HUMAN PERFORMANCE ON MODEL REFERENCE ADAPTIVE CONTROL SYSTEMS WITH HUMAN REACTION DELAY

A Thesis Presented
By
Yanchao Wang
to
The Department of Engineering

in partial fulfillment of the requirements for the degree of

Master of Science
in the field of
Mechanical Engineering

Northeastern University
Boston, Massachusetts

August 2018
Model Reference Adaptive Control (MRAC) is capable of dealing with system uncertainties effectively when controlling real-world systems. Such a capability can be beneficial in human-in-the-loop robotic control settings, where human and robot uncertainties can be handled properly to render the human-machine interface stable, and potentially high-performing.

In this thesis, an MRAC architecture is designed and experimented with volunteering human subjects, with the guidance of recent theoretical results considering human reaction delays. To this end, stabilizing designs based on linear quadratic regulators (LQR), modified LQR, and pole placement are pursued, and the stability of the closed-loop system is then analyzed with respect to human reaction delays, using TRACE-DDE toolbox assuming that the human model is given by a Neal-Smith pilot model.

The control design efforts help determine the appropriate settings for studies in experimental settings. Here, we develop three design conditions for MRAC to experiment with human subjects, under approved NU IRB protocol. Specifically, we designed a Cursor Control Game for the subjects to play, where a subject uses the computer mouse to move a cursor on the screen for tracking purposes. While performing this task, an MRAC in the background assists the subject’s mouse commands in order to help the subject in the tracking game.

According to the data collected in the experiments with human subjects, performance indexes are defined and three different designs of MRAC systems are compared based on both human performance and human control effort. The comparison shows that the faster
response and agility within the MRAC design would positively contribute to the performance of the human-in-loop MRAC system. The presentation concludes with mathematical modeling of the human behavior using Matlab’s System Identification Toolbox, identification of the robust region of model pole configurations, and discussions on the next steps in future work.
Praise the Lord for His comfort and guide with me in my life.

I am grateful for my parents’ support, emotionally and financially. I am grateful for the company of my brothers and sisters in fellowship.

Thank my advisor Dr. Rifat Sipahi, who provided me whatever is required for research and illuminate the potential further research to me.

Thank Dr. Adrián Ramírez and Dr. Min Hyong Koh for their advice in writing.

Thank Northeastern University for generously providing the compensation for volunteers in experiment with IRB#: 18-04-05.

At last, I would like to thank all the 14 participants who took the Cursor Control Game and provided us precious data, without whom the comparison of designs of MRAC system would not be possible.
# TABLE OF CONTENTS

## CHAPTER 1: INTRODUCTION

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Overview</td>
<td>1</td>
</tr>
<tr>
<td>1.1 Human Robot Interaction</td>
<td>1</td>
</tr>
<tr>
<td>1.1.1 Differences between HRI and HCI</td>
<td>1</td>
</tr>
<tr>
<td>1.1.2 Human Control of Robot Movement</td>
<td>2</td>
</tr>
<tr>
<td>1.1.3 Human Multi-Agent Robot Interaction System based on Passivity</td>
<td>3</td>
</tr>
<tr>
<td>1.1.4 Physical Human-Robot Interaction</td>
<td>3</td>
</tr>
<tr>
<td>1.2 Time Delays in Human-in-the-Loop Control Systems</td>
<td>4</td>
</tr>
<tr>
<td>1.2.1 Existence and Effects</td>
<td>4</td>
</tr>
<tr>
<td>1.2.2 Classification of Linear Time-delay Systems</td>
<td>5</td>
</tr>
<tr>
<td>1.2.3 Main Methods for Stability Analysis</td>
<td>6</td>
</tr>
<tr>
<td>1.3 Dynamic Pilot Models</td>
<td>7</td>
</tr>
<tr>
<td>1.3.1 The Precision Model</td>
<td>7</td>
</tr>
<tr>
<td>1.3.2 The Tustin-McRuer Model</td>
<td>7</td>
</tr>
<tr>
<td>1.3.3 The Gross Model</td>
<td>8</td>
</tr>
<tr>
<td>1.3.4 The Tustin Model</td>
<td>8</td>
</tr>
<tr>
<td>1.3.5 Neal-Smith Model</td>
<td>8</td>
</tr>
<tr>
<td>1.3.6 Defects of Dynamic Pilot Models</td>
<td>8</td>
</tr>
<tr>
<td>1.4 Model Reference Adaptive Control Systems</td>
<td>9</td>
</tr>
</tbody>
</table>
CHAPTER 2: DESIGN OF SYSTEM..............................................................................................................10

2.1 BLOCK DIAGRAM OF HUMAN-IN-LOOP MRAC SYSTEM ..........................................................10

2.2 MRAC SYSTEM ..............................................................................................................................11

  2.2.1 Formulations ..........................................................................................................................11

  2.2.2 Discrete-Time Equations for MRAC .....................................................................................13

  2.2.3 Boeing Airplane Model .........................................................................................................14

  2.2.4 Numerical Data for simulation of MRAC part ......................................................................15

2.3 HUMAN DYNAMICS ....................................................................................................................15

  2.3.1 Neal-Smith Model .................................................................................................................16

  2.3.2 The Design of Real Human Interaction with MRAC System ...............................................16

2.4 HUMAN-IN-LOOP MRAC SYSTEM ............................................................................................18

  2.4.1 Human Reaction Delay Effects ............................................................................................18

  2.4.2 Design of Reference Model ..................................................................................................21

  2.4.6 Results ....................................................................................................................................34

CHAPTER 3: EXPERIMENT OF REAL HUMAN WITH MRAC SYSTEM ..............................................35

3.1 SETUP OF EXPERIMENTS ............................................................................................................35

  3.1.1 Recruitment Procedure .......................................................................................................35

  3.1.2 Human Reaction Time Test ..................................................................................................37

  3.1.3 Cursor Control Game ............................................................................................................37

3.2 EXPERIMENTAL PROCEDURE ....................................................................................................39

  3.2.1 Preparation ...........................................................................................................................39

  3.2.2 Activities .............................................................................................................................39

  3.2.3 Durations, Number of Sessions, Breaks ...............................................................................40

CHAPTER 4: ANALYSIS OF EXPERIMENT DATA ...............................................................................40

4.1 HUMAN REACTION DELAY MEASUREMENT ............................................................................40

4.2 PERFORMANCE INDEX ...............................................................................................................41
LIST OF TABLES

TABLE 1 NUMERICAL DATA OF THE MRAC ................................................................. 15
TABLE 2 PARAMETERS FOR NEAL-SMITH PILOT MODEL .................................................. 16
TABLE 3 STABILITY ANALYSIS USING TRACE-DDE FOR CASES DESIGNED BY CLASSICAL LQR OF REFERENCE MODEL AND DELAY IN HUMAN MODEL .................................................................................. 23
TABLE 4 STABILITY ANALYSIS USING TRACE-DDE FOR CASES DESIGNED BY MODIFIED LQR .......................................................... 27
TABLE 5 STABILITY ANALYSIS USING TRACE-DDE FOR CASES DESIGNED BY POLE PLACEMENT .................................................... 33
# LIST OF FIGURES

