarrow f_{yaw} = E_{yaw}^{\mathrm{T}} Q_{yaw} E_{yaw} \quad (3)$$

$$E_{longitudinal} = \begin{bmatrix} ||\omega_y - \hat{\omega}_y||_2 \\ ||a_x - \hat{a}_x||_2 \end{bmatrix} \Rightarrow f_{longitudinal} = E_{longitudinal}^{\mathrm{T}} Q_{longitudinal} E_{longitudinal} \quad (4)$$

$$E_{lateral} = \begin{bmatrix} ||\omega_x - \hat{\omega}_x||_2 \\ ||a_y - \hat{a}_y||_2 \end{bmatrix} \Rightarrow f_{lateral} = E_{lateral}^{\mathrm{T}} Q_{lateral} E_{lateral} \quad (5)$$

The reference point is determined by offset from the pivot to the pilot's head since the proposed algorithm is based on human's Vestibular System. The Vestibular System comprises semicircular canals, which respond to angular acceleration and velocity, and Otoliths, which are associated with gravity.

2.1. The performance index of yaw motion

The yaw axis (z-axis) is perpendicular to the wings, and lies in the plane of the aircraft centerline. A yaw motion is a side-to-side movement of the nose of the aircraft. The error of yaw motion is defined as

$$E_{yaw} = ||\omega_{z,k+1} - \hat{\omega}_{z,k+1,highpass}||_2 \quad (6)$$

In the discrete control domain, the above error may be formulated as

$$E_{yaw} = ||\theta_{z,k+1} - \hat{\theta}_{z,k+1,highpass}||_2 \quad (7)$$

where $\hat{\theta}_{z,k+1,highpass} = \theta_{z,k} + \hat{\omega}_{z,k+1,highpass} T$.

According to Eq. (3), such two-norm error representing the performance index of yaw motion may be expressed as a quadratic equation:

$$f_{yaw} = E_{yaw}^{\mathrm{T}} Q_{yaw} E_{yaw} = [\theta_{z,k+1} - \hat{\theta}_{z,k+1,highpass}]^2 \quad (8)$$

Assume that $Q_{yaw} = 1$.

2.2. The performance index of a combination of pitch and surge motions

The pitch axis (y-axis) is perpendicular to the aircraft centerline, and lies in the plane of the wings. A pitch motion is an up or down movement of the nose of the aircraft. The 3-DOF motion simulator with yaw, pitch, and roll controls can also yield a high-frequency (onset) linear acceleration in either the x or the y direction, because the cockpit is offset from the pivot of the simulator mechanism by a height distance. The linear motion in the x-axis may be induced due to the presence of pitch motion (rotation about the y-axis).

This optimization problem attempts to improve the cues on the cockpit attitude and the linear onset cues along the x-axis simultaneously in real time. The tradeoff between these two factors is determined by the weighting matrix $Q_{longitudinal}$. According to Eq. (4), the error vector can be expressed as

$$E_{longitudinal} = \begin{bmatrix} ||\theta_{y,k+1} - \hat{\theta}_{y,k+1}||_2 \\ ||a_{x,k+1} - \hat{a}_{x,k+1}||_2 \end{bmatrix} \quad (9)$$

Additionally, the otolith organs in the human vestibular system sense both the acceleration and tilting of the pilot's head with respect to the gravity vector (Telban & Cardullo, 2005). Since the otoliths cannot discriminate between acceleration and tilt, this phenomenon, known as tilt coordination, can be adopted to improve motion simulation. This additional cue results from

passing the vehicle acceleration through a low-pass filter to produce the desired long-duration tilt cue. Tilt coordination is implemented in a motion-cueing algorithm by adding additional cross-feed channels with low-pass filters in the longitudinal (pitch/surge) and lateral (roll/sway) modes that produce the additional rotational cues. The low-frequency linear motion cues can thus be incorporated into the pitch angle to yield the following form of the error vector

$$E_{longitudinal} = \begin{bmatrix} \left\| \theta_{y,k+1} - \left(\hat{\theta}_{y,k+1} - k_s \frac{\hat{a}_{x,k+1,lowpass}}{g} \right) \right\|_2 \\ ||a_{x,k+1,highpass} - \hat{a}_{x,k+1,highpass}||_2 \end{bmatrix} \tag{10}$$

The angle obtained from low-frequency linear motions of axis x is scaled by the constant k_s based on the weight effect of the pilot. In the domain of discrete control, the above error vector can be expressed as

$$E_{longitudinal} = \begin{bmatrix} \left\| \theta_{y,k+1} - \left(\hat{\theta}_{y,k+1} - k_s \frac{\hat{a}_{x,k+1,lowpass}}{g} \right) \right\|_2 \\ ||\theta_{y,k+1} - \hat{\theta}_{y,k+1,highpass}||_2 \end{bmatrix} \tag{11}$$

The angular acceleration of the pitch can be expressed in the discrete form as

$$\hat{\alpha}_{y,k+1,highpass} = \frac{\hat{\theta}_{y,k+1,highpass} - 2\theta_{y,k} + \theta_{y,k-1}}{T^2} \tag{12}$$

The relation between $\hat{a}_{x,k+1,highpass}$ and $\hat{\alpha}_{y,k+1,highpass}$ is

$$\hat{\alpha}_{y,k+1,highpass} = \frac{\hat{a}_{x,k+1,highpass}}{\ell_{pitch,k}} \tag{13}$$

where $\ell_{pitch,k}$ denotes the moment arm to the axis y (with respect to pitch angle) at step k. Eqs. (12) and (13) can be combined to obtain

$$\hat{\theta}_{y,k+1,highpass} = \left(\frac{T^2}{\ell_{pitch,k}} \right) \hat{a}_{x,k+1,highpass} + 2\theta_{y,k} - \theta_{y,k-1} \tag{14}$$

The performance index of the pitch motion can be stated as

$$f_{longitudinal} = E^{\mathrm{T}}_{longitudinal} \begin{bmatrix} 1 & 0 \\ 0 & W_{longitudinal,k} \end{bmatrix} E_{longitudinal} \tag{15}$$

where

$$E_{longitudinal} = \begin{bmatrix} \left\| \theta_{y,k+1} - \left(\hat{\theta}_{y,k+1} - k_s \frac{\hat{a}_{x,k+1,lowpass}}{g} \right) \right\|_2 \\ ||\theta_{y,k+1} - \hat{\theta}_{y,k+1,highpass}||_2 \end{bmatrix}$$

The above equation can be expanded into

$$f_{longitudinal} = [\theta_{y,k+1} - b]^2 + W_{pitch,k}[\theta_{y,k+1} - c]^2 \tag{16}$$

where

$$b = \hat{\theta}_{y,k+1} - \frac{k_s}{g} \hat{a}_{x,k+1,lowpass} \quad \text{and} \quad c = \hat{\theta}_{y,k+1,highpass}$$

A particular weighting function is chosen as $W_{longitudinal,k} = K_p |\hat{a}_{x,k+1,highpass}| e^{-K_e|\theta_{y,k}|}$ where K_p and K_e are constants. The linear onset motion is associated with a small weighting when the simulator cab excurses to the boundary of the workspace, and a progressive increase in weighting as the simulator cab approaches its home position.

2.3. The performance index of a combination of roll and sway motions

The roll axis (x-axis) lies along the aircraft centerline. A roll motion is an up and down movement of the wings of the aircraft. According to Eq. (5), the performance index of the roll motion can be stated as

$$f_{lateral} = E^{\mathrm{T}}_{lateral} \begin{bmatrix} 1 & 0 \\ 0 & W_{lateral} \end{bmatrix} E_{lateral} \tag{17}$$

where

$$E_{lateral} = \begin{bmatrix} \left\| \theta_{x,k+1} - \left(\hat{\theta}_{x,k+1} + k_s \frac{\hat{a}_{y,k+1,lowpass}}{g} \right) \right\|_2 \\ ||\theta_{x,k+1} - \hat{\theta}_{x,k+1,highpass}||_2 \end{bmatrix}$$

The angular acceleration of the roll can be expressed in the discrete form as

$$\hat{\alpha}_{x,k+1,highpass} = \frac{\hat{\theta}_{x,k+1,highpass} - 2\theta_{x,k} + \theta_{x,k-1}}{T^2} \tag{18}$$

The relation between $\hat{a}_{y,k+1,highpass}$ and $\hat{\alpha}_{x,k+1,highpass}$ is

$$\hat{\alpha}_{x,k+1,highpass} = -\frac{\hat{a}_{y,k+1,highpass}}{\ell_{roll,k}} \tag{19}$$

where $\ell_{roll,k}$ denotes the moment arm to the axis x (with respect to roll angle) at step k. Eqs. (18) and (19) can be combined to obtain

$$\hat{\theta}_{x,k+1,highpass} = \left(-\frac{T^2}{\ell_{roll,k}} \right) \hat{a}_{y,k+1,highpass} + 2\theta_{x,k} - \theta_{x,k-1} \tag{20}$$

The above equation can be expanded into

$$f_{lateral} = [\theta_{x,k+1} - b]^2 + W_{lateral,k}[\theta_{x,k+1} - c]^2 \tag{21}$$

where

$$b = \hat{\theta}_{x,k+1} + \frac{k_s}{g} \hat{a}_{y,k+1,lowpass} \quad c = \hat{\theta}_{x,k+1,highpass}$$

The weighting function may be chosen as $W_{lateral,k} = K_p |\hat{a}_{y,k+1,highpass}| e^{-K_e|\theta_{x,k}|}$ where K_p and K_e are constants.

