

國立交通大學

機械工程研究所

博士論文

六自由度運動模擬器之電子凸輪式追蹤
運動控制策略

**A Motion Control Strategy Based on the Electronic Cam Tracking
for a Six Degree-of-Freedom Motion Simulator**

研究生：廖俊旭

指導老師：成維華 教授

中華民國九十三年五月

六自由度運動模擬器之電子凸輪式追蹤運動控制策略

**A Motion Control Strategy Based on the Electronic Cam Tracking
for a Six Degree-of-Freedom Motion Simulator**

研 究 生：廖 俊 旭 Student: Chung-Shu Liao

指 導 教 授：成 維 華 Advisor: Wei-Hua Chieng

國 立 交 通 大 學

機 械 工 程 研 究 所

博 士 論 文

A Dissertation

Submitted to Department of Mechanical Engineering

College of Engineering

National Chiao Tung University

In Partial Fulfillment of the Requirements

for the degree of Doctor of Philosophy

in

Mechanical Engineering

May 2004

Hsinchu, Taiwan, Republic of China

中 華 民 國 九 十 三 年 五 月

六自由度運動模擬器之電子凸輪式追蹤運動控制策略

學生：廖俊旭 指導老師：成維華 教授

國立交通大學機械工程研究所

摘 要

本研究的目的是，在於建立和分析一種針對六自由度運動模擬器之運動線索控制系統。文中首先介紹電子凸輪運動控制命令的產生方法；為即時地經由一系列的干擾抑制控制、主從追蹤控制與約束條件下之最佳化演算法。接著導入一種新的六自由度運動模擬器之沖刷濾波器的設計方法，此方法主要是期望工作空間被嚴格限制的運動模擬器，在保證驅動系統的強健性下，能夠在有限的空間中模擬無限的運動線索。再者，為了達成六自由度運動模擬器的軌跡追蹤精度，本研究提出一種可切換主控軸之電子凸輪式追蹤策略，應用於六自由度運動模擬器之控制系統，理論上將保證模擬器運動的軌跡不致偏移。最後，為了印證所提之運動控制策略的優異性，本研究整合了運動線索控制系統、汽車動力學系統、碰撞偵測系統、座艙操縱系統、音效與三維空間虛擬實境系統之軟硬體，以實現對六自由度運動模擬器之人機互動式的運動控制。

關鍵字：運動線索、電子凸輪、沖刷濾波器、可切換主控軸

A Motion Control Strategy Based on the Electronic Cam Tracking for a Six Degree-of-Freedom Motion Simulator

Student: Chung-Shu Liao Advisor: Wei-Hua Chieng

Institute of Mechanical Engineering National Chiao Tung University

Abstract

The purpose of this study is to establish and analyze a novel motion cuing control system of a six degree-of-freedom (DOF) motion simulator. This article first presents a method of generating the electronic cam motion, by way of a sequence of disturbances suppressed control, master-slaves tracking control and a constrained optimization algorithm in real-time. Next introduces a novel approach to designing the washout filter of the motion control for a six DOF motion simulator. The main focus of this approach is to make the motion cue feasible for use in a simulator with a restricted workspace, while ensuring the robustness of the driving system. Furthermore, for the purpose of trajectory tracking for a six DOF motion simulator, a novel master switching method for the electronic cam control is introduced. By applying the master switching method to the motion control system, the simulator's motion will be theoretically guaranteed not to deviate from the planned trajectory. Finally, in order to demonstrate the advantages of the proposed motion control strategy, the software of the proposed motion cuing system had been integrated with vehicle dynamics system, collision detection system, cabin operating system, sound effects and three-dimensional virtual reality (VR) system. Then, combine the software with the hardware to perform the human-machine interactive motion control for a six DOF motion simulator.

Keywords: motion cuing, electronic cam, washout filter, master switching

誌 謝

本論文的完成，首先要感謝我的指導老師 成維華 教授，在兩年碩士班與近四年博士班的學生生涯中，不厭其煩地給予我啟發與教誨，使我不但在學術的研究中不斷精進，亦在理論與實作之間獲得許多獨到的心得，甚且在待人處事方面亦能漸趨圓融，這種無形智慧的累積，使得即將再度踏入社會的我，自信能排除萬難、解決所有難題。

感謝所有口試委員的精闢見解與指教，您的意見使本論文的邏輯性與可閱讀性相對地提高許多。

古語有云：「父母在、不遠遊」，四年了！感謝爸、媽的體諒與認同，一切盡在不言中，不肖兒衷心地將此學位的榮耀獻給最親愛的爸、媽。

在此也將這份喜悅與妻子琳、兒子唯辰與唯廷一起分享；在我毅然決然轉變人生跑道，放棄一個一般認為不錯的事業逕而攻讀博士學位時，幾乎所有親友均感訝異甚至反對，琳總是尊重我的理想、支持我的決定，並且無怨無悔地照顧與教導兒子，使我能在無任何後顧之憂的環境下順利取得博士學位。附帶一提的是，在研究這條路上，兒子們的笑聲總能舒緩我時而煩躁的心情，家庭的溫暖與安慰給了我最大的動力。

特別感謝學弟張仰宏與吳秉霖對汽車飛躍力學、碰撞力學與力回饋方向盤的理論推導與實現，十分感謝蘇俊銘同學對 3D 虛擬實境圖像控制程式的整合，感謝學弟童永成完成音效程式的纂寫，亦要感謝江建勳學長與領航動感科技公司提供的模擬器平台；由於大家的努力不懈，使得系統整合後的運動控制，能真正實現人機互動的效果。

最後要感謝岳父、母與所有姊姊、姐夫以及親朋好友，平時不吝給我關懷與鼓勵，在此誠心地說聲：「謝謝您們！」

Contents

摘要	i
Abstract	ii
誌謝	iii
Contents	iv
List of Figures	vii
List of Tables	xi
Nomenclature	xiii
Chapter 1 Introduction	1
Chapter 2 Electronic Cam Motion Generation with Special Reference to Constrained Velocity, Acceleration and Jerk	10
2.1 Prerequisite of Electronic Cam (ECAM) Tracking Control	10
2.1.1 External disturbance estimator	11
2.1.2 Suppressing external disturbance	12
2.2 Electronic Gearing (E-Gearing) Process	13
2.3 Predicting the Position of Slaves	16
2.4 Simulation and Experimental Results	21
2.4.1 Simulation of disturbance estimator	21
2.4.2 Experimental results for tracking performance of the electronic gearing process	22
2.4.3 Performance of the electronic cam process	23
2.4.4 Computational load on the CPU of the proposed ECAM tracking control	24
Chapter 3 A Novel Washout Filter Design for a Six Degree-of-Freedom Motion Simulator	25
3.1 Inverse kinematics	25
3.2 Preview of Classical Linear Washout Filter (CLWF)	26

3.3	Adjustable Scaling Filter (ASF) and Yawing Washout Filter (YWF)	27
3.4	Dead Zone Washout Filter (DZWF)	29
3.5	Structure of the Adaptive Washout Filter (AWF)	32
3.5.1	Transformation (S to J)	32
3.5.2	Washout function	32
3.5.3	Self-Tuning Process (Shown in Figure 3.3(b))	35
3.6	Performance Index (<i>PI</i>)	37
3.7	Experimental Results and Comparison	40
Chapter 4	A Novel Master Switching Method for Electronic Cam Control with Special Reference to Multi-Axis Coordinated Trajectory Following	43
4.1	Method of Building Cam Profiles (Master-slaves Trajectories)	43
4.1.1	Polynomial curve-fitting	43
4.1.2	Poly-line Curve-Fitting	46
4.2	Infinity Norm of the Master Switching ECAM Controller	48
4.3	Applying the Proposed Control Scheme to a Six DOF Motion Simulator	49
4.3.1	Inverse Kinematics	49
4.3.2	Jacobian formulation of simulator SP-120	49
4.3.3	Calculate the loaded torque of each joint using Jacobian matrix	52
4.4	Analysis of Stability and Robustness	52
4.5	Numerical Method for the Forward Kinematics of Six DOF Flight Simulator	56
4.6	Experimental Results and Comparisons	57
Chapter 5	System Integration	60
5.1	Dynamics of Vehicle	60
5.2	Collision Detection	60

5.3	VR Programming and Sound Effects	61
5.4	Motion Control System	61
5.5	Cabin Operating System and Force Feedback Steering System	61
Chapter 6	Conclusion	63
Appendix		65
References		70
	本論文作者已發表的著作	121



List of Figures

- Figure 1.1 Flow chart of the system integration for the human-machine interactive motion
- Figure 1.2 The motion control system for a six DOF motion simulator
- Figure 2.1(a) Block diagram of the proposed ECAM system for mathematical representation
- Figure 2.1(b) Block diagram of the proposed disturbance estimator for one practical embodiment
- Figure 2.2 The estimation of external disturbance
- Figure 2.3 The external disturbance estimator and external disturbance eliminated control
- Figure 2.4 Temporal relations between the two proposed procedures
- Figure 2.5 Cubic B-spline curve, $r_{k+1,j}(u)$, $j = 1$ to 4, and its control points, $p_{k+1,0} \sim p_{k+1,6}$
- Figure 2.6 Flow chart of the optimal solution process
- Figure 2.7 The location of the optimal position command, $p_{k+1,5}^*$, for all different cases
- Figure 2.8(a) Simulated angular velocity of the master
- Figure 2.8(b) Simulated angular velocity of the master using disturbance estimator feedback control (zoom in)
- Figure 2.9 The errors between the fed torque (τ_L) and the estimated torque ($\hat{\tau}_L$) with respect to various time constants (ζ)
- Figure 2.10(a) The tracking error of the master's position for the zero-order interpolation method
- Figure 2.10(b) The tracking error of the master for the third-order polynomial tracking method
- Figure 2.10(c) The tracking error of the master for the fourth-order polynomial tracking method
- Figure 2.10(d) The tracking error of the master for the fifth-order polynomial tracking method
- Figure 2.11 The piecewise tracking trajectory of the electronic cam motion; (a) the actual master's position in real-time (b) the reference trajectory corresponding to the electronic cam motion (c) the actual cam trajectory in real-time. Note that the unit "counts" means the encoder's pulse counts. The resolutions of the master's encoder and the slave's encoder are 2000 counts/revolution and 10000 counts/revolution, respectively.

- Figure 2.12 (a) Cam profile error with the zero-order(conventional) tracking method (The maximum travel distance: 200000 encoder's counts)
- Figure 2.12(b) Cam profile error with the third-order polynomial tracking method (The maximum travel distance: 200000 encoder's counts)
- Figure 2.12(c) Cam profile error with the fourth-order polynomial tracking method (The maximum travel distance: 200000 encoder's counts)
- Figure 2.12(d) Cam profile error with the fifth-order polynomial tracking method (The maximum travel distance: 200000 encoder's counts)
- Figure 2.13(a) The tracking result of the slave velocity, acceleration and jerk purely based on the Lagrange polynomial curve-fitting with no optimization
- Figure 2.13(b) The result of the slave velocity, acceleration and jerk applying the optimization
- Figure 3.1 Classical linear washout filter (referred to Nahon and Reid, 1990)
- Figure 3.2 Block diagram of motion-cueing system, using the proposed control strategy
- Figure 3.3(a) AWF structure (to be continued)
- Figure 3.3(b) AWF structure (to continue)
- Figure 3.4 Prototype SP-120
- Figure 3.5 Kinematical skeleton of simulator platform SP-120
- Figure 3.6 Restoration process in the dead zone washout filter
- Figure 3.7 Washout trajectory along the i -axis, where the subscript i represents the three mutually orthogonal axes (x-, y-, and z-axis)
- Figure 3.8 Segmental data concerning linear accelerations along the x-axis
- Figure 3.9 Segmental errors of linear accelerations along the x-axis, between the static scaled VR dynamic output ($a_{s,x}$) and the simulator's output (a_x) using the (a) CLWF and (b) the proposed strategies
- Figure 3.10 Three segmental data concerning Euler's angular velocities
- Figure 3.11 Segmental errors of Euler's angular velocities between the static scaled VR dynamic output ($\omega_{s,x}$) and the simulator's output (ω_x) using (a) the CLWF and (b) the proposed strategies
- Figure 3.12 Three segmental data concerning linear accelerations along the x-axis

- Figure 3.13 Segmental errors of linear accelerations along the x-axis between the static scaled VR dynamic output ($a_{s,x}$) and the simulator's output (a_x) using (a) the optimal pair of weighting parameters (1, 0) and (b) the pair of weighting parameters (1, 0.5)
- Figure 3.14 Three segmental data concerning Euler's angular velocities
- Figure 3.15 Segmental errors of Euler's angular velocities between the static scaled VR dynamic output ($\omega_{s,x}$) and the simulator's output (ω_x) using (a) the optimal pair of weighting parameters (1, 0) and (b) the pair of weighting parameters (1, 0.5)
- Figure 4.1 Master switching method for q axes ECAM control
- Figure 4.2 Cam profile trajectory established using the polynomial curve-fitting method
- Figure 4.3 Cam profile trajectory established using the poly-line curve-fitting method
- Figure 4.4 Conditions on dual solutions using the polynomial curve-fitting method
- Figure 4.5 Equivalent model of slider
- Figure 4.6 Simplified control system's block diagram of control system of each slider of simulator SP-120
- Figure 4.7 Upper linear fractional transformation with $\Delta_1 = \delta_m$, $\Delta_2 = \delta_c$ and $|\delta_m| < 1$, $|\delta_c| < 1$
- Figure 4.8 Singular value frequency responses, $\sigma(G_1(j\omega))$, for various proportional gains, K_a
- Figure 4.9 Upper bounds of structured singular values, $\mu_\Delta(G_1(j\omega))$, for various proportional gains, K_a
- Figure 4.10 The piecewise ground earthquake signal involves only the translation
- Figure 4.11 Power spectrum density of the ground earthquake signal at various frequencies
- Figure 4.12 Comparison of Euler's piecewise roll angle errors
- Figure 4.13 Comparison of Euler's piecewise pitch angle errors
- Figure 4.14 Comparison of Euler's piecewise yaw angle errors
- Figure 5.1 Flow chart of the system integration of the human-machine interactive motion
- Figure 5.2 Operating mode of the car dynamics

Figure 5.3 Colliding mode of the car dynamics

Figure 5.4 Hopping mode of the car dynamics



List of Tables

- Table 2.1 An example of cam profile table, both sets of data are scaled by their largest travel distance of one cam cycle.
- Table 2.2 Experimental specifications of parameters for the ECAM control, last three data are scaled by their largest travel distance ($40\pi \text{ rad}$) of one cam cycle.
- Table 2.3 The maximum tracking error of the master's position for the N^{th} order polynomial tracking control ($N = 0 \sim 5$)
- Table 2.4 An experimental example for the maximum tracking errors of the slave's position in the encoder's counts for the N^{th} order polynomial master tracking control, the maximum travel distance is 200000 encoder's counts (equivalent to $40\pi \text{ rad}$)
- Table 2.5 An experimental example for the maximum slave velocity, acceleration and jerk based purely on the Lagrange polynomial curve-fitting and applying the optimization algorithm to the cubic B-spline curve-fitting process
- Table 3.1 Capabilities of several motion simulators
- Table 3.2 Comparison between the CLWF and the proposed washout filter in terms of the magnitudes of $RMS(E_a)$, $RMS(E_\omega)$ and PI using various static scaling factors (s_a, s_ω), used successively to scale the linear accelerations and the angular velocities of VR dynamic output
- Table 3.3 Magnitudes of $RMS(E_a)$, $RMS(E_\omega)$ and PI using various pairs of adaptive scaling factors ($\lambda_a, \lambda_\omega$); the static scaling factors (s_a, s_ω) are set to (0.8, 0.7) (which will be later used to scale the linear accelerations and Euler's angular velocities of VR dynamic output)
- Table 4.1 Maximum singular values of $\|G_1(j\omega)\|_\infty$, maximum structured singular values of $\sup_{\omega \in R} \mu_\Delta(G_1(j\omega))$ and bandwidth of control system for various proportional gains, K_a ; the upper bound, m_H , of the nominal mass is set to 250 kg
- Table 4.2 Critical upper bounds of the nominal mass for various proportional gains, K_a

Table 4.3 Root mean square (RMS) errors of Euler angles, obtained using the proposed master switching method and the conventional method for ECAM control Table executed on the simulator SP-120



Nomenclature

$a_{threshold}$	= indifference threshold for acceleration
\mathbf{a}	= acceleration received from the output of self-tuning process
\mathbf{a}_i	= the i th component of acceleration vector \mathbf{a}
\mathbf{a}_{AA}	= specific acceleration of aircraft
\mathbf{a}_{HP}	= acceleration received from the output of high-pass filter
$\mathbf{a}_{HP, i}$	= the i th component of acceleration vector \mathbf{a}_{HP}
\mathbf{a}_s	= acceleration obtained from scaling \mathbf{a}_{VR}
\mathbf{a}_{ref}	= acceleration received from the output of dead zone washout filter
$\mathbf{a}_{ref, i}$	= the i th component of acceleration vector \mathbf{a}_{ref}
\mathbf{a}_{res}	= restoring acceleration in the dead zone washout filter
$\mathbf{a}_{res, i}$	= the i th component of acceleration vector \mathbf{a}_{res}
\mathbf{a}_{VR}	= acceleration obtained by summing gravity (\mathbf{g}_1) and applied force
$\mathbf{a}_{VR, i}$	= the i th component of acceleration vector \mathbf{a}_{VR}
$\mathbf{a}_1(\square), \mathbf{a}_2(\square)$	= acceleration function in adaptive washout filter
$\mathbf{a}_{2, i}(\square)$	= the i th component of acceleration vector $\mathbf{a}_2(\square)$
E_c	= input voltage of the servo-motor
\mathbf{G}	= mass center of the cockpit
g	= acceleration due to gravity

\mathbf{g}_I	= gravity vector in the inertia reference frame
HP filter	= high-pass filter
LP filter	= low-pass filter
L	= length of linkage
\mathbf{p}_i	= position of i^{th} slider ball joint ($i = 1$ to 6)
\mathbf{q}_i	= position of i^{th} ball joint ($i = 1$ to 6)
${}^O \mathbf{q}_i$	= coordinates of \mathbf{q}_i in the frame X-Y-Z-O
${}^G \mathbf{q}_i$	= coordinates of \mathbf{q}_i in the frame x-y-z-G
${}^{S_i} \mathbf{q}_i$	= coordinates of \mathbf{q}_i in the frame $x_i - y_i - z_i - S_i$
${}^{S_i} \mathbf{p}_i = {}^{S_i} [p_{xi} \ p_{yi} \ p_{zi}]^T$	= coordinates of \mathbf{p}_i in the frame $x_i - y_i - z_i - S_i$
${}^O_G R(\alpha, \beta, \gamma)$	= rotational transformation matrix from x-y-z-G to X-Y-Z-O
${}^S_i O R$	= constant rotational transformation matrix from X-Y-Z-O to $x_i - y_i - z_i - S_i$ ($i = 1, 3, 5$)
$RMS(\bullet)$	= root mean square of \bullet
s	= Laplace operator
s_a	= static scaling factor used to scale the linear accelerations
s_ω	= static scaling factor used to scale the Euler angular velocities
s_p	= lead screw pitch
(S to I)	= transform from simulator to inertia reference frame

(S to J)	= transformation of coordinates from x-y-z-G to $x_i - y_i - z_i - S_i$
TF	= transform from angular velocity to Euler angle rates
T	= system sampling time
u	= force applied to slider
X-Y-Z-O	= inertial coordinate system
${}^o G$	= coordinates of G relative to the X-Y-Z-O frame
x-y-z-G	= cockpit coordinate system
$x_i - y_i - z_i - S_i$	= joint coordinate system ($i = 1, 3, 5$) used to represent the positions of slider ball joints \mathbf{p}_i and \mathbf{p}_{i+1}
ω_{indiff}	= indifference threshold for angular speed
ω	= angular velocity received from the output of self-tuning process
ω_{AA}	= specific angular velocity of aircraft
ω_{HP}	= angular velocity received from the output of high-pass filter
ω_s	= scaled Euler's angular velocity ω_{VR}
ω_{ref}	= angular velocity received from the output of adjustable scaling filter and yawing washout filter
ω_{VR}	= Euler's angular velocity
α	= roll angle of the platform (upper plate)
β	= pitch angle of the platform (upper plate)

γ = yaw angle of the platform (upper plate)

$\boldsymbol{\varphi} = [\alpha \ \beta \ \gamma] = [\phi_x \ \phi_y \ \phi_z]$ = Euler angle in the x-y-z-G frame

τ = output torque of AC servo-motor

θ = output angle of motor's shaft

$\dot{\theta}$ = angular velocity of motor's shaft



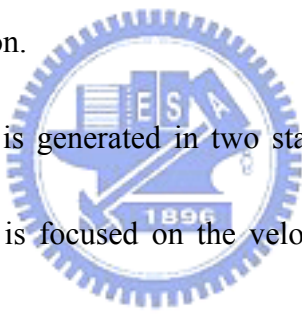
Chapter 1 Introduction

Recently, the applications of six degree-of-freedom (DOF) motion simulators have been widely developed, especially in flight simulation and vehicle simulation. The well-known examples are the ride films that generally use the motion identification method to perform the navigated motion. But there have been rarely developed for the human-machine interactive motion, especially in the simulators of small workspace. The main purpose of this dissertation is to study a motion control strategy to perform the human-machine interactive motion in a six DOF motion simulator. The system of human-machine interactive motion includes the following subsystems: cabin operating system, vehicle dynamics system, collision detection system, motion control system, sound effects and virtual reality system. Figure 1.1 shows the relations of these subsystems. This dissertation will be focused solely on the motion control scheme.

As shown in Fig 1.2, the motion control system involves motion cuing control and trajectory tracking control in a six DOF motion simulator. Motion cuing control provides a strategy of motion trajectory generation such that the motion cue can be managed and displayed in a limited workspace. Another significant concern of a six DOF motion simulator is the trajectory tracking precision rather than the positioning accurate. This study proposes a novel trajectory tracking control algorithm to guarantee that the simulator's motion always follows the trajectory of motion planning. In order to guarantee the correct motion cue, i.e.,

the feeling of linear acceleration and angular velocity, a trajectory tracking control using the master-slaves control scheme based on the electronic cam (ECAM) tracking control structure is introduced in this dissertation.

In this dissertation, chapter 2 presents an ECAM motion generation scheme with special reference to constrained velocity, acceleration and jerk; chapter 3 proposes a novel washout filter design for a six DOF motion simulator; chapter 4 uses the ECAM tracking control concepts, building a novel master switching method for ECAM control with special reference to multi-axis coordinated trajectory following; chapter 5 describes the system integration; chapter 6 concludes this dissertation.



In this study, ECAM motion is generated in two stages, the first of which is a typical electronic gearing process which is focused on the velocity tracking control of the master motor. Steven [1] specified a tracking control electronic gearing system called an “optimal feed-forward tracking controller”, concerned primarily with the design of the slave controller. However, he did not consider the output properties of the master motor, including the measurement noise, periodic errors and external harmonic disturbances. In practice, the measurement noise or the external disturbance must be controlled and eliminated by modeling the disturbances, before applying tracking control to estimate the master position. This study proposes the used of a disturbance estimator [2] to suppress the external disturbance. The design of this disturbance estimator is practical and easy to implement.

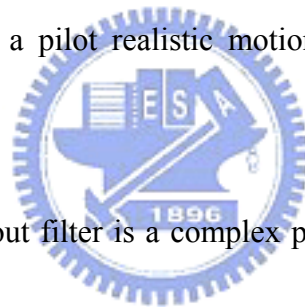
This approach to obtaining highly precise estimate of the master position involves N^{th} order polynomial tracking. For example, the fifth order polynomial estimate is more precise than the third order polynomial estimate by a factor of about 10000. The advantage of the proposed tracking method is that the low-frequency harmonic disturbances of a loaded master are very precisely estimated. Such nominal harmonic disturbances are observed in many industrial applications [3]. In practice, the frequencies of the external disturbances are expected to be far below the Nyquist frequency [4] of the real-time system.

ECAM motion regulates the slave motion to follow a predetermined trajectory, which is a function of the position of the master axis [5, 6]. A cam trajectory is generally specified by a cam profile table, which lists a set of reciprocal coordinates. Chen [7] applied B-spline [8, 9] and polynomial curve-fitting methods to generate a smooth cam profile curve function. Kim and Tsao [10] developed an electrohydraulic servo actuator for use in electronic cam motion generation, and obtained improved performance. However, some original performance limits, including velocity, acceleration or jerk constraints must be considered because motors have the lower loaded capacity relative to the hydraulic actuators. For example, for a highly precise machining tool, chattering must be avoided, so jerking in motion must be reduced.

This study proposes an optimization algorithm [11] to prevent extremely high velocities, accelerations or jerks, yielding smooth motion of the slave motor without loss of precision. The proposed tracking method presented here was experimentally verified using a real-time

program to realize the ECAM control system. The master's system uses a disturbance estimator to eliminate external disturbances. This estimator is the prerequisite for the N^{th} order polynomial tracking control. Lagrange polynomial [9] curve-fitting, cubic B-spline [9] interpolation and a constrained optimization algorithm are used to determine the position of the slaves. Consequently, a tradeoff may exist between precision and constraints, which are imposed in given order of priority.

The purpose of the washout filter is to transform trajectories generated by a dynamic model of virtual reality (VR), which incorporates very large displacements, into driving system commands that can give a pilot realistic motion cues while remaining within the simulator's limited workspace.



Designing an efficient washout filter is a complex problem. These filters are first of all complex control systems whose robustness and stability must be ensured to prevent mechanical damage to the simulator. Furthermore, designs of washout filters must take into account the spatial-disorientation of the pilot making a “realistic” simulation hard to define. The problem's complexity derives from human factors and the human-machine interaction. Many schemes have been presented in the last 20 years. Classical washout filters were developed first [12]-[14], followed by adaptive algorithms [15, 16], optimal control filters [17, 18], hybrid classical-adaptive filters [19] and robust filters [20, 21]. Importantly, even though these studies extensively address applications in simulators with relatively large workspaces,

the performance of various techniques for simulating specific VR motion in a motion simulator with a restricted workspace has rarely been discussed.

This study presents a novel washout filter design, that consists of a classical linear washout filter (CLWF), an adjustable scaling filter (ASF), a yawing washout filter (YWF), a dead zone washout filter (DZWF) and an adaptive washout filter (AWF). The CLWF separates the motion cues into high (“onset”) and low (“sustained”) frequency components so that cues can be managed and displayed within the physical confines of a given platform system. However, for motion simulators with severely restricted workspaces, even if the cutoff frequencies are properly selected [22], the position of cockpit may still exceed the platform’s workspace during a given motion because the linear accelerations and the angular velocities are mutually independent in the Cartesian coordinate system, but are coupled after applying inverse kinematics to every independent joint of the motion simulator. Moreover, the constraints on the driving system limit the performance of the motion simulator, such as the saturation for the driving current. Accordingly, the CLWF with appropriate cutoff frequencies is not always suitable for many practical platforms, especially those simulators with smaller workspace, but it remains useful in reducing the probability of leaving the limited workspace. The ASF dynamically tunes the cockpit’s angular velocities instead of the purely static scaling. Sometimes, the magnitude of linear acceleration is lower than the human sensible threshold [23]-[25], and the DZWF utilizes this moment to drive the cab stealthily back to its home

position by accelerating it under the indifference threshold [23]-[25]. An AWF that includes the coordinate transformation from simulator to joints (S to J), washout function and self-tuning process, greatly improves the motive performance for strictly confined simulators.

Two cost functions are defined to determine the performance index (PI) of VR motion. The PI quantifies the realism of the motion and is improved by inducing an empirical rule while adaptive scaling factors in the self-tuning process are tuned offline using numerical induction. Additionally, real-time software has been developed to implement the proposed criteria online in an automotive simulation for a six degree-of-freedom (DOF) motion simulator. Comparing the proposed washout filter with the CLWF shows that the experimental results indicate that the former is much more adaptive and realistic, especially for the motion simulator with a small workspace.



The important issue of the robust stability of a six DOF motion simulator concerns its six axes cross-coupled behavior: each axis pulls and drags every other such that the most heavily loaded axis may act unexpectedly; that is, the actual trajectory of the cockpit may be unexpected. This phenomenon follows from inconsistent tracking of the planned trajectory and may cause the cockpit of the motion simulator to leave its nominal workspace. Thus, a trajectory tracking controller is urgently required. Numerous papers [26]-[28] exist around the parallel robots [29], which were focused on the dynamics analysis. Several articles have also referred to the design of controllers of a six DOF motion simulators. Chung, Chang and Lin

[30] referred a fuzzy control system for a six DOF simulator and considered the hydraulic actuator system. Werner [31] introduced a robust tracking control for an unstable, linearized plant which was linearized. Plummer [32] described a nonlinear multi-variable controller for a flight simulator. The procedure for completely designing a robust controller of a nonlinear system consists of finding the nominal controlled plant [10, 33, 34, 35], which is very complicated and impractical; thus, the dynamics of the nonlinear control system must be linearized and simplified. Furthermore, to make motion planning of parallel robots, that are needed to calculate inverse and direct kinematical models. If the first one is often easy to solve, the second one is very complex for a six DOF parallel robot [36]. Usually, the selection is to make motion planning, either in joint space or in operational space [37]. This study introduces a master-slaves control strategy – master switching ECAM tracking control, based on the motion planning. The master switching ECAM control scheme presents a simplified control system for a six DOF motion simulator.

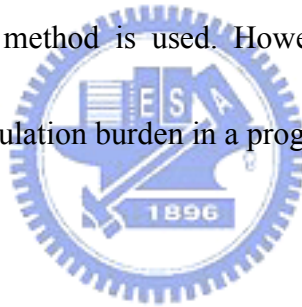
ECAM tracking is applied to a multi-axes motion control system mainly to enable the slaves to follow consistently trajectories obtained from the predicted sets of reciprocal points of master and slaves. When the master receives a position command, it will or will not be driven to the desired position, and the slaves will be moved into new positions by following the predicted cam profiles. However, in a fixed master ECAM system, the heavily loaded slaves may follow a lightly loaded master, and then the slaves may lose tracking precision as

it reaches its current (force) limit. Kim and Tsao [10] developed an electrohydraulic servo actuator for use in electronic cam motion generation, addressing the robust performance control for the fixed slave, an electrohydraulic servo actuator. Steven [1] specified an optimal feed-forward tracking controller, primarily associated with the fixed slave controller design. Each of their control schemes was demonstrated to satisfy the demands of precision and robustness, but to be valid only for its particular application.

In the master switching control scheme, the most heavily loaded axis must be predetermined before anticipative motion begins: this axis will be treated as the master and the other axes as the slaves. The master may be switched between different types of motion from time to time, to exchange the master and one of the slaves in the subsequent action. After the master is instantaneously determined, the next important task is to build ECAM profiles from the demanded ECAM tables. Two curve-fitting methods [11] are proposed to establish piecewise ECAM profile. One is the polynomial curve-fitting method which is suggested for use in cases of low frequency motion. Simulations indicate that the polynomial curve-fitting method [9, 38] performs well, if the frequencies of the active body are less than one-tenth of half the system's sampling frequency (Nyquist frequency). Restated, this method is favorable if and only if the trajectory of motion is very smooth from the viewpoint of the Nyquist frequency. The second method is the poly-line curve-fitting method, as shown in Fig. 3, which is more appropriate for higher frequency motion.

A simplified dynamics model of the simulator for analysis is proposed to model the structured perturbation with parametric uncertainties. The well-known μ tool [39] is used to analyze the robust performance of the original control system, and then to demonstrate that it is more robust and stable after the proposed control scheme is applied to the system.

Restated, real-time software was developed to implement the PC-based master switching ECAM control scheme used in the SP-120 motion simulator. Experimental results show the advantage of the proposed tracking accuracy. However, experimental analysis has also revealed that a shorter system sampling time yields more accurate tracking control, especially when the poly-line curve-fitting method is used. However, a tradeoff exists between the system sampling time and the calculation burden in a programming cycle.



Chapter 2 Electronic Cam Motion Generation with Special Reference to Constrained Velocity, Acceleration and Jerk

Electronic cam motion involves velocity tracking control of the master motor and trajectory generation of the slave motor. Special concerns such as the limits of the velocity, acceleration and jerk are beyond the considerations in the conventional electronic cam motion control. This study proposes the curve-fitting of a Lagrange polynomial to the cam profile, based on trajectory optimization by cubic B-spline interpolation. The proposed algorithms may yield a higher tracking precision than conventional master-slaves control method does, providing an optimization problem is concerned. The optimization problem contains three dynamic constraints including velocity, acceleration and jerk of the motor system.



2.1 Prerequisite of Electronic Cam (ECAM) Tracking Control

Electronic cam (ECAM) control is a well-known master-slaves system. Figure 2.1(a) schematically depicts a block diagram of the proposed ECAM control system for mathematical representation. The variables and symbols in the figure are defined in the following sections. In Fig. 2.1(a), the motion of the slave motor clearly depends on the estimated slave position command, p_{k+1} , which is generated by cubic B-spline interpolation, combined with an optimization algorithm. Such optimization is performed to meet the demands of limited performance - the constraints of velocity, acceleration and jerk. The method of cubic B-spline curve-fitting is based on substituting the estimated master position

into the cam trajectory. It is established using Lagrange's interpolation formula to generate a Lagrange polynomial curve. However, the predicted master position is estimated by the electronic gearing (E-gearing) process.

2.1.1 External disturbance estimator

External disturbances (or loads) applied to the master may directly impact the efficiency of E-gearing. Therefore, disturbances must be suppressed. A disturbance estimator, depicted in Fig. 2.1(a), is used to estimate and suppress the external loads of the master motor. However, Fig. 2.1(b) is one practical embodiment for the proposed disturbance suppressed control.

In Fig. 2.1(b), the external load (τ_L) is estimated from the input current (i_a) and the angular velocity (ω), where K_a , \hat{L}_f , \hat{R}_f , \hat{K} , \hat{J} and \hat{B} represent the nominal back electromotive force constant, the nominal armature current inductance, the nominal armature current resistance, the nominal torque constant, the nominal moment of inertia and the nominal damping coefficient of the motor, respectively. Furthermore, V_{ref} , L_f , R_f , K , J and B represent the reference voltage input, the actual armature current inductance, the actual armature current resistance, the actual (uncertain) torque constant, the actual (uncertain) moment of inertia and the actual (uncertain) damping coefficient of the motor, respectively.

Consider the dynamics of a DC motor:

$$\begin{aligned} \hat{J}\dot{\omega} + \hat{B}\omega + \tau_L &= \hat{K} \cdot i_a \\ \Rightarrow \tau_L &= \hat{K} \cdot i_a - \hat{J}\dot{\omega} - \hat{B}\omega \end{aligned} \tag{2.1}$$

According to Fig. 2.2(a), this estimator cannot be realized because of the differential term ($\hat{J}s$) of angular velocity. The estimator depicted in Fig. 2.2(a) is also very numerically sensitive to the measurement noise because it yields high gains in the high-frequency field. Accordingly, a first-order low pass filter is used to estimate the disturbance, $\hat{\tau}_L$, as shown in Fig. 2.2(b),

where

$$\hat{\tau}_L = \frac{1}{(\mathfrak{T}s + 1)} \tau_L \quad (2.2)$$

and

$$\frac{\hat{\tau}_L}{\omega} = \frac{-(\hat{J}s + B)}{\mathfrak{T}s + 1} = -\frac{\hat{J}}{\mathfrak{T}} + \frac{-\hat{B} + \hat{J} / \mathfrak{T}}{\mathfrak{T}s + 1} \quad (2.3)$$

Rearranging this external disturbance estimator in Fig. 2.3 yields no differential term. The estimated disturbance ($\hat{\tau}_L$) is then fed back to the current loop, and the external disturbance is suppressed. In practice, due to the current-loop's bandwidth is much larger than the speed-loop's bandwidth, the electric dynamics ($1/(L_f s + R_f)$) can be neglected in Fig. 2.1(b) for analysis.

2.1.2 Suppressing external disturbance

According to Fig. 2.3, the pole of the disturbance estimator equals the pole of the low pass filter, specified by Eq. (2.2). Thus, the estimated value for low delay time is obtained by reducing the time constant (\mathfrak{T}) of the low pass filter. However, the small time constant trades off the estimated precision and robustness because it suffers more on measurement noise and modeling uncertainty.

Figure 2.2(b) is equivalently transformed to Fig. 2.2(c) to elucidate the effect of the external disturbance (τ_L). According to Fig. 2.2(c), the effect of τ_L is that of passing τ_L through the filter $\mathfrak{T}s / (\mathfrak{T}s + 1)$. Accordingly, the external disturbance can be suppressed when the disturbance frequency is less than $1/\mathfrak{T}$ rad/s. Thus, the smaller time constant \mathfrak{T} yields better efficiency for suppressing high-frequency disturbances. However, a trade-off exists between estimated precision and robustness, as described in the above paragraph.