**Figure 1** Block diagram of the human-in-loop MRAC system (inspired by [47]) .................................................. 10

**Figure 2** Simplified block diagram for stability analysis and reference model design ............................................. 20

**Figure 3** Simulation results with different $R$, $\theta$ represents the pitch angle and $u$ is the command from Neal-Smith pilot model to MRAC system .................................................................................................................. 24

**Figure 4** Simulation results with different $\alpha$ ................................................................................................................... 28

**Figure 5** Zoomed-in time domain step response for pole placement method ................................................................. 31

**Figure 6** Simulation result with three sets of poles ........................................................................................................... 34

**Figure 7** Graphical User Interface of Human Experiment .................................................................................................. 38

**Figure 8** Reaction delay time, in which subject 2 and 10 are absent from the experiments ........................................... 41

**Figure 9** Average and error bar of standard deviation for system output and human command input over time .................................................................................................................................................. 42

**Figure 10** Illustration of Basic Metrics .......................................................................................................................... 43

**Figure 11** MISE average and error bar for pole sets (figures are in different range for visualization purpose) ......................................................................................................................................................... 45

**Figure 12** MISE average and error bar for test sequences (figures are in different range for visualization purpose) ..................................................................................................................................................... 47

**Figure 13** Average variance of human input and error bar for pole sets (figures are in different range for visualization purpose) ........................................................................................................................................ 49

**Figure 14** Average variance of human input and error bar for test sequences (figures are in different range for visualization purpose) ........................................................................................................................................ 50
Figure 15 Average time when state stays in 5% range (Figures are in different range for visualization purpose) .......................................................... 52

Figure 16 Average time when state stays in 20% range (Figures are in different range for visualization purpose) .......................................................... 53

Figure 17 Average number of times Human input changes direction (Figures are in different range for visualization purpose) .......................................................... 54

Figure 18 Average number of times System output changes direction (Figures are in different range for visualization purpose) .......................................................... 55

Figure 19 Numbers of cases within threshold .................................................................................................................. 57

Figure 20 Block diagram of real human with reaction delay .................................................................................................. 58

Figure 21 Comparison between real human and identified system .................................................................................. 61

Figure 22 Number of cases within threshold .................................................................................................................. 62

Figure 23 Center of poles and zeros with standard deviation on real axis .......................................................... 63
Chapter 1: Introduction

Overview

With the decrease in costs and the development in reliability, implementation of robots has been rapidly expanding in various fields, especially in labor intensive works [1]. Additionally, robots have unreplaceable advantages over real humans, especially in dangerous tasks, such as in urban search and rescue [2] [3] and in hazardous environment which might harm worker’s health [4]. Apart from above fields, applications in medical purpose are also in progress. After a long history of rehabilitation robots, a sharp growth occurred in the past decade due to the shift of ideas from assistive devices to robotic therapy [5]. In providing entertainment, Sony’s Aibo and many other devices are available for customers. The wide applications of robots make human-robot interaction an essential research topic toward designing such interactions safe and functional.

1.1 Human Robot Interaction

During the period of infancy of human-robot interactions (HRI) in the first decade of 21st century, topics on the definition, taxonomy, and models were discussed. According to the definition presented in 1992 by the Curriculum Development Group of the Association for Computing Machinery (ACM) Special Interest Group on Computer-Human Interaction (SIGCHI):" Human-computer interaction is a discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them" [6]. The robot fits the definition of the computing systems and therefore the human-robot interaction (HRI) could be considered as a subset of the area of human-computer interaction (HCI).

1.1.1 Differences between HRI and HCI

In the study [7], Fong notes that HRI is different from HCI and human-machine interaction (HMI) mainly because the robot operates in a real-world environment which is
complex and keeps changing. Despite the working environment, differences in interaction roles and from the physical nature of robots and the fact that human may interact with multiple systems simultaneously are discussed in [8].

Scholtz defined five types of roles in interaction [8]: supervisor, operator, bystander, teammate, and mechanic. In his illustration, supervisory and teammate exists when there are human-human interactions in the process; operator’s function is to control the robot behavior; mechanic type of interaction aims at adjusting physical components; and the bystander will try to understand what will happen before robot takes action. The physical nature of robots means the ability of robots to interact with physical world, which brings the requirement to observe surroundings. The third difference is the dynamic nature of the robot platform. In HCI, the computer itself is considered not to change over time. However, there is possibility for sensors on the robotic platform to degrade or fail. A dusty, poor illumination, noisy, and sometimes even shaky or unsteady place could be the working environment for urban search and rescue [9] [10], which results in the fourth difference between HRI and HCI. The fifth difference is in HRI. Human is requested to interact with several different systems simultaneously, which is different from the fact that in HCI human interacts with one system and another interaction occurs as computer-computer interaction. The final difference is in the way the computer and robot deal with commands. The implementation of planning software allows users to give robots higher level commands and decisions.

1.1.2 Human Control of Robot Movement

1.1.2.1 Hazard of Human Control of Robot Movement

With the replacement of human labor in hazardous and monotonous tasks, robots have brought new potential risks for human workers. In the analysis of robot accidents in [11], hazards can happen in any process when working with robots, including normal operation, maintenance and programming. In the 24 accidents recorded with definite cause in the cited paper, 13 cases are caused by human error directly. Additionally, poor workplace design
contributes to 20 of 32 accidents. Although regulations are published to prevent humans from being hit by robots, it is still unsolved for operators on site, see [12].

1.1.2.2 Solutions to Risks in Human Control of Robot Movement

Apart from regulating humans and restrict their access to works space, the implementation of safeguards in designing robots [12], including physical barriers, presence-sensing devices and ‘intelligent’ detection and control systems could also contribute to a safer working environment.

1.1.3 Human Multi-Agent Robot Interaction System based on Passivity

One main difference between multi-agent robots and conventional robots is the essential requirement of teleoperation. Rich knowledge in the field of bilateral teleoperation [13] contributes to the rapid development in human-swarm interactions, which has attracted considerable attention in recent years. During the long history of research in bilateral teleoperation, among many control algorithms developed in this field, passivity based control algorithm is recognized as one of the most powerful design concepts for such swarm systems [14]. The main reason for this technique being powerful is illustrated in [13]. Combining passive environment, passive human operator and passive teleoperator could guarantee the passivity of the closed-loop system. And the stability analysis of a passive system is to sum all constituting blocks’ storage function as a Lyapunov function, which means if all the blocks in the system are stable separately, the overall system is stable [14]. Using passivity based control algorithm could largely simplify the design of multi-agent robot system and the stability analysis.

1.1.4 Physical Human-Robot Interaction

In the above discussions, humans are asked to control robots instead of interacting with them. In cases of interaction, the robot is required to have the ability to make real-time decisions and even predict human behavior to provide a response, e.g., assistance. However,
due to the uncertainty of human behavior, the behavior of the robot might be based on incorrect prediction, which could possibly make things even worse. To meet this requirement, a new control approach is developed based on risk-sensitive optimal feedback control in [15]. In the experiment, the robot can change its attitude between passive and dominant behavior based on human physical interaction force, which is proved to be capable of improving human–robot interaction performance and lowering the potential risks in interacting with the robot physically.