3. 3-DOF motion simulator and motion cueing

Fig. 2 shows a 3-DOF motion simulator. Fig. 2 indicates that the yaw motion is performed by rotating the turntable through motor #1 (see Appendix A). The rotation of turntable is independent of the rest of the connecting mechanism, and thus the yaw motion is independent of the pitch and roll motion. The pitch and roll motion is performed by rotating the arm-gear through motor #2, and the base-gear through motor #3. The ratio of pitch to roll motion is determined from the amount of motor #3 rotation. These motions can be easily decoupled at the coordination control stage based on the inverse kinematic equations. The inverse kinematic equations shown in Appendix A are adopted to convert the Cartesian space motion into the joint space control commands.

The workspace of the motion simulator is restricted by the mechanical structure. The velocity of the motion simulator is limited by the driving system. A motion-cueing strategy must be able to confine the simulator cab within the workspace, provided that the driving system is not over-speeding at all instances. Since different mechanisms and driving systems may yield different bounds and limitations, this study formulates the constraints in terms of the cockpit coordinates, i.e. Z–Y–X Euler angles, for general cases.

3.1. Real-time motion simulation structure

Fig. 3 shows the real-time motion simulation structure of the 3-DOF motion simulator. The operator control inputs drive a mathematical model of the VR system, generates the vehicle states. Passing vehicle states through the real-time motion-cueing

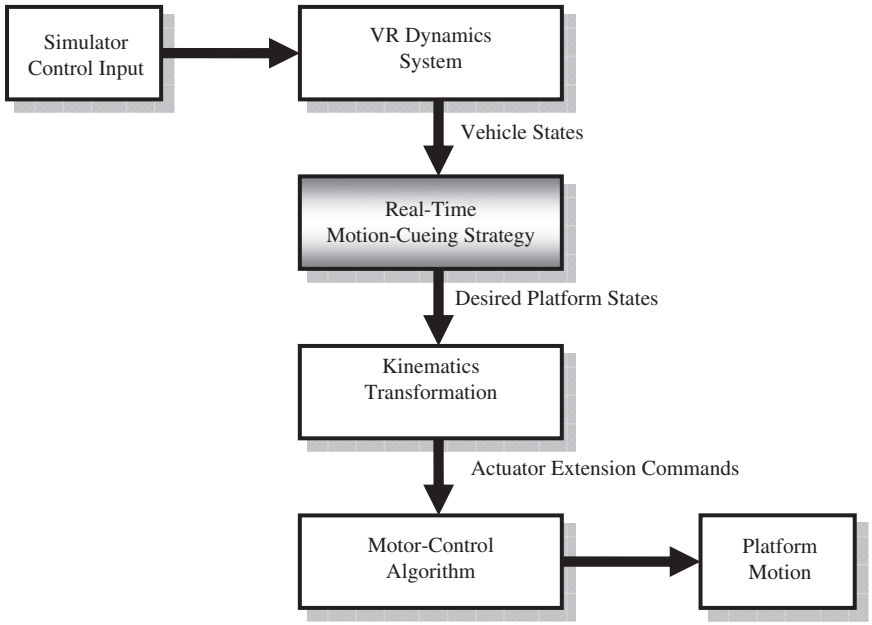

Fig. 3. 3-DOF motion simulation structure.

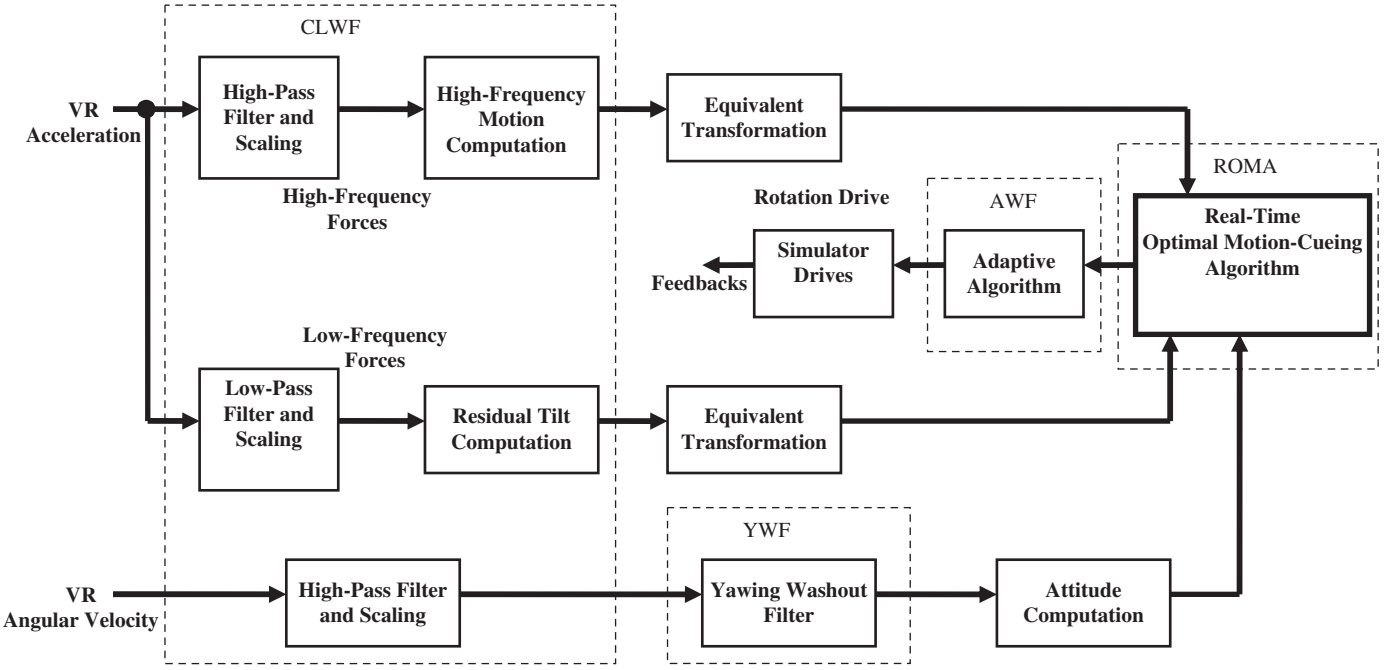

Fig. 4. Simplified schematic of the proposed motion-cueing strategy.

strategy produces the desired motion cues and platform states. The desired platform states are then transformed from DOF space to actuator space, generating the realized commands to the three actuators by kinematics transformation. The actuator motion commands serve as input to the platform dynamics, resulting in the actual simulator motion.

3.2. Motion-cueing strategy

The proposed motion-cueing strategy comprises three branches of motion cues, as shown in Fig. 4. All branches are fed into a ROMA for a motion optimization process. The first branch of the motion cue is the high-frequency (onset) linear motion; the second branch is the low-frequency (sustained) linear motion, and the third branch is the angular motion cue. The CLWF converts the sustained motion into the residual tilt (rotation angle of the cockpit). The angular motion cue is then fed into the YWF, which is discussed in Section 5, and subsequently fed into the ROMA.

4. Real-time optimal motion-cueing algorithm (ROMA)

Optimization theory and methods select the best alternative in the sense of the given objective function (Chong & Żak, 2001). The ROMA in this study involves minimizing the motion error where the inequality constraints of concerns are not violated.

4.1. K–K–T conditions

The KKT condition (Chong & Żak, 2001) is the necessary condition for the constrained optimization problem. Specifically, the points satisfying the KKT condition are considered as candidate minimizers. The cost (objective) function defined herein is in quadratic form, thus forming a convex optimization problem. The KKT condition also yields the sufficient condition of the global minimum. In sum, the KKT condition corresponding to the inequality constraints comprises four parts (two equalities and two inequalities):

$$\begin{aligned} &\text{(K1)} \quad \boldsymbol{\mu}^* \geqslant \mathbf{0} \\ &\text{(K2)} \quad Df(\mathbf{x}^*) + \boldsymbol{\mu}^{*\mathrm{T}} Dg(\mathbf{x}^*) = \mathbf{0}^{\mathrm{T}} \\ &\text{(K3)} \quad \boldsymbol{\mu}^{*\mathrm{T}} \mathbf{g}(\mathbf{x}^*) = \vec{0} \\ &\text{(K4)} \quad \mathbf{g}(\mathbf{x}^*) \leqslant \mathbf{0} \end{aligned}$$

4.2. Optimization on yaw motion

The constraint vector of mechanical boundaries and velocity limitations can be expressed as follows:

$$\mathbf{g} = \begin{bmatrix} \theta_{z,k+1}^2 - \theta_{z,bound}^2 \\ (\theta_{z,k+1} - \theta_{z,k})^2 - \omega_{z,bound}^2 T^2 \end{bmatrix} \tag{22}$$

The optimization problem can be stated as

$$\begin{aligned} &\min \quad f_{yaw} \\ &\text{s.t.} \quad \mathbf{g} \leqslant 0 \end{aligned} \tag{23}$$

Condition K1 yields

$$\boldsymbol{\mu}^* = \begin{bmatrix} \mu_1^* \\ \mu_2^* \end{bmatrix} \geqslant \mathbf{0} \tag{24}$$

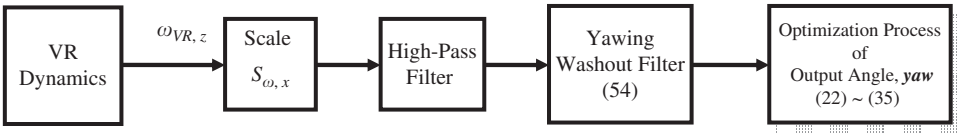

Fig. 5. Block diagram of optimization process of yaw motion.