Due to considerations of robustness, the measurement noise and the modeling uncertainty must also be considered in determining the time constant \mathfrak{T} . Appendix A discusses the sensitivities, $S_K^{G_c}$, $S_J^{G_c}$ and $S_B^{G_c}$ to the uncertainties, where $S_K^{G_c}$, $S_J^{G_c}$ and $S_B^{G_c}$ are the sensitivities of the current loop transfer function G_c to the uncertain parameters K , J and B , respectively. Moreover, the effect of measurement noise is discussed with reference to a numerical simulation in Section 2.4.1.

2.2 Electronic Gearing (E-Gearing) Process

The electronic gearing (E-Gearing) differentiating itself from the mechanical gearing for that the E-Gearing system employed only electronic means to achieve the constant input/output velocity ratio. It is assumed that the output velocity control system is stiff and the main issue for the electronic gearing is to predict the future master velocity from its past. The velocity of the slave (output) motor is controlled according to the velocity of the master (input)

motor.

The velocity of the master motor varies when loads or other external disturbances are applied. Therefore, the master velocity is not usually constant and may exhibit harmonics. Even the amplitudes of the harmonic velocity are greatly reduced by using the proposed disturbance estimator, there still exists velocity variations. The procedure for estimating the master position and/or velocity is an important step for E-gearing. Methods of tracking control have been developed in various fields, and include radar tracking control [40] and others. This study proposes an N^{th} order polynomial tracking method to perform the E-gearing process.

According to the N^{th} order polynomial, the master velocity at time t can be expressed as,



$$\omega = \sum_{i=0}^N c_i t^i \quad (2.4)$$

To determining the above coefficients (c_0, c_1, \dots, c_N) in real-time, two procedures are proposed.

(I) Initial procedure, $t = kT$, $1 \leq k \leq N + 1$, is the various order $((k - 1)^{th}$ order) polynomial extrapolation, where the symbol k is a real-time counter of time base, T is the PC-based programming sampling time and kT denotes the present time over all this dissertation.

$$\sum_{i=0}^{k-1} c_i (l \cdot T)^i = \omega_l, l = 0 \text{ to } k-1 \quad (2.5)$$

Here use the assumption of $0^0 = 1$.

(II) Main procedure, $t = kT$, $k > N+1$, is the fixed N^{th} order polynomial extrapolation:

$$\sum_{i=0}^N c_i (j \cdot T)^i = \omega_l, \quad l = (k-N-1) \text{ to } (k-1), \quad j = l - (k-N-1) \quad (2.6)$$

Similarly, the symbol k is a real-time counter of time base, T is the PC-based programming sampling time. Where $\omega_l (= (x_{l+1} - x_l) / T)$ are the measured angular velocities during the interval $[lT, (l+1)T]$, x_l are the recorded positions of the master measured from the encoder at the past time lT . Furthermore, Figure 2.4 shows the temporal relations of the two proposed procedures.

Rewriting Eq. (2.6) in matrix form yields,

$$\begin{aligned} \mathbf{M} \cdot \mathbf{C}_k &= \mathbf{\Omega} \\ \Rightarrow \mathbf{C}_k &= \mathbf{M}^{-1} \cdot \mathbf{\Omega} \end{aligned} \quad (2.7)$$

where $\mathbf{M} \in \mathbf{R}^{(N+1) \times (N+1)}$, $\mathbf{C}_k \in \mathbf{R}^{N+1}$ and $\mathbf{\Omega}$ in this chapter are the obtained time matrix, the matrix of polynomial coefficients and the matrix of measured angular velocities, respectively.

Moreover, the element of \mathbf{M} in the i^{th} row and j^{th} column can be expressed as,

$$m_{i,j} = ((i-1)T)^{j-1} \quad (2.8)$$

$$\mathbf{C}_k = [c_0, c_1, \dots, c_N]^T, \quad \mathbf{\Omega} = [\omega_{k-N-1}, \omega_{k-N}, \dots, \omega_{k-1}]^T \quad (2.9)$$

In Eq. (2.8), \mathbf{M} is a constant matrix and \mathbf{M}^{-1} exists; the computation involves no numerical degeneracy. Then the estimated velocity $\hat{\omega}_k$ during the time interval $[kT, (k+1)T]$ can be calculated as

$$\hat{\omega}_k = [1, NT, \dots, (NT)^N] \cdot \mathbf{C}_k \quad (2.10)$$

However, the estimated initial angular velocity may be chosen as the reference master velocity ($\bar{\omega}$) which is the desired velocity of the master, i.e. $\hat{\omega}_0 = \bar{\omega}$.

Then, the estimated position of the master is,

$$\hat{x}_{k+1} = x_k + \hat{\omega}_k \cdot T \quad (2.11)$$

where x_k and \hat{x}_{k+1} are the measured position of the master at the present sample time kT and the estimated position at the next sample time $(k+1)T$, respectively.

2.3 Predicting the Position of Slaves

This study uses Lagrange's interpolation formula to establish piecewise cam trajectories. If the piecewise reciprocal master-slave's coordinates, (x_i, y_i) , obtained from the given cam profile table, specify $n+1$ points, where $i = 0$ to n , and $x_0 < x_1 < \dots < x_n$, then the n th-degree Lagrange polynomial is,

$$f_L(x) = \sum_{i=0}^n L_i(x) y_i \quad (2.12)$$

where

$$L_i(x) = \prod_{j=0, j \neq i}^n \left(\frac{x - x_j}{x_i - x_j} \right) \quad (2.13)$$

are the Lagrange interpolation coefficients. Table 2.1 is an example of a cam profile table.

Substituting Eq. (2.11) into Eq. (2.12), yields the next ideal cam profile position of the slave:

$$f_L(\hat{x}_{k+1}) = \sum_{i=0}^n L_i(\hat{x}_{k+1}) y_i \quad (2.14)$$

However, in Eq. (2.14), as the order n increased, ripples and oscillations may occur [9].

Furthermore, the design of the cam profile may not consider the dynamic capability of the control plant in advance. Some dynamic limitations that degrade the slave motion generally apply; for example, a cutting machine tool may chatter due to over-large jerk, so the jerk has to be limited during the cutting process. Restated, maximal velocity and acceleration are limited by the motor and servo drive system. Consequently, the actual trajectory of slave motion may not be fulfilled Eq. (2.14), but must be close to the ideal trajectory provided fitting the specified constraints. Given its low sensitivity, the piecewise trajectory of the actual slave motion with respect to time is proposed to follow a cubic B-spline curve of fourth

degree [9], as shown in Fig. 2.5:

$$r_{k+1,j}(u) = F_{1,4}(u)p_{k+1,j-1} + F_{2,4}(u)p_{k+1,j} + F_{3,4}(u)p_{k+1,j+1} + F_{4,4}(u)p_{k+1,j+2} \quad (2.15)$$

where $r_{k+1,j}(u)$ represents the j^{th} segment of the $(k+1)^{\text{th}}$ time interval; $j \in [1:4]$ denotes the curve segment number and $u = 0$ to 1 within each curve segment.

$p_{k+1,j-1} \sim p_{k+1,j+2}$ are the control points of the spline. $F_{1,4}(u) \sim F_{4,4}(u)$ are the blending functions.

The fourth degree cubic B-spline, as shown in Fig. 2.5, exhibits second-order continuity.

All the variables of the B-spline are defined below.

(I) $p_{k+1,0}$ ($= p_{k-4,5}$) denotes the initial control point of the $(k+1)^{\text{th}}$ time interval, where $p_{k-4,5}$ is the previous position command of the slave at time $(k-4)T$ and equivalently the fifth

control point of the $(k - 4)^{th}$ time interval.

(II) $p_{k+1,1}(= p_{k-3,5})$ denotes the 1st control point of the $(k + 1)^{th}$ time interval, where $p_{k-3,5}$ is the previous position command of the slave at time $(k-3)T$ and equivalently the fifth control point of the $(k - 3)^{th}$ time interval.

(III) $p_{k+1,2}(= p_{k-2,5})$ denotes the 2nd control point of the $(k + 1)^{th}$ time interval, where $p_{k-2,5}$ is the previous position of the slave at time $(k-2)T$ and equivalently the fifth control point of the $(k - 2)^{th}$ time interval.

(IV) $p_{k+1,3}(= p_{k-1,5})$ denotes the 3rd control point of the $(k + 1)^{th}$ time interval, where $p_{k-1,5}$ is the previous position of the slave at time $(k-1)T$ and equivalently the fifth control point of the $(k - 1)^{th}$ time interval.

(V) $p_{k+1,4}(= p_{k,5})$ denotes the 4th control point of the $(k + 1)^{th}$ time interval, where $p_{k,5}$ is the previous position of the slave at time kT and equivalently the fifth control point of the k^{th} time interval.

(VI) $p_{k+1,5}$ denotes the position command of the slave motor yet to be determined, and is equivalently the fifth control point of the $(k + 1)^{th}$ time interval.

(VII) $p_{k+1,6}(= f_L(\hat{x}_{k+2}))$ denotes the sixth control point of the $(k + 1)^{th}$ time interval, where $f_L(\hat{x}_{k+2})$ is derived from the cam profile position at time $(k+2)T$, as indicated in Eq. (2.14).

Statements (I) ~ (VII) include a total of seven unknowns and six independent equalities. There is an extra degree of freedom left for the following optimization problem:

The slave's position error between the next unknown position command $p_{k+1,5}$ and the ideal cam profile position command $f_L(\hat{x}_{k+1})$ at time $(k+1)T$ can be expressed as,

$$e_{k+1} = p_{k+1,5} - f_L(\hat{x}_{k+1}) \quad (2.16)$$

The objective error function is defined in quadratic form as,

$$\begin{aligned} E_{k+1} &= \|e_{k+1}\|_2^2 \\ &= \|p_{k+1,5} - f_L(\hat{x}_{k+1})\|_2^2 \end{aligned} \quad (2.17)$$

To ensure that the velocity, acceleration and jerk do not exceed the maximal values, (V_{\max} , A_{\max} and $Jerk_{\max}$) allowed for the motor's system, three inequality constraints are imposed on the optimization. The first-, second- and third-differentiation of the cubic B-spline curve at the start, $u = 0$, of the fourth segment, can be expressed as follows [9].

$$r_{k+1,4}^u(0) = -0.5p_{k+1,3} + 0.5p_{k+1,5} \quad (2.18a)$$

$$r_{k+1,4}^{uu}(0) = p_{k+1,3} - 2p_{k+1,4} + p_{k+1,5} \quad (2.18b)$$

$$r_{k+1,4}^{uuu}(0) = -p_{k+1,3} + 3p_{k+1,4} - 3p_{k+1,5} + p_{k+1,6} \quad (2.18c)$$

Minimizing the objective error function subject to the constraints on velocity, acceleration and jerk, yields the one dimensional constrained optimization problem:

$$\text{Minimize } \|p_{k+1,5} - f_L(\hat{x}_{k+1})\|_2^2 \quad (2.19a)$$

$$\text{subject to } \begin{cases} |r_{k+1,4}^u(0)| \leq V_{\max} & (2.19b) \\ |r_{k+1,4}^{uu}(0)| \leq A_{\max} & (2.19c) \\ |r_{k+1,4}^{uuu}(0)| \leq Jerk_{\max} & (2.19d) \end{cases}$$

The constrained optimization problem of a quadratic cost function has an easy-to-find

optimal solution, $p_{k+1,5}^* = f_L(\hat{x}_{k+1})$ with zero cost, when none of the constraints is violated.

According to equation (2.19a) ~ (2.19d), the optimization problem may be reformulated as an unconstrained minimization problem as follows:

$$\text{Minimize } \|p_{k+1,5} - f_L(\hat{x}_{k+1})\|_2^2 + W_v g_v(p_{k+1,5}) + W_a g_a(p_{k+1,5}) + W_j g_j(p_{k+1,5}) \quad (2.20)$$

where W_v , W_a and W_j are the weighting factors of velocity constraint, acceleration constraint and jerk constraint, respectively, and

$$g_v(p_{k+1,5}) = \begin{cases} ||r_{k+1,4}^u(0) - V_{\max}|, & \text{if } |r_{k+1,4}^u(0)| < V_{\max} \\ 0, & \text{if } |r_{k+1,4}^u(0)| \geq V_{\max} \end{cases} \quad (2.21a)$$

$$g_a(p_{k+1,5}) = \begin{cases} ||r_{k+1,4}^{uu}(0) - A_{\max}|, & \text{if } |r_{k+1,4}^{uu}(0)| < A_{\max} \\ 0, & \text{if } |r_{k+1,4}^{uu}(0)| \geq A_{\max} \end{cases} \quad (2.21b)$$

$$g_j(p_{k+1,5}) = \begin{cases} ||r_{k+1,4}^{uuu}(0) - Jerk_{\max}|, & \text{if } |r_{k+1,4}^{uuu}(0)| < Jerk_{\max} \\ 0, & \text{if } |r_{k+1,4}^{uuu}(0)| \geq Jerk_{\max} \end{cases} \quad (2.21c)$$

In an extreme case that $W_v \gg W_a \gg W_j$, the minimization problem implies a constraint violation priority that g_v is much more important than g_a and g_j . In practice, equation (2.19) is highly nonlinear, existing techniques to find the global optimization is not guaranteed. It needs to enumerate all the possible cases for the global solution. Figure 2.7 shows all the possible optimal solution for the extreme case that $W_v \gg W_a \gg W_j$. The bounds of $p_{k+1,5}$ for each of the constraints may be easily calculated from equations (2.19b), (2.19c) and (2.19d) by substituting the inequality sign into equality sign, as follows.

$$p_{k+1,5} = 2 \cdot \text{sign}(r_{k+1,4}^u(0)) \cdot V_{\max} + p_{k+1,3} \quad (2.22a)$$

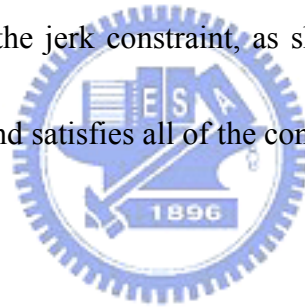
$$p_{k+1,5} = \text{sign}(r_{k+1,4}^{uu}(0)) \cdot A_{\max} - p_{k+1,3} + 2p_{k+1,4} \quad (2.22b)$$

$$p_{k+1,5} = -\frac{1}{3} \text{sign}(r_{k+1,4}^{uuu}(0)) \cdot \text{Jerk}_{\max} - \frac{1}{3} p_{k+1,3} + p_{k+1,4} + \frac{1}{3} p_{k+1,6} \quad (2.22c)$$

The optimal solution process may be depicted in the flow chart as shown in Fig. 2.6.

According to the flow chart, the solution of the optimization problem is unique, and thus guarantees to be the global optimum.

In Fig. 2.6, all possible cases are enumerated and categorized as follows. (i) The ideal cam profile position command violates the velocity constraint, as shown in Figs. 2.7 (a) ~ (l); (ii) the ideal cam profile position command violates acceleration constraint and does not violate velocity constraint, as shown in Figs. 2.7(m) and 2.7(n); (iii) the ideal cam profile position command violates only the jerk constraint, as shown in Figs. 2.7(o) ~ (q); (iv) the ideal cam profile position command satisfies all of the constraints, as shown in Fig. 2.7(r).



2.4 Simulation and Experimental Results

2.4.1 Simulation of disturbance estimator

For simulation purposes, the nominal external disturbance is assumed to be a square wave function.

$$\tau_L = \begin{array}{c} \downarrow \\ \text{---} \square \text{---} \square \text{---} \square \text{---} \\ \uparrow \\ \text{Amp} \end{array} \quad (2.23)$$

The torque amplitude Amp of the square wave is set to $4.8773 \text{ (N} \cdot \text{m)}$ and the frequency of the square wave is 1 Hz. Figures 2.8(a) and (b) present the master's simulated angular velocity obtained using the proposed disturbance estimator feedback control and without

using the disturbance estimator. The nominal parameters of the master motor defined in Section 2.1.1, are $\hat{K} = 0.55 (N \cdot m / A)$, $\hat{J} = 0.093 (kg \cdot m^2)$, $\hat{B} = 0.008 (N \cdot m \cdot s / rad)$, $L_f = 0.046 H$, $R_f = 1 \Omega$ and $K_a = 0.55 V \cdot s / rad$. The sampling time of the current loop is set to $0.001s$ in the simulation. Furthermore, the amplitude of the disturbance load torque is $4.8773 (N \cdot m)$ and the torque constant is $0.55 (N \cdot m / A)$, that is, the operating current is about $8.9 (A)$, the $i_a^2 R_f$ power loss is around $78.6 (W)$ (provided by the paper reviewer). However, the power loss of $78.6 (W)$ is not that serious for the applications with motors up to several kW .

Figures 2.9(a) and (b) show the maximum errors between the fed torque (τ_L) and the estimated torque ($\hat{\tau}_L$) for various time constants (\mathfrak{T}). In Fig. 2.9(a), a smaller \mathfrak{T} yields a smaller mean torque error. However Fig. 2.9(b) reveals that a lower \mathfrak{T} yields a larger measurement noise. Furthermore, the measurement noise was assumed to be a zero-mean, normally (Gaussian) distributed random signal in the simulation.

Both a larger mean torque error and a larger measurement noise reduce the tracking performance of the master, so the time constant must be neither too small nor too large. In the experiment, the time constant (\mathfrak{T}) of the disturbance estimator was set to ten times the current loop sampling time. As depicted in Fig. 2.8(b), the time constant (\mathfrak{T}) and the current loop sampling time are set to $0.01 sec$ and $0.001 sec$, respectively.

2.4.2 Experimental results for tracking performance of the electronic gearing process

Table 2.2 lists the parameter settings of the ECAM control. The accuracy of the tracking of the master's velocity is characterized by the maximum error between the actual position and the estimated position. Figures 2.10(a) ~ 2.10(d) show that the maximum tracking error of the master's position, using the fifth order polynomial tracking control method, is zero when the master's nominal mean speed is $10\pi \text{ rad/s}$. Table 2.3 shows the maximum tracking error of the master's position for polynomial tracking control methods of various orders ($N = 0$ to 5).

2.4.3 Performance of the electronic cam process

Figure 2.11(b) shows an example of a reference trajectory that corresponds to the electronic cam motion. According to a constant master speed of $10\pi \text{ rad/s}$ and a maximum slave travel distance of $40\pi \text{ rad}$, the reference trajectory yields a $100\pi \text{ rad/s}$ maximum slave speed, $630\pi \text{ rad/s}^2$ maximum acceleration and $4060\pi \text{ rad/s}^3$ maximum jerk. The master's speed is generally not constant and may be harmonic, as shown in Figs. 2.8(a) and (b). The speed will exhibit the actual position of the master and the ideal cam trajectory, as shown in Figs. 2.11(a) and 2.11(c), respectively. This piecewise cam trajectory contains 191 points. Three performance indices are used to quantify the accuracy and consistency. The tracking accuracy of the slave motion is characterized by the maximum error and the root mean square (RMS) error. The consistency of the cam tracking – that is, the cycle-to-cycle variation - is characterized by the RMS difference between the particular error response and

the error response averaged over a number of cycles. Fifty cycles of tracking error data were collected. Figures 2.12(a) ~ 2.12(d) summarize the results of slave position. Table 2.4 lists the maximum tracking errors of the slave's position in encoder counts, using the N^{th} order polynomial tracking control method and the pure Lagrange polynomial curve-fitting method. Furthermore, Fig. 2.13 shows the partial results of the slave's tracking velocity, acceleration and jerk, according to Lagrange polynomial curve-fitting with or without the aforementioned optimization. Similarly, Table 2.5 indicates the tracking control performance, also for the Lagrange polynomial curve-fitting method with or without the aforementioned optimization.

2.4.4 Computational load on the CPU of the proposed ECAM tracking control

The selection of N depends on the accuracy demanded. As stated above, tracking using a higher order polynomial yields higher precision; however a tradeoff exists between the “order” of the polynomial used and the CPU time required. In practice, the computational time of the proposed algorithm (fifth-order tracking) is about 0.02 *ms* in a programming cycle on an Intel Pentium III 900MHz CPU. The computational time of a programming cycle is much less than the PC-based sampling time, 10 *ms*.

Chap 3 A Novel Washout Filter Design for a Six Degree-of-Freedom Motion Simulator

The motion cue performed by a motion simulator is restrained by the workspace of the simulator structure. A typical reasoning is then to build a large motion simulator unless it is for entertainment, which involves in small simulators. This study proposes a novel approach to designing the washout filter of the motion control of a six degree-of-freedom motion simulator for entertainment purposes. Using information obtained from the inverse kinematics of the simulator, the workspace boundary, detected in real-time, is fed into the washout filter as a reference for the motion planning. The main focus of this approach is to make the motion cue feasible for use in a simulator with a restricted workspace, while ensuring the robustness of the driving system. In this paper, different indices are established to specify the performance of the motion cue. A classical linear washout filter was implemented and compared with the proposed washout filter using the performance indices to demonstrate the benefits of the latter.

3.1 Inverse Kinematics

The motion cue control may be also called the cockpit position control, because the position of the cockpit, including both translation and rotation, must be transformed into the coordinates of the six sliders' ball joints (S to J) using inverse kinematics. The inverse kinematics of the motion simulator SP-120 (Fig. 3.4) is presented as follows.

As shown in Fig. 3.5, the coordinate of ${}^{Si}\mathbf{p}_i = {}^{Si}[p_{xi} \ p_{yi} \ p_{zi}]^T$ is determined by

$$\| {}^{S_i} \mathbf{q}_i - {}^{S_i} \mathbf{p}_i \|^2 = L^2 \quad (3.1)$$

in which all parameters are fixed in the S_i coordinate system, where

$${}^{S_i} \mathbf{q}_i = {}^{S_i} O + {}^{S_i} R \cdot [{}^O G + {}^O R(\alpha, \beta, \gamma) \cdot {}^G \mathbf{q}_i] \quad (3.2)$$

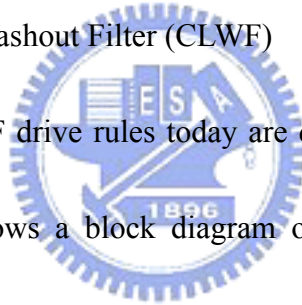
and ${}^O R(\alpha, \beta, \gamma)$ is the transformation matrix of the Euler angle and can be expressed as,

$${}^O R(\alpha, \beta, \gamma) = \begin{bmatrix} c\beta c\gamma & -c\beta s\gamma & s\beta \\ s\alpha s\beta c\gamma + c\alpha s\gamma & -s\alpha s\beta s\gamma + c\alpha c\gamma & -s\alpha c\beta \\ -c\alpha s\beta c\gamma + s\alpha s\gamma & c\alpha s\beta s\gamma + s\alpha c\gamma & c\alpha c\beta \end{bmatrix} \quad (3.3)$$

and $c\beta = \cos \beta, s\alpha = \sin \alpha, \dots$ and so on.

3.2 Preview of Classical Linear Washout Filter (CLWF)

The most widely used CLWF drive rules today are derived from the design of Schmidt and Conrad [13]. Figure 3.1 shows a block diagram of the typical implementation [14].



Modern implementations tend to drive the simulation with angular rates rather than angular accelerations, since this method has generally been found to produce a more realistic cue. As

shown in Fig. 3.1, specific acceleration (\mathbf{a}_{AA}) is transformed to the inertial reference frame (S to I) and converted into acceleration (\mathbf{a}_{VR}), obtained by summing gravity (\mathbf{g}_I) and applied

forces before the high-pass filter operation is performed. This approach uses a more convenient frame of reference for generating the commands of the simulator's driving system.

Similarly, high-pass onset filtering is applied to the scaled Euler's angular velocity ($\boldsymbol{\omega}_s$). The

low-frequency specific acceleration components are low-pass filtered, and operated upon by a

"tilt coordinate" block, much like that in Schmidt and Conrad's residue-tilt design; the tilt coordinate cross-feed is rate-limited to ensure that the commanded rates do not exceed the pilot's indifference threshold, which is set to 3 *deg/s* [23]. As stated above, the CLWF technique is combined with the following auxiliary washout filters, yielding a novel washout filter, presented in Fig. 3.2.

3.3 Adjustable Scaling Filter (ASF) and Yawing Washout Filter (YWF)

The limited workspace of the simulator constrains the Euler angles, including roll, pitch and yaw. Therefore, this paper proposes that the angular velocity of the cockpit must be adjusted by a dynamic tuning process called adjustable scaling filtering which involves a nonlinear filter, rather than by purely static scaling down, which would also reduce the active intensity even if the angular rates are originally lower. Applying this nonlinear filter can guarantee that the signals of angular velocities, fed to the simulator are more realistic than those associated with traditional static scaling down, unless the limited workspace is sufficiently large that the magnitude of the static scaling factor is approximately unity. Restated, the degree of scaling down is traded off with the limited size of the workspace, but can be greatly reduced after the ASF is used. The algorithm is as follows.

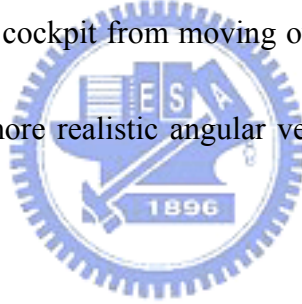
$$\begin{cases} \phi_{i,k+1} = \phi_{HP,i,k+1}, & \text{if } \|(\phi_{HP,k+1})\|_2 \leq \phi_{critical} \\ \phi_{i,k+1} = \phi_{i,k+1} \cdot \phi_{critical} / \|\phi_{HP,k+1}\|_2, & \text{if } \|(\phi_{HP,k+1})\|_2 > \phi_{critical} \end{cases} \quad (3.4)$$

and,

$$\omega_{i,k} = (\phi_{i,k+1} - \phi_{i,k}) / t_{VR} \quad (3.5)$$

where $\phi_{i,k}$ and $\phi_{HP,i,k}$ both represent the present Euler angles, the latter of which is received from the high-pass filter at time kt_{VR} ; $\phi_{critical}$ is the magnitude of a given critical Euler angle; $\|\bullet\|_2$ represents the 2-norm of \bullet ; t_{VR} is the sampling period of VR; both $\omega_{i,k}$ and $\omega_{HP,i,k}$ are present Euler angular velocities, the latter of which is received from the output of the high-pass filter at time kt_{VR} . The subscript i indicates the i -axis ($i = x, y$ or z).

Applying the above algorithm greatly improves the rotational performance of the motion simulator, not only to prevent the cockpit from moving outside the limited workspace during pure rotation but also to obtain more realistic angular velocities or attitudes of the platform during real-time VR motion.



Importantly, the platform's attitude in terms of roll and pitch involves an actual tilt coordination that enables the pilot to feel the component of gravity; thus, the roll or pitch cannot be arbitrarily changed during the restoration unless the attitude is obtained by low-pass filtering of the acceleration along the y- or x-axis, as in residue tilt. Contrarily, the yaw angle is not important: the only concern is the yawing velocity. Therefore, a yawing washout algorithm is proposed as follows.

$$\text{if } (\phi_{z,k+1} < \phi_{z,critical} \text{ and } |\omega_{HP,z,k}| > \omega_{indiff}) \text{ or } \text{sign}(\omega_{HP,z,k}) = -\text{sign}(\phi_{z,k})$$

$$\Rightarrow \omega_{z,k} = \omega_{HP,z,k} \quad (3.6)$$

if $(\phi_{z,k+1} \geq \phi_{z,critical} \text{ or } |\omega_{HP,z,k}| < \omega_{indiff})$ and $sign(\omega_{HP,z,k}) = sign(\phi_{z,k})$, and $t \leq t_{res,yaw}$

$$\Rightarrow \omega_{z,k} = -sign(\phi_{z,k}) \cdot \omega_{indiff} \quad (3.7)$$

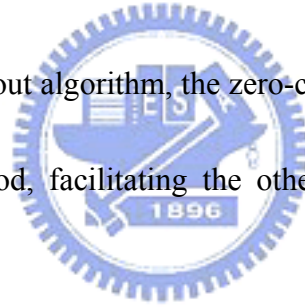
where $\phi_{z,critical}$ is the given critical yaw about the simulator; ω_{indiff} is the indifference threshold for angular speed; t and $t_{res,yaw}$ are the present restoring time and the total periodic restoring time, respectively, and

$$t_{res,yaw} = |\phi_{z,k}| / \omega_{indiff}, \quad (3.8)$$

where,

$$\begin{cases} sign(\bullet) = 1, \text{ if } (\bullet) > 0 \\ sign(\bullet) = -1, \text{ if } (\bullet) < 0 \\ sign(\bullet) = 0, \text{ if } (\bullet) = 0 \end{cases}$$

Applying the above yawing washout algorithm, the zero-crossing phenomenon does not occur during the yawing washout period, facilitating the other actions including roll, pitch and translation.

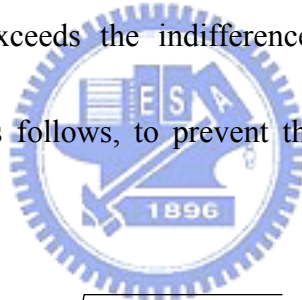


3.4 Dead Zone Washout Filter (DZWF)

During the restoration period, the limitations on linear acceleration and angular velocity [24, 25] almost prohibit the restoration of the cockpit to its home position except by extending the restoration period, or when the original motion in VR are at sufficiently low frequencies. Clearly, extending the restoration time may cause some significant motion to be lost, so this strategy is not favored. Translation with lower frequencies may give enough time to carry the cockpit back stealthily after proper high-pass filtering is performed, but generally, such a

motion implies unexcited motion and poorly represents most normal actions. Consequently, a proposed strategy, called dead zone washout filtering, is adept at utilizing time. Dead time is defined as the period during which the linear acceleration is lower than the pilot's sensible threshold [23]-[25]. Restated, the acceleration enters the dead area during this dead time; otherwise, it is in the scaled area.

During the dead time, the cockpit is translated to its home position rather than being scaled to zero. Every component of the restoring acceleration $\mathbf{a}_{res} \in R^3$ must be lower than the indifference threshold $a_{threshold}$ (0.17~0.28 m/s^2 [25], here set to 0.017g). Even if the restoring acceleration slightly exceeds the indifference threshold, an adaptive restoring acceleration must be modified as follows, to prevent the workspace boundary from being touched.



$$\begin{cases} |a_{res,i}| = v_{0,i}^2 / 2S_{max,i}, & \text{if } |v_{0,i}| \geq \sqrt{2a_{threshold}S_{max,i}} \\ |a_{res,i}| = a_{threshold}, & \text{if } |v_{0,i}| < \sqrt{2a_{threshold}S_{max,i}} \end{cases} \quad (3.9)$$

where $S_{max,i}$ is the maximum distance from the present position to the nominal workspace boundary in the direction of the present velocity $v_{0,i}$; subscript i indicates the i -axis ($i = x, y$ or z).

The next urgent task is to determine the maximum restoration period. Figure 3.6(a) indicates a situation in which the present velocity has the same direction as the present displacement, with reference to the home position. Figure 3.6(b) presents a situation in which the present velocity is in the opposite direction from the displacement. These two cases are

both treated by the basic law of kinematics, yielding,

$$t_1 = \text{sign}(P_{cur,i}) \cdot v_{0,i} / |a_{res,i}| + (1/2) \sqrt{2v_{0,i}^2 / a_{res,i}^2 + 4d_i / |a_{res,i}|} \quad (3.10)$$

$$t_2 = \text{sign}(P_{cur,i}) \cdot v_{0,i} / |a_{res,i}| + \sqrt{2v_{0,i}^2 / a_{res,i}^2 + 4d_i / |a_{res,i}|} \quad (3.11)$$

and,

$$d_i = |P_{home,i} - P_{cur,i}| \quad (3.12)$$

where d_i is the distance from the present position to the home position along the i -axis;

$P_{home,i}$ and $P_{cur,i}$ are the home position and the present position along the i -axis,

respectively; t_1 is the period of deceleration and t_2 is the total restoration period. The

velocity is importantly guaranteed to be zero at the restoring time t_2 .

During the restoration period, the restoring action will continue unless the direction of acceleration along the i -axis is opposite neither that of the present velocity nor the present position. Then, the active acceleration along the i -axis can be expressed as,

$$\begin{cases} a_i = a_{HP,i}, & \text{if } a_{HP,i} > a_{threshold} \text{ and } \text{sign}(a_{HP,i}) = -\text{sign}(v_{cur,i}) = -\text{sign}(P_{cur,i}) \\ a_i = a_{res,i}, & \text{others} \end{cases} \quad (3.13)$$

where $a_{HP,i}$ represents the linear acceleration along the i -axis, received from the output of high-pass filter, and $v_{cur,i}$ is the present linear velocity along the i -axis.

Like the yawing washout, the DZWF procedure involves no zero-crossing and improves the rotational performance. Restated, it greatly reduces the cross coupling of rotation and translation.

3.5 Structure of the Adaptive Washout Filter (AWF)

This DZWF algorithm cannot always guarantee that the cockpit of simulator does not leave the actual workspace because not all of the workspace boundaries are very explicit. Therefore, adding an adaptive washout filter is proposed to compensate for the insufficiency of the prior proposed filters and thus accommodate the more severe restrictions, such as the smaller workspace and the limited driving current. The AWF involves the transformation (S to J), the washout function (Fig. 3.3(a)) and the self-tuning process (Fig. 3.3(b)).

3.5.1 Transformation (S to J)

The transformation (S to J) is presented using inverse kinematics, as described in section

3.1.



3.5.2 Washout Function

The purpose of the washout function (Fig. 3.3(a)) is to prevent the cockpit from exiting its limited workspace. Figure 3.7 depicts a trajectory along the i -axis proposed to plan the washout motion, initially making the pilot feel an instantaneous linear acceleration and later carrying the cockpit to its starting position stealthily in the period $t_d \leq t \leq t_f$. The planned trajectory is as follows.

The continuous trajectory $\mathbf{P}(t) \in R^3$ of the translation of the cockpit consists of two cubic polynomial segments.

$$\mathbf{P}(t) = \mathbf{P}_1(t) + \mathbf{P}_2(t) \quad \text{for} \quad 0 \leq t \leq t_f \quad (3.14)$$

where the vectors $\mathbf{P}_1(t)$ and $\mathbf{P}_2(t)$ are in R^3 ;

$$\mathbf{P}_1(t) = \mathbf{0} \quad \text{for } t_d \leq t \quad \text{and}$$

$$\mathbf{P}_2(t) = \mathbf{0} \quad \text{for } t < t_d$$

The $\mathbf{P}(t)$ at t_f is the desired target position and that at t_d is the transition position. At least eight constraints on $\mathbf{P}(t)$ are evident. The initial and final values of the function are constrained.

$$\mathbf{P}(0) = \mathbf{0}, \mathbf{P}(t_f) = \mathbf{0}.$$

Continuity at the transition position yields,

$$\mathbf{P}_1(t_d^-) = \mathbf{P}_2(t_d^+).$$

The function yields continuous velocities, implying that the initial and transitional velocities' vectors (R^3) are both continuous and the final velocity's vector is zero in the procedure of washout motion planning, and can be expressed as,

$$\mathbf{v}(0) = \mathbf{v}_0, \mathbf{v}(t_f) = \mathbf{0}, \mathbf{v}_1(t_d^-) = \mathbf{v}_2(t_d^+).$$

A further constraint is that the transition acceleration must be continuous:

$$\mathbf{a}_1(t_d^-) = \mathbf{a}_2(t_d^+) \in R^3.$$

The following equation implies that the initial acceleration must fulfill the demands of VR – to ensure that the pilot feels an instantaneous linear acceleration.

$$\mathbf{a}_1(0) = \mathbf{a}_{ref} \in R^3$$

where \mathbf{a}_{ref} is the linear acceleration received from the output of the DZWF. These eight

constraints can be used to determine two consecutive cubic polynomial segments, since two such segments have exactly eight coefficient vectors.

$$\mathbf{P}_1(t) = \mathbf{b}_0 + \mathbf{b}_1 t + \mathbf{b}_2 t^2 + \mathbf{b}_3 t^3 \quad (3.15)$$

$$\mathbf{P}_2(t) = \mathbf{c}_0 + \mathbf{c}_1 t + \mathbf{c}_2 t^2 + \mathbf{c}_3 t^3 \quad (3.16)$$

where the eight vectors, \mathbf{b}_0 , \mathbf{b}_1 , \mathbf{b}_2 , \mathbf{b}_3 , \mathbf{c}_0 , \mathbf{c}_1 , \mathbf{c}_2 , and \mathbf{c}_3 are all in R^3 . The vectors of velocities and accelerations along the path are derived as follows.

$$\mathbf{v}_1(t) = \dot{\mathbf{P}}_1(t) = \mathbf{b}_1 + 2\mathbf{b}_2 t + 3\mathbf{b}_3 t^2 \quad (3.17)$$

$$\mathbf{v}_2(t) = \dot{\mathbf{P}}_2(t) = \mathbf{c}_1 + 2\mathbf{c}_2 t + 3\mathbf{c}_3 t^2 \quad (3.18)$$

$$\mathbf{a}_1(t) = \ddot{\mathbf{P}}_1(t) = 2\mathbf{b}_2 + 6\mathbf{b}_3 t \quad (3.19)$$

$$\mathbf{a}_2(t) = \ddot{\mathbf{P}}_2(t) = 2\mathbf{c}_2 + 6\mathbf{c}_3 t \quad (3.20)$$

Combining Eqs. (3.15) to (3.20) with the eight constraints yields eight-by-three constrained equations in eight-by-three unknowns. Let $t_d = \kappa t_f$, where $0 < \kappa < 1$. Now,

$$\mathbf{b}_0 = \mathbf{0} \quad (3.21)$$

$$\mathbf{b}_1 = \mathbf{v}_0 \quad (3.22)$$

$$\mathbf{b}_2 = \mathbf{a}_{ref} / 2 \quad (3.23)$$

$$\mathbf{b}_3 = (2\mathbf{a}_{ref} \kappa t_f + 2\mathbf{v}_0 \kappa + \mathbf{a}_{ref} t_f + 4\mathbf{v}_0) / (6 \kappa t_f^2) \quad (3.24)$$

$$\mathbf{c}_0 = -\kappa^2 t_f (\mathbf{a}_{ref} t_f + 4\mathbf{v}_0) / (6(1-\kappa)^2) \quad (3.25)$$

$$\mathbf{c}_1 = (2\mathbf{v}_0 (1+\kappa^2) + \mathbf{a}_{ref} \kappa t_f) / (2(1-\kappa)^2) \quad (3.26)$$

$$\mathbf{c}_2 = (\mathbf{a}_{ref} \kappa^2 t_f - 4\mathbf{v}_0 - 2\mathbf{a}_{ref} \kappa t_f) / (2 t_f (1-\kappa)^2) \quad (3.27)$$

$$\mathbf{c}_3 = (-\mathbf{a}_{ref} \kappa t_f (3-2\kappa) - 2\mathbf{v}_0 (3-\kappa^2)) / (6(1-\kappa)^2 t_f^2) \quad (3.28)$$

From Eq. (3.20), the maximum linear deceleration is $\mathbf{a}_2(t_d)$, and,

$$a_{2,i}(t_d) \leq a_{threshold} \quad (3.29)$$

where the subscript i represents the three mutually orthogonal axes (x-, y-, and z-axis), and $a_{2,i}(t_d)$ is the component of $\mathbf{a}_2(t_d)$ along the i -axis. Equation (3.28) implies that κ can be treated as a ratio to constrain deceleration during the restoration. The magnitude of deceleration must be constrained below an indifference threshold $a_{threshold}$ to prevent the pilot from becoming aware of this restoration [25].