1.2 Time Delays in Human-in-the-loop Control Systems

1.2.1 Existence and Effects

In many dynamical systems, delay exists in acquiring information, making decision and executing decisions [16] which contribute to the result that events do not happen simultaneously. Systems with delays widely exist in engineering, biology, physics, operations research, and economics [16].

In [17], delays in human brain and its effects are illustrated. According to Stepan's example, vibrations exist in every human body, for example, during balancing. Healthy humans could easily suppress vibrations and maintain stability. However, due to the malfunctioning of the neural system, an increased delay [18] could cause unmanageable shifts in the phase of neural signals. It is widely discussed that tremor in the fingers, the arm and the body; difficulties in balancing; the increased danger of falling over for elder people and even motion disorders in the case of bursts of epilepsy, multiple sclerosis, Parkinson disease and so on are in part caused by the abnormal increase in delay in human neural system [17].

In human-in-the-loop systems, human reaction time is always present because humans need time to observe the situation and make decisions before operating the system. Take traffic-flow model as an example. The authors in [19] define three characteristic times
in traffic, one of which is human reaction time of about 0.5-1.5s. This delay time is a dead time for drivers they need to become aware of their surroundings, plan and initiate actions. Such a delay could in part contribute to traffic jams and stop-and-go waves – a major research topic. Research by Helbing in [20] shows that, in Europe, drivers spend several days standing in traffic jams each year. Economic losses are at the level of billions of dollars [21]. Additionally, the emission of $SO_2$, $NO_x$, $CO$, $CO_2$, dust particles, smog, and noise have reached alarming levels [22].

The existence of delay does not necessarily contribute to the instability of a system. In some cases, the presence of delays could help stabilize the system [16]. In [23], a controller is designed to stabilize the multi-agent system by intentionally introducing delays in the controller. Discussions on the stabilizing effects of delay can be found in [16].

1.2.2 Classification of Linear Time-delay Systems

1.2.2.1 Retarded Type

Retarded type of time-delay systems could be described by the following set of delay differential equations (DDEs):

$$\dot{x}(t) = A_0 x(t) + \sum_{i=1}^{m} A_i x(t - \tau_i)$$

where $x(t) \in \mathbb{R}^n$ is the state variable at time $t$, $A_i \in \mathbb{R}^{n \times n}, i = 0, 1, \ldots, m$, and $0 < \tau_1 < \tau_2 \cdots < \tau_m$ is the time delays in the system. The system is called 'retarded' since the term $\dot{x}(t)$ is not influenced by delays, but only $x(t)$ [24] [25].

1.2.2.2 Neutral Type

Neutral type could be illustrated by the following DDEs:

$$\frac{d}{dt} \left( x(t) + \sum_{k=1}^{m} H_k x(t - \tau_k) \right) = A_0 x(t) + \sum_{k=1}^{m} A_k x(t - \tau_k)$$
where $x(t) \in \mathbb{R}^n$ is the state variable at time $t$, $A_i \in \mathbb{R}^{n \times n}, i = 0, 1, \ldots, m$, $H_i \in \mathbb{R}^{n \times n}, i = 0, 1, \ldots, m$, and $0 < \tau_1 < \tau_2 \cdots < \tau_m$ is the time delays in the system. Different from retarded type, neutral type systems carry the effects of delays in $\dot{x}(t)$ [24].

### 1.2.2.3 Advanced Type

Advanced type DDEs carry an ‘advance’ term with $x(t + \text{delay})$, that is,

$$\dot{x}(t) = A_0 x(t) + \sum_{i=1}^{m} A_i x(t + \tau_i)$$

where $x(t) \in \mathbb{R}^n$ is the state variable at time $t$, $A_i \in \mathbb{R}^{n \times n}, i = 0, 1, \ldots, m$, and $0 < \tau_1 < \tau_2 \cdots < \tau_m$ is the time delays in the system. This type of system requires states in the future to determine the states in the next time instant. Since this does not respect the principal of causality, advanced type DDEs do not arise in real world engineering problems.

### 1.2.3 Main Methods for Stability Analysis

Methods for analyzing stability for time-delay systems could be categorized into frequency domain methods and time domain methods. In time domain, Lyapunov stability theory is utilized [26] along with convex optimization methods and linear matrix inequalities (LMIs) [27] to analyze the stability of DDEs. Moreover, since the location of the characteristic roots on the complex plane determine stability, the location of these roots with respect to the imaginary axis can be studied for stability assessment. This approach is also known as the “frequency domain” approach. Such approaches primarily build on D-subdivision and $\tau$-decomposition theorems [28]. The Cluster Treatment of Characteristic Roots (CTCR) [29] which is a frequency domain-based approach, has been widely utilized to study stability of LTI systems influenced by delays; see also [16] for a review of the state of the art.

In this thesis, stability of linear system with time delay is determined via frequency domain techniques, mainly by using the Matlab package TRACE-DDE [30] based on pseudo-
spectral approach [31] [32]. This package is able to compute the rightmost roots of the
dynamics, which can be used to assess both stability and approximate system’s settling time.

1.3 Dynamic Pilot Models

In flight control, because the smallest mistake could possibly contribute to a lethal
disaster, special effort is devoted in this area to design airplane systems by considering
human behavior models.

In the last century, Robert McRuer showed that pilot’s behavior could be described by
linear dynamic system with some simplifying hypothesis. During 1960s, McRuer proposed
The Crossover Model in [33]. Based on this Crossover law, pilot models can be created.
Quasi-linear analytical model was taken as one of the most useful models in describing pilot
control behavior and analyzing the potential performance of pilot-plane system.

1.3.1 The Precision Model

The precision model is created by McRuer [34] which describes a human neuromuscular
system and could be represented by the following formula.

\[ F_H(s) = K_H \cdot \frac{T_L s + 1}{T_I s + 1} \cdot \frac{1}{s^2 + \frac{2 \zeta_N}{\omega_N} s + 1} \cdot e^{-\tau s} \]

where \( K_H \) is the gain of pilot, \( T_L \) is the time lead constant in second, \( T_I \) is the time lag
constant in second, \( \omega_N \) represents the natural frequency of human neuromuscular system,
\( \zeta_N \) represents the damping of the system, \( \tau \) is the pilot reaction delay in second, and \( s \)
represents the Laplace operator.

1.3.2 The Tustin-McRuer Model

The Tustin-McRuer model is created based on the precision model and becomes one
of the most widely used pilot model now [35]. In Tustin-McRuer model, the oscillation part
of the neuromuscular system is substituted by a neuromuscular constant $T_N$ in second. Transfer function could be illustrated in following formula [36].

$$F_H(s) = K_H \cdot \frac{T_l s + 1}{(T_I s + 1)(T_N s + 1)} \cdot e^{-ts}$$

1.3.3 The Gross Model

The Gross model applied another simplification on the precision model by considering the oscillation part as a new pilot response delay $\tau_N$ in second [34] [37] [38]. Transfer function is in following form.

$$F_H(s) = K_H \cdot \frac{T_l s + 1}{T_I s + 1} \cdot e^{-(\tau + \tau_N)s}$$

1.3.4 The Tustin Model

In 1944, Arnold Tustin created the Tustin model. According to the description in [39], the Tustin model could be expressed in the following form.

$$F_H(s) = K_H \cdot \frac{T_l s + 1}{s} \cdot e^{-ts}$$

1.3.5 Neal-Smith Model

In 1970, Neal and Smith designed a frequency domain technique for the analysis of aircraft [40], where the pilot pitch compensator could be modeled as

$$H(s) = k_p \cdot \frac{T_p s + 1}{T_z s + 1} e^{-ts}$$

where $k_p$ is the Neal-Smith pilot model gain, $T_p$ is the Neal-Smith pilot model lead time constant, $T_z$ is the Neal-Smith pilot model lag time constant, and $\tau$ represents the delay between the moment when pilot observed the state and pilot produces an action.