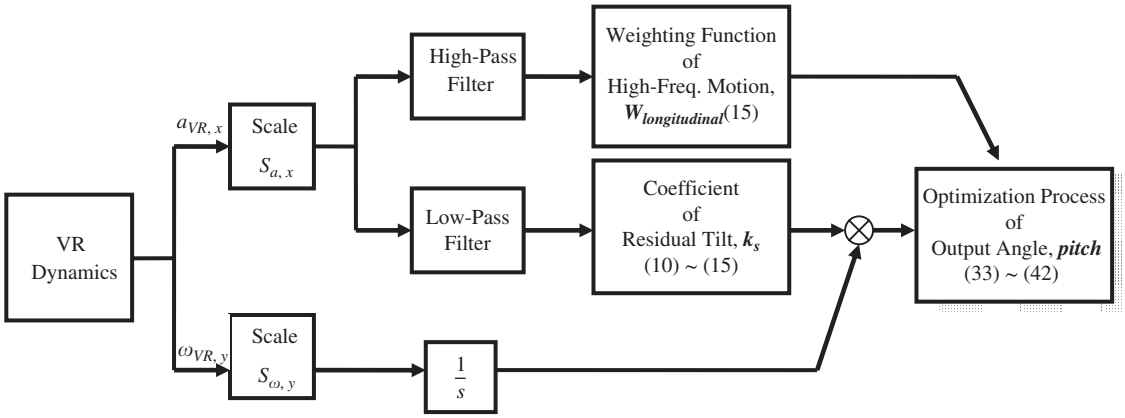

Fig. 6. Block diagram of optimization process of pitch and surge (longitudinal) motion.

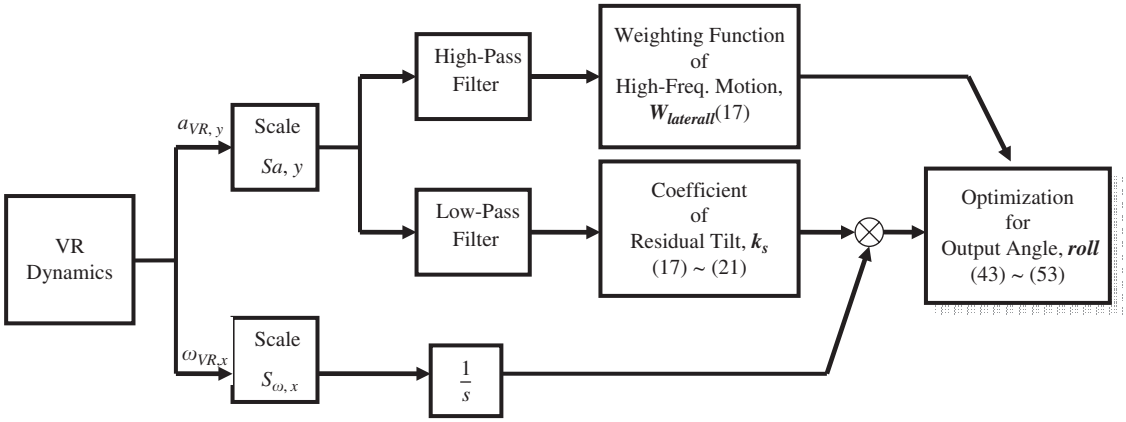

Fig. 7. Block diagram of optimization process of roll and sway (lateral) motion.

Condition K2 yields

$$\theta^*_{z,k+1} = \frac{\hat{\theta}_{z,k+1,highpass} + \mu^*_2 \theta_{z,k}}{1 + \mu^*_1 + \mu^*_2} \tag{25}$$

Condition K3 yields

$$\mu^*_1(\theta^{*2}_{z,k+1} - \theta^2_{z,bound}) + \mu^*_2[(\theta^*_{z,k+1} - \theta_{x,k})^2 - (\omega_{z,bound}T)^2] = 0 \tag{26}$$

Condition K4 yields

$$\begin{cases} \theta^{*2}_{z,k+1} - \theta^2_{z,bound} \leqslant 0 \\ (\theta^*_{z,k+1} - \theta_{x,k})^2 - (\omega_{z,bound}T)^2 \leqslant 0 \end{cases} \tag{27}$$

Substituting the conditions of Eq. (27) into Eq. (26) indicates that conditions $\mu^*_1 > 0$ and $\mu^*_2 > 0$ cannot both exist simultaneously. Thus, the optimum solution can be obtained from one of the three cases as follows:

Case 1: $\mu^*_1 = 0$, $\mu^*_2 = 0$. Eq. (25) yields

$$\theta^*_{z,k+1} = \hat{\theta}_{z,k+1,highpass} \tag{28}$$

Case 2: $\mu^*_1 > 0$, $\mu^*_2 = 0$. Eqs. (25) and (26) yield

$$\theta^*_{z,k+1} = \pm\theta_{z,bound} \tag{29}$$

provided that

$$\mu^*_1 = \frac{\hat{\theta}_{z,k+1,highpass} - \theta^*_{z,k+1}}{\theta_{z,bound}} > 0 \tag{30}$$

Case 3: $\mu^*_1 = 0$, $\mu^*_2 > 0$. Eqs. (25) and (26) yield

$$\theta^*_{z,k+1} = \theta_{z,k} \pm \omega_{z,bound}T \tag{31}$$

provided that

$$\mu^*_2 = \frac{\hat{\theta}_{z,k+1,highpass} - \theta^*_{z,k+1}}{\omega_{z,bound}T} > 0 \tag{32}$$

Fig. 5 shows the yaw motion optimization process with the YWF discussed in Section 5.

4.3. Optimization on pitch and surge motion

The constraint vector of mechanical boundaries and velocity limitations can be expressed as follows:

$$\mathbf{g} = \begin{bmatrix} \theta^2_{y,k+1} - \theta^2_{y,bound} \\ (\theta_{y,k+1} - \theta_{y,k})^2 - \omega^2_{y,bound}T^2 \end{bmatrix} \tag{33}$$

The optimization problem can be stated as

$$\begin{aligned} &\min \quad f_{longitudinal} \\ &\text{s.t.} \quad \mathbf{g} \leqslant 0 \end{aligned} \tag{34}$$

Condition K1 yields

$$\boldsymbol{\mu}^* = \begin{bmatrix} \mu^*_1 \\ \mu^*_2 \end{bmatrix} \geqslant \mathbf{0} \tag{35}$$

Condition K2 yields

$$\theta^*_{y,k+1} = \frac{b + W_{longitudinal,k}c + \mu^*_2 \theta_{y,k}}{1 + W_{longitudinal,k} + \mu^*_1 + \mu^*_2} \tag{36}$$

Condition K3 yields

$$\mu^*_1(\theta^{*2}_{y,k+1} - \theta^2_{y,bound}) + \mu^*_2[(\theta^*_{y,k+1} - \theta_{y,k})^2 - (\omega_{y,bound}T)^2] = 0 \tag{37}$$

Condition K4 yields

$$\begin{cases} \theta^{*2}_{y,k+1} - \theta^2_{y,bound} \leqslant 0 \\ (\theta^*_{y,k+1} - \theta_{y,k})^2 - (\omega_{y,bound}T)^2 \leqslant 0 \end{cases} \tag{38}$$

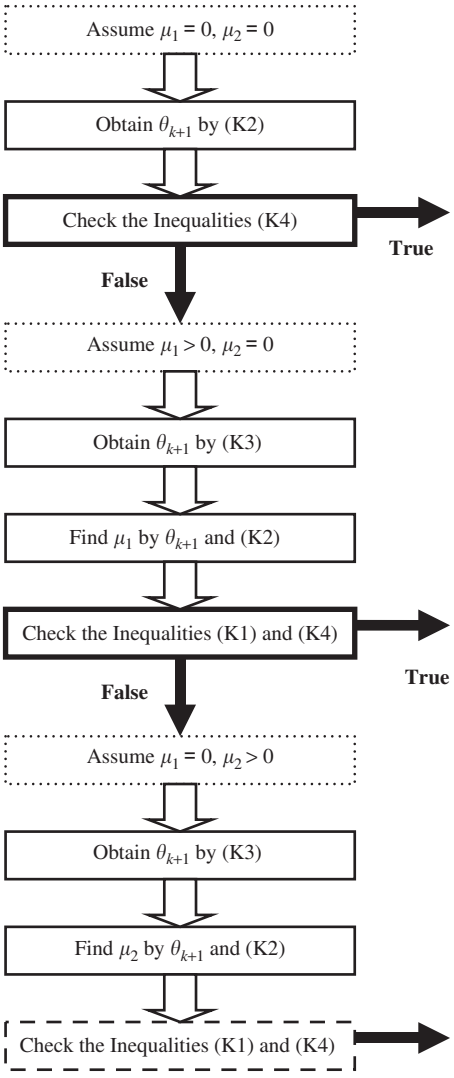

Fig. 8. Procedure of applying KKT conditions in practice.