The maximum displacement is at a stationary value when the velocity is zero. For the second segment of the polynomial,

$$\mathbf{v}_2(t) = \mathbf{c}_1 + 2\mathbf{c}_2 t + 3\mathbf{c}_3 t^2 = \mathbf{0}$$



which yields,

$$t_{2,P_{MAX}} = t_f (2v_{0,i} (1+\kappa^2) + a_{ref,i} \kappa t_f) / (2v_{0,i} (3-\kappa^2) + a_{ref,i} \kappa t_f (3-2\kappa)) > t_d \quad (3.30)$$

The maximum displacement along the i -axis is obtained by substituting $t_{2,P_{MAX}}$ into the i -axis washout function $P_i(t)$, such that,

$$P_{MAX,i} = P_i(t_{2,P_{MAX}}) \quad (3.31)$$

which will be used to determine whether the washout planning is executed.

3.5.3 Self-Tuning Process (Fig. 3.3(b))

The saturation of the driving current also constrains the performance of the simulator,

and may cause the angular speed of the servo motor to exceed its critical value and, causing a problem related to the robustness of the driving system. Additionally, motion may sometimes still violate the workspace after filtering, because the indifference threshold of deceleration always limits the washout efficiency. Thus, a final check on the self-tuning process is proposed to guarantee that the system of motion simulator is robust. The following steps determine the rules.

1. Calculate whether the commanded velocity fed to the driving system exceed the critical value.
2. Calculate whether the cockpit will be outside the limited workspace.
3. If at least one of the answers to the preceding questions is positive, let the linear acceleration and the angular velocity be multiplied by two appropriately predetermined scaling functions, $p_a(\lambda_a)$ and $p_\omega(\lambda_\omega)$, respectively. Then, redo steps 1 and 2 until the answers are both negative or the iterative loop is performed more than n times, where n is the number limit in this chapter preset by considering whether the total calculation time will meet the demands of real-time programming.

The following two simple equations represent the above strategies.

$$\mathbf{a} = \mathbf{a}_{ref} \cdot p_a^n(\lambda_a) \quad \text{and} \quad \boldsymbol{\omega} = \boldsymbol{\omega}_{ref} \cdot p_\omega^n(\lambda_\omega) \quad (3.32)$$

where $\mathbf{a} \in R^3$ and $\boldsymbol{\omega} \in R^3$ represent the cockpit's present linear acceleration and angular velocity, respectively; \mathbf{a}_{ref} and $\boldsymbol{\omega}_{ref}$ (Fig. 3.2) are the linear acceleration received from the

output of DZWF and the angular velocity that combines the output of ASF with the tilt coordination, respectively. The functions $p_a(\lambda_a)$ and $p_\omega(\lambda_\omega)$ are the adaptive scaling functions of linear acceleration and angular velocity, respectively. The adaptive scaling factors λ_a and λ_ω are properly predicted before the first simulation test and are later tuned offline by inducing an empirical rule to obtain a heuristically selected pair of adaptive scaling factors, as described in the next section.

3.6 Performance Index

In this paper, the VR motion fed to the specific motion simulator-SP120 is specified by a performance index (PI) combined with two cost functions, $E_{a,k}$ and $E_{\omega,k}$.

$$E_{a,k} = [\| \mathbf{a}(kt_{VR}) - \mathbf{a}_{ref}(kt_{VR}) \|_2 / RMS(\mathbf{a}_{ref})] \cdot RMS(\mathbf{a}_{VR}) / RMS(\mathbf{a}_{ref}) \quad (3.33)$$

$$= RMS(\mathbf{a}_{VR}) \cdot (1 - p_a^{n_k}(\lambda_a)) \| \mathbf{a}_{ref}(kt_{VR}) \|_2 / [RMS(\mathbf{a}_{ref})]^2$$

$$E_{\omega,k} = [\| \boldsymbol{\omega}^2(kt_{VR}) - \boldsymbol{\omega}_{ref}^2(kt_{VR}) \|_2 / RMS(\boldsymbol{\omega}_{ref}^2)] \cdot RMS(\boldsymbol{\omega}_{VR}) / RMS(\boldsymbol{\omega}_{ref}) \quad (3.34)$$

$$= RMS(\boldsymbol{\omega}_{VR}) \cdot (1 - p_\omega^{2n_k}(\lambda_\omega)) \| \boldsymbol{\omega}_{ref}^2(kt_{VR}) \|_2 / [RMS(\boldsymbol{\omega}_{ref}^2) \cdot RMS(\boldsymbol{\omega}_{ref})]$$

and,

$$PI = W_a \cdot RMS(E_a) + W_\omega \cdot RMS(E_\omega) \quad (3.35)$$

$$RMS(E_a) = \sqrt{(\sum_{k=0}^N E_{a,k}^2) / N}, \quad RMS(E_\omega) = \sqrt{(\sum_{k=0}^N E_{\omega,k}^2) / N} \quad (3.36)$$

where W_a and W_ω are the weighting parameters of $RMS(E_a)$ and $RMS(E_\omega)$, respectively; $RMS(\bullet)$ means the root mean square of \bullet ; N is the total number of samples of VR motion; n_k is the total number of self-tuning iterations at time kt_{VR} and is

determined by the self-tuning process online. By considering the dimensions of Eqs. (3.33) and (3.34), the adaptive scaling functions $p_a(\lambda_a)$ and $p_\omega(\lambda_\omega)$ can be properly defined,

$$\begin{cases} p_a(\lambda_a) = \lambda_a^2, & 0 \leq \lambda_a \leq 1 \\ p_\omega(\lambda_\omega) = \lambda_\omega, & 0 \leq \lambda_\omega \leq 1 \end{cases} \quad (3.37)$$

where λ_a and λ_ω are both set to constant values during one test, provided that the total number of iterations exceeds zero, such that $n_k \geq 1$. Otherwise, $\lambda_a = \lambda_\omega = 1$, provided that the answers in both steps 1 and 2 in the first loop of the self-tuning process are negative, such that $n_k = 0$.

The magnitudes of the two weighting parameters W_a and W_ω represent the relative significances of linear acceleration and angular velocity, respectively, determined by the pilot's response. The proper values are set to (0.5, 0.5), after consultation with ten pilots. A smaller PI implies more realistic motion. To yield a smaller PI , the effects of using many different pairs of adaptive scaling factors are observed and an empirical rule induced off-line to obtain a heuristically selected pair $(\lambda_a^*, \lambda_\omega^*)$, which is in future tests to be substituted in the self-tuning process and used instead of the old adaptive scaling factors.

Equations (3.33) ~ (3.37) state that the PI is function of λ_a , λ_ω and n_k , where n_k is also coupled with the adaptive scaling factors and varies irregularly with the sampling number k ; that is, the PI is not an explicit function of λ_a and λ_ω , so obtaining an optimal pair of adaptive scaling factors by directly minimizing the PI is difficult. Furthermore, the PI is determined instantaneously after online testing, which in turn is performed after the

adaptive scaling factors are determined off-line. Thus, these two factors must be correctly predetermined. The values of these two cost functions after many test runs using different adaptive scaling factors, indicate that a tradeoff exists between λ_a and λ_w to reduce the magnitude of PI . One of the rules of thumb is that the extreme values of Eq. (3.35) may be at the boundaries of λ_a and λ_w . Accordingly, the corresponding set of adaptive scaling factors may be,

$$(\lambda_a^*, \lambda_w^*) = (1, 0) \text{ or } (0, 1) \quad (3.38)$$

where the set (0, 0) is irrational. This result implies that the heuristically selected set of adaptive scaling factors always tends to one direction (toward (1, 0) or (0, 1)), determined by comparing the magnitudes of $RMS(E_a)$ and $RMS(E_w)$.

For example, if $RMS(E_w)$ exceeds $RMS(E_a)$ in the preceding test, then the degree of self-tuning of the angular velocities may be too small to perform the more difficult specific actions. Therefore, in the following test, the adaptive scaling factors are adjusted using above heuristically selected results ((1, 0)) to reduce the magnitude of $RMS(E_w)$. If $RMS(E_w)$ is reduced normally but still exceeds $RMS(E_a)$ after the second test, then the probably optimal set of adaptive scaling factors is (1, 0); otherwise, performing the third test by substituting the set (0, 1) into the self-tuning process, and then comparing the magnitudes of PI obtained in these two tests, enables the other set of adaptive scaling factors to be heuristically selected. Restated, the set of adaptive scaling factors must be sought at least twice, implying that one of

the magnitudes of PI in the second and third tests is heuristically selected.

3.7 Experimental Results and Comparison

A specific VR motion of an automotive system and its dynamics are considered to apply the proposed washout filter to the motion simulator SP-120. Moreover, real-time software was developed to demonstrate the advantages of the proposed control strategy and this technique is compared with the classical one. The following figures represent Euler's angular velocities ω_x and ω_y as pure rotational velocities. That is, the residue-tilting effect was omitted during the processing of data.

Figures 3.8 and 3.10 show the data concerning linear accelerations along the x-axis and Euler's angular velocities (ω_x) for the three segments. These data were obtained from the scaled VR dynamic output ($\mathbf{a}_s, \boldsymbol{\omega}_s$) and the simulator's two outputs using the control strategies of CLWF and the proposed washout filter. Figures 3.9 and 3.11 present the segmental errors of linear accelerations along the x-axis and Euler's angular velocities (ω_x), respectively. The two sets of errors are both between the scaled VR dynamic output and the simulator's outputs obtained using the control strategies of CLWF and the proposed washout filter. As shown in these figures, the maximum acceleration along the x-axis is around 0.3g and most of the scaled linear accelerations (\mathbf{a}_s) and scaled angular velocities ($\boldsymbol{\omega}_s$) can be simulated by applying the novel washout filtering to the simulator, which works in a relatively

small workspace, as stated in Table 3.1 [22]. In contrast, the CLWF technique performs very poorly with such a simulator. Sometimes, the acceleration may drop back to zero because the current of the driving system is saturated. Linear accelerations or angular velocity may have been maintained in a particular direction for so long that the current of the driving system may exceed the critical value. The proposed strategy maximally suppresses cases in which the linear acceleration drops back to zero.

As shown in Figs. 3.8 to 3.11, the proposed washout filter outperforms the CLWF in the SP-120 motion simulator. Table 3.2 compares the CLWF with the proposed washout filter in terms of the magnitudes of $RMS(E_a)$, $RMS(E_\omega)$ and PI , using various static scaling factors (s_a , s_ω) successively, to scale the linear accelerations and the angular velocities of the dynamic output of VR. In Table 3.2, the performance obtained when CLWF is applied to the simulator SP-120 shows that the efficiency associated with the simulated scaled data (a_s , ω_s) is better when the static scaling factor is smaller. This finding implies that the CLWF technique may be suited to a simulator with a large workspace but not one that operates in a more restricted workspace, such as the SP-120. Furthermore, using the proposed washout filter, even for this restricted simulator, the performance in terms of reality, strength and practicability remain excellent in many repeated tests in real time.

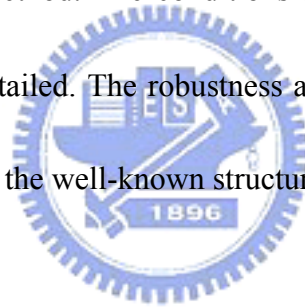
Figures 3.12 ~ 3.15 compare the use of the heuristically selected pair of adaptive scaling factors (1, 0) with the arbitrary pair (1, 0.5). The results reveal the advantages of using the

former pair. Table 3.3 presents the magnitudes of $RMS(E_a)$, $RMS(E_\omega)$ and PI obtained using various pairs of adaptive scaling factors $(\lambda_a, \lambda_\omega)$. The PI is a heuristically selected value when the pair of adaptive scaling factors is $(1, 0)$.



Chapter 4 A Novel Master Switching Method for Electronic Cam Control with Special Reference to Multi-Axis Coordinated Trajectory Following

Multi-axis coordinated trajectory following is important in CNC machines and metal cutting tools. Recently, flight simulators with electrical actuators have been in increasing demand. However, the coordinate control scheme affects the accuracy of the motion because motors have an insufficient load capacity relative to the hydraulic actuators. The electronic cam (ECAM) is typically used to perform coordinated control. However, selection of the master may determine potentially very different characteristics of motion. This study proposes an automatic master switching method. The conditions and results of the master switching method for electronic cam are detailed. The robustness and stability of the proposed control system is also demonstrated using the well-known structured perturbation analysis tool, μ .



4.1 Method of Building Cam Profiles (Master-slaves Trajectories)

4.1.1 Polynomial curve-fitting

A polynomial curve-fitting method is proposed to build a continuous curve in order to fit a known discrete signal, and the established curve is treated as the piecewise-continuous cam profile (master-slaves trajectories). As presented in Fig. 4.2, T is the sampling time of the driving system and t_{VR} is the period of motion planning. The predictive planned N points are the known discrete commands for which t_{VR} equals N times T ; the cam profile of each axis can be expressed as a function of time index t , which describes the common relationship

between master and slaves, for $0 \leq t \leq N \cdot T$, and

$$f_i(t) = \sum_{n=0}^{N-1} c_{i,n} \cdot t^n, \quad i = 0 \text{ to } q \quad (4.1)$$

in which q is the numbers of all axes including master and all slaves. By expanding Eq. (4.1),

then

$$\begin{bmatrix} 1 & 0 & \dots & 0 \\ 1 & T & \dots & T^{N-1} \\ \dots & \dots & \dots & \dots \\ 1 & (N-1)T & \dots & [(N-1)T]^{N-1} \end{bmatrix} \cdot \begin{bmatrix} c_{i,0} \\ c_{i,1} \\ \dots \\ c_{i,N-1} \end{bmatrix} = \begin{bmatrix} f_i(0) \\ f_i(T) \\ \dots \\ f_i((N-1)T) \end{bmatrix}$$

$$\Rightarrow \mathbf{T}_{matrix} \cdot \mathbf{C}_i = \mathbf{F}_i \quad (4.2)$$

$$\Rightarrow \mathbf{C}_i = \mathbf{T}_{matrix}^{-1} \cdot \mathbf{F}_i$$

where $f_i(t)$ is the position of the i th axis motion planning with respect to time index t , which normally equals the planning time kT , unless an external equivalent force acts on an axis exceeds the critical value, and further, \mathbf{T}_{matrix} , \mathbf{C}_i and \mathbf{F}_i are the constant time matrix, the polynomial parameters of the i th axis and the predicted positions of the i th axis, respectively. The matrix \mathbf{T}_{matrix} is constant and nonsingular so \mathbf{T}_{matrix}^{-1} exists. Adequately estimating the master's next position $f_m(t)$ enables the above equation to be used to determine the time index t , and then the estimated positions of all of the slaves $f_{i,s}(t)$ ($i = 1$ to $q-1$) are determined by substituting t into Eq. (4.1). Note that the subscript m and i,s over all this chapter mean the axis of the master and the i th axis of the slaves, respectively. The algorithm includes the following steps.

1. Estimating the next position of the master is an electronic gearing process, and the proper

estimate is expressed follows.

$$f_m(\hat{t}_{k+1}) = \hat{x}_{k+1} = x_k + \hat{v}_k \cdot T \quad (4.3)$$

where \hat{x}_{k+1} is the estimated position of the master; x_k is the present measured position of the master, and \hat{v}_k is the velocity estimated during the process of motion planning.

2. Substituting the estimated position \hat{x}_{k+1} of the master into Eq. (4.1) yields,

$$f_m(\hat{t}_{k+1}) = \hat{x}_{k+1} = \sum_{n=0}^{N-1} c_{m,n} \cdot \hat{t}_{k+1}^n \quad (4.4)$$

This equation generally has $N-1$ solutions, and only one real rational solution is correct. A proper constraint $\hat{t}_k < \hat{t}_{k+1} \leq (k+1)T$ is added to Eq. (4.4) to limit the region in which the solution may be found. Sometimes, two solutions satisfy this constraint, but identifying the correct one is not difficult. According to the properties of the polynomial curve and the planned velocities of the master, the sign of the slope of the curve plotted against the time index \hat{t}_{k+1} must be the same as that of the ideal velocity \hat{v}_k . For example, in Fig. 4.4, the solution near $(k+1)T$ is the correct one.

The master velocity in terms of the time index \hat{t}_{k+1} is expressed as,

$$f_m'(\hat{t}_{k+1}) = d\hat{x}_{k+1} / dt = \sum_{n=0}^{N-1} n c_{m,n} \cdot \hat{t}_{k+1}^{n-1} \quad (4.5)$$

such that,

$$\text{sign}\left(\sum_{n=0}^{N-1} n c_{m,n} \cdot \hat{t}_{k+1}^{n-1}\right) = \text{sign}(\hat{v}_k) \quad (4.6)$$

where,

$$\text{sign}(\cdot) = \begin{cases} 1, & \text{as } (\cdot) \text{ is positive} \\ -1, & \text{as } (\cdot) \text{ is negative} \\ 0, & \text{as } (\cdot) \text{ is zero} \end{cases}$$

3. The time index is estimated in the preceding steps, and the estimated position of the i th slave is represented as,

$$f_{i,s}(\hat{t}_{k+1}) = \sum_{n=0}^{N-1} c_{i,n} \hat{t}_{k+1}^n, \quad i = 1 \text{ to } 5 \quad (4.7)$$

4.1.2 Poly-line Curve-Fitting

The poly-line curve-fitting method is used to fit the signal of higher frequency according to the viewpoint of Nyquist frequency. And then yields a poly-line curve as shown in Fig. 4.3. If the number of motion planning points equals N , then as in the section 4.1.1, the cam profile can be expressed as a function of the time index t .

$$f_i(t) = \sum_{n=0}^{N-1} c_{i,n} \cdot |t - nT|, \quad i = 0 \text{ to } q \quad (4.8)$$

Expanding Eq. (24) yields

$$\begin{bmatrix} 0 & T & \dots & (N-1)T \\ T & 0 & \dots & (N-2)T \\ \dots & \dots & \dots & \dots \\ (N-1)T & (N-2)T & \dots & 0 \end{bmatrix} \cdot \begin{bmatrix} c_{i,0} \\ c_{i,1} \\ \dots \\ c_{i,N-1} \end{bmatrix} = \begin{bmatrix} f_i(0) \\ f_i(T) \\ \dots \\ f_i((N-1)T) \end{bmatrix}$$

$$\Rightarrow \mathbf{T}_{matrix} \cdot \mathbf{C}_i = \mathbf{F}_i \quad (4.9)$$

$$\Rightarrow \mathbf{C}_i = \mathbf{T}_{matrix}^{-1} \cdot \mathbf{F}_i$$

where the parameters in Eqs. (4.8) and (4.9) are all defined as in the above section. Similarly,

matrix \mathbf{T}_{matrix} is constant and nonsingular; thus, \mathbf{T}_{matrix}^{-1} exists.

The next time index \hat{t}_{k+1} is properly determined by substituting the estimated position \hat{x}_{k+1} of the master into Eq. (4.8) and considering the following conditions.

Case 1: $0 \leq t_{k+1} \leq T$

$$(c_0 - \sum_{i=1}^{N-1} c_i)t_{k+1} + \sum_{i=1}^{N-1} ic_i T = \hat{x}_{k+1}$$

Case 2: $T < t_{k+1} \leq 2T$

$$(\sum_{i=0}^1 c_i - \sum_{i=2}^{N-1} c_i t_{k+1}) + (-c_1 + \sum_{i=2}^{N-1} ic_i)T = \hat{x}_{k+1}$$

...

Case N-1: $(N-2)T < t_{k+1} \leq (N-1)T$

$$(\sum_{i=0}^{N-2} c_i - c_{N-1})t_{k+1} + (-\sum_{i=1}^{N-2} ic_i + (N-1)c_{N-1})T = \hat{x}_{k+1}$$

Under these conditions, the general formulation is as follows.

$$\hat{t}_{k+1} = [\hat{x}_{k+1} + \sum_{n=1}^{N-1} \text{sign}(\hat{t}_{k+1} - nT)c_n \cdot nT] / \sum_{n=0}^{N-1} c_n \cdot \text{sign}(\hat{t}_{k+1} - nT) \quad (4.10)$$

This equation is solved first by determining whether the value of $(\hat{t}_{k+1} - nT)$ is positive or negative. Restated, the probable region of \hat{t}_{k+1} must be determined correctly. The region $t_k < \hat{t}_{k+1} \leq (k+1)T$ is the correct choice, where t_k is the actual time index obtained by substituting the actual master's position x_k into Eq. (4.10) at time kT . Multi solutions may be in this region, so the correct solution of Eq. (4.10) must next be identified. As aforementioned, the sign of the slope of the poly-line function of the time index \hat{t}_{k+1} must be the same as that of the ideal velocity \hat{v}_k . That is,

$$\text{sign}(df(\hat{t}_{k+1})/dt) = \text{sign}(\hat{v}_k) \quad (4.11)$$

From the above analysis, the time index \hat{t}_{k+1} can be estimated; then, the estimated position of the i th slave can be represented as,

$$f_{i,s}(\hat{t}_{k+1}) = \sum_{n=0}^{N-1} c_{n,i} \cdot |\hat{t}_{k+1} - nT|, \quad i = 1 \text{ to } 5 \quad (4.12a)$$

4.2 Infinity Norm of the Master Switching ECAM Controller

By rearranging Eq. (4.12a) in a matrix form, the control input of the i th axis of the master switching method can be expressed as

$$y_{c,i} = f_{i,s}(t) = \Gamma_t \cdot C_i = \Gamma_t \cdot T_{matrix}^{-1} \cdot F_i, \quad (4.12b)$$

where $\Gamma_t = [|\hat{t}|, |\hat{t} - T|, \dots, |\hat{t} - (N-1)T|]$

Then, from the characteristics of the master switching ECAM control scheme, the actual speed of each axis theoretically does not exceed its reference speed. Therefore, the control displacement $y_{c,i}$ is confined by $|y_{c,i}| \leq |y_{r,i}|$, where $y_{r,i}$ is reference displacement input of the i th axis and is also the component of vector F_i ; that is,

$$|\Gamma_t \cdot T_{matrix}^{-1} \cdot F_i| \leq |y_{r,i}| \quad (4.13a)$$

and

$$|F_i|_{\infty} \geq |y_{r,i}|_{\infty} \quad (4.13b)$$

which implies the infinity norm of the controller $G_2 = \Gamma_t \cdot T_{matrix}^{-1}$ is confined by,

$$\|\Gamma_t \cdot T_{matrix}^{-1}\|_{\infty} \leq 1 \quad (4.13c)$$

4.3 Applying the Proposed Control Scheme to a Six DOF Motion Simulator

This proposed master switching ECAM control scheme is applied to the control system of multi-axes mechanisms to demonstrate its advantages. In this dissertation, the six DOF motion simulator SP-120 (Fig. 3.4) is used to implement the generalized ECAM tracking technique. If the current (force) of the most heavily loaded axis reaches its critical value, then the cockpit cannot easily execute its planned motion easily by directly feeding individual, planned commands to each axis. Rather, the cockpit may sometimes leave its nominal workspace. Accordingly, the master switching ECAM control scheme is better suited than the master fixed ECAM method to this application.

The master of the flight simulator is predetermined the heaviest loaded axis, so the Jacobian matrix [41] of the simulator must be calculated and updated from time to time. To find the most heavily loaded axis of the six DOF motion simulator, SP-120, the Jacobian of the simulator should be calculated by the following sub-sections.

4.3.1 Inverse Kinematics

Inverse kinematics of the simulator SP-120 is stated in section 3.1.

4.3.2 Jacobian formulation of simulator SP-120

From Eq. (3.1),

$$q_{xi} \cdot dq_{xi}/dt + (q_{yi} - p_{yi}) \cdot d(q_{yi} - p_{yi})/dt + q_{zi} \cdot dq_{zi}/dt = 0 \quad (4.14)$$

$$\begin{aligned} dp_{y_i}/dt &= [q_{x_i}/(q_{y_i} - p_{y_i}) \ 1 \ q_{x_i}/(q_{y_i} - p_{y_i})]^{S_i} [dq_{x_i}/dt \ dq_{y_i}/dt \ dq_{z_i}/dt]^T \\ &= [r_{x_i} \ r_{y_i} \ r_{z_i}]^{S_i} [dq_{x_i}/dt \ dq_{y_i}/dt \ dq_{z_i}/dt]^T, \quad i = 1 \text{ to } 6 \end{aligned} \quad (4.15)$$

where, $[r_{x_i} \ r_{y_i} \ r_{z_i}] = [q_{x_i}/(q_{y_i} - p_{y_i}) \ 1 \ q_{x_i}/(q_{y_i} - p_{y_i})]$; the superscript “T” represents the transpose of the matrix and all the parameters are considered in the S_i coordinate frame. From Eq. (3.2)

$$\begin{aligned} {}^{S_i}[dq_{x_i}/dt \ dq_{y_i}/dt \ dq_{z_i}/dt]^T &= R(\theta_{x_i}, \theta_{y_i}, \theta_{z_i}) \{ [dX_G/dt \ dY_G/dt \ dZ_G/dt]^T \\ &\quad + (R_\alpha \cdot d\alpha/dt + R_\beta \cdot d\beta/dt + R_\gamma \cdot d\gamma/dt) \cdot {}^G[q_{x_i} \ q_{y_i} \ q_{z_i}]^T \} \end{aligned} \quad (4.16)$$

where R_α is the partial derivative of ${}^0_G R(\alpha, \beta, \gamma)$ with respect to α .

$$R_\alpha = \begin{bmatrix} 1 & 0 & 0 \\ c\alpha s\beta c\gamma - s\alpha s\gamma & -c\alpha s\beta s\gamma - s\alpha c\gamma & -c\alpha c\beta \\ s\alpha s\beta c\gamma + c\alpha s\gamma & -s\alpha s\beta s\gamma + c\alpha c\gamma & -s\alpha c\beta \end{bmatrix} \quad (4.17)$$

R_β is the partial derivative of ${}^0_G R(\alpha, \beta, \gamma)$ with respect to β .

$$R_\beta = \begin{bmatrix} -s\beta c\gamma & s\beta s\gamma & c\beta \\ s\alpha c\beta c\gamma & -s\alpha c\beta s\gamma & s\alpha s\beta \\ -c\alpha c\beta c\gamma & c\alpha c\beta s\gamma & -c\alpha s\beta \end{bmatrix} \quad (4.18)$$

R_γ is the partial derivative of ${}^0_G R(\alpha, \beta, \gamma)$ with respect to γ .

$$R_\gamma = \begin{bmatrix} -c\beta s\gamma & -c\beta c\gamma & 0 \\ -s\alpha s\beta s\gamma + c\alpha c\gamma & -s\alpha s\beta c\gamma - c\alpha s\gamma & 0 \\ c\alpha s\beta s\gamma + s\alpha c\gamma & c\alpha c\beta s\gamma - s\alpha s\gamma & 0 \end{bmatrix} \quad (4.19)$$

Substituting Eq. (4.16) into Eq. (4.15) yields

$$\begin{aligned} dp_{y_i}/dt &= [r_{x_i} \ r_{y_i} \ r_{z_i}] \cdot R(\theta_{x_i}, \theta_{y_i}, \theta_{z_i}) \{ [dX_G/dt \ dY_G/dt \ dZ_G/dt]^T \\ &\quad + (R_\alpha \cdot d\alpha/dt + R_\beta \cdot d\beta/dt + R_\gamma \cdot d\gamma/dt) \cdot {}^G[q_{x_i} \ q_{y_i} \ q_{z_i}]^T \} \\ &= [r_{x_i} \ r_{y_i} \ r_{z_i}] \cdot R(\theta_{x_i}, \theta_{y_i}, \theta_{z_i}) \{ [dX_G/dt \ dY_G/dt \ dZ_G/dt]^T \} \end{aligned}$$

$$+[R_\alpha \cdot Q_i, R_\beta \cdot Q_i, R_\gamma \cdot Q_i] \cdot [d\alpha/dt, d\beta/dt, d\gamma/dt]^T \} \quad (4.20)$$

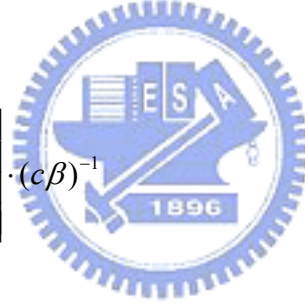
where $Q_i = {}^G [q_{xi} \ q_{yi} \ q_{zi}]^T$. However, α , β , and γ are the Euler angles measured in the body embedded coordinate frame, which is the cockpit coordinate system. When dealing with angular velocity, the inertial frame must be the reference frame. Let $[\varpi_x \ \varpi_y \ \varpi_z]^T$ be the cockpit angular velocity measured in the inertial frame X-Y-Z-O. Then,

$$[\varpi_x \ \varpi_y \ \varpi_z]^T = \begin{bmatrix} 1 & 0 & s\beta \\ 0 & c\alpha & -s\alpha c\beta \\ 0 & s\alpha & c\alpha c\beta \end{bmatrix} \cdot \begin{bmatrix} d\alpha/dt \\ d\beta/dt \\ d\gamma/dt \end{bmatrix} = \mathbf{M} \cdot \begin{bmatrix} d\alpha/dt \\ d\beta/dt \\ d\gamma/dt \end{bmatrix} \quad (4.21)$$

$$\therefore [d\alpha/dt \ d\beta/dt \ d\gamma/dt]^T = \mathbf{M}^{-1} \cdot [\varpi_x \ \varpi_y \ \varpi_z]^T \quad (4.22)$$

where,

$$\mathbf{M}^{-1} = \begin{bmatrix} c\beta & s\alpha s\beta & -c\alpha s\beta \\ 0 & c\alpha c\beta & s\alpha c\beta \\ 0 & -s\alpha & c\alpha \end{bmatrix} \cdot (c\beta)^{-1} \quad (4.23)$$



Substituting Eq. (4.22) into Eq. (4.20) yields,

$$\begin{aligned} dp_{yi}/dt &= [r_{xi} \ r_{yi} \ r_{zi}] \cdot R(\theta_{xi}, \theta_{yi}, \theta_{zi}) \{ [dX_G/dt \ dY_G/dt \ dZ_G/dt]^T \\ &\quad + [R_\alpha \cdot Q_i, R_\beta \cdot Q_i, R_\gamma \cdot Q_i] \cdot \mathbf{M}^{-1} [\varpi_x \ \varpi_y \ \varpi_z]^T \} \end{aligned} \quad (4.24)$$

According to the definition of Jacobian matrix \mathbf{J} , the joint space is converted into Cartesian space, such that,

$$d\mathbf{X}/dt = \mathbf{J} \cdot d\Theta/dt \quad (4.25)$$

where $X_{6 \times 1} = [X_G \ Y_G \ Z_G \ \alpha \ \beta \ \gamma]^T$, and $\Theta = [p_{y1}, p_{y2}, \dots, p_{y6}]^T$, and

$$d\Theta/dt = \mathbf{J}^{-1} \cdot d\mathbf{X}/dt \quad (4.26)$$

As mentioned above, the angular velocity in the inertial frame is more meaningful than that measured in the body embedded frame. Form Eq. (4.26), the elements of \mathbf{J}^{-1} can be summarized directly as follows.

The first part of Eq. (4.26) comprises the first three columns of \mathbf{J}^{-1}

$$[J_{i1}^{-1} \ J_{i2}^{-1} \ J_{i3}^{-1}] = [r_{xi} \ r_{yi} \ r_{zi}] \cdot R(\theta_{xi}, \theta_{yi}, \theta_{zi}), \quad i = 1 \text{ to } 6 \quad (4.27)$$

The second part of Eq. (4.26) comprises the last three columns of \mathbf{J}^{-1}

$$[J_{i4}^{-1} \ J_{i5}^{-1} \ J_{i6}^{-1}] = [r_{xi} \ r_{yi} \ r_{zi}] \cdot R(\theta_{xi}, \theta_{yi}, \theta_{zi}) + [R_{\alpha} \cdot Q_i, \ R_{\beta} \cdot Q_i, \ R_{\gamma} \cdot Q_i] \cdot M^{-1} \quad (4.28)$$

4.3.3 Calculate the loaded torque of each joint using Jacobian matrix

The relationship between the 6×1 joint torque vector $\boldsymbol{\tau}$, $\boldsymbol{\tau} = [\tau_1, \tau_2, \tau_3, \tau_4, \tau_5, \tau_6]^T$, and the 6×1 equivalent Cartesian force-moment vector \mathbf{F} , $\mathbf{F} = [m\mathbf{a}, \mathbf{I}_{3 \times 3} \boldsymbol{\alpha}]^T$, acting at the mass center of the upper plate, can be written in the form [41]

$$\boldsymbol{\tau} = \mathbf{J}^T \mathbf{F}. \quad (4.29)$$

where m denotes the mass of the simulator's cockpit in this section; $\mathbf{a} \in \mathbf{R}^3$ and $\boldsymbol{\alpha} \in \mathbf{R}^3$ are the linear acceleration vector and the angular acceleration vector of the cockpit, respectively;

$\mathbf{I}_{3 \times 3}$ represents the moment of inertia of the cockpit.

4.4 Analysis of Stability and Robustness

The dynamics of each slider of the SP-120 motion simulator (Figs. 3.4 and 3.5) can be modeled by parametric uncertainties, using the linear fractional transformation (LFT)

representation. An equivalent mass, m , and a nominal damping coefficient, c , are introduced to simplify the dynamics of the slider motion and to decouple the components of the system's nonlinear terms, to explicate the stability and the robust performance of the system. Thus, a simplified dynamic model of each slider is,

$$\tau = u \cdot s_p / 2\pi = -K_n \dot{\theta} + K_c E_c, \quad (4.30a)$$

and

$$x = \theta \cdot s_p / 2\pi \quad (4.30b)$$

where x is the displacement of each slider in this section; s_p is the lead screw pitch; u represents the force applied to slider; $\dot{\theta}$ is the angular velocity of motor's shaft; K_c is the AC servo motor constant (here is 0.0529); K_n is the AC servo motor constant (here is 0.00242552); E_c is the input voltage of the servo-motor

$$u = -K_f^2 K_n \dot{x} + K_f K_c E_c \quad (4.31)$$

where $K_f = 2\pi / s_p$ and \dot{x} represent the machine constant and the slider's linear velocity, respectively. As presented in Fig. 4.5, the slider's linear acceleration can be expressed as

$$\ddot{x} = -(c/m)\dot{x} + u/m \quad (4.32)$$

Suppose that the physical parameters m and c are not known exactly, but are believed to lie in known intervals. Assume,

$$m = \bar{m} + \Delta_m \delta_m, c = \bar{c} + \Delta_c \delta_c \quad (4.33)$$

where the nominal mass is $\bar{m} = (m_H + m_L) / 2$, and the nominal damping is

$\bar{c} = (c_H + c_L)/2$; the maximum variation of mass is $\Delta_m = (m_H - m_L)/2$, and the maximum variation of damping is $\Delta_c = (c_H - c_L)/2$; the perturbations δ_m and δ_c are confined by $|\delta_m| < 1$ and $|\delta_c| < 1$, respectively, in which $m_H = 250$ kg, $c_H = 15$ kg/s and $m_L = 50$ kg, $c_L = 5$ kg/s are in practice the upper and lower bounds of the slider's nominal mass and damping, respectively.