1.3.6 Defects of Dynamic Pilot Models

Despite the developments in pilot models and the increasing precision in describing
pilot's behavior, it was found that human behavior cannot be described as a pure integrator [34]. Above models can be suitable to gather a qualitative understanding of the human-in-the-loop system, but one is cautioned that real-world human dynamics can have nonlinearities as well. Thus, in the following simulations, the Neal-Smith pilot model is implemented in human-in-the-loop system to obtain some understanding of the human-in-the-loop systems and to perform stability analysis.

1.4 Model Reference Adaptive Control Systems

Generally speaking, adaptive control system is the system which uses the information collected in real time feedback of system state to adjust the controller automatically and, as a result, for an unknown plant model, the adjusted controller could maintain the system having a desired performance. The adaptive control could be divided into two categories, direct adaptive control and indirect adaptive control [41]. The direct adaptive control theory was developed by Whitaker’s team and published in 1958 in [42] for aircraft control. In this type of adaptive control, the adjustment of controller is based on the observation of the difference between the reference model and the output of the system. The main purpose of direct adaptive control system is to maintain the behavior of unknown plant model similar to the designed reference model. For the indirect adaptive control, the main method is estimating the plant model to design the controller parameters in real time [43]. The indirect adaptive control is also called the ad-hoc certainty equivalence principle or ad-hoc separation theorem [41], which is further developed by Åström and Wittenmark [44] and Clarke and Gawthrop [45] into self-tuning controller.

The first approach to analysis of stability of MRAC system is using the Lyapunov function in [46], which was generalized in [47]. The input-output method based on passivity concepts was used for design purpose [48]. In this paper, Lyapunov method is selected for stability analysis of the nonlinear part in MRAC system.
Chapter 2: Design of System

2.1 Block diagram of Human-in-Loop MRAC System

Figure 1 Block diagram of the human-in-loop MRAC system (inspired by [49])

Figure 1 is the human-in-loop MRAC system designed in [49]. In the figure, the entire human-in-loop MRAC system is separated into two parts, outer loop and inner loop.

The outer loop part represents the process for human to take actions according to the reference input (the goal human intended to reach) and feedback from Uncertain Dynamical System (the real state of the system) being controlled. In real situations, human dynamics should be a real human controlling the system when human understands what the goal is and what the output of the system is at any moment. Human models could be implemented here to simulate the human-in-the-loop MRAC system to see how the system might react prior to human subject experiments.

The inner loop part represents the core of MRAC system. The reference model is an ideal mathematical model which is designed by the control designer to meet certain requirements related to transient and steady state. The uncertain dynamical system is the real-world system which the control designer wants to control. However, because parameters of the system are unknown which is common, the controller cannot be designed
to ensure the behavior for real-world system to be exactly the same as that of reference model. Thus, parameters in controller block are to be adjusted by the parameter adjustment mechanism which will observe the differences between the output from reference model and from uncertain dynamic model and determine how to change the controller to decrease the difference so that the real-world uncertain dynamic system could behave similar as the reference model.

2.2 MRAC system

This section illustrates the design and parameter selection for the inner loop, see [49] for details.

2.2.1 Formulations

In the inner loop, the uncertain dynamical system is given by

$$\dot{x}_p = A_p x(t) + B_p \Lambda u(t) + B_p \delta_p \left(x_p(t)\right), \quad x_p(0) = x_{p0}$$

(1)

where $x_p(t) \in \mathbb{R}^{n_p}$ is the state vector, $u(t) \in \mathbb{R}^m$ is the input to the system, $\delta_p : \mathbb{R}^{n_p} \rightarrow \mathbb{R}^m$ is an uncertainty function, $A_p \in \mathbb{R}^{n_p \times n_p}, B_p \in \mathbb{R}^{n_p \times m}$ are two known system matrices, $\Lambda \in \mathbb{R}^{m \times m}$ is an unknown control effectiveness matrix. In the above system, $A_p$ and $B_p$ are assumed to be controllable and $x_p$ represents the accessible state of the system. The uncertainty function $\delta_p$ is defined as

$$\delta_p \left(x_p(t)\right) = W_p^T \sigma_p \left(x_p(t)\right)$$

(2)

where $W_p \in \mathbb{R}^{s \times m}$ is an unknown weight matrix and $\sigma_p : \mathbb{R}^{n_p} \rightarrow \mathbb{R}^s$ is a known basis function $\sigma_p(x_p) = \left[\sigma_{p_1}(x_p), \sigma_{p_2}(x_p), ..., \sigma_{p_s}(x_p)\right]^T$.

Then let $x_c(t) \in \mathbb{R}^{n_c}$ be the integrator state which is

$$\dot{x}_c(t) = E_p x_p(t) - c(t), \quad x_c(0) = x_{c0}$$

(3)
where $E_p \in \mathbb{R}^{n_c \times n_p}$ which is used to choose part of the state vector $x_p(t)$, $c(t) \in \mathbb{R}^{n_c}$ is the command in Figure 1 produced by human.

According to equations (1) and (3), following equation could be derived.

$$\dot{x}(t) = Ax(t) + B\Lambda u(t) + BW_p^T \sigma_p \left( x_p(t) \right) + B_r c(t), x(0) = x_0 \quad (4)$$

where

$$A \triangleq \begin{bmatrix} A_p & 0_{n_p \times n_c} \\ E_p & 0_{n_c \times n_c} \end{bmatrix} \in \mathbb{R}^{n \times n} \quad (5)$$

$$B \triangleq \begin{bmatrix} B_p^T \\ 0_{n_c \times m} \end{bmatrix}^T \in \mathbb{R}^{n \times m} \quad (6)$$

$$B_r \triangleq \begin{bmatrix} 0_{n_p \times n_c} \\ -I_{n_c \times n_c} \end{bmatrix}^T \in \mathbb{R}^{n \times n_c} \quad (7)$$

$$x(t) \triangleq \begin{bmatrix} x_p^T(t) \\ x_c^T(t) \end{bmatrix} \in \mathbb{R}^n, \quad x_0 \triangleq \begin{bmatrix} x_{p0}^T \\ x_{c0}^T \end{bmatrix} \in \mathbb{R}^n \quad (8)$$

The controller part is designed based on MRAC theory. Thus, the signal from controller to plant could be considered as

$$u(t) = u_n(t) + u_a(t) \quad (9)$$

where $u_n(t) \in \mathbb{R}^m$ is the input derived from nominal control law and $u_a(t) \in \mathbb{R}^m$ is the input based on adaptive control law. The $u_n(t)$ could be designed as

$$u_n(t) = -Kx(t) \quad (10)$$

where $K \in \mathbb{R}^{m \times n}$. $A_r \triangleq A - BK$ is assumed to be a stable matrix.