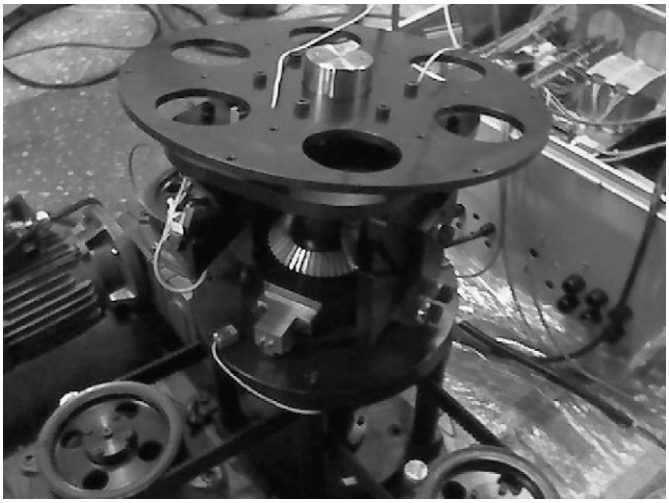

Fig. 9. Implementation of the control and driving system (courtesy of IMON Corp.).

Substituting the conditions of Eq. (38) into Eq. (37) indicates that conditions $\mu_1^* > 0$ and $\mu_2^* > 0$ cannot both exist simultaneously. Thus, the optimum solution can be obtained from one of the three cases as follows:

Case 1: $\mu_1^* = 0$, $\mu_2^* = 0$. Eq. (36) yields

$$\theta_{y,k+1}^* = \frac{b + W_{longitudinal,k}c}{1 + W_{longitudinal,k}} \tag{39}$$

Case 2: $\mu_1^* > 0$, $\mu_2^* = 0$. Eqs. (36) and (37) yield

$$\theta_{y,k+1}^* = \pm\theta_{y,bound}$$

provided that

$$\mu_1^* = \frac{b + W_{longitudinal,k}c - (1 + W_{longitudinal,k})\theta_{y,k+1}^*}{\theta_{y,bound}} > 0 \tag{40}$$

Case 3: $\mu_1^* = 0$, $\mu_2^* > 0$. Eqs. (36) and (37) yield

$$\theta_{y,k+1}^* = \theta_{y,k} \pm \omega_{y,bound}T \tag{41}$$

provided that

$$\mu_2^* = \frac{b + W_{longitudinal,k}c - (1 + W_{longitudinal,k})\theta_{y,k+1}^*}{\omega_{y,bound}T} > 0 \tag{42}$$

Fig. 6 shows the pitch motion optimization process.

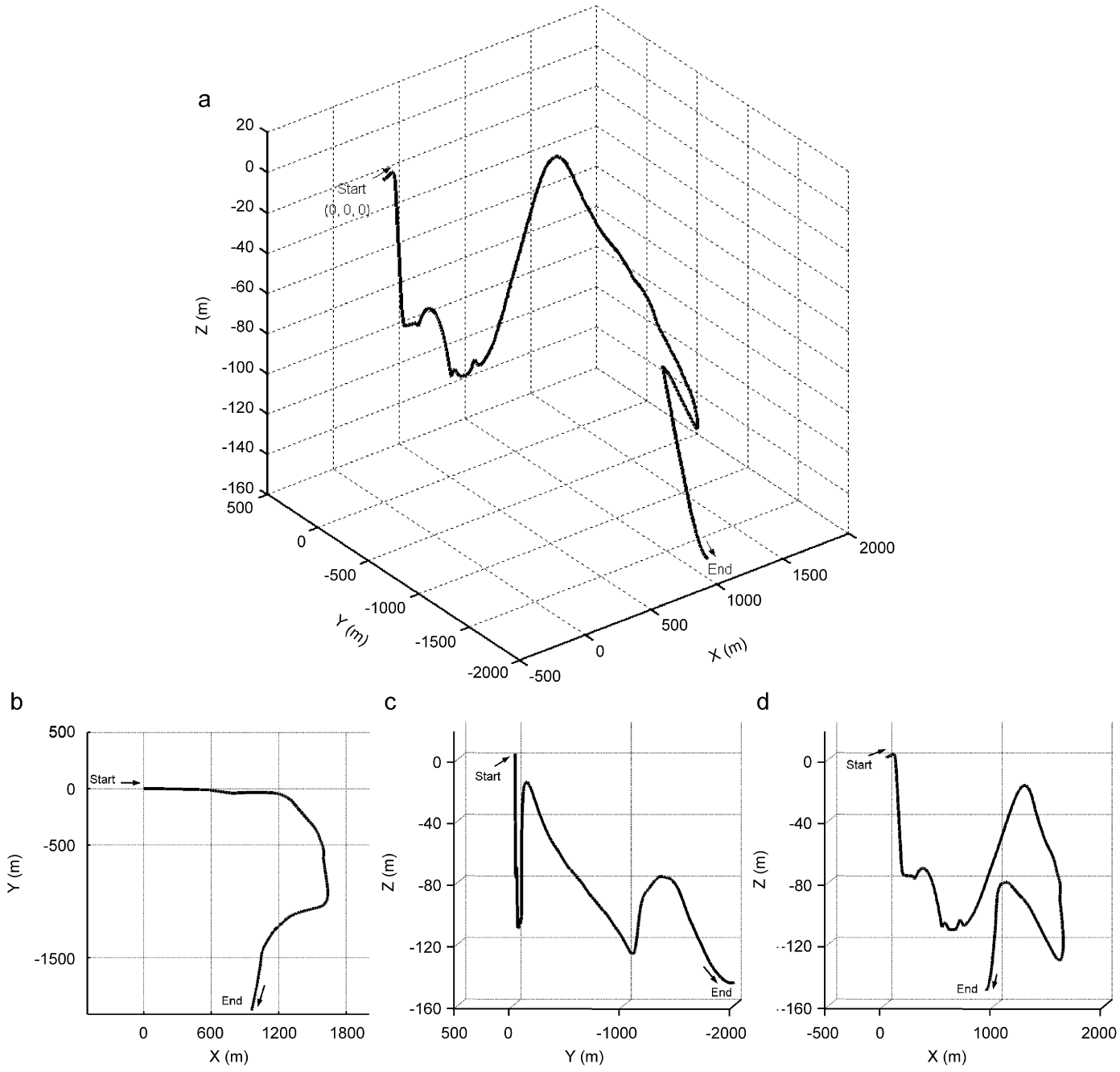

Fig. 10. Trajectory of flight simulation in the view of (a) 3-D view, (b) top view (X–Y plane), (c) front view (Y–Z plane), and (d) side view (X–Z plane).

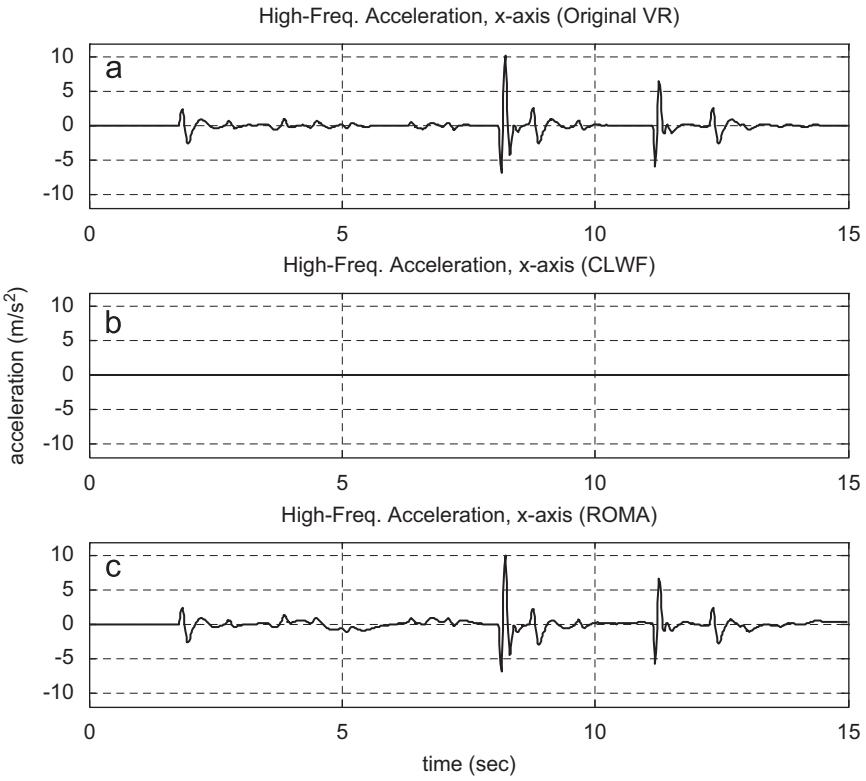

Fig. 11. Comparison of linear high-frequency acceleration along *x*-axis between: (a) original VR dynamic output, (b) simulator output using classical washout filter (CLWF), and (c) simulator output using proposed algorithm (ROMA).