Figure 4.6 presents the system's block diagram according to the foregoing dynamical equations. Suppose the control input is $[w_1, w_2, y_{c,i}]^T$ and the output is $[z_1, z_2, y]^T$. Then, using the Doyle's representation, the transformation matrix can be represented as below.

$$\mathbf{M} = \left[\begin{array}{ccc|ccc}
 0 & 1 & 0 & 0 & 0 & 0 \\
 -K_c K_f^2 K_a / \bar{m} & -(c + K_f^2 K_n) / \bar{m} & -1 / \bar{m} & -1 / \bar{m} & K_c K_f^2 K_a / \bar{m} & \\
 \hline
 -\Delta_m K_c K_f^2 K_a / \bar{m} & -\Delta_m (c + K_f^2 K_n) / \bar{m} & -\Delta_m / \bar{m} & -\Delta_m / \bar{m} & \Delta_m K_c K_f^2 K_a / \bar{m} & \\
 0 & \Delta_c & 0 & 0 & 0 & 0 \\
 \hline
 1 & 0 & 0 & 0 & 0 & 0
 \end{array} \right] \quad (4.34)$$

$$= \left[\begin{array}{c|c}
 M_{11} & M_{12} \\
 \hline
 M_{21} & M_{22}
 \end{array} \right] \equiv G_1(s)$$

and the transfer function is

$$G_1(s) = M_{22} + M_{21}(sI - M_{11})^{-1}M_{12} \quad (4.35)$$

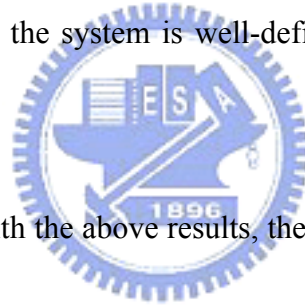
where K_a is the proportional gain in position loop, and the system including the perturbations δ_m and δ_c , can be represented using LFT. That is,

$$y = \mathfrak{F}_u(M, \Delta)y_r, \Delta = \begin{bmatrix} \delta_m & \\ & \delta_c \end{bmatrix} \in RH_\infty, \quad w = \begin{bmatrix} w_1 \\ w_2 \end{bmatrix} = \Delta \cdot \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \Delta \cdot z \quad (4.36)$$

where $\mathfrak{F}_u(M, \Delta)$ is the upper LFT, as shown in Fig. 4.7, and $\Delta \in RH_\infty$ is the structured uncertainty. Stability is often not the only property of a closed-loop system that must be

robust to perturbations. The most well-known use of μ as a robustness analysis tool is in the frequency domain. Figures 4.8 and 4.9 show the singular value frequency responses of $G_1(j\omega)$ and the structured singular values, $\mu_\Delta(G_1(j\omega))$, respectively, for each frequency with $\Delta \in C^{2 \times 2}$, obtained by adjusting the proportional gain, K_a . These figures are obtained by programming the theorem of μ [2]. Moreover, the bounds of $\mu_\Delta(G_1(j\omega))$ are formulated within the reference book [39] as presented in appendix B. In Figs. 4.8 and 4.9, the maximum singular value of $G_1(j\omega)$ is increased by decreasing the proportional gain, and the maximum structured singular value is increased by increasing the proportional gain. Table 4.1 presents the maximum singular values $\|G_1(j\omega)\|_\infty$, the maximum structured singular values $\sup_{\omega \in R} \mu_\Delta(G_1(j\omega))$ and the bandwidth of the control system for various proportional gains. Moreover, if the upper bound of the nominal mass exceeds a critical value, then the maximum structured singular value will be larger than unity, possibly causing the requirement for robust performance to be unsatisfied. Table 4.2 presents the critical upper bounds of m_H for various proportional gains, K_a . The critical upper bound increases as the proportional gain decreases. Combining Table 4.1 and Table 4.2 reveals that the system is more robustly stable at a lower proportional gain, but the time constant of the system responses is higher. Thus, a tradeoff exists between the robustness and the performance of the system's response. Nevertheless, by carefully considering this tradeoff, the most suitable proportional gain can be conveniently adjusted to fit the specific demands of the control. In

this paper, m_H is estimated to be around 250 kg by transforming the maximum torque of each joint of the motion simulator SP-120 to the equivalent mass. The maximum torque is obtained by applying the critical velocity and the maximum tolerable acceleration to drive the slider of the flight simulator provided traveling most the nominal workspace of the simulator. Moreover, for example, if the damping ratio is set to 0.707, then the proportional gain must be adjusted to 6.3, and the maximum structured singular value is then calculated as 0.801358. Clearly, the sufficient and necessary condition for robust performance is satisfied. That is, the maximum structured singular value must be less than unity. Consequently, according to the theorem of μ and μ -synthesis, the system is well-defined and internally stable under the structured perturbation, $\|\Delta\|_\infty < 1$.



By combining Eq. (4.13c) with the above results, the maximum structured singular value of the entire system, G_1G_2 , is confined by the following inequality.

$$\sup_{\omega \in R} \mu_\Delta(G_1(j\omega)G_2(j\omega)) \leq \sup_{\omega \in R} \mu_\Delta(G_1(j\omega)) < 1 \quad (4.37)$$

Restated, the master switching control system is more robustly stable than the original stable system.

4.5 Numerical Method for the Forward Kinematics of Six DOF Flight Simulator

The cockpit trajectories obtained using conventional tracking control and the proposed tracking control, are compared to demonstrate the precision of the proposed control scheme.

Therefore, the six sliders must be transformed into the cockpit positions off-line; that is, forward kinematics will be used to transform the six axis coordinates into the cockpit's coordinates, including translation components and rotation components (and representing a transformation from J to S). However, direct forward kinematics is difficult to formulate for a six DOF flight simulator. Therefore, this study proposes the use of a numerical method, such as Newton's method to execute the transformation (J to S) indirectly. The following iterative steps describe the numerical, steepest descent approach [11, 42].

1. Set $k = 0$, and set the initial cockpit position, x_0 , to the cockpit home position.
2. Calculate the present Jacobian matrix J_k , according to the algorithm presented in the Appendix A.
3. Calculate the estimated errors in the positions of the six sliders as,

$$\alpha_k = p_y - p_{est,k} \in R^6 \quad (4.38)$$

where p_y is the actual positions of six slider, $p_{est,k}$ is the estimated positions of the six sliders, calculated by inverse kinematics, and α_k is the chosen step size.

4. Calculate the next estimated cockpit position,

$$x_{k+1} = x_k + J_k \cdot \alpha_k \quad (4.39)$$

where the Jacobian J_k matrix is the equivalent gradient matrix.

5. If $\|x_{k+1} - x_k\|_2 < \varsigma$ or $\|\alpha_k\|_2 < \xi$, terminate the iteration; the approximate cockpit position is x_{k+1} , where ς and ξ are the set maximum tolerable errors.

6. Set $k = k + 1$; repeat steps 2 to 5.

The convergence of this algorithm takes about two to three iterative loops, given the setting

$\zeta = 1e-12$ and $\xi = 1e-12$.

4.6 Experimental Results and Comparisons

In this study, the proposed ECAM tracking scheme is used on the SP-120 simulator to simulate ground earthquake signal received at Shui-Li Primary School on September 21, 1998.

Figure 4.10 shows a part of this ground earthquake signal. Figure 4.11 presents the power spectrum density of this signal at various frequencies. As aforementioned, the frequencies of the signal are not all less than one-tenth of the Nyquist frequency (here is 50 Hz). Accordingly, poly-line curve-fitting method is used in the proposed control scheme.

Figures 4.12 ~ 4.14 compare the Euler's roll angle errors, pitch angle errors and yaw angle errors, respectively, between the conventional and proposed method. This ground earthquake signal involves only the translation; restated, the simulator's output attitude must not include a rotational component. However as stated above, the six axes may mutually pull and drag each other, causing rotational motion during this pure translation. Table 4.3 presents the root mean square (RMS) errors of Euler angles for using the proposed ECAM tracking scheme and the master fixed ECAM tracking method executed on the simulator SP-120. In this simulation, the poly-line curve-fitting method is used to establish the ECAM profile and

the positioning accuracy depends on the system sampling time: a smaller sampling time yields greater accuracy. However, a tradeoff exists between the calculation time and the system sampling frequency. For example, with a calculation time of around $0.5 \sim 1 \text{ ms}$, the system sampling frequency may be set to 100 Hz. Therefore, some small errors still occur (as shown in Figs. 4.12 ~ 4.14) even if the master switching tracking control is applied to the simulator system. Thus, higher performance computers clearly track more precisely.



Chapter 5 System Integration

To demonstrate the adaptability of the proceeding motion control algorithm in a six DOF motion simulator, as shown in Fig. 1.1, this study integrates the dynamics of several various vehicles, the collision detection algorithm, the VR programming techniques, the sound effects, the motion control system, the cabin operating system and the force feedback steering system, to perform a human-machine interactive motion in real time.

5.1 Dynamics of Vehicle

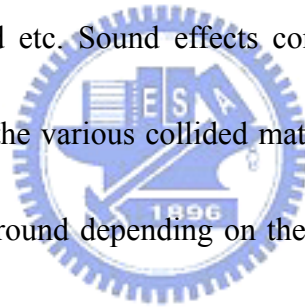
The dynamics of vehicle include operating mode, colliding mode and hopping mode. The operating mode must firstly calculate the engine's torque output with respect to the engine's rotation speed and the gear; then, the car's size, weight, static sliding friction coefficient of wheels, dynamic sliding friction coefficient of wheels, drag force of wheels' rolling friction with respect to the car speed, the wind drag force with respect to the car speed and the engine brake, all have to be considered; finally, suspension effects are also applied to the dynamics. Colliding mode include the dynamics of car vs. wall colliding, car vs. car colliding and car vs. other objects colliding. Hopping mode include the dynamics of starting hopping, hopping and grounding. Figs. 5.1 ~ 5.3 depict the results of these three dynamic modes.

5.2 Collision Detection

In order to make wheels always run on the surface of the virtual ground, the wheels have to detect the ground and modify the positions of the wheels. Also to avoid the car crossing the virtual wall, another car or other objects, the car must detect if it collides the virtual wall, another car or other objects.

5.3 VR Programming and Sound Effects

Virtual reality (VR) include the building of 3D models and textures of scene, data reading of 3D model, drawing, frame rate control, and special effects such as lighting, shadow, explosion, smoke, spark, ..., and etc. Sound effects consist of background music control, collision sound effect relative to the various collided material and the various collided force, the sound of wheels contacting ground depending on the ground material and the car speed, the sounds of other cars, ..., and etc.



5.4 Motion Control System

By combining the proposed motion cuing algorithm with the master switching ECAM control yields the motion control system, as presented in chapter 3 and chapter 4.

5.5 Cabin Operating System and Force Feedback Steering System

The cabin operating system consists of the display system, lighting system, control button

system and joystick system. The force feedback steering system is embedded in the cabin operating system, which includes the algorithm of wheels' force feedback and the driving system of torque control.



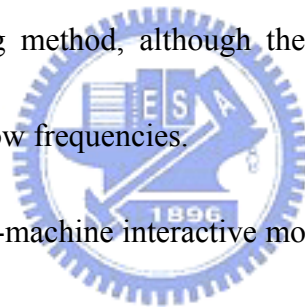
Chapter 6 Conclusion

In chapter 2, the proposed disturbance estimator can effectively suppress the external disturbance and the high frequency measurement noise, trading off delay time and the robustness of the estimator. As a result, higher order polynomial fitting must be adapted for a cam profile with a farther travel distance. The cam profile tracking is formulated as optimization in real-time control. A deterministic and unique solution is derived for all possible cases of tracking control. The proposed method is effective for general motion tracking control and guarantees a global optimal solution for practical control.

In chapter 3, the proposed washout strategy is a general method and can be used in many simulators with different workspaces and driving systems. However, it is particular effective in a simulator with a small workspace. This approach is practical and efficient, especially for use in motion simulators used for entertainment, with restricted workspaces. Furthermore, this study establishes the performance index to conveniently quantify the efficiency of motion as the reference for realism. Repeated tests were performed online; they demonstrated that the proposed washout filter yields much more realistic motion cues than the classical technique for a motion simulator with a restricted workspace and an inexpensive driving system.

In chapter 4, the master switching electronic cam tracking control is combined with the motion planning of a six DOF motion simulator. The displacements of slaves of the electronic cam control system depend on the displacement of the master; the master switching method

selects the most heavily loaded axis to be the master in real-time. The trajectory following speed yielded by the master switching method can be less than the speed yielded by the conventional (master fixed) method. Precision and robustness are the key concerns and the proposed method is sound. As aforementioned, by adjusting the proportional gain, a tradeoff exists between the robust stability and the velocity response of the control system. Using the well-known μ analysis of structured uncertainty, a most appropriate proportional gain may be chosen to satisfy the demand of control performance, provided robust stability is guaranteed. Furthermore, the poly-line curve-fitting method requires less computational time than the polynomial curve-fitting method, although the latter one may theoretically yield higher precision for a motion of low frequencies.



To perform a realistic human-machine interactive motion for entertainment demands, this dissertation finally roughly describes how to perform the system integration. That is to integrate the dynamics of several various vehicles, the collision detection algorithm, the VR programming techniques, the sound effects, the proceeding motion control system, the cabin operating system and the force feedback steering system.

This study uses the structural parametric method to model the dynamics of each ball joint slider; that may be just used to design the proportion gain, but is not direct and complete for designing a multivariable control system. Thus, the future research may focus on the dynamics and the design of multivariable controller for a six DOF motion simulator.

Appendix A

The physical parameters of the motor may be dynamically varied, so the effect of parameter uncertainty must also be discussed. The well-known analysis of the modeling uncertainty is the μ analysis [39]. A more direct method is to analyze the sensitivities ($S_{\bar{K}}^{G_c}$, $S_{\bar{J}}^{G_c}$ and $S_{\bar{B}}^{G_c}$) of the transfer function G_c to the motor's uncertain parameters, \bar{K} , \bar{J} and \bar{B} , respectively, where

$$G_c = \frac{K\bar{K}(\mathfrak{T}s + 1)}{K\bar{J}\mathfrak{T}s^2 + (K\bar{B}\mathfrak{T} + \bar{K}J)s + \bar{K}B} \quad (\text{A.1})$$

$$S_{\bar{K}}^{G_c} = \frac{\partial G_c}{\partial \bar{K}} \cdot \frac{\bar{K}}{G_c} = \frac{K\bar{J}\mathfrak{T}s^2 + K\bar{B}\mathfrak{T}s}{K\bar{J}\mathfrak{T}s^2 + (K\bar{B}\mathfrak{T} + \bar{K}J)s + \bar{K}B} \quad (\text{A.2})$$

$$S_{\bar{J}}^{G_c} = \frac{\partial G_c}{\partial \bar{J}} \cdot \frac{\bar{K}}{G_c} = \frac{-K\bar{J}\mathfrak{T}s^2}{K\bar{J}\mathfrak{T}s^2 + (K\bar{B}\mathfrak{T} + \bar{K}J)s + \bar{K}B} \quad (\text{A.3})$$

$$S_{\bar{B}}^{G_c} = \frac{\partial G_c}{\partial \bar{B}} \cdot \frac{\bar{K}}{G_c} = \frac{-K\bar{B}\mathfrak{T}s}{K\bar{J}\mathfrak{T}s^2 + (K\bar{B}\mathfrak{T} + \bar{K}J)s + \bar{K}B} \quad (\text{A.4})$$

Figures A.1 (a) ~ (c) show the magnitudes of the three sensitivities in relation to the input frequency, where the parameters of the master motor are all set as in Section 3.5. According to Eqs.(A.2) and (A.3), the magnitudes of the sensitivities, $S_{\bar{K}}^{G_c}$ and $S_{\bar{J}}^{G_c}$, are both small for low-frequency motion. Figures A.1 (a) and (b) reveal that the magnitudes of the sensitivities, $S_{\bar{K}}^{G_c}$ and $S_{\bar{J}}^{G_c}$, are both less than 0.707 while the input frequency is lower than $1/\mathfrak{T}$ (Hz). Furthermore, according to Fig.A.1(c), the magnitude of the sensitivity $S_{\bar{B}}^{G_c}$ is less than 0.00086 over the entire frequency domain. From Eqs.(A.1) ~ (A.3) and the foregoing discussion, the low time constant (\mathfrak{T}) of the disturbance estimator suppresses the sensitivities, $S_{\bar{K}}^{G_c}$, $S_{\bar{J}}^{G_c}$ and $S_{\bar{B}}^{G_c}$.

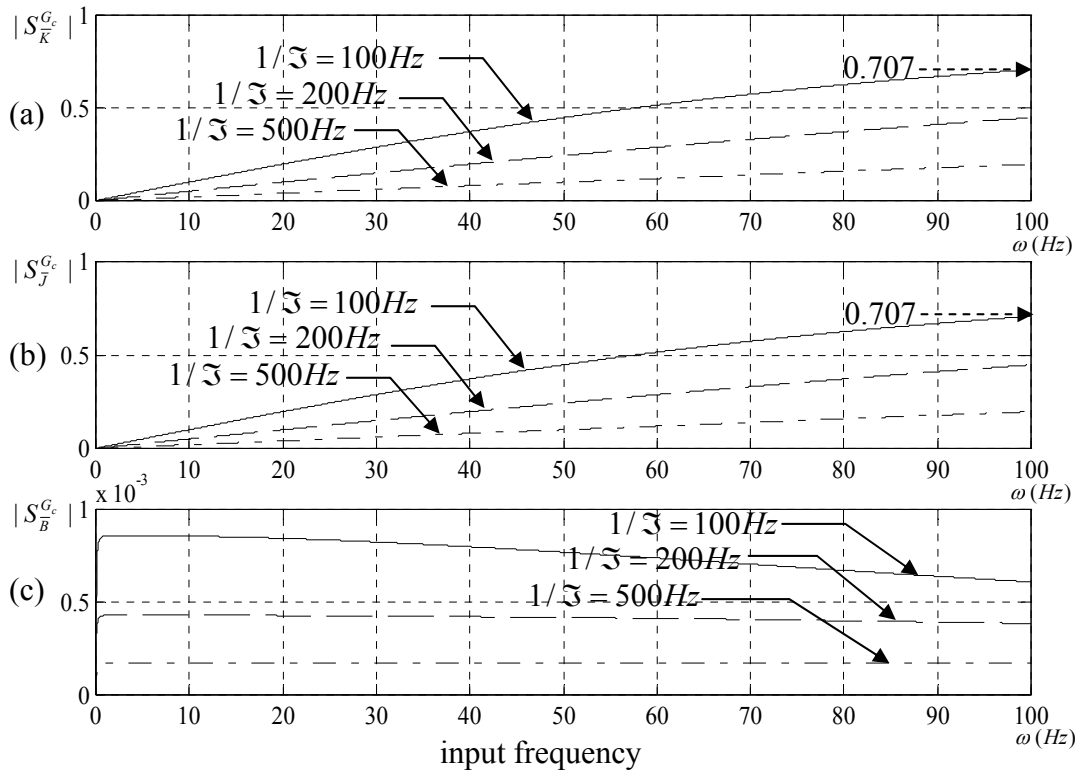


Figure A.1 Magnitudes of the sensitivities: (a) $S_K^{G_c}$, (b) $S_J^{G_c}$ and (c) $S_B^{G_c}$, in relation to the input frequency (ω) at various time constants (τ)

Appendix B

The following analysis of the structured singular value follows reference [39].

Suppose that the uncertainty block is given by

$$\Delta = \begin{bmatrix} \delta_m & \\ & \delta_c \end{bmatrix} \in RH_\infty, \text{ where } |\delta_m| < 1 \text{ and } |\delta_c| < 1,$$

with $\|\Delta\|_\infty < 1$ and that the interconnection model, $G_1(s)$ is rearranged as,

$$G_1(s) = \begin{bmatrix} G_{11}(s) & G_{12}(s) \\ G_{21}(s) & G_{22}(s) \end{bmatrix} \in RH_\infty \quad (\text{B.1})$$

Then, the closed-loop system is well-posed and internally stable iff $\sup_{\omega \in R} \mu_\Delta(G_1(j\omega)) \leq 1$. Let

$$D_\omega = \begin{bmatrix} d_\omega I & \\ & I \end{bmatrix}, d_\omega \in R_+ \quad (\text{B.2})$$

Then,

$$D_\omega G_1(j\omega) D_\omega^{-1} = \begin{bmatrix} G_{11}(j\omega) & d_\omega G_{12}(j\omega) \\ \frac{1}{d_\omega} G_{21}(j\omega) & G_{22}(j\omega) \end{bmatrix} \quad (\text{B.3})$$

Hence, by the theorem of Packard and Doyle [43], at each frequency ω ,

$$\mu_\Delta(G_1(j\omega)) = \inf_{d_\omega \in R_+} \bar{\sigma} \left(\begin{bmatrix} G_{11}(j\omega) & d_\omega G_{12}(j\omega) \\ \frac{1}{d_\omega} G_{21}(j\omega) & G_{22}(j\omega) \end{bmatrix} \right) \quad (\text{B.4})$$

where the function $\bar{\sigma}(\cdot)$ expresses the maximum singular value of (\cdot) . Since the

minimization is convex in $\log d_\omega$ (see, Doyle [44, 45]), the optimal d_ω can be found by a

search; however, two approximations [39] to d_ω can be obtained easily by approximating

the right-hand side of Eq. (B.4):

B.1 First approximation

$$\begin{aligned}
\mu_{\Delta}(G_1(j\omega)) &\leq \inf_{d_{\omega} \in \mathbb{R}^+} \bar{\sigma} \left(\begin{bmatrix} \|G_{11}(j\omega)\| & d_{\omega} \|G_{12}(j\omega)\| \\ \frac{1}{d_{\omega}} \|G_{21}(j\omega)\| & \|G_{22}(j\omega)\| \end{bmatrix} \right) \\
&\leq \sqrt{\inf_{d_{\omega} \in \mathbb{R}^+} \left(\|G_{11}(j\omega)\|^2 + d_{\omega}^2 \|G_{12}(j\omega)\|^2 + \frac{1}{d_{\omega}^2} \|G_{21}(j\omega)\|^2 + \|G_{22}(j\omega)\|^2 \right)} \\
&= \sqrt{\|G_{11}(j\omega)\|^2 + \|G_{22}(j\omega)\|^2 + 2 \cdot \|G_{12}(j\omega)\| \cdot \|G_{21}(j\omega)\|}
\end{aligned} \tag{B.5}$$

with minimized d_{ω} given by,

$$\hat{d}_{\omega} = \begin{cases} \sqrt{\|G_{21}(j\omega)\| / \|G_{12}(j\omega)\|}, & \text{if } G_{12} \neq 0 \ \& \ G_{21} \neq 0, \\ 0 & \text{, if } G_{21} = 0, \\ \infty & \text{, if } G_{12} = 0. \end{cases} \tag{B.6}$$

B.2 Alternative approximation can be obtained by using the Frobenius norm:

$$\begin{aligned}
\mu_{\Delta}(G_1(j\omega)) &\leq \inf_{d_{\omega} \in \mathbb{R}^+} \left\| \begin{bmatrix} G_{11}(j\omega) & d_{\omega} G_{12}(j\omega) \\ \frac{1}{d_{\omega}} G_{21}(j\omega) & G_{22}(j\omega) \end{bmatrix} \right\|_F \\
&= \sqrt{\inf_{d_{\omega} \in \mathbb{R}^+} \left(\|G_{11}(j\omega)\|_F^2 + d_{\omega}^2 \|G_{12}(j\omega)\|_F^2 + \frac{1}{d_{\omega}^2} \|G_{21}(j\omega)\|_F^2 + \|G_{22}(j\omega)\|_F^2 \right)} \\
&= \sqrt{\|G_{11}(j\omega)\|_F^2 + \|G_{22}(j\omega)\|_F^2 + 2 \cdot \|G_{12}(j\omega)\|_F \cdot \|G_{21}(j\omega)\|_F}
\end{aligned} \tag{B.7}$$

with minimized d_{ω} given by,

$$\tilde{d}_{\omega} = \begin{cases} \sqrt{\|G_{21}(j\omega)\|_F / \|G_{12}(j\omega)\|_F}, & \text{if } G_{12} \neq 0 \ \& \ G_{21} \neq 0, \\ 0 & \text{, if } G_{21} = 0, \\ \infty & \text{, if } G_{12} = 0. \end{cases} \tag{B.8}$$

Hence an approximation of μ can be obtained as

$$\mu_{\Delta}(G_1(j\omega)) \leq \bar{\sigma} \left(\begin{bmatrix} G_{11}(j\omega) & \hat{d}_{\omega} G_{12}(j\omega) \\ \frac{1}{\hat{d}_{\omega}} G_{21}(j\omega) & G_{22}(j\omega) \end{bmatrix} \right) \tag{B.9}$$

or, alternatively,

$$\mu_{\Delta}(G_1(j\omega)) \leq \bar{\sigma} \left(\begin{bmatrix} G_{11}(j\omega) & \tilde{d}_{\omega} G_{12}(j\omega) \\ \frac{1}{\tilde{d}_{\omega}} G_{21}(j\omega) & G_{22}(j\omega) \end{bmatrix} \right) \quad (\text{B.10})$$

These approximated μ are now determined.



References

- [1] Steven Chingyei Chung, Tracking Control for the Electric Gear System, National Science Council of the Republic of China 845-012-011, 1995, pp. 1-35.
- [2] Zhao Qing-Feng, Advanced Design for Automatic Control System, Chan Hwa Science and Technology Book Co., Chapter 4, Ltd, 2000, pp. 1-26.
- [3] Gavrilovic A. and Heath A. J., Prediction and Mitigation of System Disturbances to Industrial Loads, Sources and Effects of Power System Disturbances, IEE Conference publication, 1982, pp. 227-230.
- [4] Charles L. Phillips and H. Troy Nagle, Digital Control System Analysis and Design, Prentice-Hall International Editions, 1994, pp. 103-144.
- [5] H.S. Yan, M.C. Tsai and M.H. Hsu, An experimental study of the effects of cam speed on cam-follower systems, The Int. Journal of Mechanism and Machine Theory, vol.31, 1996, pp.397-412.
- [6] H.S. Yan, M.C. Tsai and M.H. Hsu, A variable-speed method for improving motion characteristics of cam-following system, ASME Journal of Mechanism Design, vol.118, 1996, pp.250-258.
- [7] Chen Li-Shan, Follower Motion Design in a Variable-Speed Cam System, A Thesis Submitted to Institute of Mechanical Engineering College of Engineering National Chiao Tung University, T NCTU MEENG pt.11:3, 1995, pp. 1-49.

- [8] Dierchx and Paul, Curve and Surface Fitting with Splines, Oxford University Press, 1993, pp. 75-91.
- [9] Mortenson Michael E., Geometric Modeling, Wiley-Interscience Publication, John Wiley & Sons, INC., 1985, pp. 91-146.
- [10] Kim Dean H. and Tsao Tsu-Chin, Robust Performance Control of Electrohydraulic Actuators for Electronic Cam Motion Generation, IEEE Transactions on Control Systems Technology, vol. 8, No. 2, 2000, pp 220-227.
- [11] Edwin K. P. Chong and Stanislaw H. ŻAK , An Introduction to Optimization, A Wiley-Interscience Publication, John Wiley & Sons, INC., 1996, pp. 101-164.
- [12] J. B. Sinacori, "A practical approach to motion simulation", Visual and Motion Simulation Conference, AIAA Paper 73-931, Palo Alto (CA), September 1973.
- [13] S.F. Schmidt and Conrad Bjorn, "Motion Drive Signals for Piloted Flight Simulators," Analytical Mechanics Associated, Technical Report Contract NAS2-4869, May 1970.
- [14] M. A. Nahon and L. D. Reid, "Simulator Motion-Drive Algorithms: A Designer's Perspective," Journal of Guidance, Control, and Dynamics, vol. 13, 1990, pp. 356-362.
- [15] D. Ariel and R. Sivan, "False Cue Reduction in Moving Flight Simulators," IEEE Transactions on Systems, Man and Cybernetics, SMC-14, No. 4, 1984, pp. 665-671.
- [16] R.L. Bowles, R.V. Parrish, J.E. Dieudonne, "Coordinated Adaptive Washout for Motion Simulators," Journal of Aircraft, vol. 12, No. 1, 1975, pp. 44-50.

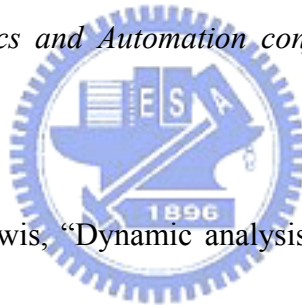
- [17] R. Sivan, J. Ish-shalom and J. K. Huang, "An Optimal Control Approach to the Design of Moving Flight Simulators," IEEE Transactions on Systems, Man and Cybernetics, SMC-12, No. 6, 1982, pp. 818-827.
- [18] W. Wu, and F. M. Cardullo, "Is There an Optimum Cueing Algorithm?" AIAA Modeling and Simulation Technologies Conference, New Orleans, LA, August 11-13, 1997, pp. 23-29.
- [19] L. D. Reid, M. A. Nahon and J. Kirdeikis, "Adaptive Simulator Motion Software with Supervisory Control," Journal of Guidance, Control, and Dynamics, Vol. 15, No. 2, 1992, pp. 376-383.
- [20] Moshe Idan and David Sahar, "Robust Controller for a Dynamic Six Degree of Freedom Flight Simulator," In AIAA Proceeding Of Conference On Flight Simulator Technologies, 1996, p. 53-60.
- [21] I. Moshe and M. A. Nahon, "Offline Comparison of Classical and Robust Flight Simulator Motion Control," Journal of Guidance, Control, and Dynamics, Vol. 22, No. 5, 1999, pp. 702-709.
- [22] Edward A. Martin, "Motion and Force Cuing, Part I: Whole Body Motion," Flight & Ground Simulation Update, State University of New York, Binghamton NY, 2000, pp. 5-18.
- [23] B. Conrad and S. F. Schmidt, "A Study of Techniques for Calculating Motion Drive

Signals for Flight Simulators,” NASA CR-114345, July 1971.

[24] L. D. Reid, M. A. Nahon, “Flight Simulator Motion-based Drive Algorithm: Part 3 – Pilot Evaluations,” Technical Report UTIAS Report 319, Univ. of Toronto – Canada, December 1986.

[25] Nicolas A. Pouliot, Clement M. Gosselin and Meyer A. Nahon, “Motion Simulation Capabilities of Three-Degree-of-Freedom Flight Simulators,” *Journal of Aircraft*, Vol. 35, No. 1, 1998, pp. 9-17.

[26] W. Khalil and S. Guegan, “A novel solution for the dynamic modeling of Gough-Stewart manipulators,” *IEEE Robotics and Automation conference*, Washington D.C., U.S.A., May 2002, pp. 817-822.



[27] G. Lebet, K. Liu and F. Lewis, “Dynamic analysis and control of a Stewart platform manipulator,” *J. of Robotic Systems*, July 1993, 10(5):629-655.

[28] Gosselin C. “Parallel computational algorithms for the kinematics and dynamics of planar and spatial parallel manipulators”, *ASME J. of Dynamic Systems, Measurement and Control*, 1996, 118(1):22-28.

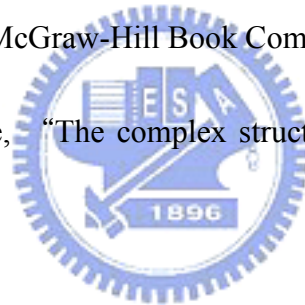
[29] J.-P. Merlet, “*Parallel Robots*,” Kluwer Academic Publishers, 2000.

[30] Chung I-Fang, Chang Hung-Hsiang & Lin Chin-Teng, Fuzzy Control of a Six-degree Motion Platform with Stability Analysis. *IEEE International Conference on Systems*, vol. 1, 1999, pp 325-330.

- [31] Werner H., Robust control of a laboratory flight simulator by non-dynamic multi-rate output feedback. *Decision and Control, Proceedings of the 35th IEEE Conference on*, vol.2, 1996, pp 1575-1580.
- [32] Plummer A.R., Non-linear control of a flight simulator motion system. *Control Applications, Proceedings of the Third IEEE Conference on*, vol.1, 1994, pp 365-370.
- [33] Dixon R., Pike A.W., (2002). Application of condition monitoring to an electromechanical actuator: a parameter estimation based approach. *Computing & Control Engineering Journal*, vol. 13 Issue: 2, April 2002, pp 71-81.
- [34] Al-Muthairi N.F., Bingulac S., Zribi M., (2002), Identification of discrete-time MIMO systems using a class of observable canonical-form. *Control Theory and Applications, IEE Proceedings*, vol. 149, Issue: 2, March 2002, pp 125-130.
- [35] Zhiwen Zhu, Leung H., Identification of linear systems driven by chaotic signals using nonlinear prediction. *Circuits and Systems I: Fundamental Theory and Applications, IEEE Transactions on*, vol. 49 Issue: 2, Feb. 2002, pp 170-180.
- [36] D. Lazard, "Stewart platform and Gröbner basis", In *ARK*, Ferrare, 7-9 September 1992, pp. 136-142.
- [37] W. Khalil and E. Dombre, "Modeling, identification and control of robots," Hermes Sciences Europe, 2002.
- [38] Reich, Jens-Georg, *C curve fitting and modeling for scientists and engineers*. New York

McGraw-Hill.

- [39] Zhou Kemin, (1998), *Essentials of Robust Control*, Prentice Hall, Inc., Upper Saddle River, New Jersey, 1998, pp. 55-62, pp. 129-211.
- [40] Samuel Blackman and Robert Popoli, *Design and Analysis of Modern Tracking Systems*, Artech House Publishers, 1999, pp. 147-195.
- [41] John J. Craig, (1989). *Introduction to Robotics Mechanics and Control – 2nd ed.* Addison-Wesley Publishing Company, Inc., 1989, pp 211-215.
- [42] Garret N. Vanderplaats, (1984). *Numerical Optimization Techniques for Engineering Design: With Applications*. McGraw-Hill Book Company, 1984.
- [43] Packard, A. and J. C. Doyle, “The complex structured singular value,” *Automatica*, Vol. 29, 1993, pp. 71-109.
- [44] Doyle, J. C., “Analysis of Feedback Systems with Structured Uncertainties,” *IEE Proceedings*, Part D, Vol. 133, 1982, pp. 45-56.
- [45] Doyle, J. C., J. Wall and G. Stein, “Performance and Robustness Analysis for Structured Uncertainty,” in *Proc. IEEE Conf. Dec. Contr.*, 1982, pp. 629-636.