Implement (9), (10) in (4).

$$\dot{x}(t) = Ax(t) + B\Lambda(-Kx(t) + u_a(t)) + BW_p^T \sigma_p \left( x_p(t) \right) + B_r c(t) \; (11a)$$

Rearrange (11)

$$\dot{x}(t) = A_r x(t) + B_r c(t) + B\Lambda[u_a(t) + (\Lambda^{-1} - I)Kx(t) + \Lambda^{-1}W_p^T \sigma_p \left( x_p(t) \right)] \quad (11b)$$
Simplify (11b) by defining the new basis function as \( \sigma^T(x(t)) \triangleq \begin{bmatrix} \sigma_p^T(x_p(t)) \\ x^T(t) \end{bmatrix} \in \mathbb{R}^{s+n} \).

\[
\dot{x}(t) = A_rx(t) + B_rc(t) + BA[u_a(t) + W^T\sigma(x(t))] \tag{11c}
\]

where \( W^T \triangleq [\Lambda^{-1}W_p^T \ (\Lambda^{-1} - I_{m\times m})K] \in \mathbb{R}^{(s+n)\times m} \) is an unknown weight matrix.

Design the adaptive control law as

\[
u_a(t) = -\hat{W}^T(t)\sigma(x(t)) \tag{15}\]

\[
\hat{W}(t) = \gamma \sigma(x(t))e^T(t)PB, \quad \hat{W}(0) = \hat{W}_0 \tag{16}
\]

where \( \hat{W}^T(t) \in \mathbb{R}^{(s+n)\times m} \) is an estimation of \( W \), \( \gamma \in \mathbb{R}_+ \) is the learning rate and the system error is defined as

\[
e(t) \triangleq x(t) - x_r(t) \tag{17}\]

in which \( x_r(t) \in \mathbb{R}^n \) represents the state vector of the reference model which yields

\[
\dot{x}_r = A_r x_r(t) + B_r c(t), \quad x_r(0) = x_{r0}. \tag{18}\]

\( P \in \mathbb{R}^{n\times n}_+ \cap \mathbb{S}^{n\times n} \) in (16) is a solution of the Lyapunov equation

\[
0 = A_r^T P + PA_r + R \tag{19}
\]

with \( R \in \mathbb{R}^{n\times n}_+ \cap \mathbb{S}^{n\times n} \). Note that for a given \( R \), there exists a unique \( P \) when \( A_r \) is stable [49].

### 2.2.2 Discrete-Time Equations for MRAC

To prepare for the simulation in Matlab, the above MRAC system is transformed into discrete time. Following equations are presented to prepare for the simulations.

From (17), error is calculated in the following discrete formula. Error is calculated first

\[
e[n] = x[n-1] - x_r[n-1]
\]
where $e[n]$ represents the error and $x[n]$, and $x_r[n]$ are state vector of uncertain dynamical system and state vector of reference model, respectively. Based on the derived error and equation (16), we have

$$\tilde{W}[n] = \tilde{W}[n - 1] + \gamma \sigma(x[n - 1]) e^T[n] P B \Delta t$$

where $\tilde{W}[n]$ is the estimation of $W$, $\sigma(x[n])$ is the basis function as $\sigma^T(x[n]) \equiv [\sigma_p^T(x_p[n]) \ x^T[n]]$ and $x_p[n]$ is the state vector of the plant. $\Delta t$ represents the time interval between two simulation steps. Thus, after implementing (10) and (15) to (9), the control signal from controller to plant is

$$u[n] = -K x[n - 1] - \tilde{W}^T[n] \sigma(x[n - 1]).$$

The discrete form of differential equation (18) which simulates the state of designed reference model is

$$x_r[n] = x_r[n - 1] + (A_r x_r[n - 1] + B_r c[n]) \Delta t$$

with $c[n]$ as the sampled human command input from human dynamics to MRAC system. Implement (15) in (11b) to get the state of uncertain dynamical system.

$$x[n] = x[n - 1] + (A_r x[n - 1] + B_r c[n] + B \Lambda [-\tilde{W}^T[n] + W^T] \sigma(x[n - 1])) \Delta t$$

In above difference equations, $\gamma, P, B, K, A_r, B_r, \Lambda, W$ are the same as those in continuous system which are defined in Section Inner Loop Formulations.

### 2.2.3 Boeing Airplane Model

To simulate the above formulations of MRAC system, a Boeing 747 airplane model is selected as the uncertain dynamical system whose parameters are acquired in the book of Bryson [50] in 1994. To simplify the calculation and experiment design, only the longitudinal motion is taken into consideration and human is requested to control the pitch angle alone.

Example 2-747 in [50], shows the longitudinal equation of motion for Boeing 747 at 40kft height and velocity Mach number 0.80 given by
\[ \dot{x} = A_p x(t) + B_p u(t) + W^T \sigma(x(t)), \quad x(0) = x_0 \]

where \( x(t) = [u, w, q, \theta]^T \). \( u \) and \( w \) represent respectively the velocity along \( x \) axis and \( z \) axis. \( q \) represents the angular velocity with respect to \( y \) axis in \( \text{rad/sec} \) and \( \theta \) is the pitch Euler angle of the body axis with respect to the reference axis in \( \text{rad} \) (1 \( \text{rad} = 0.01 \text{rad} \)). \( W \) is an unknown weighting matrix and \( \sigma(x(t)) = [1, u(t), w(t)]^T \) is a known basis function.

2.2.4 Numerical Data for simulation of MRAC part

For simulations, numerical data is borrowed from [49] and [50] and also summarized in Table 1.

| \( A_p \) | \[
\begin{pmatrix}
-0.003 & 0.039 & 0 & -0.322 \\
-0.065 & -0.319 & 7.74 & 0 \\
0.0201 & -0.101 & -0.429 & 0 \\
0 & 0 & 1 & 0
\end{pmatrix}
\] |
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>( B_p )</td>
<td>([0.01, -0.18, -1.16, 0]^T)</td>
</tr>
<tr>
<td>( W )</td>
<td>([0.1, 0.3, -0.3]^T)</td>
</tr>
<tr>
<td>( \Lambda )</td>
<td>1</td>
</tr>
<tr>
<td>( \gamma )</td>
<td>50</td>
</tr>
<tr>
<td>( E_p )</td>
<td>([0,0,0,1])</td>
</tr>
</tbody>
</table>

Table 1 Numerical data of the MRAC

The design of the \( K \) in the reference model will be discussed later in Section Design of Reference Model.

2.3 Human Dynamics

In the block diagram (Figure 1), human dynamics receives two signals, the reference
input and the output of the real-world plant with uncertainties.

2.3.1 Neal-Smith Model

In this thesis, Neal-Smith Model [40] which could simulate the behavior of a pilot in controlling pitch angle is selected as the human dynamics model in simulation and for design purpose. Despite possible inaccuracy of this model to describe a real pilot behavior, it is still useful as a design tool.