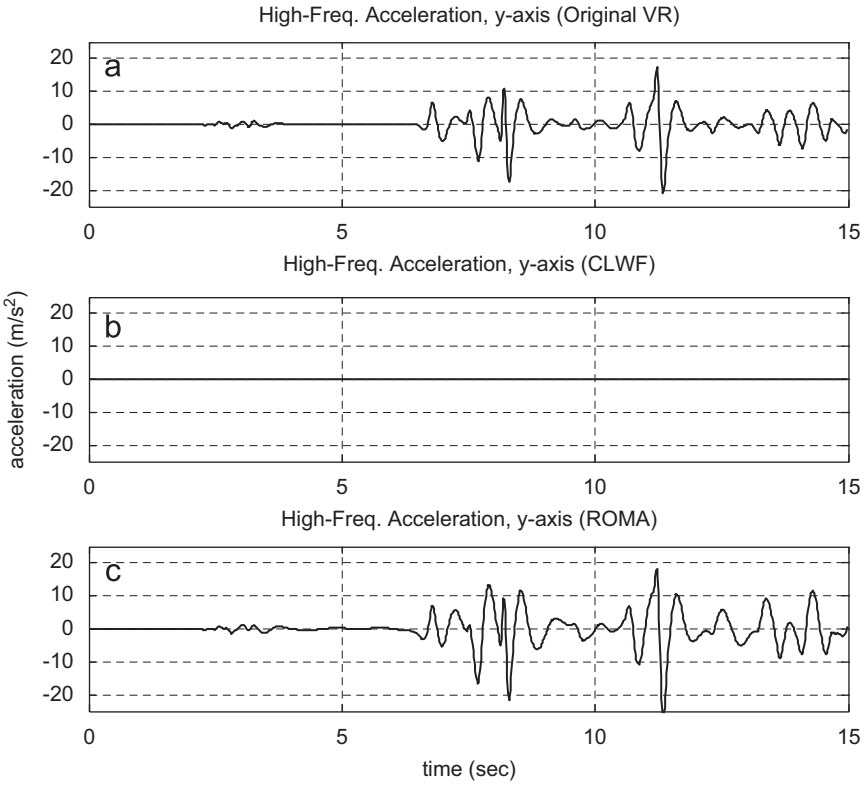

Fig. 12. Comparison of linear high-frequency acceleration along *y*-axis between: (a) original VR dynamic output, (b) simulator output using classical washout filter (CLWF), and (c) simulator output using proposed algorithm (ROMA).

4.4. Optimization on roll and sway motion

The constraint vector of mechanical boundaries and velocity limitations can be expressed as follows:

$$\mathbf{g} = \begin{bmatrix} \theta_{x,k+1}^2 - \theta_{x,bound}^2 \\ (\theta_{x,k+1} - \theta_{x,k})^2 - \omega_{x,bound}^2 T^2 \end{bmatrix} \tag{43}$$

The optimization problem can be stated as

$$\begin{aligned} &\min \quad f_{lateral} \\ &\text{s.t.} \quad \mathbf{g} \leqslant 0 \end{aligned} \tag{44}$$

Condition K1 yields

$$\boldsymbol{\mu}^* = \begin{bmatrix} \mu_1^* \\ \mu_2^* \end{bmatrix} \geqslant \mathbf{0} \tag{45}$$

Condition K2 yields

$$\theta_{x,k+1}^* = \frac{b + W_{lateral,k} c + \mu_2^* \theta_{x,k}}{1 + W_{lateral,k} + \mu_1^* + \mu_2^*} \tag{46}$$

Condition K3 yields

$$\mu_1^* (\theta_{x,k+1}^{*2} - \theta_{x,bound}^2) + \mu_2^* \left[(\theta_{x,k+1}^* - \theta_{x,k})^2 - (\omega_{x,bound} T)^2 \right] = 0 \tag{47}$$

Condition K4 yields

$$\begin{cases} \theta_{x,k+1}^{*2} - \theta_{x,bound}^2 \leqslant 0 \\ (\theta_{x,k+1}^* - \theta_{x,k})^2 - (\omega_{x,bound} T)^2 \leqslant 0 \end{cases} \tag{48}$$

Similarly, substituting the conditions of Eq. (48) into Eq. (47) indicates that conditions $\mu_1^* > 0$ and $\mu_2^* > 0$ cannot both exist simultaneously. Thus, the optimum solution can be obtained from one of the three cases as follows:

Case 1: $\mu_1^* = 0$, $\mu_2^* = 0$. Eq. (46) yields

$$\theta_{x,k+1}^* = \frac{b + W_{lateral,k} c}{1 + W_{lateral,k}} \tag{49}$$

Case 2: $\mu_1^* > 0$, $\mu_2^* = 0$. Eqs. (46) and (47) yield

$$\theta_{x,k+1}^* = \pm \theta_{x,bound} \tag{50}$$

provided that

$$\mu_1^* = \frac{b + W_{lateral,k} c - (1 + W_{lateral,k}) \theta_{x,k+1}^*}{\theta_{x,bound}} > 0 \tag{51}$$

Case 3: $\mu_1^* = 0$, $\mu_2^* > 0$. Eqs. (46) and (47) yield

$$\theta_{x,k+1}^* = \theta_{x,k} \pm \omega_{x,bound} T \tag{52}$$

provided that

$$\mu_2^* = \frac{b + W_{lateral,k} c - (1 + W_{lateral,k}) \theta_{x,k+1}^*}{\omega_{x,bound} T} \tag{53}$$

Fig. 7 shows the roll motion optimization process. Fig. 8 shows the procedure of applying the KKT condition in practice.

4.5. Physical meaning of each case of the optimization

Each optimization task has three cases when applying the KKT conditions. Taking pitch motion as an example, if the pitch motion reaches its workspace bound then $\mu_1^* > 0$, otherwise $\mu_1^* = 0$. If the velocity of the pitch motion reaches its maximum allowable value, then $\mu_2^* > 0$, otherwise $\mu_2^* = 0$.

Subject to individual optimization tasks, in case #1, $\mu_1^* > 0$ and $\mu_2^* > 0$ is applied at the instance of real-time computation that neither the workspace boundary nor the velocity constraint is violated, while in case #2, $\mu_1^* > 0$ and $\mu_2^* = 0$ is applied at the instance that the velocity constraint is not violated when the workspace bound is reached. For example, the pitch angle is sustained at its maximum pitch angle when the maximum allowable pitch angle is reached; otherwise, the corresponding position constraint is violated. Case #3: $\mu_1^* = 0$ and $\mu_2^* > 0$ is applied when the velocity constraint is violated, thus limiting the corresponding motion to the maximum velocity.

5. Washout filtering

Washout must provide a high-pass filtering scheme to limit the simulator cab excursions. The washout filters in the proposed algorithm include the yaw and adaptive filters as stated in the following sections.

5.1. Yawing washout filter

The YWF is applied to prevent the yawing angle at step (k+1) from passing beyond the limits of the motion simulator

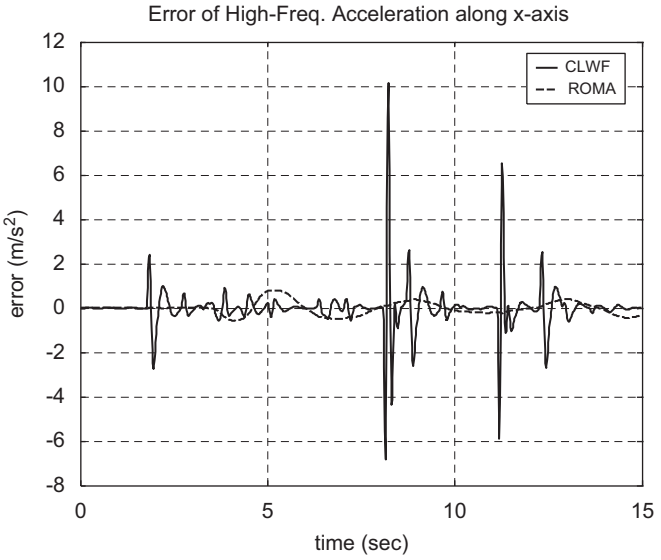

Fig. 13. Segmental error of linear high-frequency (onset) acceleration along x-axis using classical method (CLWF) and the proposed algorithm (ROMA).