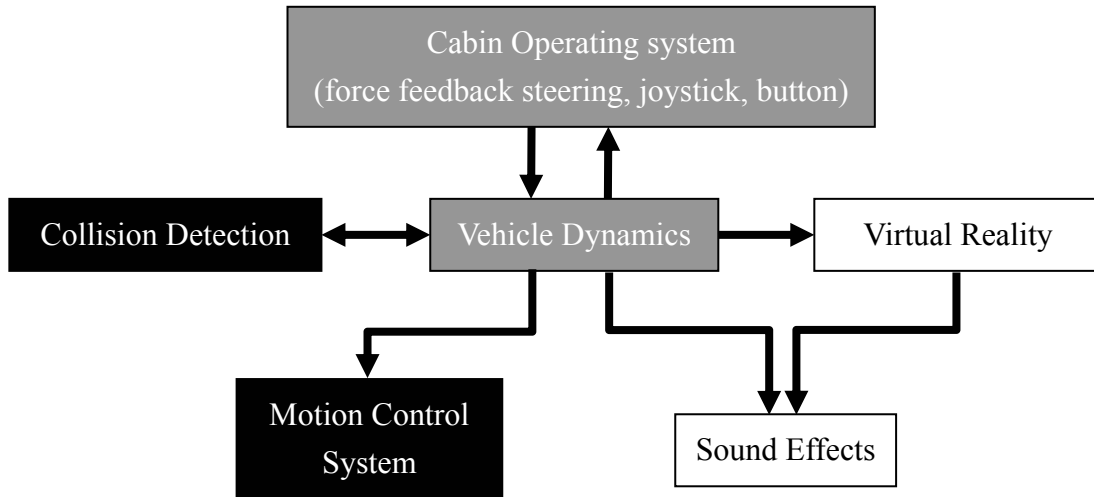


Fig. 1.1 Flow chart of the system integration for the human-machine interactive motion

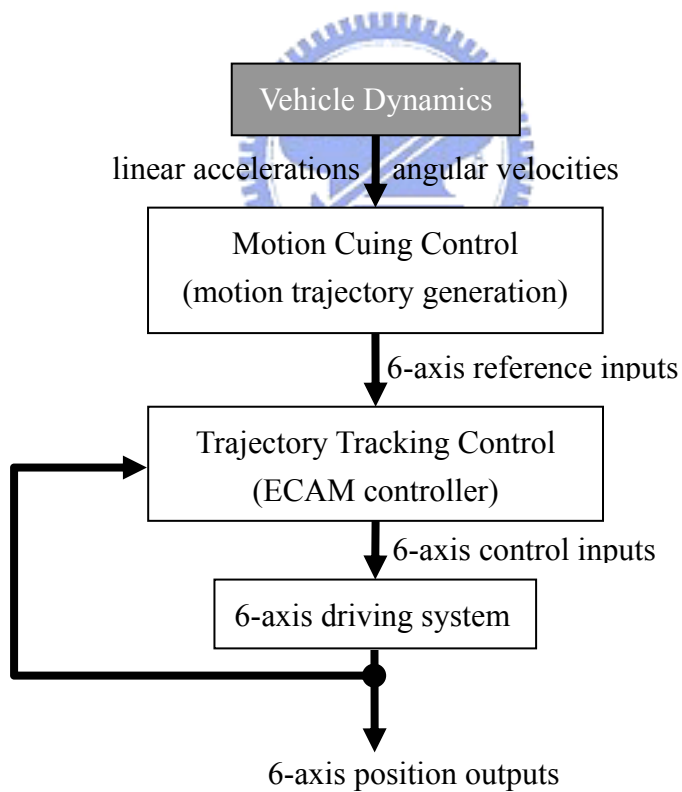


Fig. 1.2 The motion control system for a six DOF motion simulator

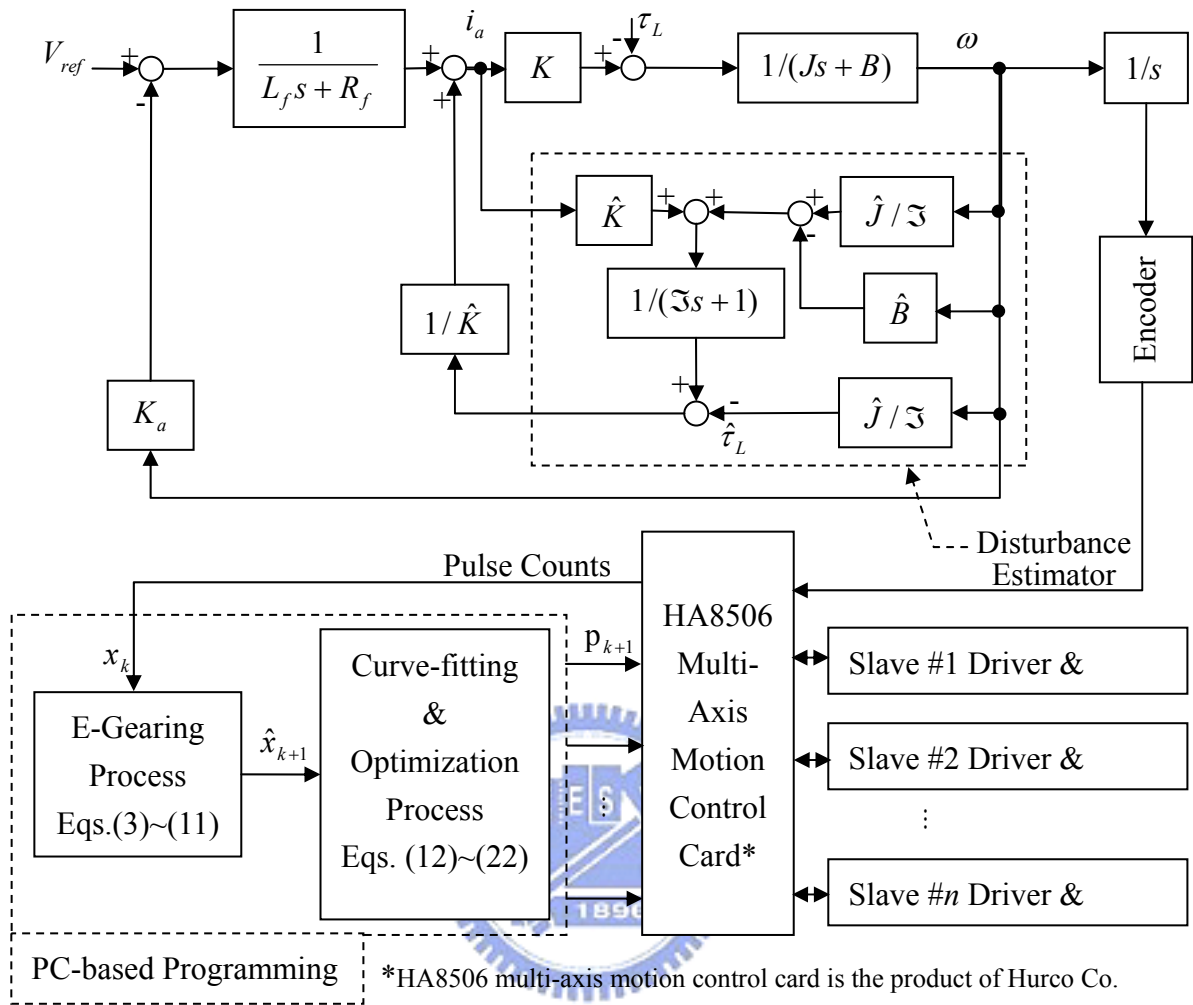


Fig. 2.1(a) Block diagram of the proposed ECAM system for mathematical representation

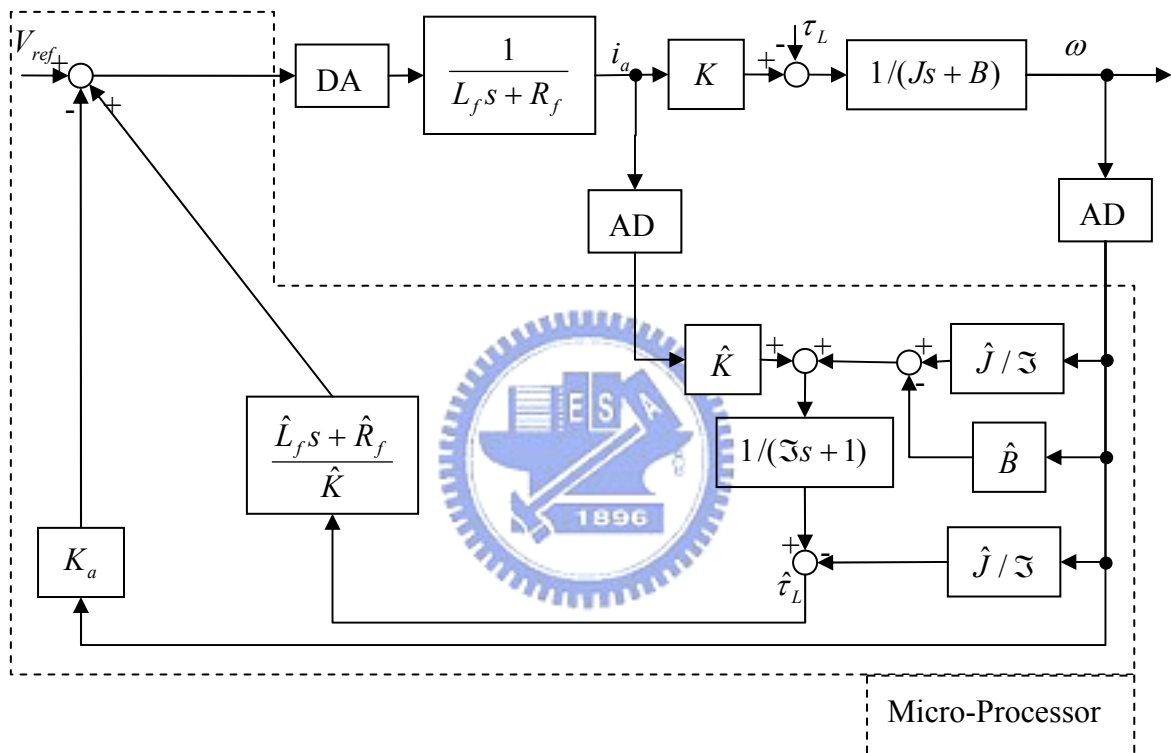


Fig. 2.1(b) Block diagram of the proposed disturbance estimator for one practical embodiment

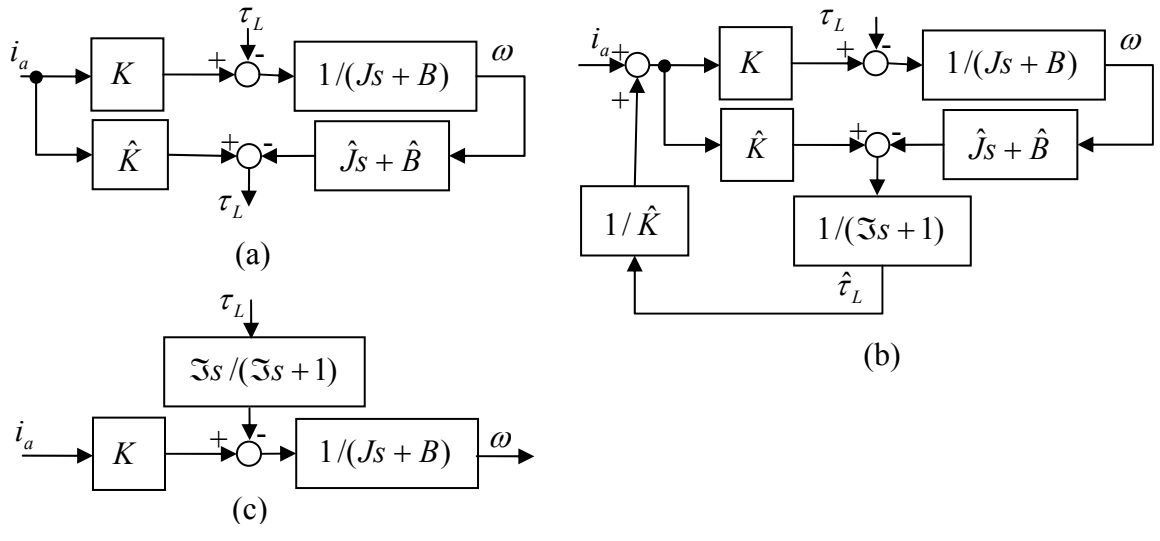


Fig. 2.2 The estimation of external disturbance

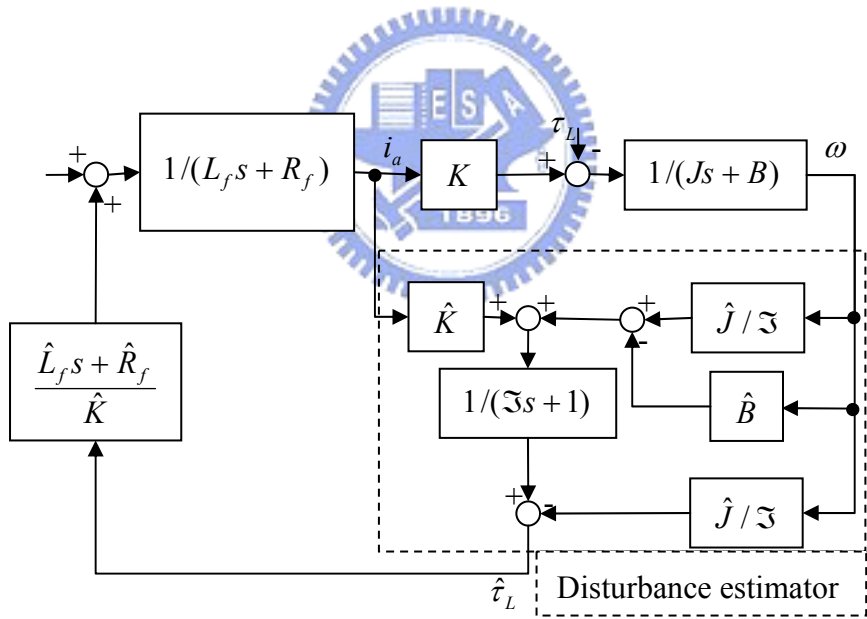


Fig. 2.3 The external disturbance estimator and external disturbance eliminated control

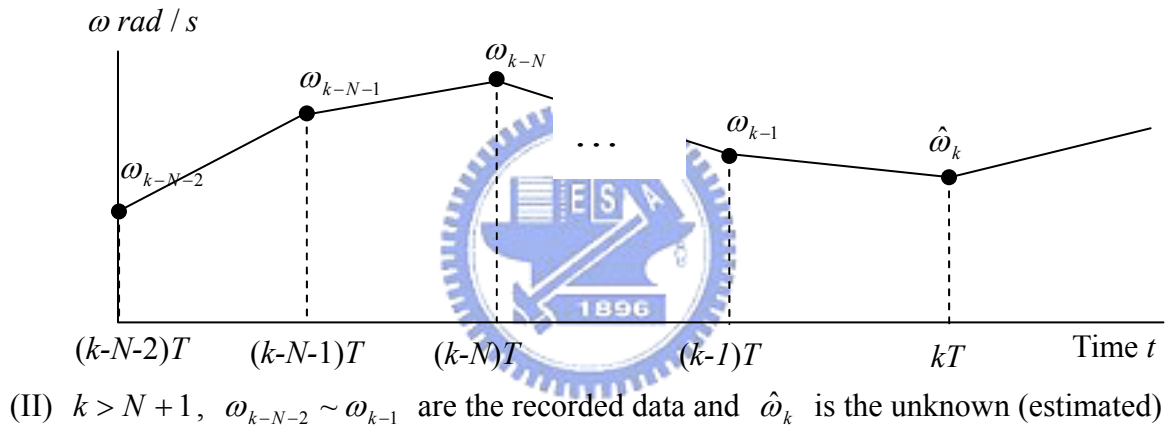
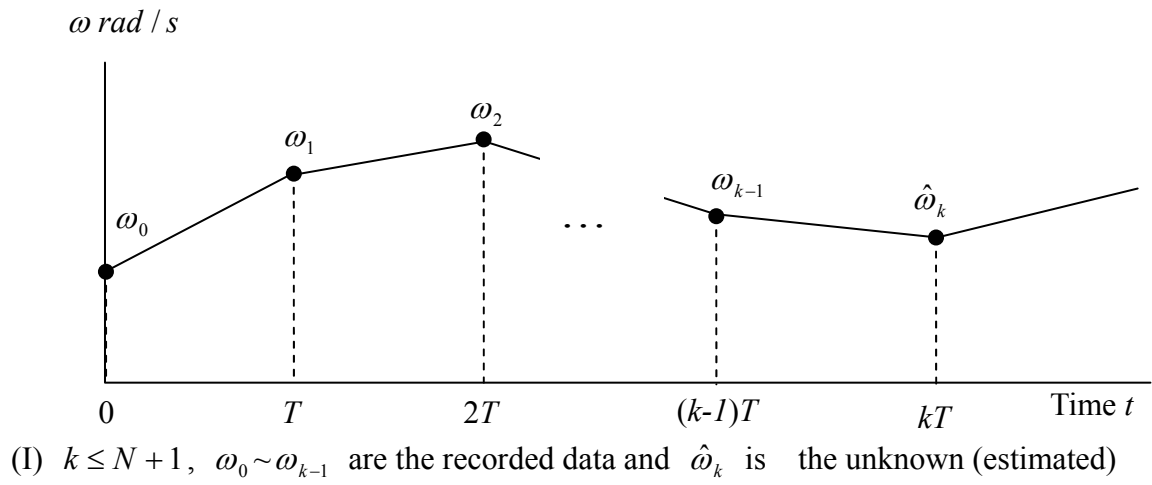
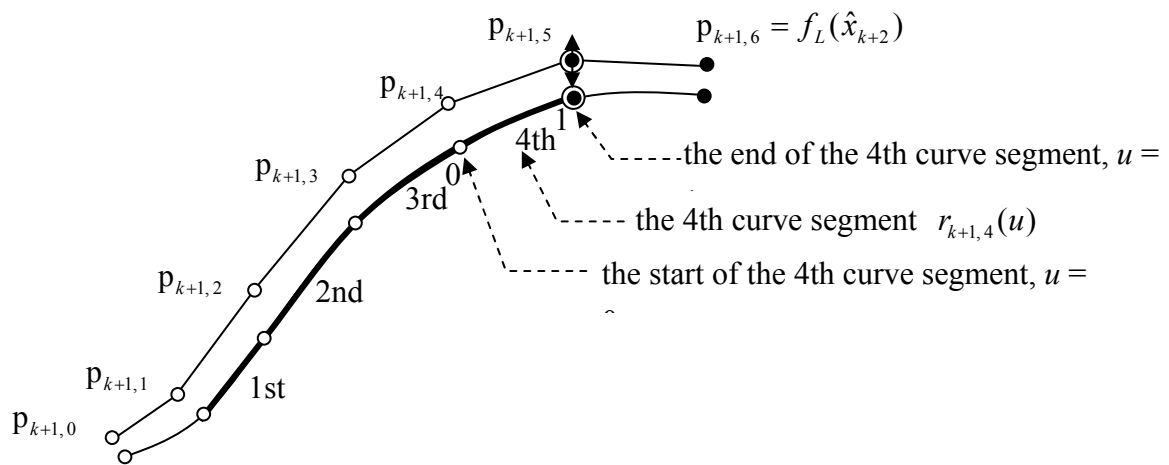
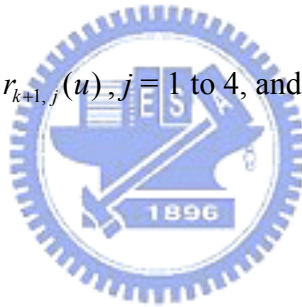


Fig. 2.4 Temporal relations between the two proposed procedures



Note: Modulate the spline curve by adjusting the control point $p_{k+1,5}$.

Fig. 2.5 Cubic B-spline curve $r_{k+1,j}(u)$, $j = 1$ to 4 , and its control points, $p_{k+1,0} \sim p_{k+1,6}$



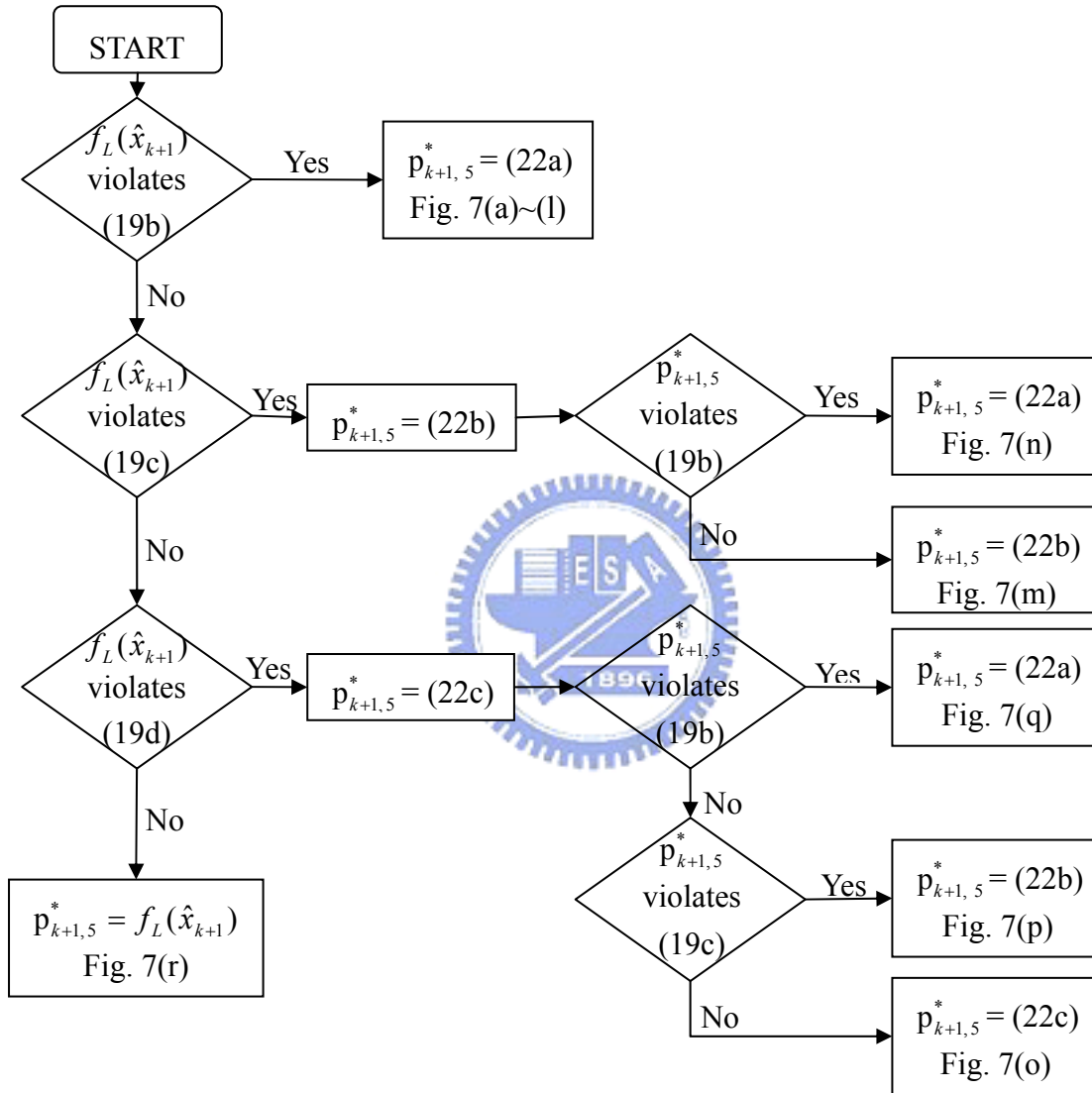
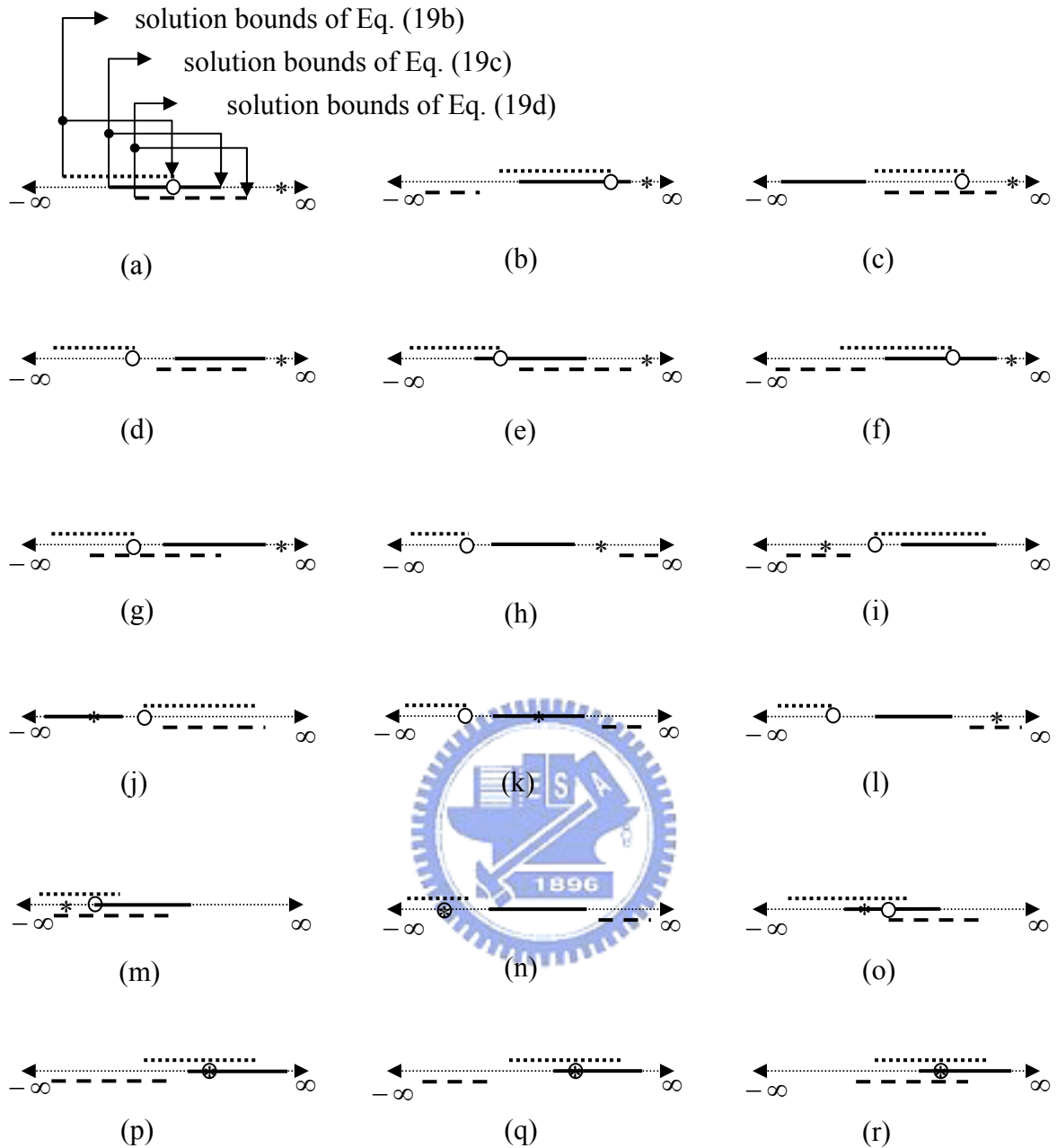


Fig. 2.6 Flow chart of the optimal solution process



- **A**: solution set of the inequality in Eq. (19b)—velocity constraint
- **B**: solution set of the inequality in Eq. (19c)—acceleration constraint
- - - **C**: solution set of the inequality in Eq. (19d)—jerk constraint
- * : The ideal cam profile position command, $f_L(\hat{x}_{k+1})$
- O: The optimal position command, $p_{k+1,5}^*$
- ⊗ : Optimal position, $p_{k+1,5}^*$, coincided with the ideal position command, $f_L(\hat{x}_{k+1})$

Note: The above categories are in the extreme case that $W_v \gg W_a \gg W_j$

Fig. 2.7 The location of the optimal position command, $p_{k+1,5}^*$, for all different cases

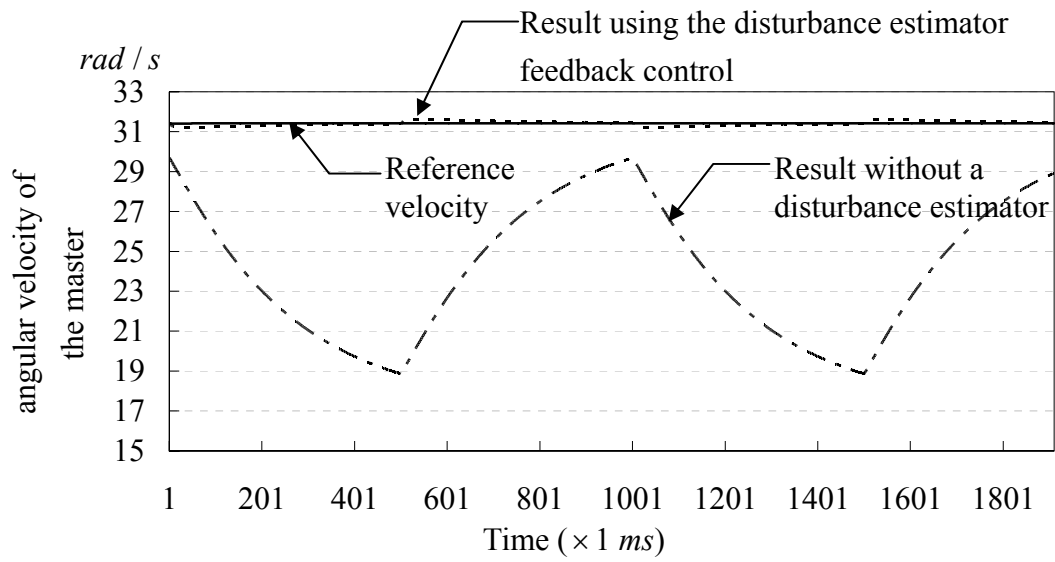


Fig. 2.8(a) Simulated angular velocity of the master

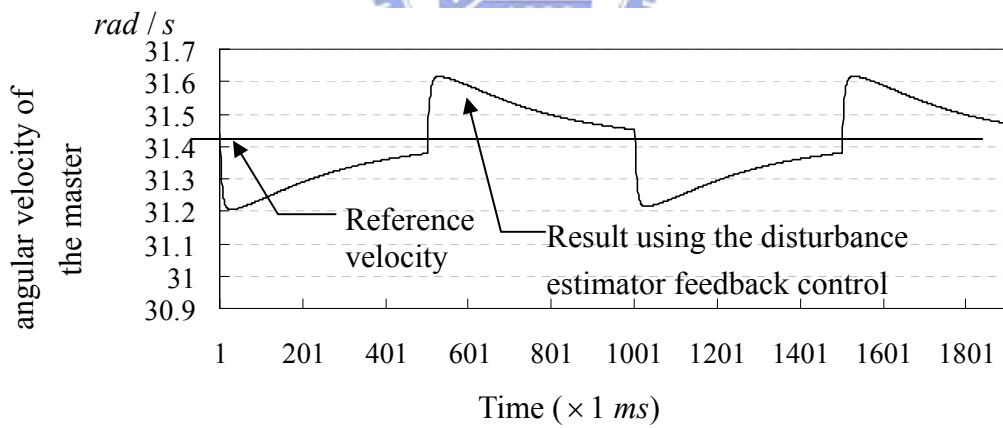


Fig. 2.8(b) Simulated angular velocity of the master using disturbance estimator feedback control (zoom in)

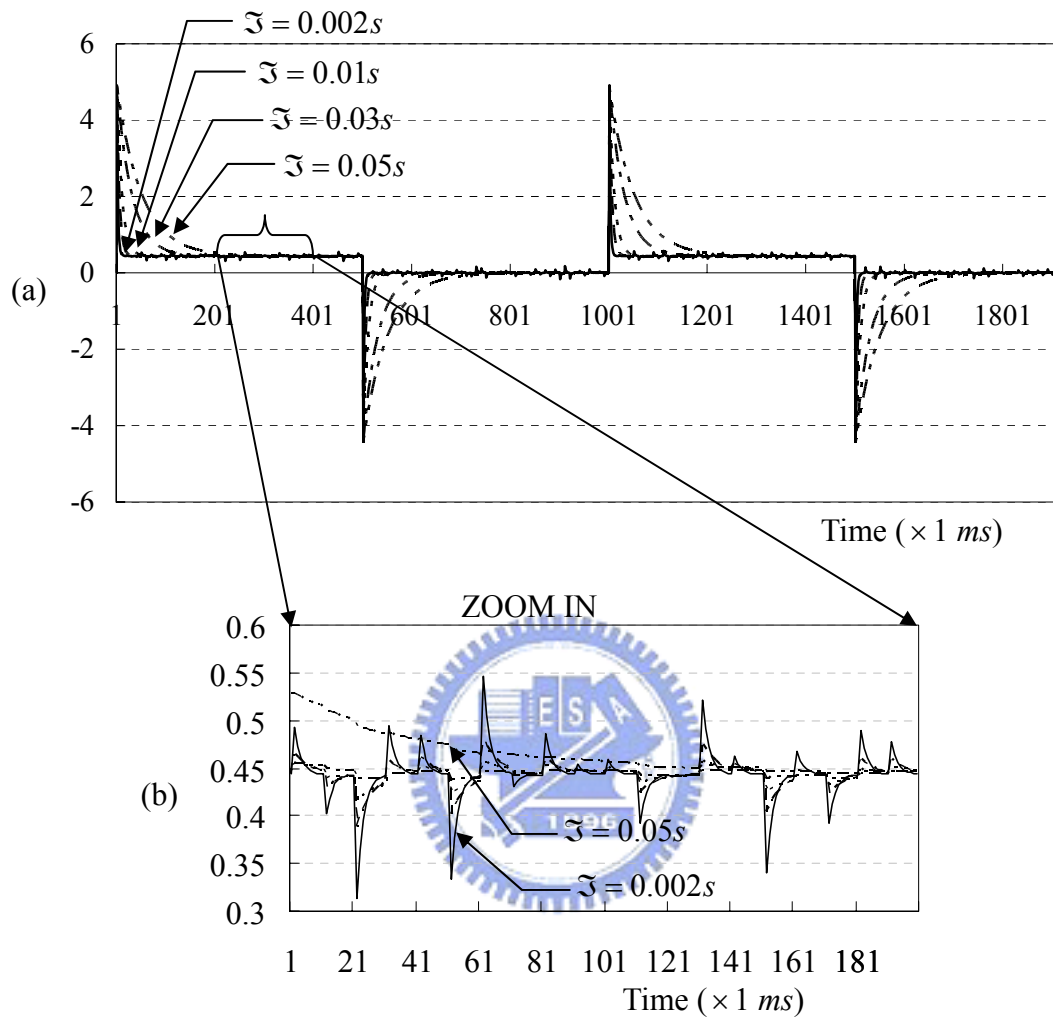


Fig. 2.9 The errors between the fed torque (τ_L) and the estimated torque ($\hat{\tau}_L$) with respect to various time constants (τ)

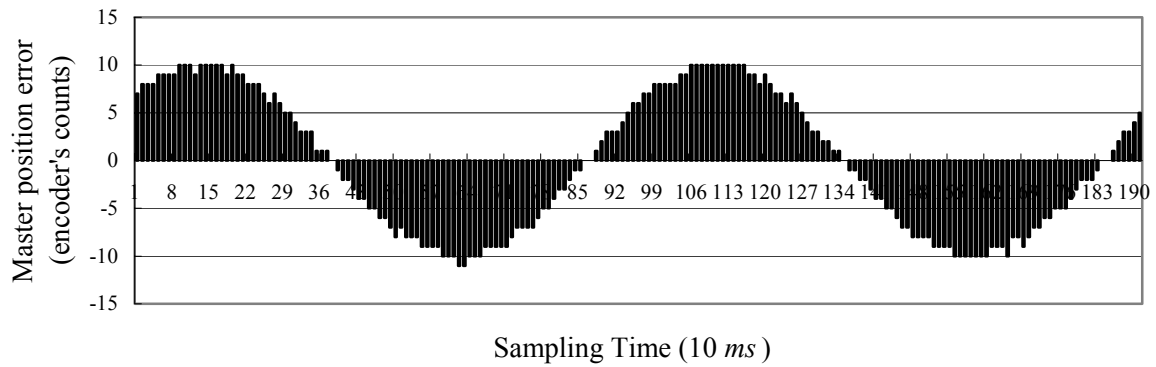


Fig. 2.10(a) The tracking error of the master's position for the zero-order interpolation method

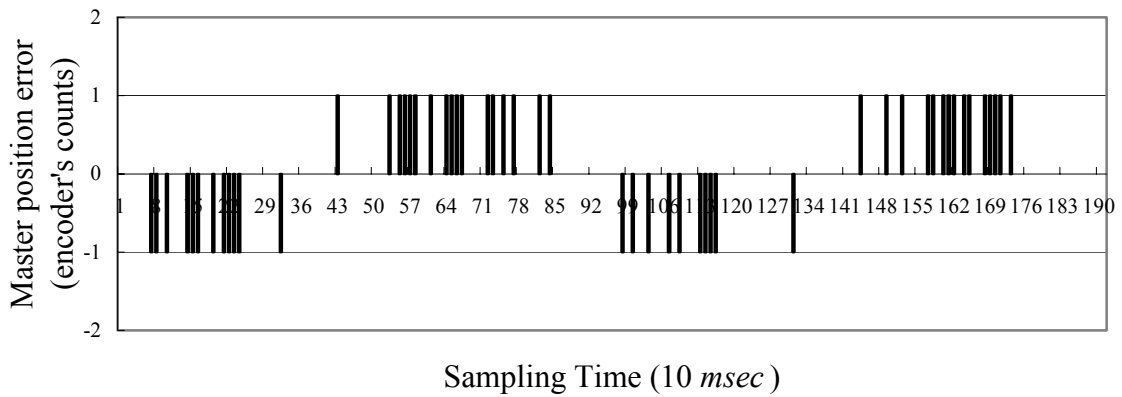


Fig. 2.10(b) The tracking error of the master for the third-order polynomial tracking method.