The Neal-Smith model is given as

\[ H(s) = k_p \frac{T_p s + 1}{T_z s + 1} e^{-\tau s} \]  \hspace{1cm} (20)

According to the definition provided by [33], the delay in pilot model is defined as \( \tau_e \) which is the effective delay, including transport delays and high frequency neuromuscular lags. In the study [35], the typical value of response delay for a skilled pilot ranges from 0.1 to 0.5s. Since the potential participants in experiments are untrained, delay is likely to be larger, so it is selected to be 0.5s which is the worst case in controlling. Parameters in this model are selected as shown in Table 2 for simulation purposes.

| \( k_p \) | 4.5 |
| \( T_p \) | 1 |
| \( T_z \) | 7.4 |
| \( \tau \) | 0.5 |

Table 2 Parameters for Neal-Smith pilot model

2.3.2 The Design of Real Human Interaction with MRAC System
Due to the purpose of allowing real human participants to control Human-in-the-loop model reference adaptive control system, the outer loop part should be designed for human to control the mathematical dynamic MRAC model.

As shown in Figure 1, reference input and real-time state (the pitch angle) of uncertain dynamical system are required. Commands from participants should be transmitted to the controller and reference model simultaneously to control the MRAC system. Take the control on a computer using a mouse as an example. Due to the limitation of device and the purpose to simplify the experiment, only the pitch angle is required to be controlled by participants. Thus, for the requirement of reference input, the reference should be presented on the screen as a figure or simply a number. The reference could be a constant number or a trajectory. For the feedback of the pitch angle, the signal should also be presented on the screen simultaneously. For the convenience of participants to observe the error, two sets of data are presented in one figure with different types of lines. The command from participants need to be transferred from analog signal (the movement of mouse) to digital signal with proper sampling time and a filter should be implemented to prevent the effect of high frequency oscillation from human hand in controlling a mouse. Because of the limited calculation speed and the demand to simulate in real time, the sampling time should be selected both enabling control and real-time simulation. Thus, the sampling time should be selected based on the calculation speed of computer which varies a lot. Because the filter is designed to attenuate the high frequency oscillation signals, a first order lowpass filter is selected as

\[
H_L(s) = \frac{\omega_c}{s + \omega_c} = \frac{1}{1 + \frac{s}{\omega_c}}
\]

in which \(\omega_c/(2\pi)\) represents the cutoff frequency. Because the simulation will be in discrete time, the low-pass filter must be in discrete form

\[
y[n] = y[n - 1] + \frac{(x[n - 1] - y[n - 1])\Delta t}{T}
\]
where $\Delta t$ is the sampling time and $T = \frac{1}{40} s$ and $\omega_c = 40 \text{rad}/s$ which means the cutoff frequency is about $6.366 \text{Hz}$.

### 2.4 Human-in-Loop MRAC system

In previous sections, human dynamics and MRAC system have been illustrated and designed except the reference model. Since the purpose of this thesis is to analyze human performance under different design of the MRAC system, in this section we implement the Neal-Smith model as human dynamics and simulate the effect of different design of $K$s in the reference model. In the comparison part, only human performance is focused. The main indexes for performance are the time required to reach a stable steady state and the overshoot. The magnitude of input command to MRAC system, the total energy consumption and the scale of comfort for passengers are neglected.

#### 2.4.1 Human Reaction Delay Effects

For the stability analysis of nonlinear systems, Cao & Frank presented an approach based on linear Takagi-Sugeno fuzzy models [51]. However, due to the special characteristic of MRAC system, [49] provided another approach in the stability analysis of the human-in-the-loop MRAC system. The human dynamical model is given as

$$
\begin{align*}
\dot{x}(t) &= A_h x(t) + B_h \theta(t - \tau), x(0) = x_0 \\
c(t) &= C_h x(t) + D_h \theta(t - \tau) \\
\theta(t) &\triangleq r(t) - E_h x(t)
\end{align*}
$$

where $x(t) \in \mathbb{R}^{n_x}$ is the internal human state vector, $A_h \in \mathbb{R}^{n_x \times n_x}$, $B_h \in \mathbb{R}^{n_x \times n_r}$, $C_h \in \mathbb{R}^{n_c \times n_x}$, $D_h \in \mathbb{R}^{n_c \times n_r}$, $c(t) \in \mathbb{R}^{n_c}$ is the command from human, $\theta(t) \in \mathbb{R}^{n_r}$, $r(t) \in \mathbb{R}^{n_c}$, $E_h \in \mathbb{R}^{n_r \times n}$ is used to select proper state. Because the pitch angle is the fourth term of $x(t)$, $E_h = [0,0,0,1,0]$.

According to the Theorem 3.1 in [49], \textit{"Consider the uncertain dynamical system (1) subjects to (2), the reference model given by $A_r \triangleq A - BK$, the feedback control law given}
by (9), (10), (15), and (16), and the human dynamics given by (23), (24), (25). Then, \( e(t) \in \mathcal{L}_\infty \) and \( \tilde{W}(t) \in \mathcal{L}_\infty \). If, in addition, there exist \( P \in \mathbb{R}_+(n+n_\xi) \times (n+n_\xi) \) and \( S \in \mathbb{R}_+(n+n_\xi) \times (n+n_\xi) \) such that the linear matrix inequality (LMI) given by \( F = \begin{bmatrix} P + P A_0 + S & PA_1 \\ A_1^T P & -S \end{bmatrix} < 0 \) holds, then \( x_r(t) \in \mathcal{L}_\infty \), \( \xi(t) \in \mathcal{L}_\infty \) and \( \lim_{t \to \infty} e(t) = 0 \)." where \( \tilde{W}(t) \triangleq \tilde{W}(t) - W \in \mathbb{R}^{(s+n) \times m} \). The above theorem requires fundamental stability limit (FSL) which could be equivalently written in the equality form (26).

\[
0 = A_0^T P + P A_0 + PA_1 S^{-1} A_1^T P + S + Q \tag{26}
\]

where

\[
A_0 \triangleq \begin{bmatrix} A_r & B_r C_h \\ 0_{n \times n} & A_h \end{bmatrix} \in \mathbb{R}^{(n+n_\xi) \times (n+n_\xi)} \tag{27}
\]

\[
A_1 \triangleq \begin{bmatrix} -B_r D_h E_h & 0_{n \times n_\xi} \\ -B_h E_h & 0_{n_\xi \times n_\xi} \end{bmatrix} \in \mathbb{R}^{(n+n_\xi) \times (n+n_\xi)} \tag{28}
\]

and \( P \in \mathbb{R}_+(n+n_\xi) \times (n+n_\xi) \cap S(n+n_\xi) \times (n+n_\xi) \), \( S \in \mathbb{R}_+(n+n_\xi) \times (n+n_\xi) \cap S(n+n_\xi) \times (n+n_\xi) \), \( Q \in \mathbb{R}_+(n+n_\xi) \times (n+n_\xi) \cap S(n+n_\xi) \times (n+n_\xi) \).