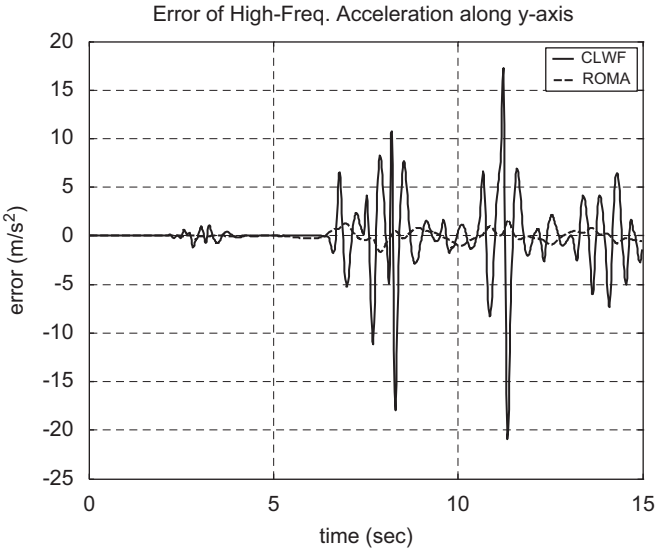

Fig. 14. Segmental error of linear high-frequency (onset) acceleration along y-axis using the classical method (CLWF) and the proposed algorithm (ROMA).

workspace before the optimization process ROMA takes place, as shown in Fig. 4. If the norm of the new yawing angular velocity is less than the indifference threshold, then the proposed yawing washout motion is in action

$$\text{If } |\hat{\omega}_{z,k+1,highpass}| \leqslant \omega_{z,indiff} \text{ then}$$
$$\begin{cases} \omega_{z,k+1} = -sign(\theta_{z,k})\omega_{z,indiff} \\ \text{and} \\ \theta_{z,k+1} = \theta_{z,k} - [sign(\theta_{z,k})\omega_{z,indiff}T] \end{cases} \quad (54)$$

The YWF continuously returns the cockpit to its home position, where the dexterity of motion is highest.

5.2. Adaptive washout filter

The optimization algorithm guarantees that the simulator cockpit does not exceed the workspace. However, this guarantee may fail in practice for the following reasons:

(1) The sampling frequency may not be stable, due to the high CPU loading subjected to the VR rendering.
(2) The workspace boundaries may be too complex to calculate the exact mechanical bounds. The actual boundaries of the 3-DOF motion simulator are associated with the *Z–Y–Z* Euler angles, i.e. α', β', and γ' (see the inverse kinematics in the appendix).

An AWF, as shown in Fig. 4, is proposed to compensate for the insufficiency of the prior proposed optimization process, and this accommodates more severe restrictions, such as the small workspace and the limited driving current. For yaw motion, the washout filter is implemented as follows:

$$\theta_{z,k+1} = \theta_{z,k} - [sign(\theta_{z,k})\omega_{z,indiff}T] \quad \text{if} \quad |\theta_{z,k}| > \theta_{z,softwarebound} \quad (55)$$

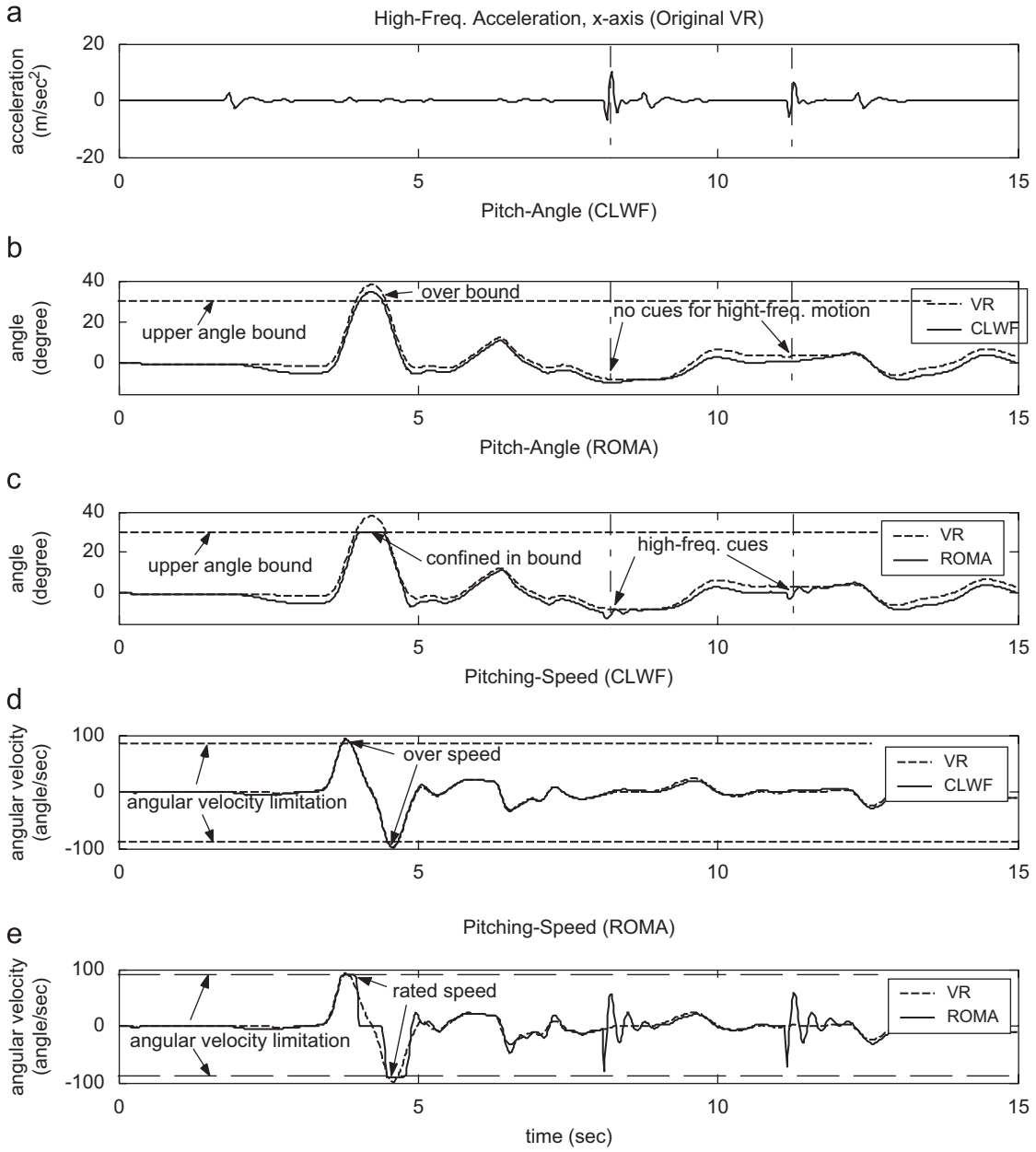

Fig. 15. Comparison of simulator output, longitudinal motion: (a) linear acceleration, *x*-axis (high-frequency), (b) pitch-angle, using classical washout filter (CLWF), (c) pitch-angle, using proposed algorithm (ROMA), (d) pitching-speed, using CLWF, and (e) pitching speed, using proposed algorithm (ROMA).

The following filter is applied to prevent the hunting motion from being possibly associated with the washout motion

$$\theta_{z,k+1}=0 \text{ if } \begin{cases} \theta_{z,k}\theta_{z,k+1}<0 \\ \text{or} \\ |\theta_{z,k+1}|\leqslant\varepsilon \end{cases} \quad (56)$$

where ε denotes a sufficiently small number. For the pitch and roll motion, further motion is prohibited as soon as the software limit of the workspace (in contrast to the hardware limit (actual boundary) of the workspace) is reached. These filters may be formulated as follows:

$$\theta_{x,k+1}=\theta_{x,k} \quad \text{if } |\theta_{x,k}|>\theta_{x,softwarebound} \quad (57)$$

$$\theta_{y,k+1}=\theta_{y,k} \quad \text{if } |\theta_{y,k}|>\theta_{y,softwarebound} \quad (58)$$

6. Experimental results and comparison

A simulation of Pseudo-Flight-Object (PFO) produced by IMON Corp. was applied in this study. The geocentric position and the body acceleration data of the aircraft produced from the equation of motion, were taken as inputs to the proposed motion-cueing strategy in the PFO software. The outputs of the experiment were the motor position commands to the 3-DOF motion simulator, as shown in Fig. 9. The results were compared to the CLWF. Fig. 10 consists of four plots used to demonstrate the flight trajectory in 3-D view, front view, side view, and top view, respectively. The flight data including longitudinal (pitch+x-acceleration), lateral (roll+y-acceleration), and yaw motions. Data of individual Euler angles (yaw, pitch and roll) were provided simultaneously to the proposed ROMA algorithm to yield the motion cue of the pilot.