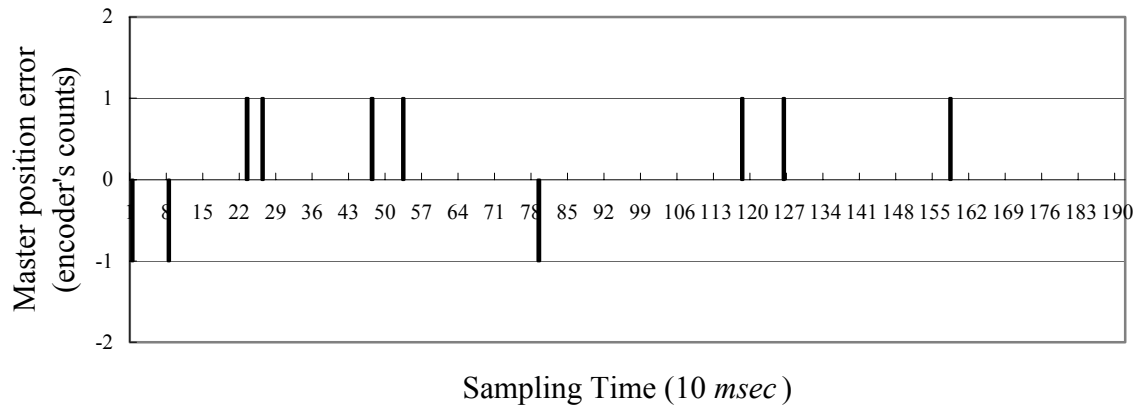


Fig. 2.10 (c) The tracking error of the master for the fourth-order polynomial tracking method

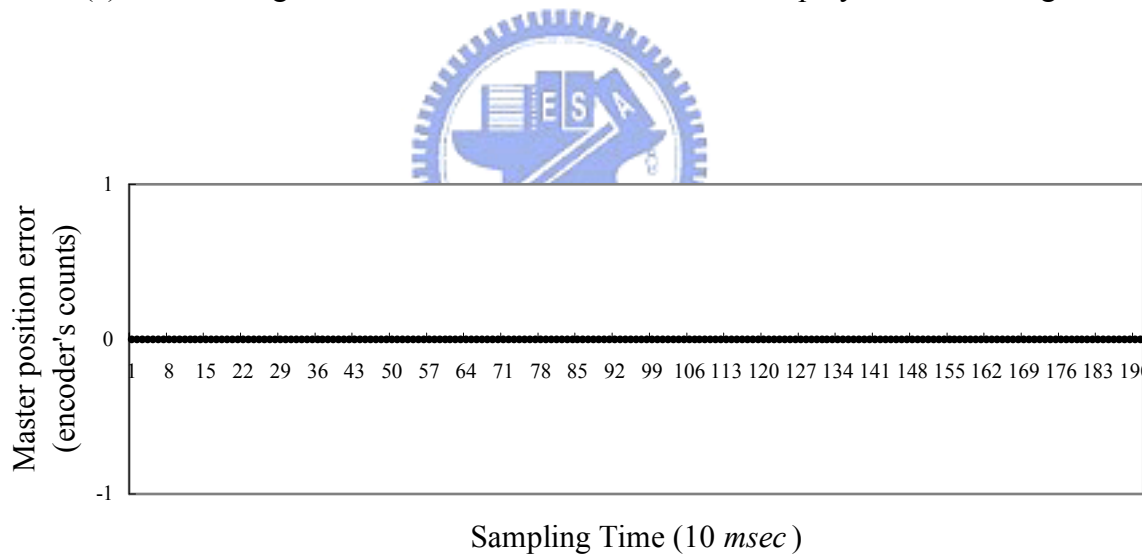


Fig. 2.10 (d) The tracking error of the master for the fifth-order polynomial tracking method

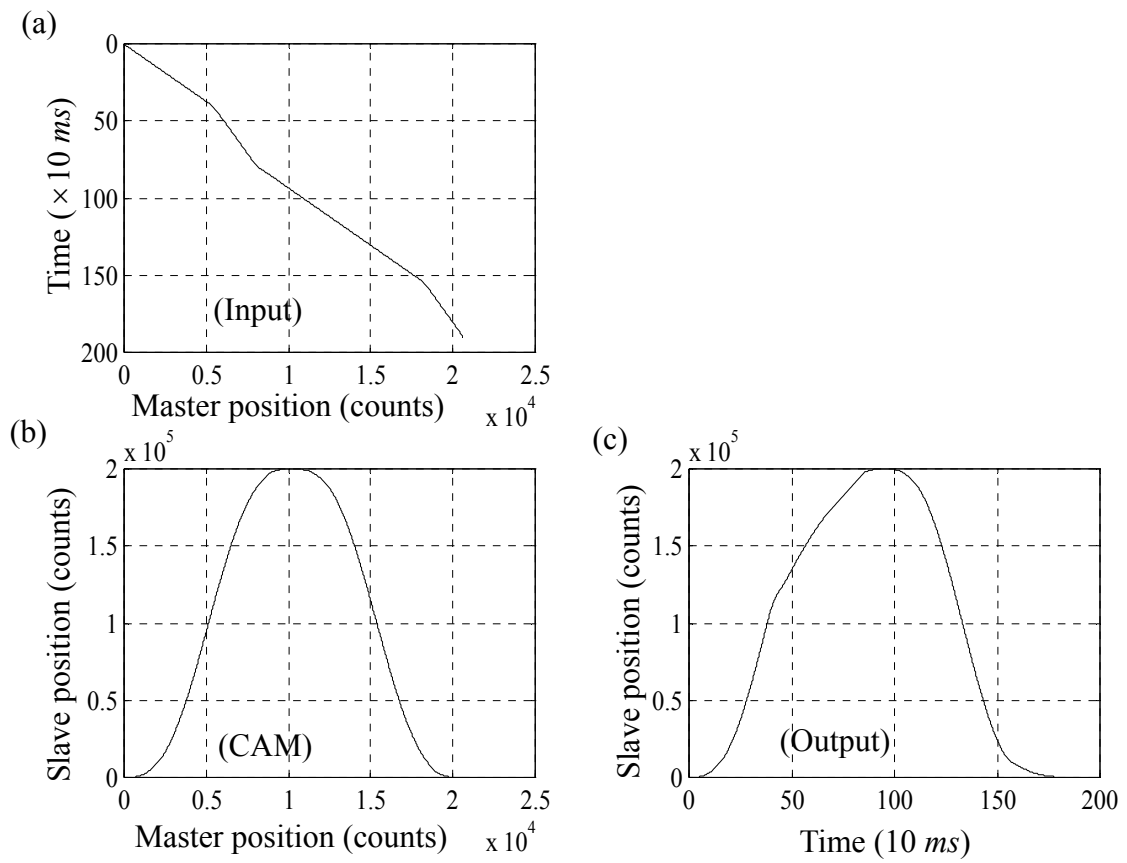


Fig. 2.11 The piecewise tracking trajectory of the electronic cam motion

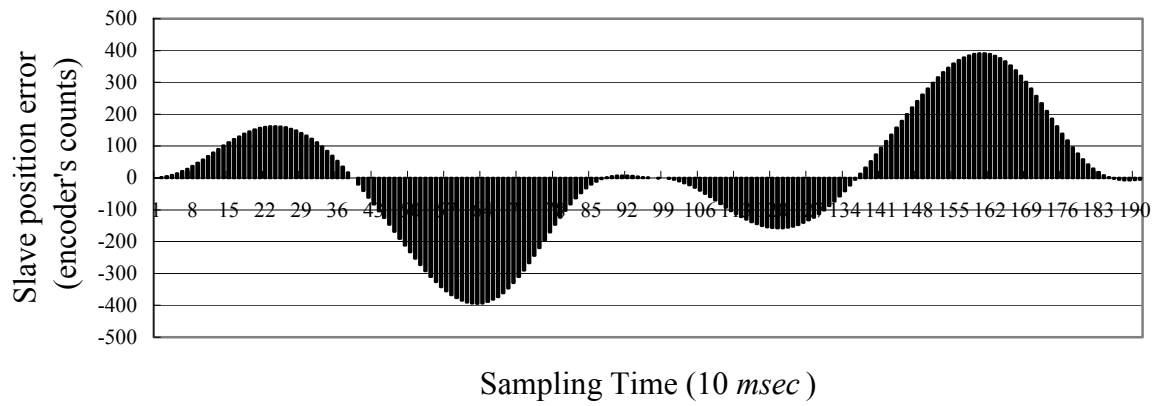


Fig. 2.12 (a) Cam profile error with the zero-order(conventional) tracking method
 (The maximum travel distance: 200000 encoder's counts)

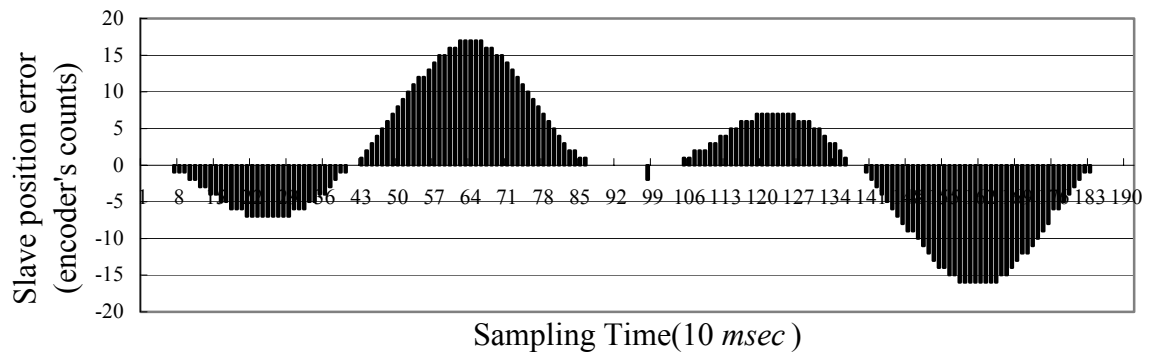


Fig. 2.12(b) Cam profile error with the third-order polynomial tracking method
 (The maximum travel distance: 200000 encoder's counts)

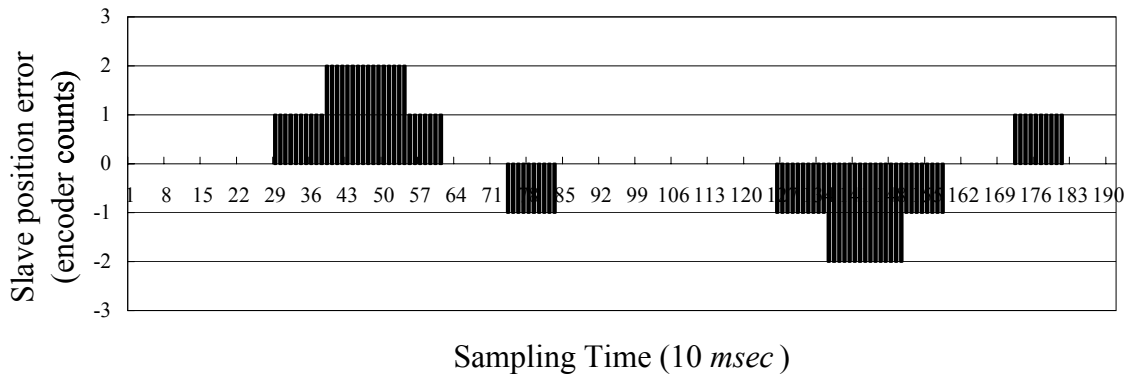


Fig. 2.12(c) Cam profile error with the fourth-order polynomial tracking method
(The maximum travel distance: 200000 encoder's counts)

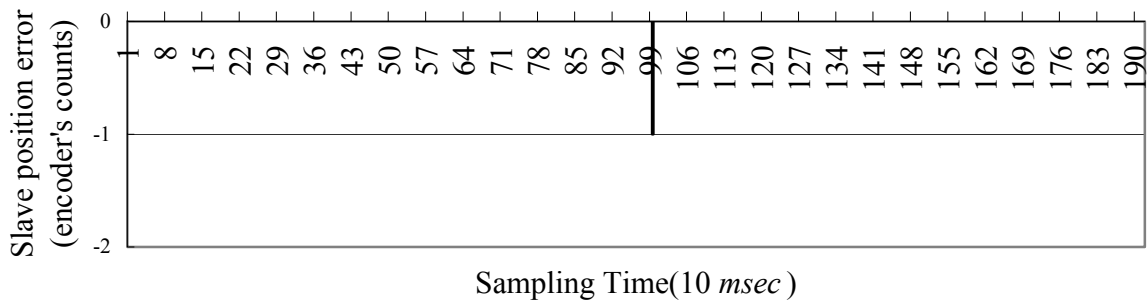


Fig. 2.12(d) Cam profile error with the fifth-order polynomial tracking method
(The maximum travel distance: 200000 encoder's counts)

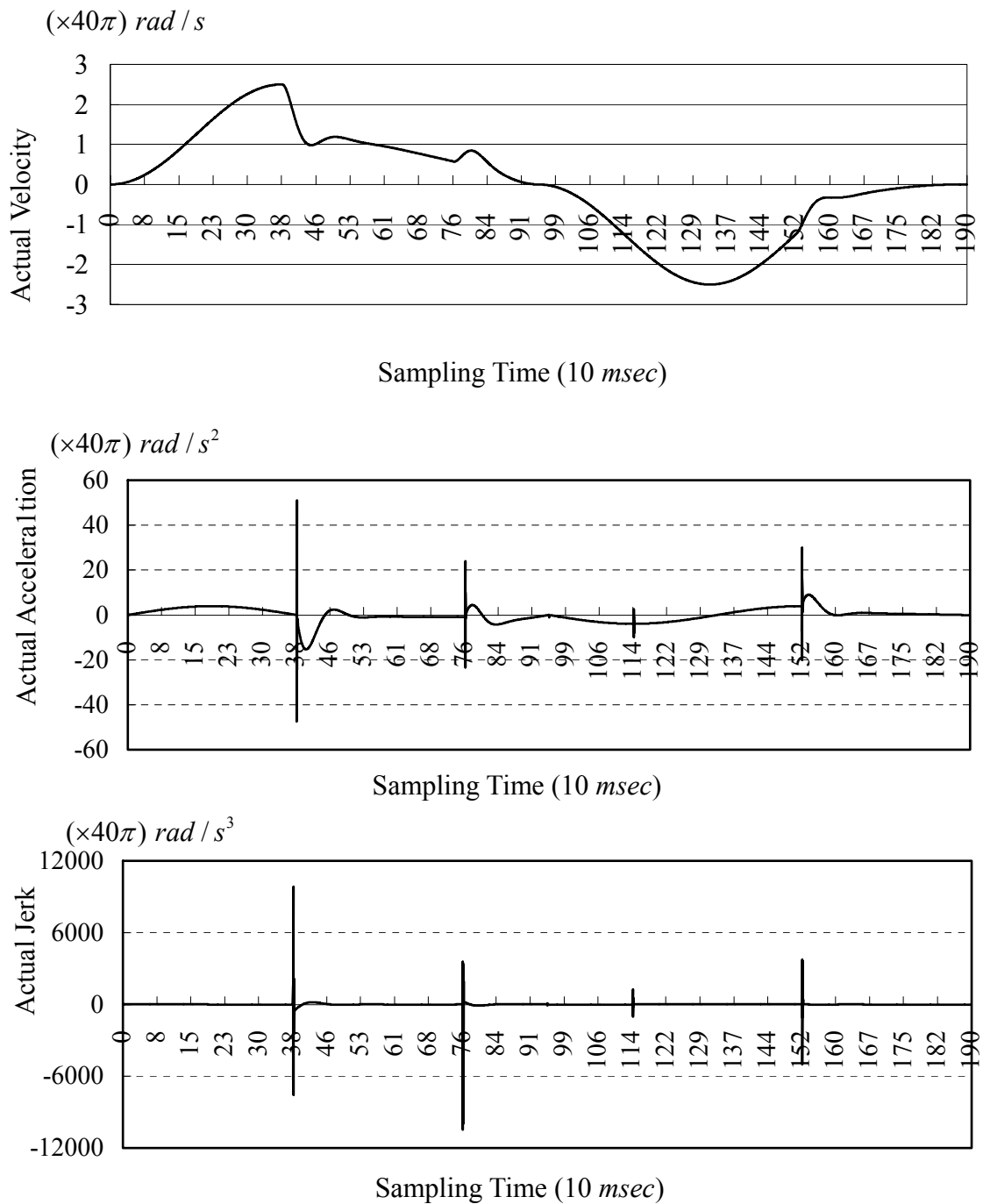


Fig. 2.13(a) The tracking result of the slave velocity, acceleration and jerk purely based on the Lagrange polynomial curve-fitting with no optimization

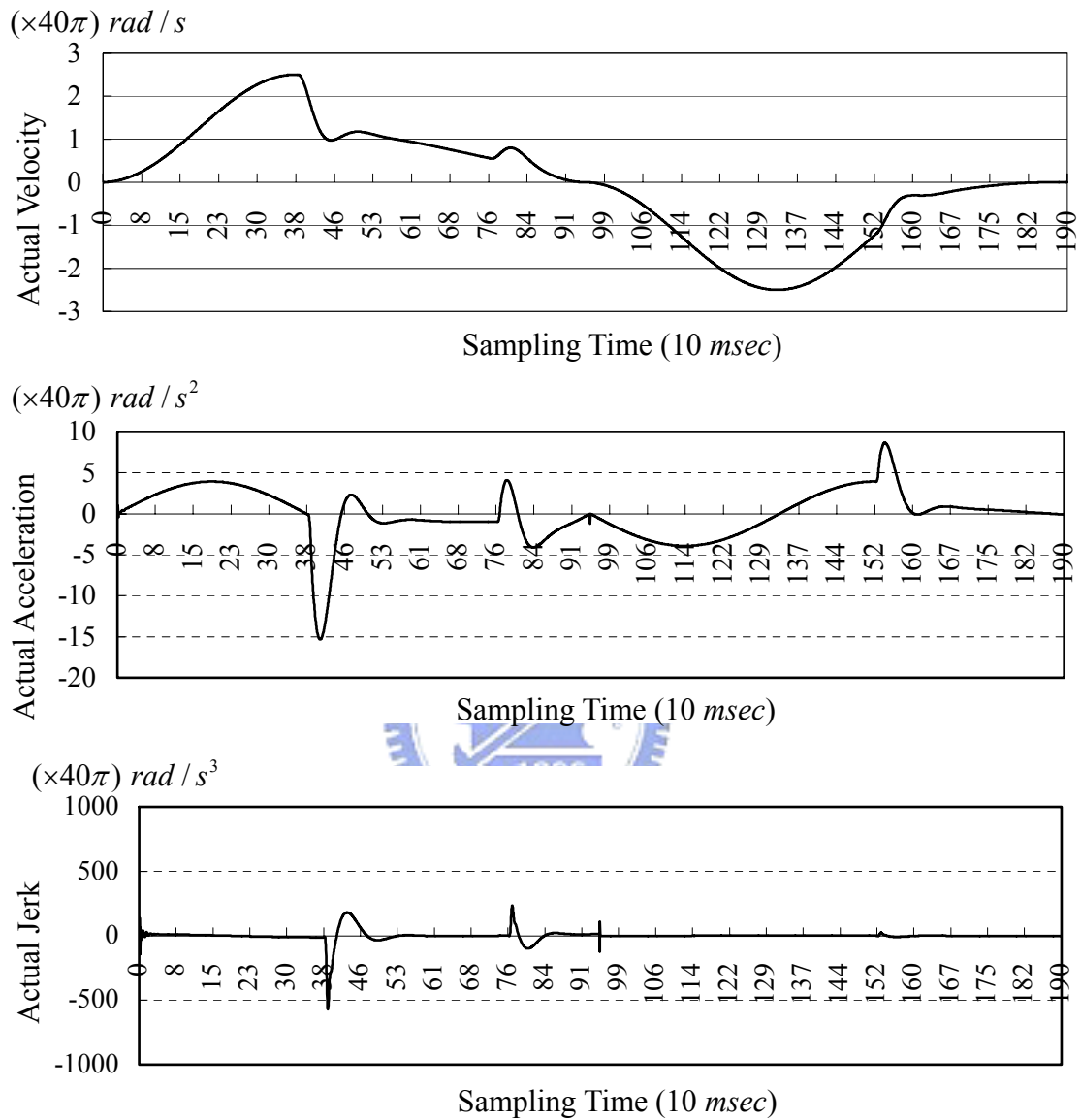


Fig. 2.13(b) The result of the slave velocity, acceleration and jerk applying the optimization

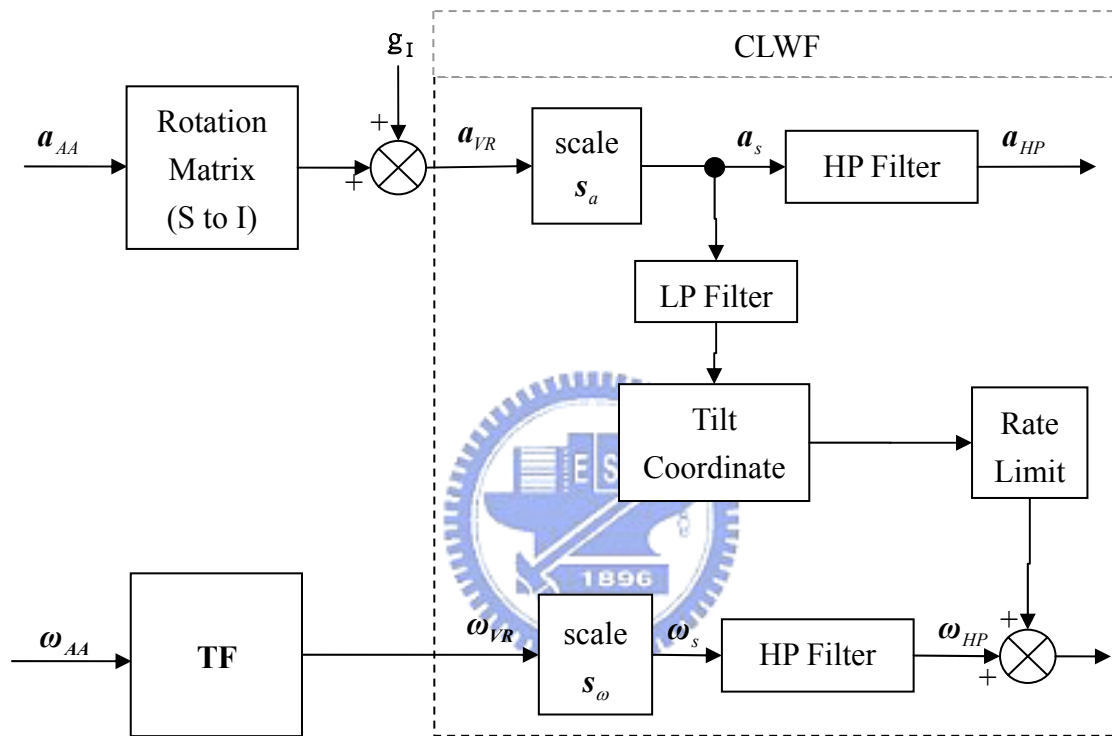


Fig. 3.1 Classical linear washout filter (referred to Nahon and Reid, 1990)

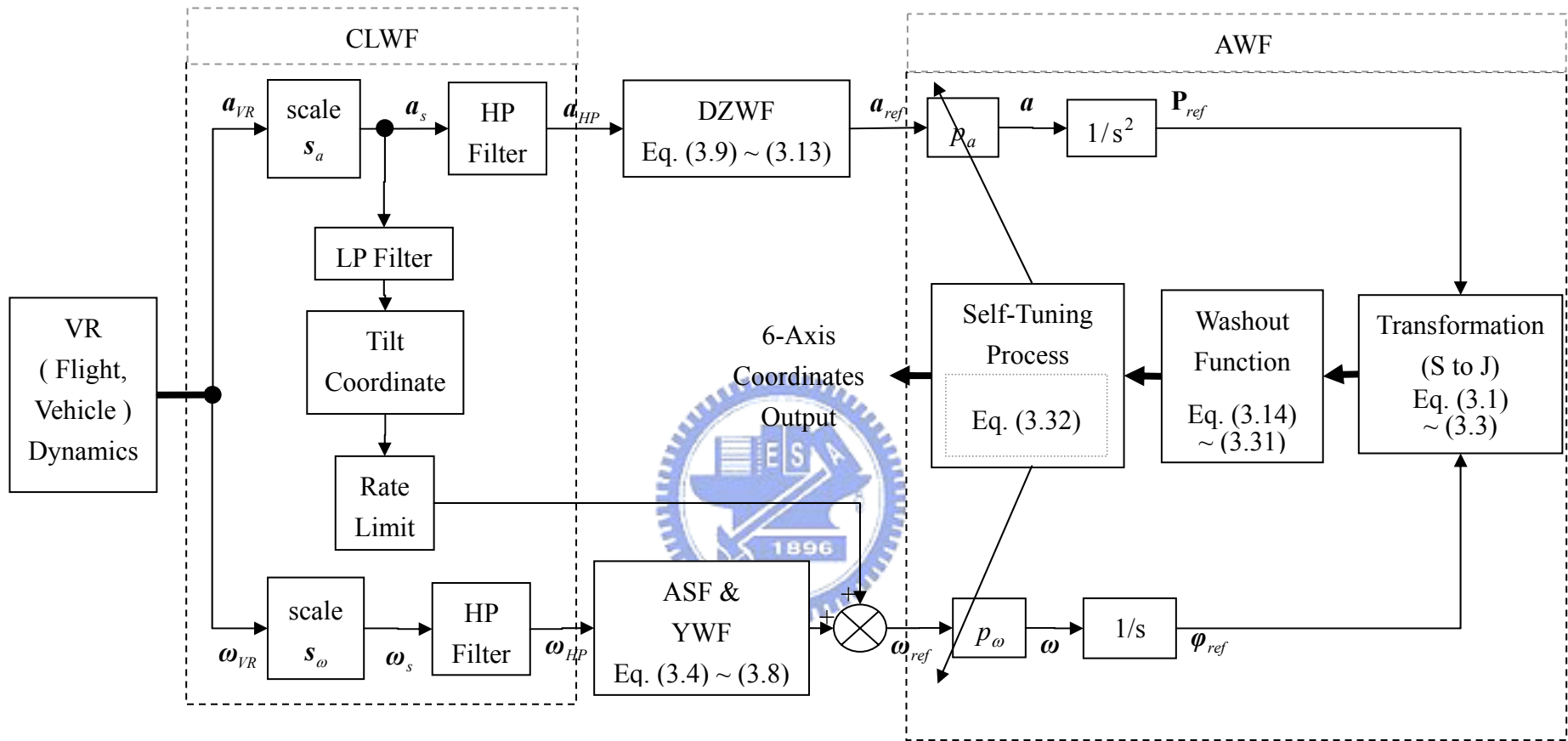


Fig. 3.2 Block diagram of motion-cueing system, using the proposed control strategy

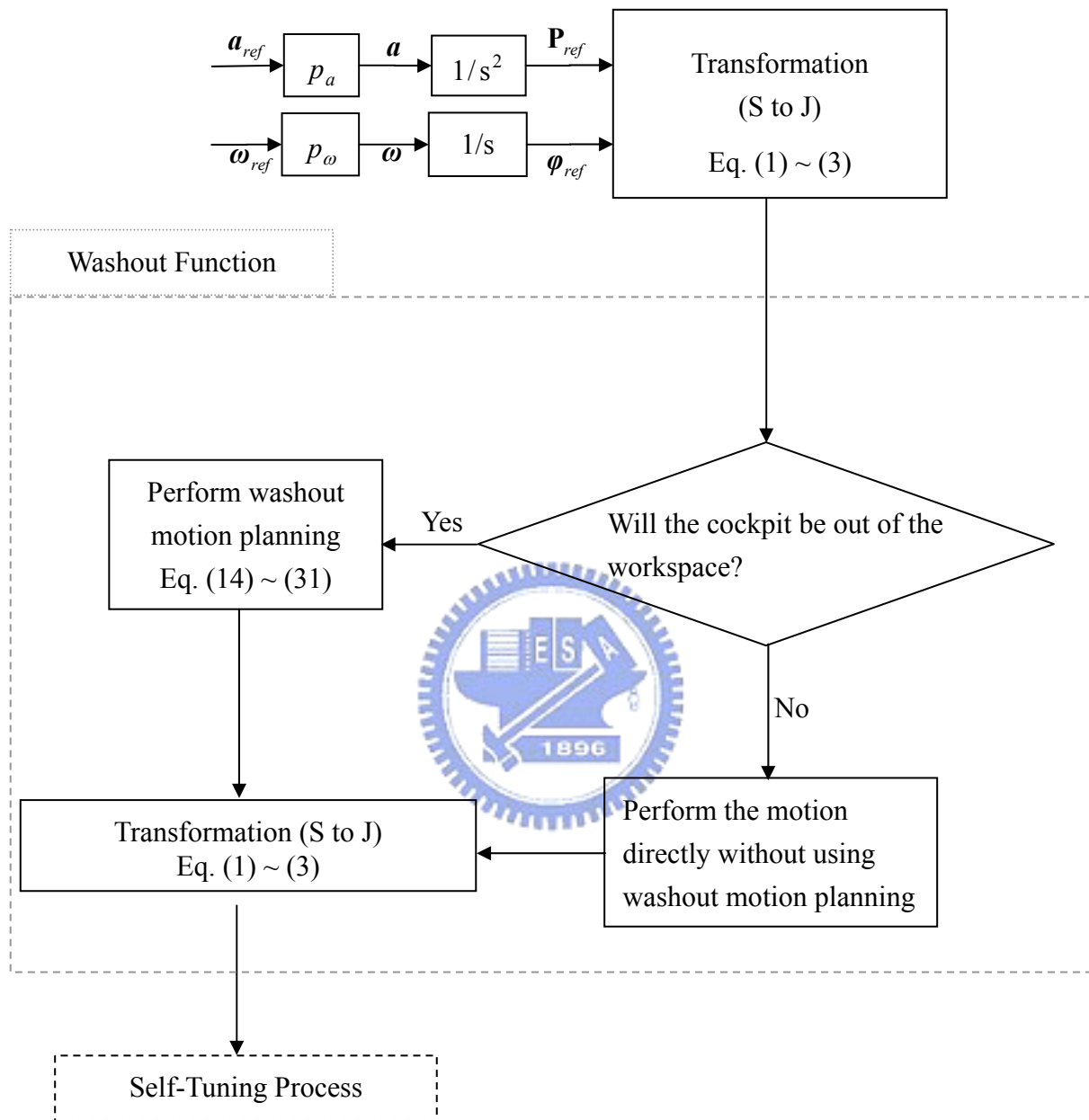


Fig. 3.3(a) AWF structure (to be continued)

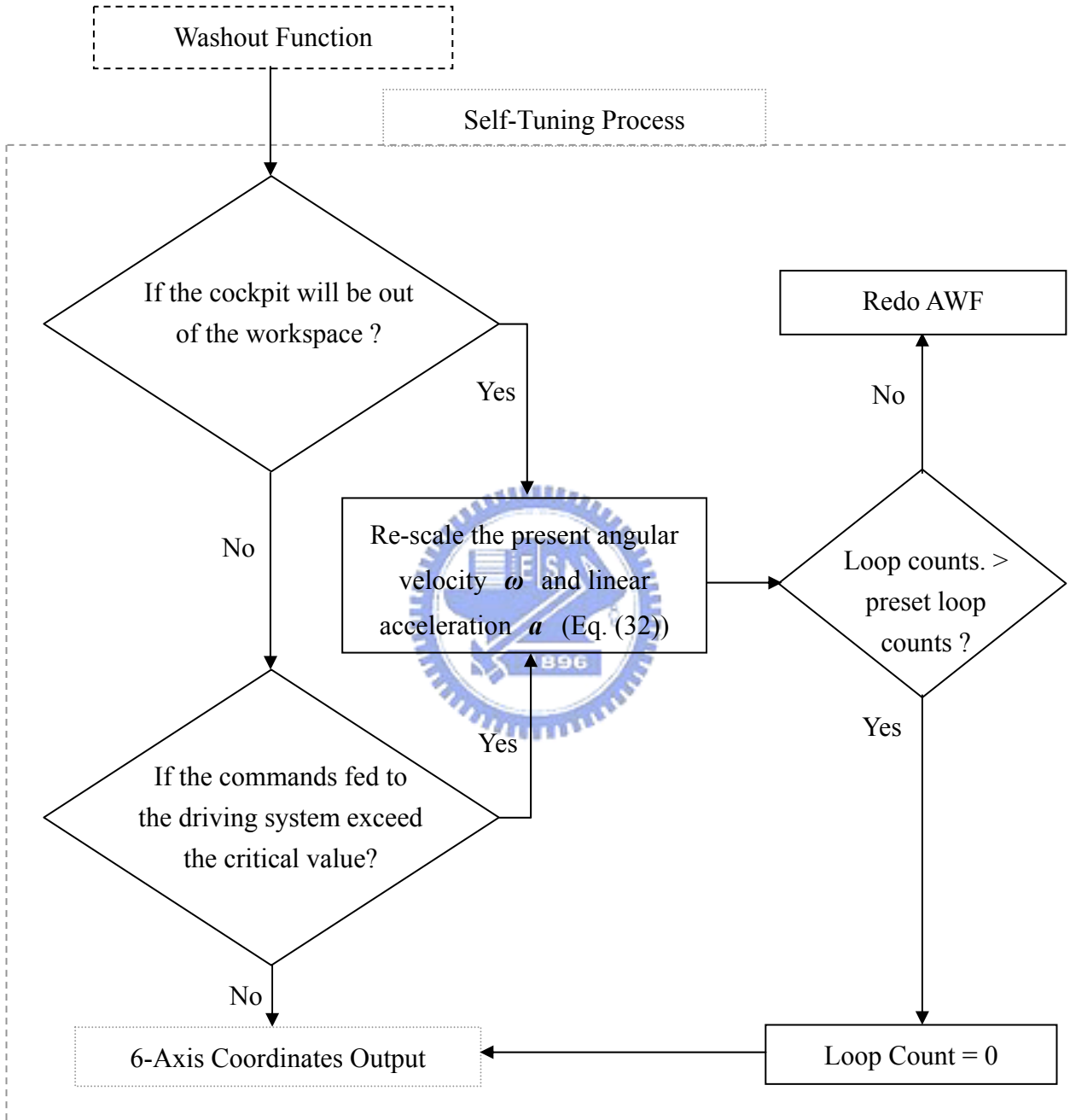


Fig. 3.3(b) AWF structure (to continue)