In conclusion, if the designed reference model meets the LMI condition \( F < 0 \) then the output of the nonlinear closed-loop system with uncertainties and model reference adaptive controllers are guaranteed to be stable. The main idea behind this approach is to divide the problem into two parts; in part (i), one designs a standard MRAC for a plant with uncertainties and nonlinearities but without delays, and in part (ii), the problem is to stabilize a linear dynamics with human reaction delay where this linear dynamics is a combination of human linear model and linear reference model. In the rest of this chapter, we focus on the more challenging part (ii). This part whose block diagram is shown in Figure 2 is challenging mainly due to the presence of human reaction delay.
Stability analysis of the linear problem, i.e., part (ii), defined with (27), and (28) can further be separated into two parts as discussed in [49], where the linear system is either delay-independent or delay-dependent stable.

It is shown in [49] that for single-input single-output systems (reference scalar input to the human and a scalar single output of the uncertain plant) the condition $k_p < 1$ guarantees delay independent stability of part (ii).

In delay-dependent stability analysis of part (ii), the rightmost pole (RMP) of the system is to be acquired from the characteristic equation

$$\det(sI - (A_0 + A_t e^{-\tau})) = 0$$

To analyze the stability and the behavior of RMPs when $\tau$ changes, TRACE-DDE [30] is used. According to the instructions of Matlab package TRACE-DDE, the system of $m$ linear delay differential equations (DDE) with multiple discrete and distributed delays are given as:

$$y'(t) = L_0 y(t) + \sum_{i=0}^{k} L_i y(t - \tau_i) + \int_{-\tau}^{0} M(\theta)y(t + \theta)d\theta, t \geq 0$$

(29)

where $L_0, L_1, \ldots, L_k \in \mathbb{C}^{m \times m}$, $0 = \tau_0 < \tau_1 < \cdots < \tau_k = \tau$, and $M: [-\tau, 0] \rightarrow \mathbb{C}^{m \times m}$ is a piecewise smooth function.

For convenience, filter is neglected in RMP calculation. Based on equations (18), (23), (24), (25), delay differential equations are given as:
\[ \dot{x}_r = A_r x_r(t) + B_r C_h \dot{\xi}(t) - B_r D_h E_h x_r(t - \tau) + B_r D_h r(t - \tau) \quad (30) \]
\[ \dot{\xi}(t) = A_h \dot{\xi}(t) - B_h E_h x_r(t - \tau) + B_h r(t - \tau) \quad (31) \]

Thus, assume new state \( y = [x_r^T, \xi^T]^T \in \mathbb{R}^{n+n_\xi} \) and neglect terms with \( r(t - \tau) \).

\[ y'(t) = L_0 y(t) + L_1 y(t - \tau) \quad (32) \]

where,

\[
L_0 = \begin{bmatrix}
A_r & B_r C_h \\
0_{n_\xi \times n} & A_h
\end{bmatrix} \in \mathbb{R}^{(n+n_\xi) \times (n+n_\xi)}
\]
\[
L_1 = \begin{bmatrix}
-B_r D_h E_h & 0_{n \times n_\xi} \\
-B_h E_h & 0_{n_\xi \times n_\xi}
\end{bmatrix} \in \mathbb{R}^{(n+n_\xi) \times (n+n_\xi)}
\]

2.4.2 Design of Reference Model

In [49], LQR is used to find a \( K \) to ensure that \( A_r = A - BK \) is a Hurwitz Matrix for convenient purpose. To modify the \( K \) and compare the effects of different \( K \)'s on human performance, three methods are implemented to derive the \( K \) in reference model.

2.4.3 Classical Linear-Quadratic Regulator

In this section, LQR method is implemented to design the reference model. The quadratic cost function is given as:

\[ J(u) = \int_0^\infty (x^T Q x + u^T R u) dt \quad (33) \]

where \( u = -K x \), \( Q = \text{diag}([0,0,0,Q_1,Q_2]) \) shows the desire to control the pitch angle precisely and smoothly, \( R \) represents the level of intention to lower the total control effort. The \( K \) is designed to minimize the quadratic cost function with different values of weighting factors. In Yucelen’s design, \( Q_1 = 1, Q_2 = 2.5, \) and \( R = 1 \). In this method, the \( K \) in this reference model is designed by the same \( Q \) in [49] and \( R \) will be changed to show different level of attention on the control effort. Ideally, the bigger the \( R \) selects, the slower the pitch angle will approach the reference input.
Four cases are selected and simulated. The RMP calculated by TRACE-DDE with equation (32) are in Table 3. Here, RMDP refers to the rightmost dominant pole, and RMNP refers to the "second" right most dominant pole (denoted as "rightmost non-dominant pole" to avoid confusion).

<table>
<thead>
<tr>
<th>$R$</th>
<th>RMDP</th>
<th>RMNP</th>
<th>Figure for 6 RMPs</th>
<th>Stability</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.1</td>
<td>-0.0117</td>
<td>-0.2943</td>
<td><img src="image1" alt="Diagram" /></td>
<td>Stable</td>
</tr>
<tr>
<td>1</td>
<td>-0.0117</td>
<td>-0.1907</td>
<td><img src="image2" alt="Diagram" /></td>
<td>Stable</td>
</tr>
</tbody>
</table>
Table 3 Stability analysis using TRACE-DDE for cases designed by classical LQR of reference model and delay in human model

In Table 3, RMDP and RMNP represents the real part of rightmost dominant pole and the real part of rightmost non-dominant pole respectively. Despite the $R$ changes from 0.1 to 100, the rightmost root does not move and the rightmost non-dominant pole moves from $-0.2943$ to $-0.1064$. The reason of the immobilization of RMDP is mainly the specification of the plant chosen in this system (Boeing 747 Airplane Model).

To see how the $K$'s derived by LQR works in the human-in-loop model reference adaptive control system, system in the block diagram in Figure 1 with Neal-Smith model as the human dynamics is simulated in Matlab.
Figure 3 Simulation results with different $R$, $\theta$ represents the pitch angle and $u$ is the command from Neal-Smith pilot model to MRAC system.

As is shown in Figure 3, although the dominant poles are same in four cases, the dynamic response to a constant reference input varies a lot. As is predicted above, the higher $R$ is selected, the more limitation on control efforts are given and the slower the state of plane will be changed. As a result, for $R = 0.1$, overshoot of the system is the smallest (1.044) and it takes much shorter time (13.15s) than other cases to stabilize the system. Additionally, the command signal from pilot model is much smaller than other cases, which means for the same settings pilots need to move the Yoke less (the control wheel on airplane). However, a larger $R$ contributes to smoother trajectory, less power required and in compensation, pilot might feel harder to change the state of airplane.

Comparing the time to reach steady state and overshoot, the case when $R = 0.1$ is the best design of reference model in four cases of classical LQR method, which represents the
circumstance that the energy consumption is the least emphasized. The result shows that in designing reference model using LQR, the more energy consumed, the better performance the system will attain with the Neal-Smith model.