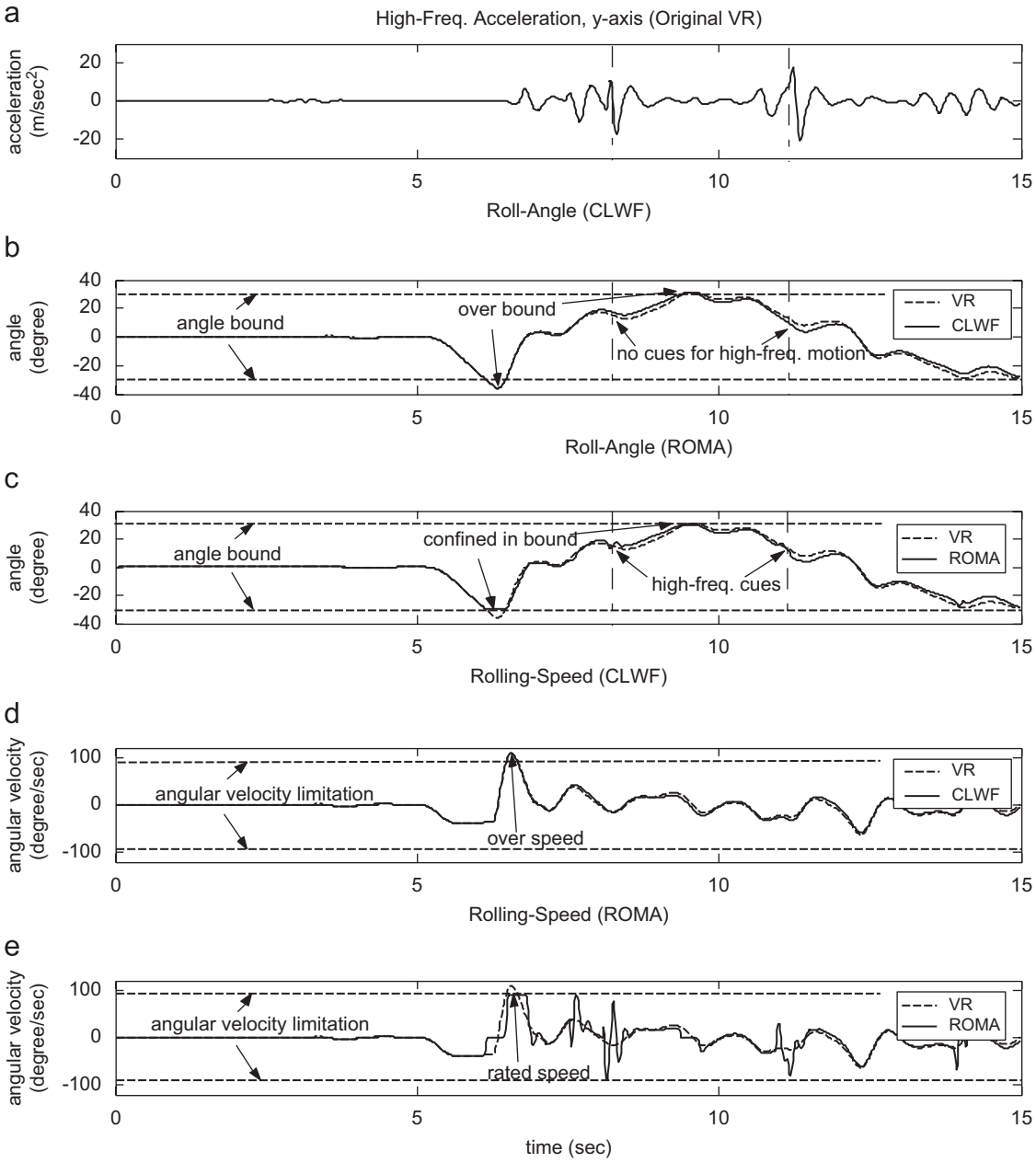

Fig. 16. Comparison of simulator output, lateral motion: (a) linear acceleration, y-axis (high-frequency), (b) roll angle, using classical washout filter (CLWF), (c) roll angle, using proposed algorithm (ROMA), (d) rolling speed, using CLWF, and (e) rolling speed, using proposed algorithm (ROMA).

Hence, one complex simulation was performed to test all motion cues simultaneously.

Various aspects of the experimental results are shown. The ROMA introduced in this paper is derived from the CLWF, which is depicted in Fig. 4. ROMA should perform similar sustained motions, or low-frequency linear motions, to CLWF. The sustained motion activates only the attitude and the residual tilt control, for which the calculation is mainly derived from the CLWF method.

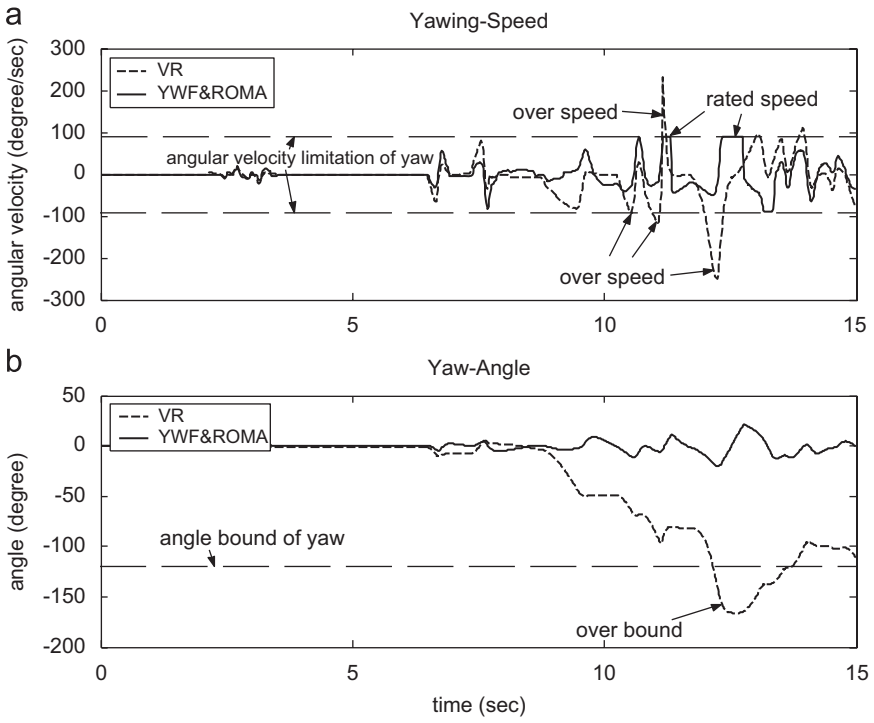

Fig. 17. Comparison of simulator output, yaw-motion: (a) yawing speed, using proposed algorithm (ROMA) and (b) yaw angle, using proposed algorithm (ROMA).

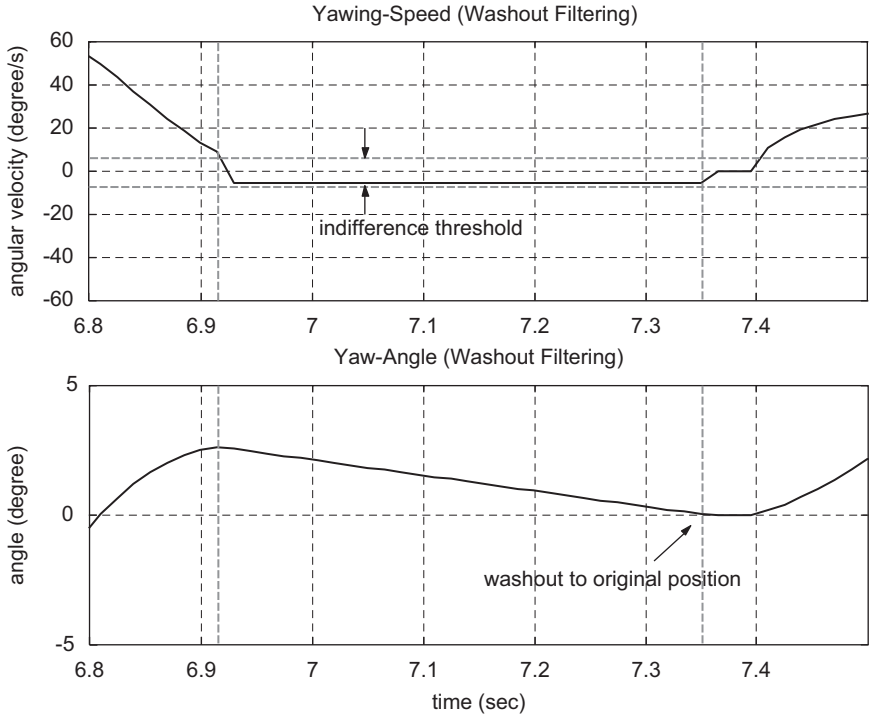

Fig. 18. Yawing speed and yaw angle after washout filtering (YWF).

There is merely difference when comparing the performances of sustained motion along the x and y axes between these two methods.

However, the CLWF is designed for the general 6-DOF motion simulator. The ROMA is designed for a 3-DOF flight simulator, specifically the rotational motion simulator with insufficient spatial DOF. The CLWF shows poor performance on the high-frequency linear motion when implemented on the rotational motion simulator.

Figs. 11 and 12 show the comparison of the high-frequency (onset) linear motion cues along the x and y axes. In this case, the CLWF generates no output to the rotation motion. The proposed algorithm ROMA eventually converts the onset linear motion to a rotation command based on Eqs. (14) and (20), and presents the onset linear motion on the motion simulator. Figs. 13 and 14 show the error between VR commands and actual linear acceleration by different motion-cueing algorithms. The data indicate that the error rises rapidly as the frequency of the linear motion increases when adopting the classical method.

Figs. 15 and 16 show the results of a mixture of sustained and onset motion. The ROMA could optimize the motion cueing without violating the inequality equation, and remained within to the mechanical bounds of a motion simulator. The CLWF failed to do so; therefore, the mechanical structure of the motion simulator can be damaged by CLWF. Nevertheless, these figures indicate that the onset motion cue can be generated by ROMA rather than CLWF.