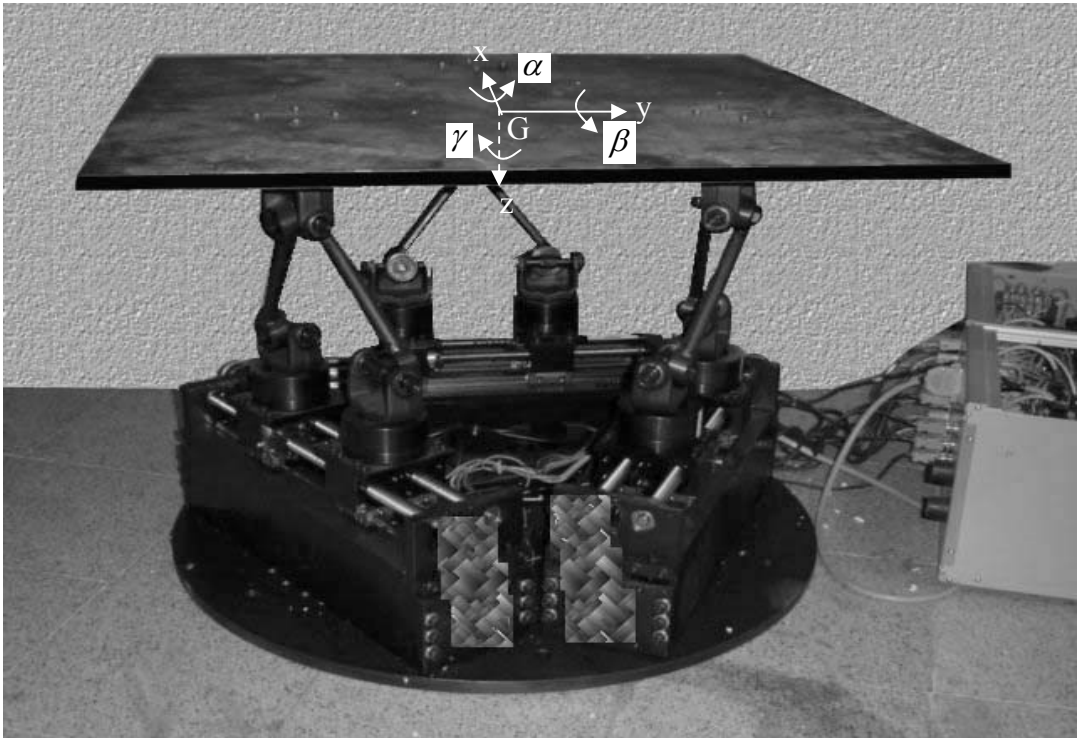


Fig. 3.4 Prototype SP-120

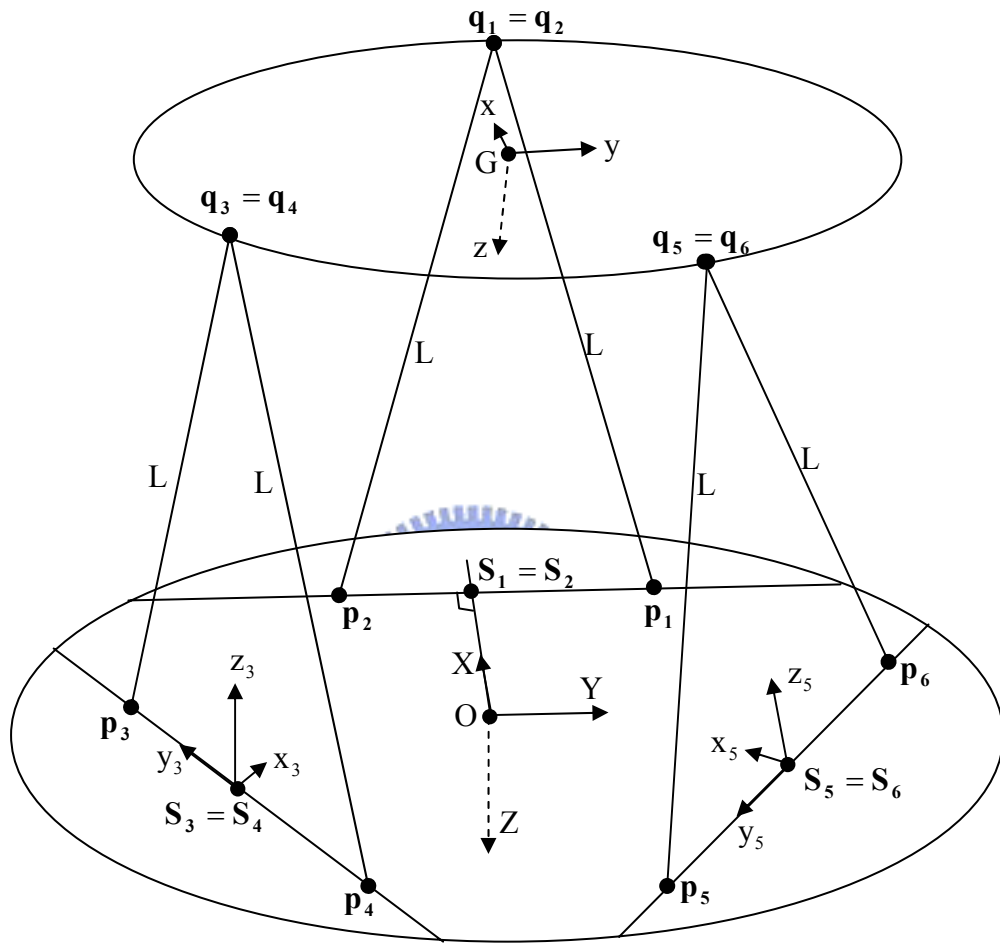


Fig. 3.5 Kinematical skeleton of simulator platform SP-120

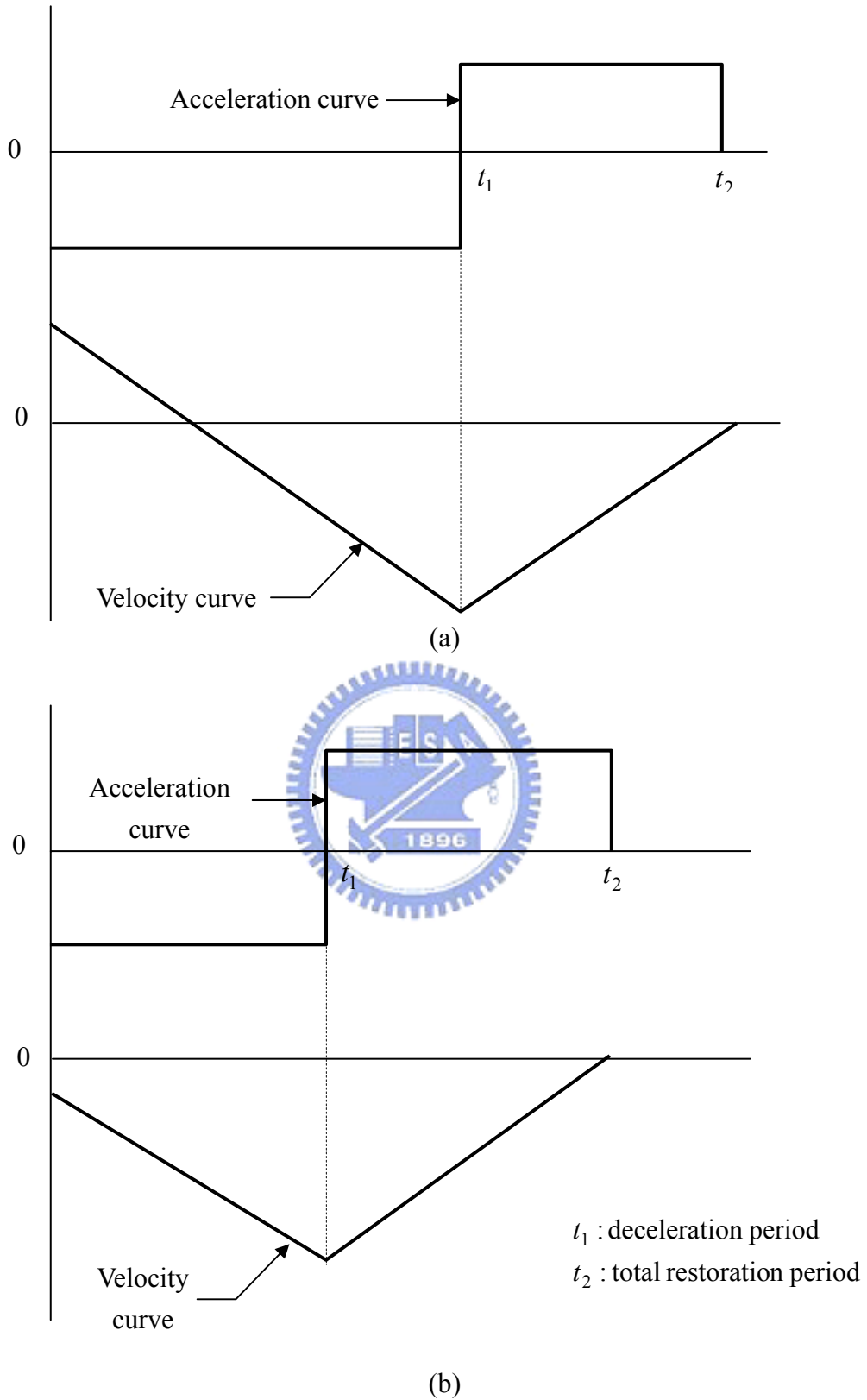


Fig. 3.6 Restoration process in the dead zone washout filter

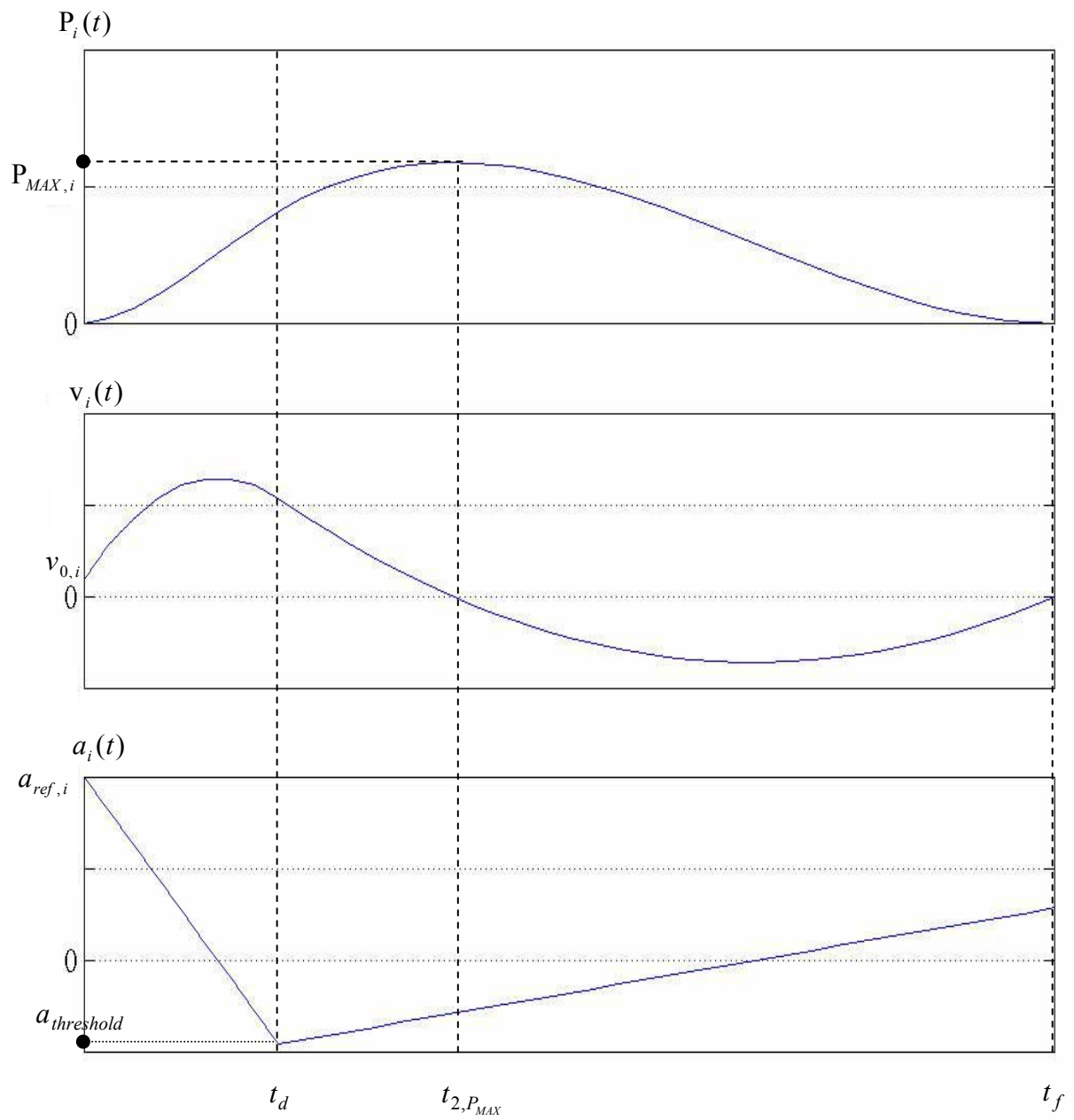


Fig. 3.7 Washout trajectory along the i -axis, where the subscript i represents the three mutually orthogonal axes (x-, y-, and z-axis)

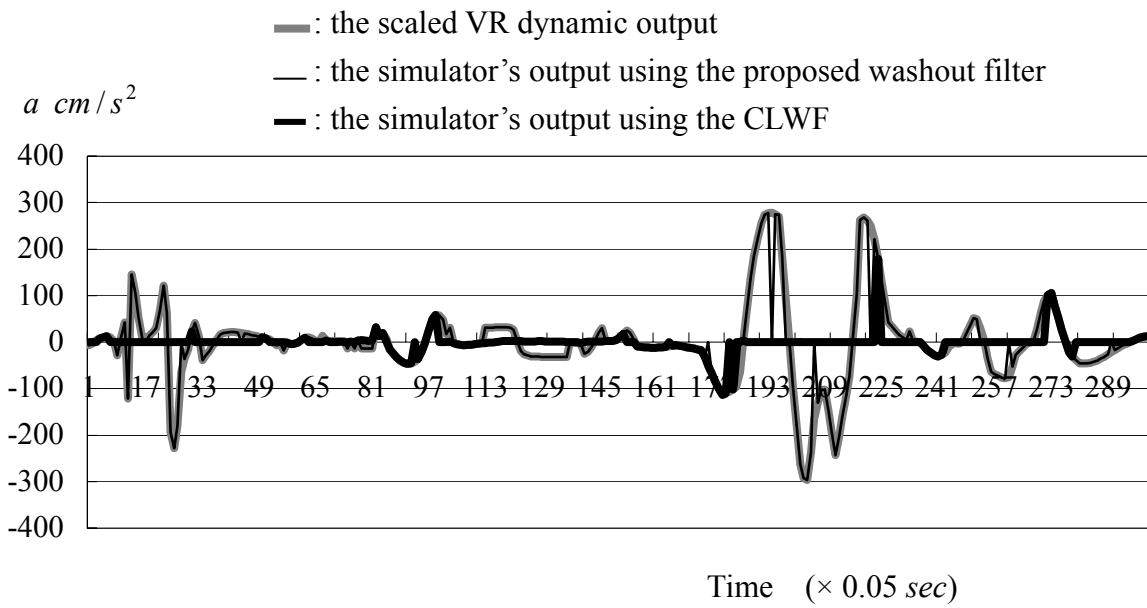


Fig. 3.8 Segmental data concerning linear accelerations along the x-axis

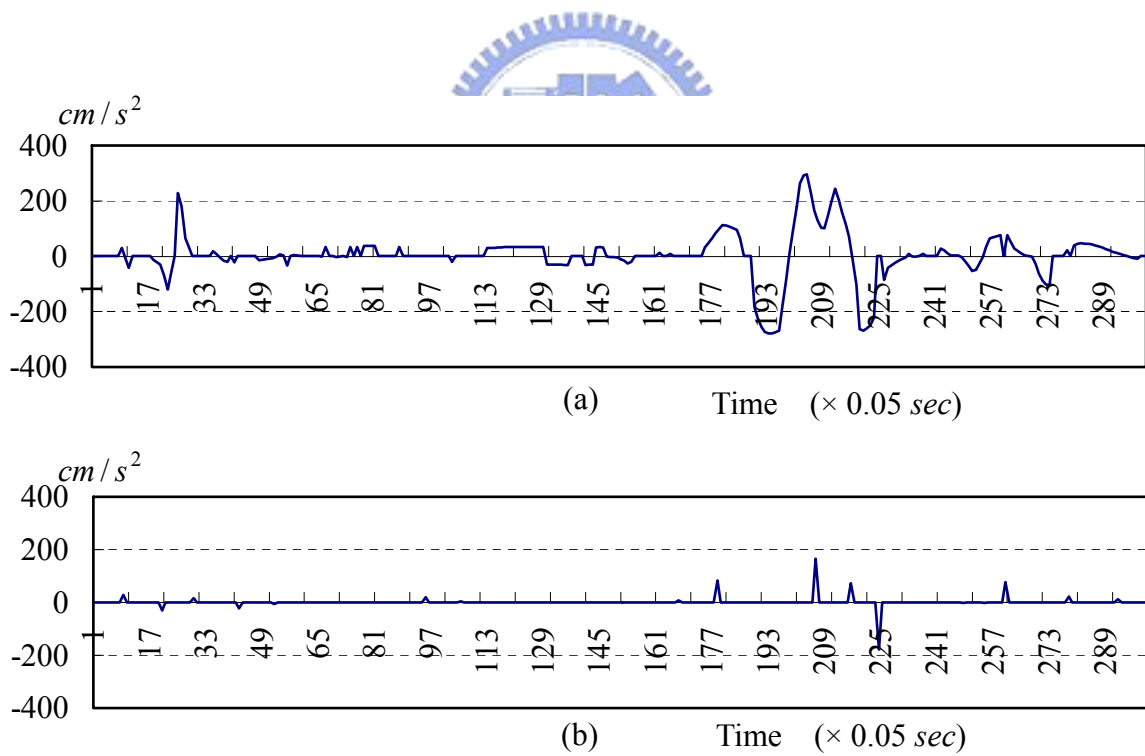


Fig. 3.9 Segmental errors of linear accelerations along the x-axis, between the static scaled VR dynamic output ($a_{s,x}$) and the simulator's output (a_x) using the (a) CLWF and (b) the proposed strategies

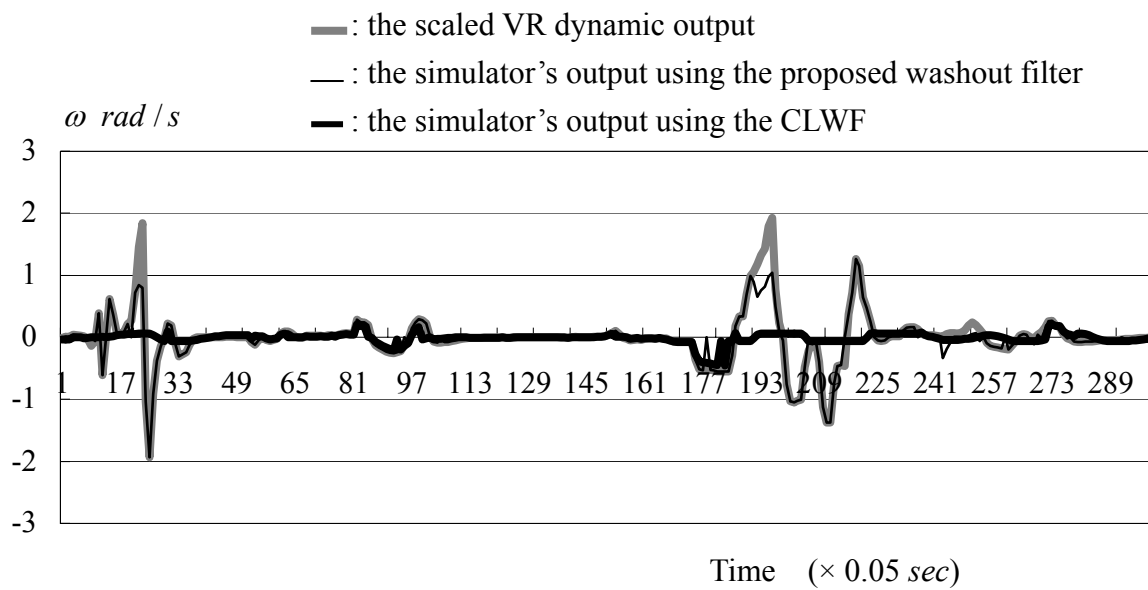


Fig. 3.10 Three segmental data concerning Euler's angular velocities

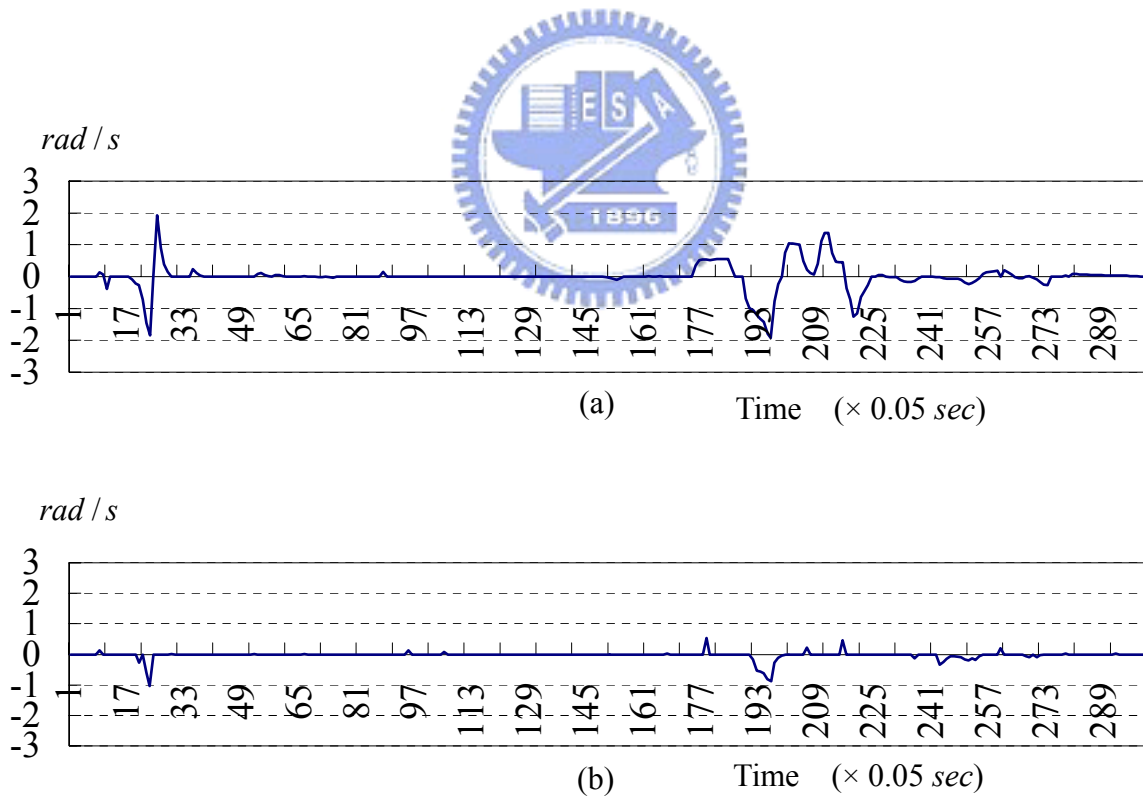


Fig. 3.11 Segmental errors of Euler's angular velocities between the static scaled VR dynamic output ($\omega_{s,x}$) and the simulator's output (ω_x) using (a) the CLWF and (b) the proposed strategies

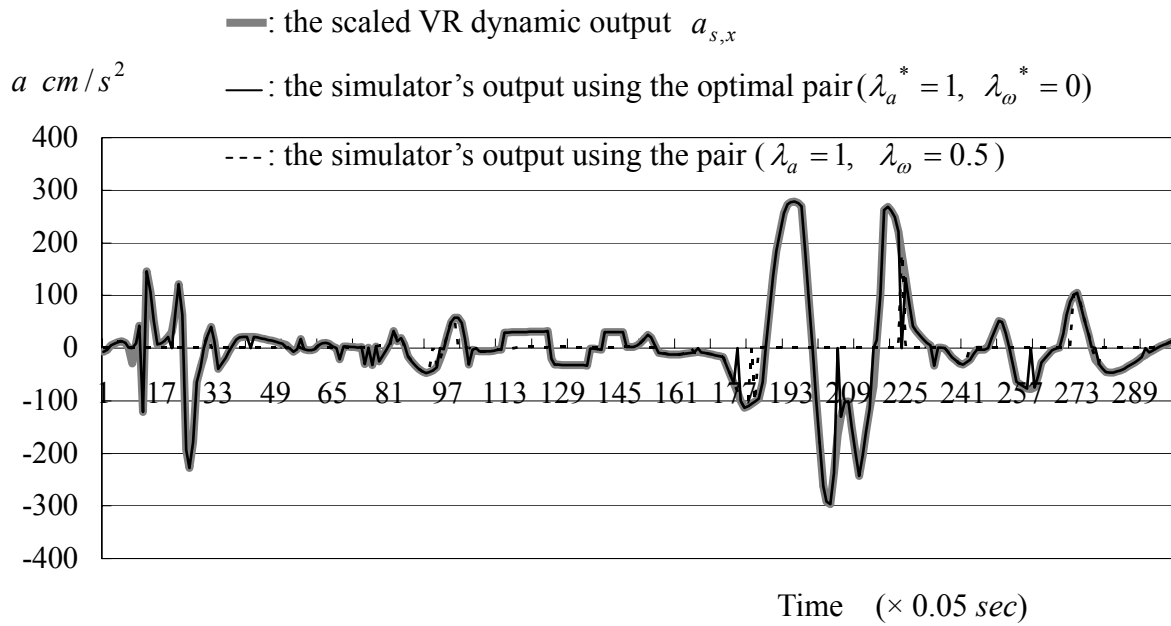


Fig. 3.12 Three segmental data concerning linear accelerations along the x-axis

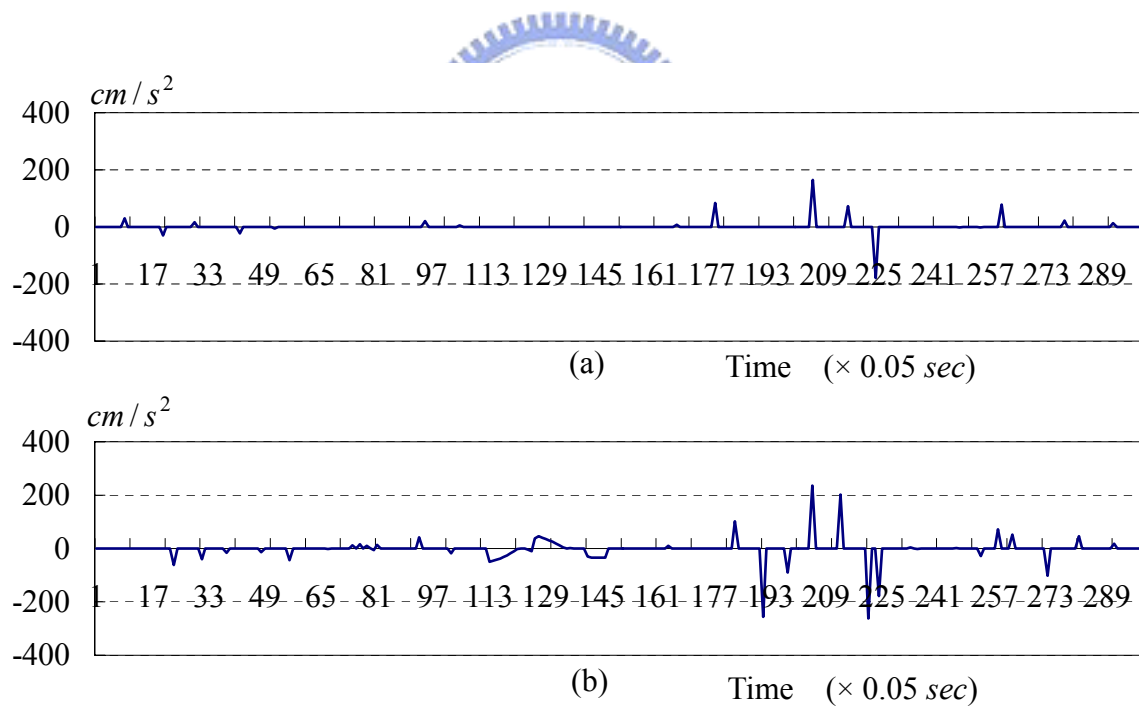


Fig. 3.13 Segmental errors of linear accelerations along the x-axis between the static scaled VR dynamic output ($a_{s,x}$) and the simulator's output (a_x) using (a) the optimal pair of weighting parameters (1, 0) and (b) the pair of weighting parameters (1, 0.5)

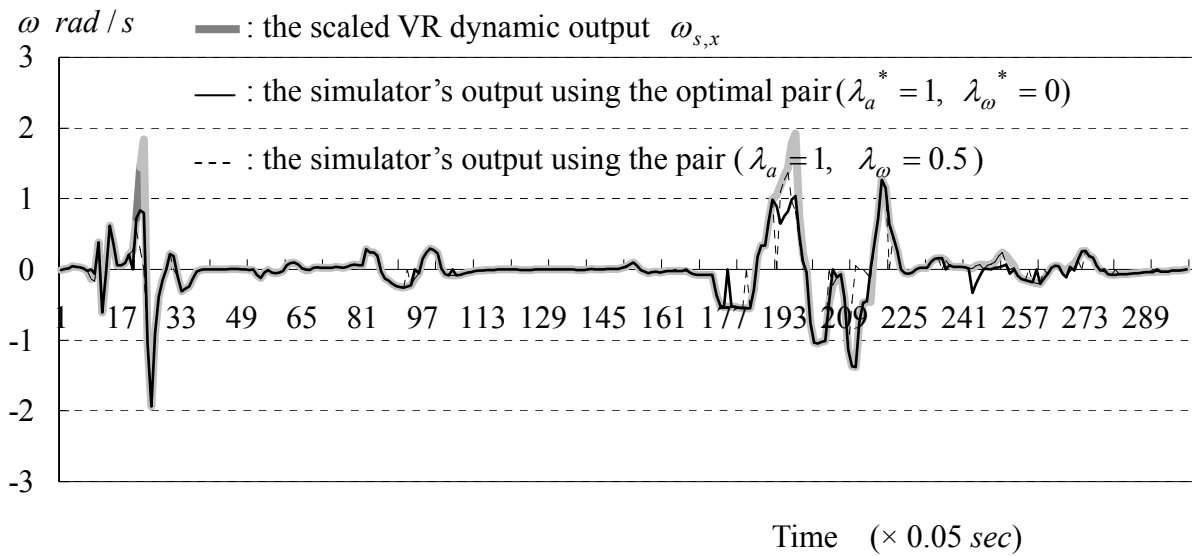


Fig. 3.14 Three segmental data concerning Euler's angular velocities

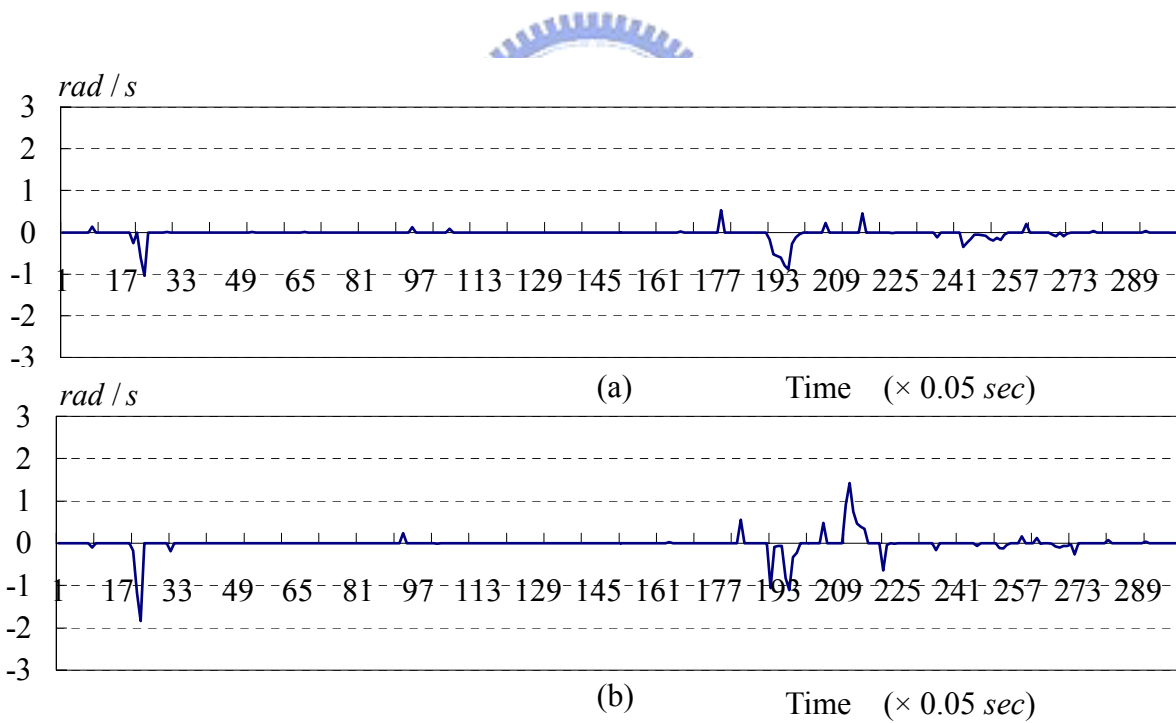


Fig. 3.15 Segmental errors of Euler's angular velocities between the static scaled VR dynamic output ($\omega_{s,x}$) and the simulator's output (ω_x) using (a) the optimal pair of weighting parameters (1, 0) and (b) the pair of weighting parameters (1, 0.5)

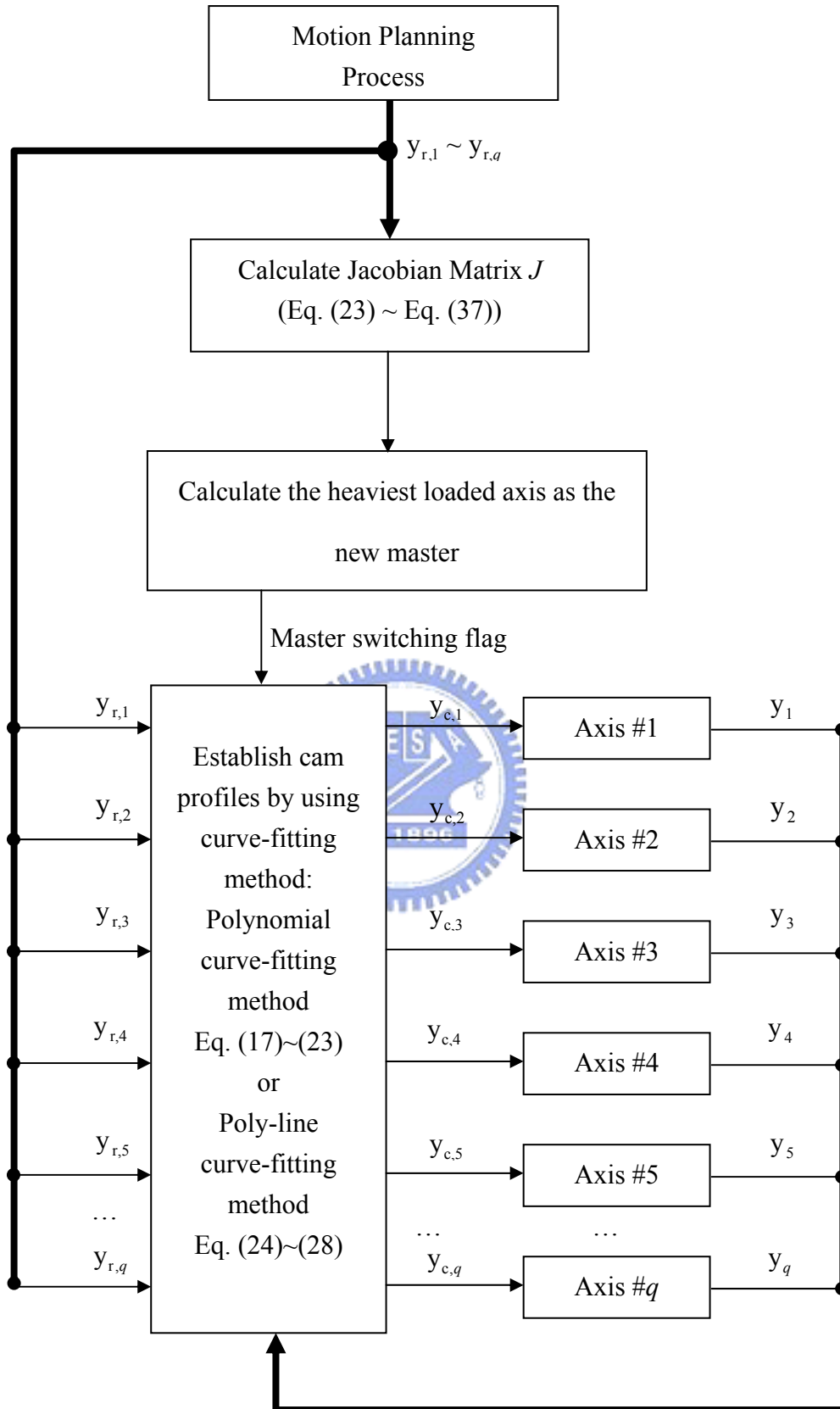


Fig. 4.1 Master switching method for q axes ECAM control

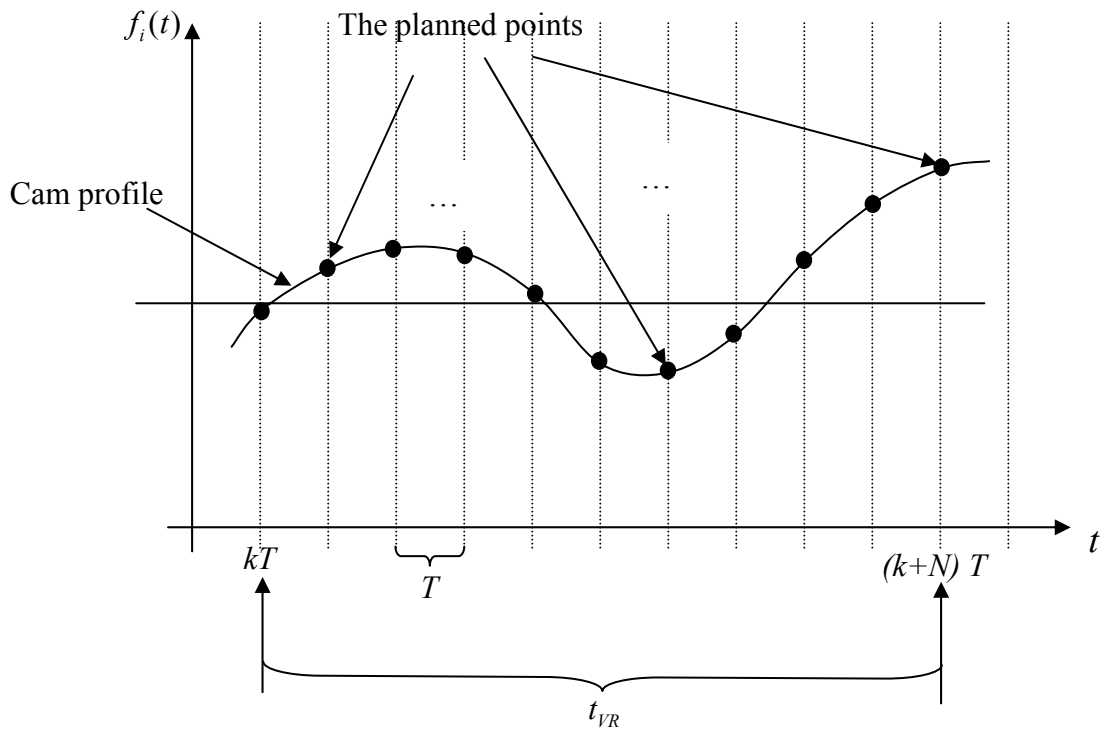


Fig. 4.2 Cam profile trajectory established using the polynomial curve-fitting method