Fig. 17 illustrates the effect of the yawing washout and ROMA. The washout motion continuously returns the cockpit to its home position when the indifference threshold is detected as in Eq. (54). The washout motion moving the cockpit back to its home position is performed with a velocity at the indifference threshold, as revealed in Fig. 18.

7. Conclusion

This study establishes a performance index of the motion performed by the motion simulator. This motion performance index is useful for planning motion control. By introducing the constrained optimization algorithm ROMA, the motion control can not only yield a precise cue to the pilot, but also avoid damage to the mechanical structure of the motion simulator. In addition to the motion optimization process, washout filters are applied to ensure dexterity of motion, and to prevent unexpected damage from loss of control of motions due to high CPU loading. Repeated tests were performed online, and indicate that the proposed motion-cueing strategy yields much more realistic motion than the classical technique with a 3-DOF motion simulator. The proposed motion-cueing strategy is generally applicable to all motion simulators with three rotational DOF.

Acknowledgments

The authors would like to thank the National Science Council of the Republic of China, Taiwan for financially supporting this research under Contract No. NSC_95-2218-E-009-003-.

Appendix A. Inverse kinematics of the polar-star mechanism

To solve the inverse kinematics problem, the Cartesian space is mapped to joint space. The X–Y–Z fixed angle frame is introduced with angles of yaw α, pitch β, and roll γ

$$\begin{aligned}
{}^{G}_{C}\mathbf{T} &= \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \\
&= {}^{G}_{C}\mathbf{R}_{XYZ}(\alpha,\beta,\gamma) \\
&= \begin{bmatrix} c\alpha c\beta & c\alpha s\beta s\gamma - s\alpha c\gamma & c\alpha s\beta c\gamma + s\alpha s\gamma \\ s\alpha c\beta & s\alpha s\beta s\gamma + c\alpha c\gamma & s\alpha s\beta c\gamma - c\alpha s\gamma \\ -s\beta & c\beta s\gamma & c\beta c\gamma \end{bmatrix} \\
&= {}^{G}_{C}\mathbf{R}_{Z'Y'Z'}(\alpha',\beta',\gamma') \\
&= \begin{bmatrix} c\alpha' c\beta' c\gamma' - s\alpha' s\gamma' & -c\alpha' c\beta' s\gamma' - s\alpha' c\gamma' & c\alpha' s\beta' \\ s\alpha' c\beta' c\gamma' + c\alpha' s\gamma' & -s\alpha' c\beta' s\gamma' + c\alpha' c\gamma' & s\alpha' s\beta' \\ -s\beta' c\gamma' & s\beta' s\gamma' & c\beta' \end{bmatrix}
\end{aligned} \tag{A.1}$$

then, if $\sin\beta' \neq 0$, that is, if $r_{33} \neq 1$. It follows that

$$\beta' = \tan^{-1}\left(\frac{\sqrt{r_{31}^2 + r_{32}^2}}{r_{33}}\right) \tag{A.2}$$

$$\alpha' = \tan^{-1}\left(\frac{r_{23}/s\beta'}{r_{13}/s\beta'}\right) \tag{A.3}$$

$$\gamma' = \tan^{-1}\left(\frac{r_{32}/s\beta'}{r_{31}/s\beta'}\right) \tag{A.4}$$

If $\beta' = 0$, that is, $r_{33} = 1$. Then a solution can be calculated to be

$$\beta' = 0 \tag{A.5}$$

$$\alpha' = 0 \tag{A.6}$$

$$\gamma' = \tan^{-1}\left(\frac{-r_{12}}{r_{11}}\right) \tag{A.7}$$

The following table shows the relationships between the Z–Y–Z Euler angles and the output angles of the motors:

	θ_1 (Motor#1 Output angle)	θ_1 (Motor#2 Output angle)	θ_3 (Motor#3 Output angle)
$R_{Z'}(\alpha')$	α'	α'	α'
$R_{Y'}(\beta')$	0	0	β'
$R_{Z'}(\gamma')$	γ'	0	0

where α', β', and γ' denote the Z–Y–Z Euler angles. The table above yields

$$\begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 1 \\ 1 & 0 & 0 \\ 1 & 1 & 0 \end{bmatrix} \begin{bmatrix} \alpha' \\ \beta' \\ \gamma' \end{bmatrix} \tag{A.8}$$

which implies

$$\begin{bmatrix} \alpha' \\ \beta' \\ \gamma' \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -1 & 1 \\ 1 & -1 & 0 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix} \tag{A.9}$$

where θ_1, θ_2, and θ_3 denote the output angles of motors 1, 2, and 3, respectively. Thus, the transfer matrix between the Z–Y–Z Euler angles and the output angles of motors is obtained.

Appendix B. Experimental setup, specification, and quantification

Software (Spec of processor: CPU: Pentium D, 3.20 GHz, RAM: 2.00 GB):

Specification	Description
Interrupt of motion control (ROMA) programming	10 ms
Interrupt of virtual reality (PFO) programming	30 ms
Calculation time of motion control (ROMA)	Less than 1 ms
CPU load	less than 1%(motion control)+50∼60% (*VR* rendering process)

Servo drive system:

Specification	Unit	Value
Rated output	kw	1.5
Rated speed	rpm	3000
Max. speed	rpm	5000
Rated voltage	V	3, 200 V
Rated current	A	9.6
Max. current	A	28.8
Rated torque	N m	4.78
Max. torque	N m	14.3
Rotor inertia	$kg\,m^2$	4.51×10^{-4}
Insulation class F	Class	F
Weight	kg	7
Ambient temp.	°C	−10∼+40

Mechanism system:

Specification	Unit	Value
Max. weight	Kg	200
Offset from pivot	M	0.50∼0.60 (depends on the position of pilot's head)
Max. pitch	deg	−30∼+30
Max roll	deg	−30∼+30
Max. yaw	deg	−120∼+120

References

Ariel, D., & Sivan, R. (1984). False cue reduction in moving flight simulators. *IEEE Transactions on Systems, Man and Cybernetics*, *SMC-14*(12), 665–671.

Bowles, R. L., Parrish, R. V., & Dieudonne, J. E. (1975). Coordinated adaptive washout for motion simulators. *Journal of Aircraft*, *12*(1), 44–50.

Chang, Y.-H., Chieng, W.-H., Liao, C.-S., & Jeng, S.-L. (2006). A novel master switching method for electronic cam control with special reference to multi-axis coordinated trajectory following. *Control Engineering Practice*, *14*(2), 107–120.

Chong, E. K.-P., & Żak, S. H. (2001). *An introduction to optimization*. A Wiley-Interscience Publication, John Wiley & Sons, Inc. pp. 397–410.

Conrad, B., & Schmidt, S. F. (1971). *A study of techniques for calculating motion drive signals for flight simulators*. NASA CR-114345.

Idan, M., & Sahar, D. (1996). Robust controller for a dynamic six degree of freedom flight simulator. *In AIAA Proceeding of conference on flight simulator technologies* (pp. 53–60).

Liao, C.-S., Huang, C.-F., & Chieng, W.-. H. (2004). A novel washout filter design for a six degree-of-freedom motion simulator. *JSME International Journal Series C*, *47*(2), 626–636.

Martin, E. A. (2000). Motion and force cueing, Part I: whole body motion. *Flight & ground simulation update* (pp. 5–18). Binghamton, NY: State University of New York.

Moshe, I., & Nahon, M. A. (1999). Offline comparison of classical and robust flight simulator motion control. *Journal of Guidance, Control, and Dynamics*, *22*(5), 702–709.

Nahon, M. A., & Reid, L. D. (1990). Simulator motion-drive algorithms: A designer's perspective. *Journal of Guidance, Control, and Dynamics*, *13*, 356–362.

Pouliot, N. A., Gosselin, C. M., & Nahon, M. A. (1998). Motion simulation capabilities of three-degree-of-freedom flight simulators. *Journal of Aircraft*, *35*(1), 9–17.

Reid, L. D., & Nahon, M. A. (1986). Flight simulator motion-based drive algorithm: Part 3 – pilot evaluations. *Technical report UTIAS report 319*, University of Toronto, Canada.

Reid, L. D., Nahon, M. A., & Kirdeikis, J. (1992). Adaptive simulator motion software with supervisory control. *Journal of Guidance, Control, and Dynamics*, *15*(2), 376–383.

Schmidt, S. F., & Bjorn, C. (1970). *Motion drive signals for piloted flight simulators*. Analytical mechanics associated, Technical report contract NAS2-4869.

Sinacori, J. B. (1973). A practical approach to motion simulation. In *Visual and motion simulation conference, AIAA paper*, Palo Alto, CA (pp. 73–931).

Sivan, R., Ish-shalom, J., & Huang, J. K. (1982). An Optimal control approach to the design of moving flight simulators. *IEEE Transactions on Systems, Man and Cybernetics*, *SMC-12*(6), 818–827.

Telban, R. J., & Cardullo, F. M. (2005). *Motion cueing algorithm development: Human-centered linear and nonlinear algorithms*. NASA CR-2005-213747, NASA Langley Research Center, Hampton, VA.

Wu, W., & Cardullo, F. M. (1997). Is there an optimum cueing algorithm? *AIAA modeling and simulation technologies conference*, New Orleans, LA (pp. 23–29).