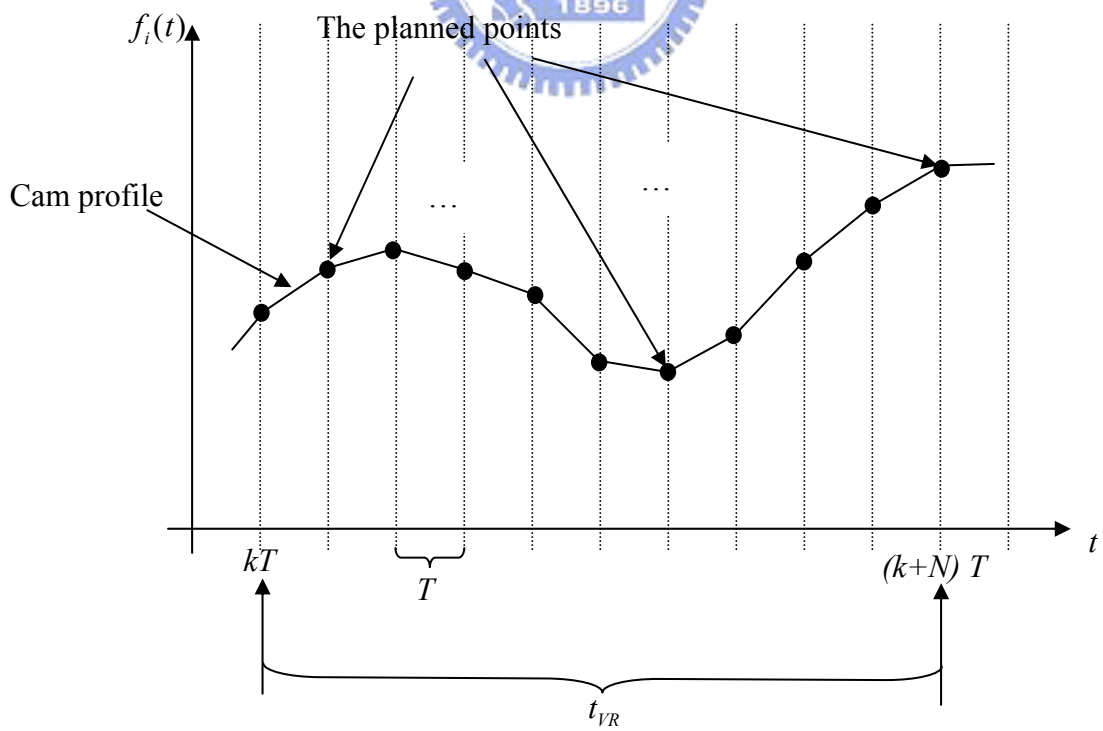


Fig. 4.3 Cam profile trajectory established using the poly-line curve-fitting method

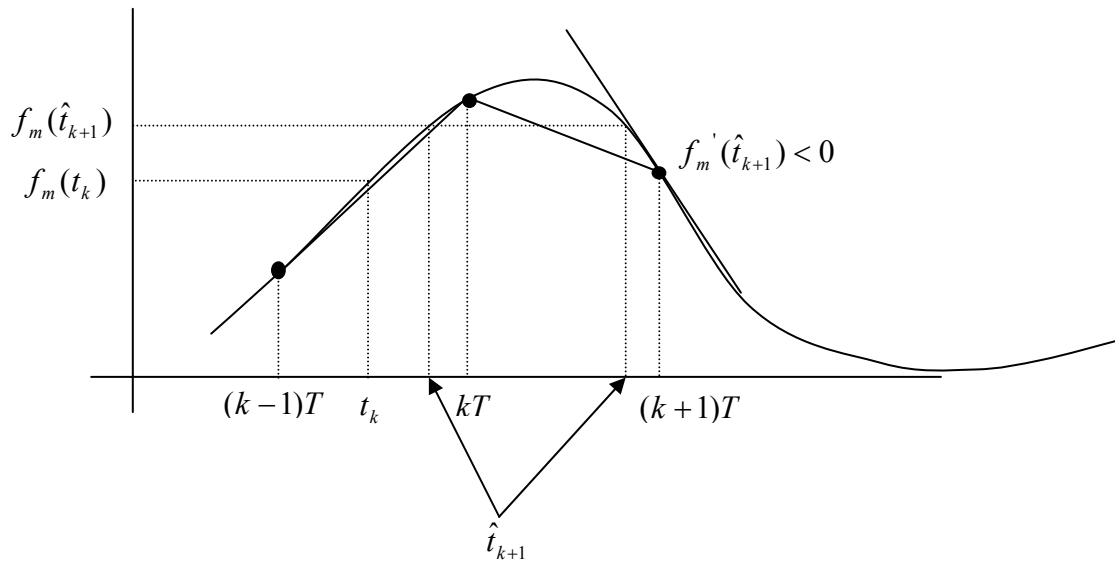


Fig. 4.4 Conditions on dual solutions using the polynomial curve-fitting method

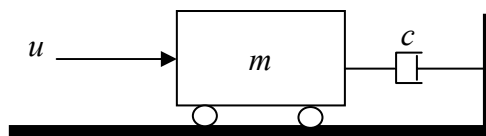


Fig. 4.5 Equivalent model of slider

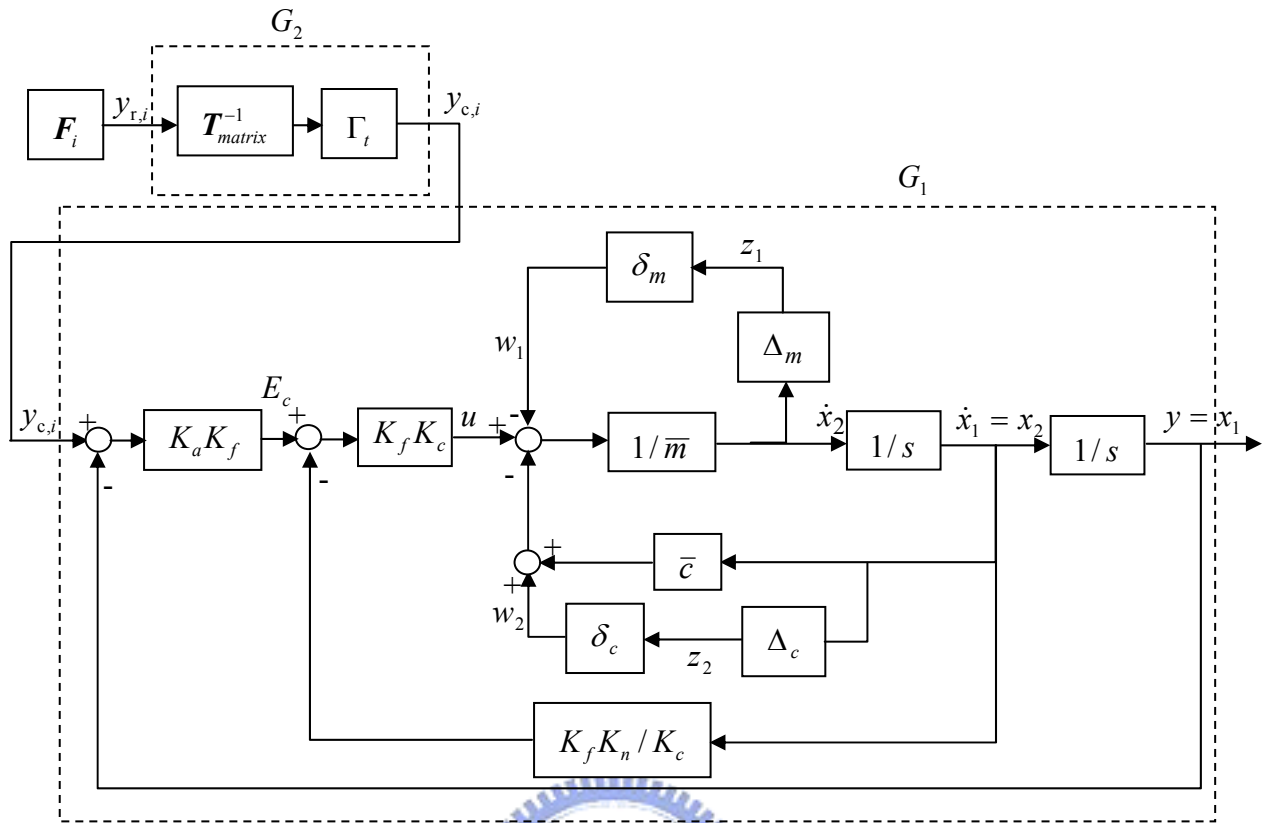


Fig. 4.6 Simplified control system's block diagram of control system of each slider of simulator SP-120

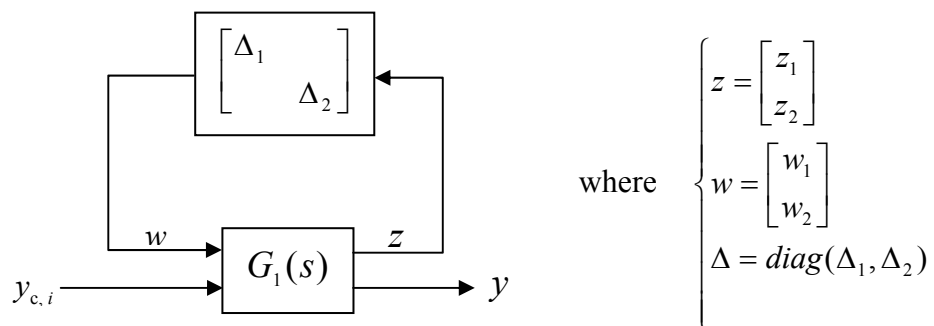


Fig. 4.7 Upper linear fractional transformation with $\Delta_1 = \delta_m$, $\Delta_2 = \delta_c$ and $|\delta_m| < 1, |\delta_c| < 1$

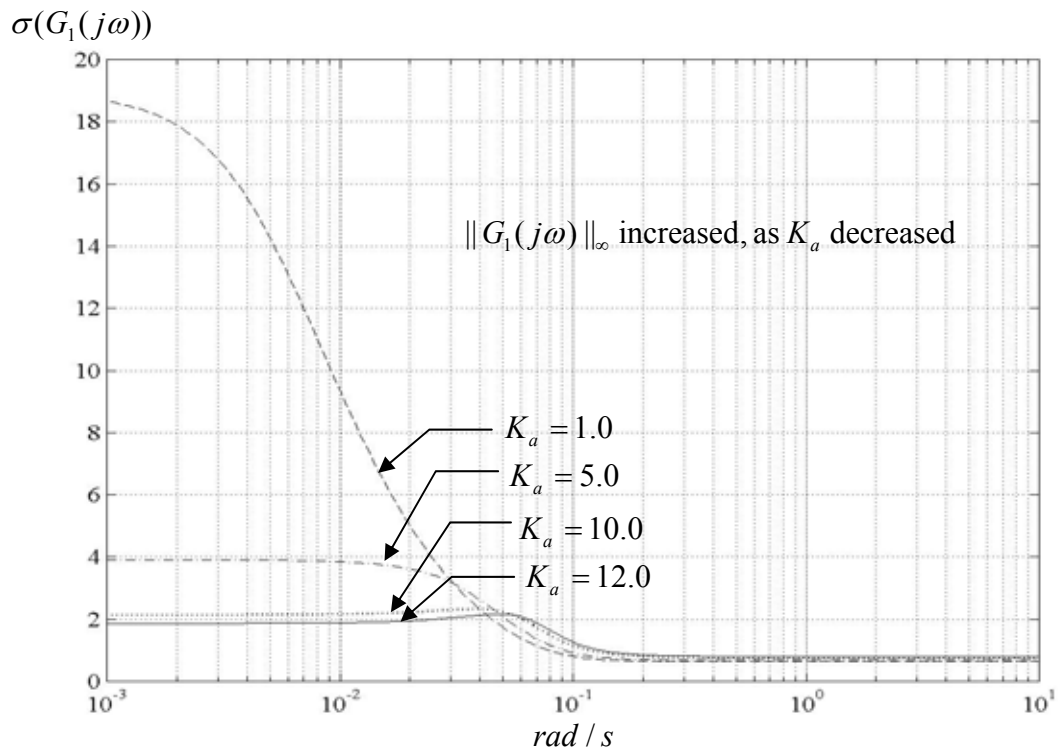


Fig. 4.8 Singular value frequency responses, $\sigma(G_1(j\omega))$, for various proportional gains,

K_a

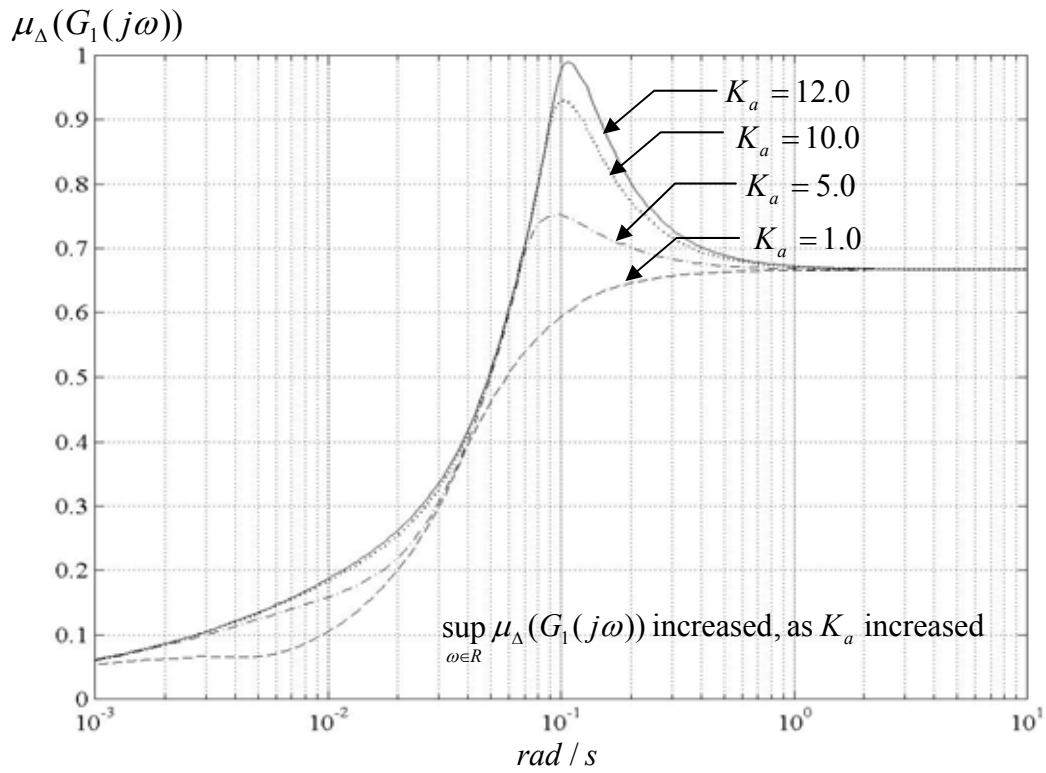


Fig. 4.9 Upper bounds of structured singular values, $\mu_\Delta(G_1(j\omega))$, for various proportional

gains, K_a

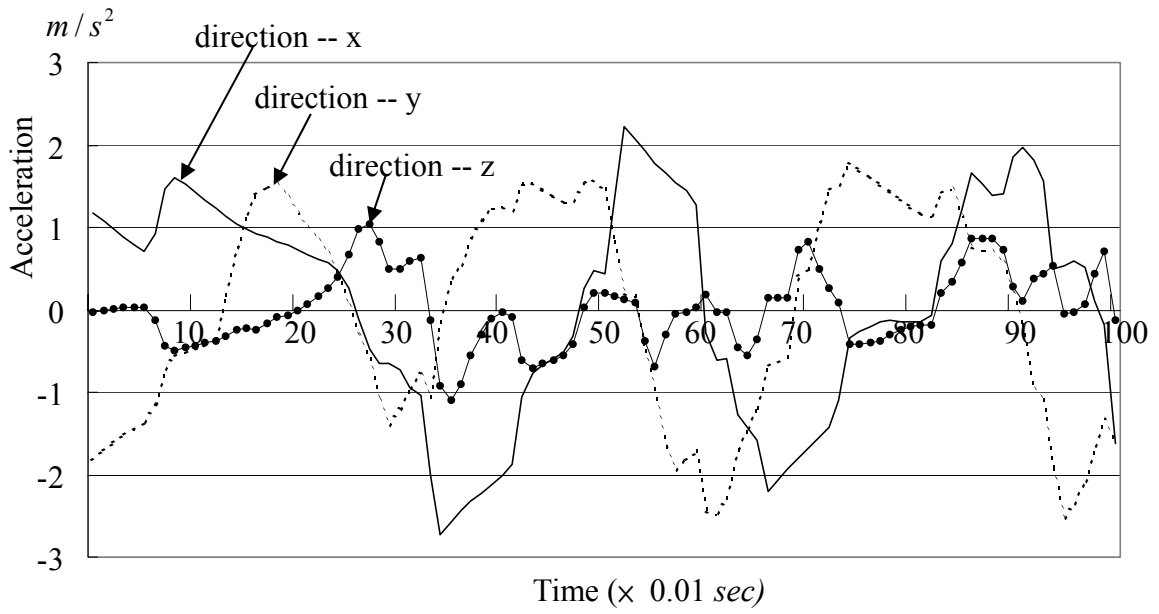


Fig. 4.10 The piecewise ground earthquake signal involves only the translation

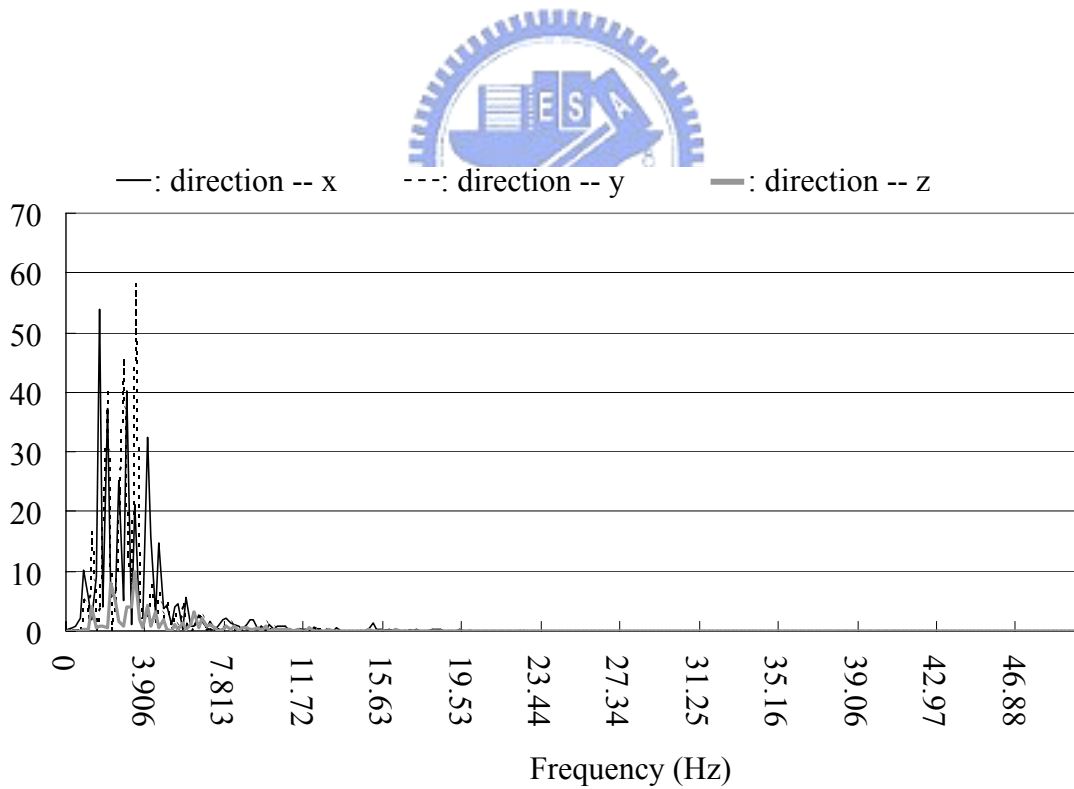


Fig. 4.11 Power spectrum density of the ground earthquake signal at various frequencies

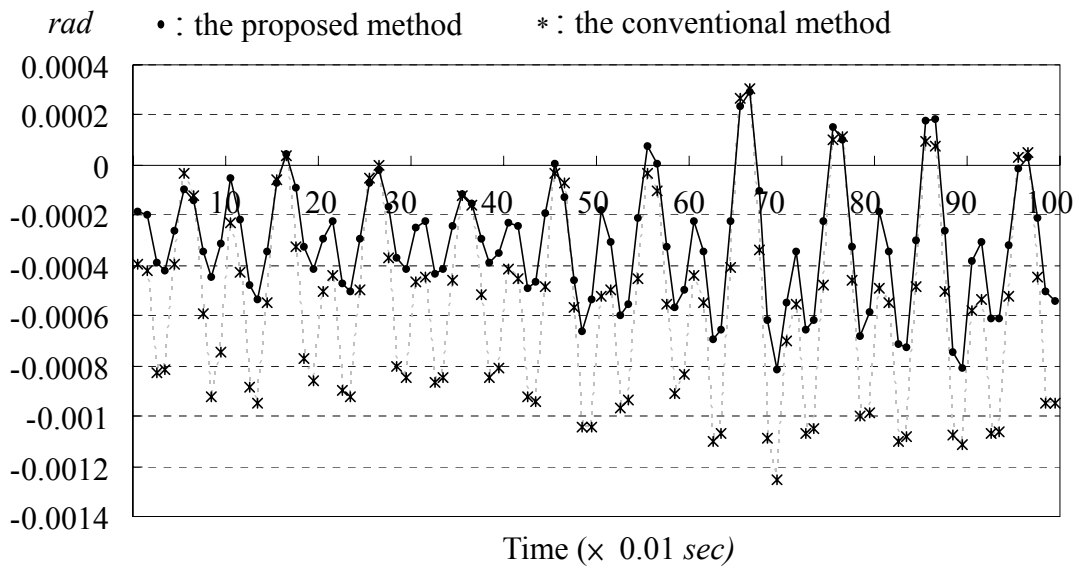


Fig. 4.12 Comparison of Euler's piecewise roll angle errors

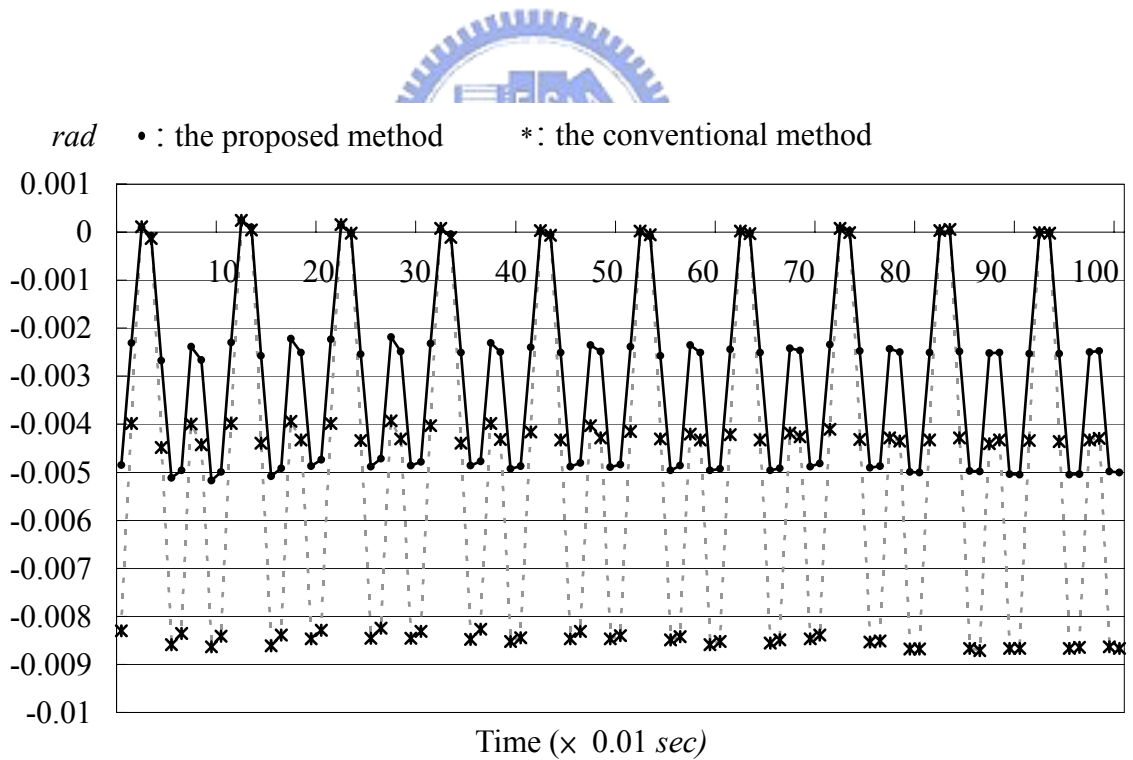


Fig. 4.13 Comparison of Euler's piecewise pitch angle errors

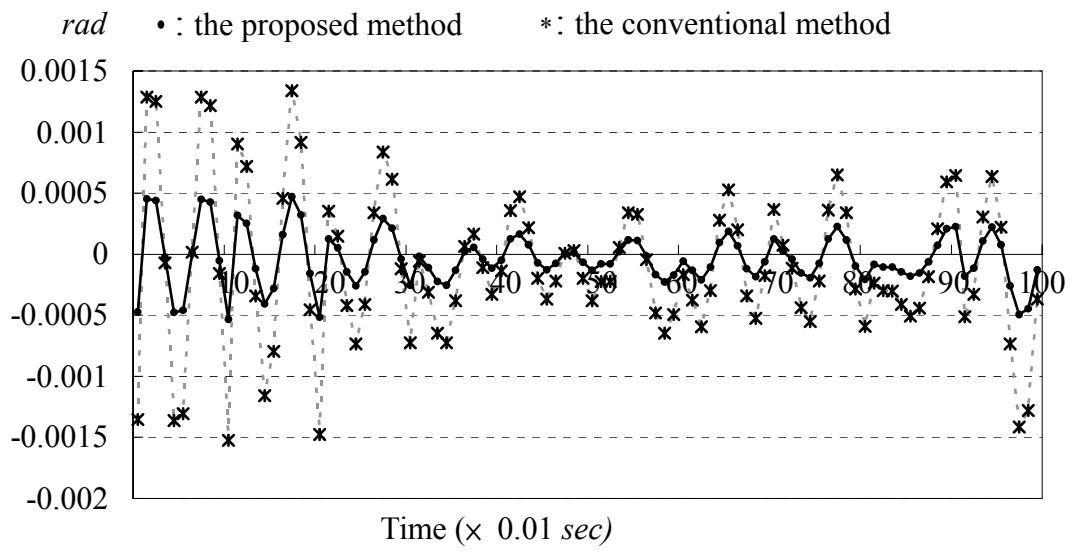


Fig. 4.14 Comparison of Euler's piecewise yaw angle errors





Fig. 5.1 Operating mode of the vehicle dynamics



Fig. 5.2 Colliding mode of the vehicle dynamics



Fig. 5.3 Hopping mode of the vehicle dynamics



Table 2.1 An example of cam profile table, both sets of data are scaled by their largest travel distance of one cam cycle.

Master Position X	0	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	1
Slave Position $f(x)$	0	0.006 45	0.048 63	0.148 63	0.306 45	0.5	0.693 55	0.851 37	0.951 37	0.993 55	1

Table 2.2 Experimental specifications of parameters for the ECAM control, last three data are scaled by their largest travel distance ($40\pi \text{ rad}$) of one cam cycle.

PC-based Programming Sampling Time T	Polynomial Order N	Critical Slave Velocity (C_v)	Critical Slave Acceleration (C_a)	Critical Slave Jerk (C_j)
0.01 s	0 ~ 5	2.5	55	573

Table 2.3 The maximum tracking error of the master's position for the N^{th} order polynomial tracking control ($N = 0 \sim 5$)

Order	0 th	1 th	2 th	3 th	4 th	5 th
Max. Err. (Counts/ $20\pi \text{ rad}$)	11	17	3	1	1	0

Table 2.4 An experimental example for the maximum tracking errors of the slave's position in encoder's counts (maximum travel distance: 200000 counts or 40π rad)

Order	0 th	1 th	2 th	3 th	4 th	5 th
Max. Error (encoder's counts)	395	655	83	17	2	1
RMS Error	194.936 8	325.6317	45.233 8	8.191 5	0.968 1	0.144 715
Cycle-to-cycle Variation	0.590 083	0.449 440	0.140 121	0.075 427	0.129 316	0.072 357

Table 2.5 An experimental example for the maximum slave velocity, acceleration and jerk

Performance Index Conditions	Maximum Velocity	Maximum Acceleration	Maximum Jerk
Applying the optimization algorithm to the cubic B-spline curve-fitting process	2.5(scaled) (equivalent to 314.16 rad / s)	15.75 (scaled) (equivalent to 1979.20 rad / s ²)	573.00 (scaled) (equivalent to 7.2005e4 rad / s ³)
purely curve-fitting using Lagrange polynomial	2.5(scaled) (equivalent to 314.16 rad / s)	51.09(scaled) (equivalent to 6420.16 rad / s ²)	10476.50 (scaled) (equivalent to 1.316516e6 rad / s ³)

Table 3.1 Capabilities of several motion simulators

Maximum Displacement Simulator Platform	Heave (feet)	Sway (feet)	Surge (feet)	Roll (degree)	Pitch (degree)	Yaw (degree)
*FSAA 6-DOF	10	100	8	90	44	60
*VMS 6-DOF	50	35	8	36	36	48
*NADS	4	90	30	80	80	continuous
*LAMARS 5-DOF	20	20	--	50	50	50
*MIL-STD 1558 6-DOF	5.6	5.6	5.6	40	50	40
SP-120	0.3	0.3	0.3	8	8.6	11.5

* Edward A. Martin, "Motion and Force Cuing, Part I: Whole Body Motion," Flight & Ground Simulation Update, State University of New York, Binghamton NY, January 2000, pp. 5-18.

Table 3.2 Comparison between the CLWF and the proposed washout filter in terms of the magnitudes of $RMS(E_a)$, $RMS(E_w)$ and PI using various static scaling factors

(s_a, s_w)

Method	Performance items (s_a, s_w)	The magnitude of $RMS(E_{a,k})$	The magnitude of $RMS(E_{w,k})$	Performance Index (PI)
C L W F	(0.8, 1.0)	0.9313	0.8762	0.90375
	(0.8, 0.7)	0.9052	0.6984	0.95145
	(0.8, 0.5)	0.9394	0.7671	0.96970
	(0.8, 0.3)	0.8968	0.7254	0.94840
	(0.8, 0.1)	0.5569	0.5916	0.77845
The proposed washout filter	(0.8, 1.0)	0.2790	0.4831	0.38105
	(0.8, 0.7)	0.2790	0.3536	0.39205
	(0.8, 0.5)	0.2797	0.2356	0.37545
	(0.8, 0.3)	0.3087	0.1263	0.36485
	(0.8, 0.1)	0.2733	0.2630	0.63665

Note: The results were obtained after the first test using an arbitrary pair of adaptive scaling factors set to (1, 0.5).

Table 3.3 Magnitudes of $RMS(E_a)$, $RMS(E_\omega)$ and PI using various pairs of adaptive scaling factors (λ_a , λ_ω)

Test Sequence	Performance items	$RMS(E_{a,k})$	$RMS(E_{\omega,k})$	PI
	$\lambda_a, \lambda_\omega$			
1st	1.0, 1.0	0.3854	0.3178	0.3516
2nd	1.0, 0.5	0.3113	0.1904	0.2509
3rd	0.5, 1.0	0.3769	0.2793	0.3281
4th	1.0, 0.3	0.3093	0.1376	0.2235
5th	1.0, 0.1	0.2655	0.1569	0.2112
6th	1.0, 0.0	0.2095	0.1834	0.1965

Note: the static scaling factors (s_a, s_ω) here are set to (0.8, 0.7).

Table 4.1 Maximum singular values of $\|G_1(j\omega)\|_\infty$, maximum structured singular values of

$\sup_{\omega \in R} \mu_\Delta(G_1(j\omega))$ and bandwidth of control system for various proportional gains,

K_a ; the upper bound, m_H , of the nominal mass is set to 250 kg

K_a	0.1	1.0	5.0	10.0	12.0
$\ G_1(j\omega)\ _\infty$	88.939455	18.651650	3.910155	2.323414	2.168739
$\sup_{\omega \in R} \mu_\Delta(G_1(j\omega))$	0.666652	0.666658	0.752879	0.929482	0.989907
Bandwidth (rad)	0.00594	0.0188	0.0420	0.0594	0.0651

Table 4.2 Critical upper bounds of the nominal mass for various proportional gains, K_a

K_a	0.1	1.0	5.0	10.0	12.0
critical upper bounds (kg)	10,400.0	1414.9	433.6	281.8	254.1

Table 4.3 Root mean square (RMS) errors of Euler angles, obtained using the proposed master switching method and the conventional method for ECAM control executed on the simulator SP-120

Error items	RMS error of roll	RMS error of pitch	RMS error of yaw
conventional method	0.0015150 rad	0.003427 rad	0.0004285 rad
master switching method	0.0007445 rad	0.001988 rad	0.0001499 rad

本論文作者已發表的著作：

Reviewed paper

J.-M.LAI, Chung-Shu Liao and Wei-Hua Chieng, “*Modeling and Analysis of Nonlinear Guideway for Double-Ball Bar (DBB) Measurement and Diagnosis*,” Int. J. Mach. Tools Mamufact. in Elsevier Science Ltd, Vol. 5, pp. 687-707, 1997.

Chung-Shu Liao, Shyr-Long Jeng and Wei-Hua Chieng, “*Tracking Control for Electronic Cam Motion Generation with Special Reference to Constrained Velocity, Acceleration and Jerk*”, ISA Transactions, accepted in September 2003.

Chung-Shu Liao, Chih-Fang Huang, Wei-Hua Chieng, “*A Novel Washout Filter Design for a Six Degree-of-Freedom Motion Simulator*,” JSME International Journal, accepted in February 2004.

Yang-Hung, Chang, Chung-Shu Liao, Shyr-Long Jeng and Wei-Hua Chieng, “*A Novel Master Switching Method for Electronic Cam Control with Special Reference to Multi-Axis Coordinated Trajectory Following*”, Elsevier journals, in Control Engineering Practice, revised in May 2004.



Chung-Shu Liao, Wu-Jong Yu and Shyr-Long Jeng, “*Space Vector PWM Synthesis by Decomposing 3D Geometrical Vectors*,” Elsevier journals, in Electric Power Systems Research, submitted in December 2003.

Conference paper

Chung-Shu Liao, Jun-Rong Chen and Shyr-Long Jeng, “*3D Geometrical Mapping for Space Vector PWM*”, 2002 工業自動控制與電力應用研討會, pp. D2-18~D2-23, 高雄, October, 2002.