2.4.4 Modified Linear-Quadratic Regulator

Because the classical LQR method cannot determine the RMDP, a modified linear-quadratic regulator is implemented to design $K$. The new quadratic cost function is given as

$$J(u) = \int_0^\infty e^{2\alpha t} (x^T Q x + u^T R u) dt$$

(34)

This method could shift the imaginary axis by $\alpha$ and guarantee that all the poles are placed on the left side of the line $Re(s) = -\alpha$. To implement (34) in Matlab using lqr command, variables need to be changed as follows

$$\hat{x} = xe^{\alpha t}$$

(35)

$$\hat{u} = ue^{\alpha t}$$

(36)

$$\hat{x} = [A + I\alpha]\hat{x} + B\hat{u}$$

(37)

In this section, $R = 1$ and $Q = diag([0,0,0,1,2.5])$. Because in classical LQR design, real part of dominant pole stays at $-0.0117$, $\alpha$ is selected as 0.012 or more. Due to the presence of delay, for a given $\alpha$, the real part of the rightmost pole shifted smaller than $\alpha$. And the effects of time delay have to be calculated to see the position of poles.

Four cases are selected and simulated. The RMP calculated by TRACE-DDE with equation (32) are in Table 4.
<table>
<thead>
<tr>
<th>α</th>
<th>RMDP</th>
<th>RMNP</th>
<th>Figure for 6 RMPs</th>
<th>Stability</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.012</td>
<td>-0.0118</td>
<td>-0.1741</td>
<td><img src="image1.png" alt="Graph" /></td>
<td>Stable</td>
</tr>
<tr>
<td>0.013</td>
<td>-0.0121</td>
<td>-0.1147</td>
<td><img src="image2.png" alt="Graph" /></td>
<td>Stable</td>
</tr>
<tr>
<td>0.014</td>
<td>-0.0123</td>
<td>-0.0642</td>
<td><img src="image3.png" alt="Graph" /></td>
<td>Stable</td>
</tr>
</tbody>
</table>
Comparing with the classical LQR method, the modified LQR method could move the RMDP as designed. But when the human model with delay is considered, we find that the RMNP are shifted closer to the imaginary axis, which will bring negative effects on the performance of the whole system. The increase in the imaginary part of poles would contribute to more oscillation. In the case of $\alpha = 0.016$, the system becomes unstable.

To see how the selection of $\alpha$ changes the system response, the designed $K$ (unstable cases are excluded) derived by modified LQR are implemented in time simulations in Matlab.
According to the simulation results in Figure 4, the system response when $\alpha = 0.012$ has the least overshoot and takes the shortest time to reach steady state. And for the case with $\alpha = 0.015$, the simulation result shows that the system oscillates a lot and hard to stabilize when coupled with the human model although it has alone the best design of dominant pole. Comparing the stability analysis in Table 4, the displacement of dominant pole to left should have resulted in a better performance with faster speed to reach steady state. However, the worse performance with $\alpha = 0.015$ shows that the small left shift of RMDP cannot improve the system response when coupled with the Neal-Smith pilot model. Because other non-dominant poles do not shift a lot in changing the $\alpha$, and the imaginary part of RMNP are almost the same, the right shift of RMNP may contribute to the result.

In the comparison of time to reach steady state and overshoot, $\alpha = 0.012$ turned out to be the best design of MRAC reference model using modified LQR method. The result turns
out that the increase in $\alpha$ contributes very little to the left shift of RMDP and causes the right shift of RMNP which will jeopardize the performance of system.

2.4.5 Pole Placement

For further investigating the effects of the design of $K$ in system response, pole placement method is implemented in the feedback loop of the reference model. The advantage of this method is the possibility to change every pole freely and directly. However, this method may not be practical for large scale design problems due to the presence of too many parameters to be tuned.

Since this thesis aims to find the relationship between the design of the reference model and human performance, we are not concerned here with finding an optimal method to design an MRAC system. Energy consumption and other factors are neglected in the design and the focus is mainly to understand how humans interact with MRAC with different reference models. In the following, pole placement method is pursued to design the reference model considering the above discussions.

In the design and simulation results of classical LQR and modified LQR, the dominant pole is hard to shift and thus the non-dominant poles contribute to the differences. Because the transfer function with delay consists of a quasi-polynomial, but not a polynomial, it is difficult to design the poles without delay and predict the poles with nonzero delay. Thus, poles selection follows certain criteria in time domain.

(i) Different reference models should share the same dominant pole
(ii) Different reference models should have similar time domain step response, for example, the same peak value. But some specification in time domain should be different for comparison, for example, different length of time required to reach the peak

Because the reference model is a fifth-order system, let:
\[ P = [p_1, p_2, p_3, p_4, p_5] \]

\[ K = [k_1, k_2, k_3, k_4, k_5] \]

\[ A_r = A - BK \]

Implementing the parameters in Table 1 into equations (5) and (6), poles of the transfer function which are roots of the characteristic equation are obtained from

\[ |sI - A_r| = 0 \tag{38} \]

where \( I \) is a 5 \times 5 identity matrix. According to the definition of pole placement method, the roots \( s \) of (38) are from the pole set \( P \). Thus, the characteristic equation should be

\[ (s + p_1)(s + p_2)(s + p_3)(s + p_4)(s + p_5) = 0 \tag{39} \]

Because both (38) and (39) are characteristic equations of the same system,

\[ |sI - A_r| = (s + p_1)(s + p_2)(s + p_3)(s + p_4)(s + p_5) \tag{40} \]

from which the relationship between \( k_1 \ldots k_5 \) and \( p_1 \ldots p_5 \) could be derived. Take \( P_1 \ldots P_5 \) as variables in the equation, then the transfer function could be derived from (40) as

\[ H(s) = C(sI - A)^{-1}B + D \tag{41} \]

\[ TF(s) = \frac{-289.4563 \cdot p_1 \cdot p_2 \cdot p_3 \cdot p_4 \cdot p_5 \cdot (s + 0.2944) \cdot (s + 0.0117)}{(s + p_1) \cdot (s + p_2) \cdot (s + p_3) \cdot (s + p_4) \cdot (s + p_5)} \tag{42} \]

To simplify the calculation of step response to shorten the time required for searching for ideal elements of \( P \), the time domain response of the transfer function (42) is required. Using the Inverse Laplace, time domain step response is acquired.
\[ y(t) = p_1 e^{-p_1 t} \frac{1.995 \cdot 10^{21} p_1^2 p_2 p_3 p_4 p_5 - 6.107 \cdot 10^{20} p_1 p_2 p_3 p_4 p_5 + 6.872 \cdot 10^{18} p_2 p_3 p_4 p_5}{(6.872 \cdot 10^{18} p_1 - 6.872 \cdot 10^{18} p_5)(p_1 - p_2)(p_1 - p_3)(p_1 - p_4)} \]
\[ - p_2 e^{-p_2 t} \frac{1.995 \cdot 10^{21} p_1^2 p_2 p_3 p_4 p_5 - 6.107 \cdot 10^{20} p_1 p_2 p_3 p_4 p_5 + 6.872 \cdot 10^{18} p_1 p_3 p_4 p_5}{(6.872 \cdot 10^{18} p_2 - 6.872 \cdot 10^{18} p_5)(p_1 - p_2)(p_2 - p_3)(p_2 - p_4)} \]
\[ + p_3 e^{-p_3 t} \frac{1.995 \cdot 10^{21} p_1^2 p_2 p_3^2 p_4 p_5 - 6.107 \cdot 10^{20} p_1 p_2 p_3 p_4 p_5 + 6.872 \cdot 10^{18} p_1 p_2 p_4 p_5}{(6.872 \cdot 10^{18} p_3 - 6.872 \cdot 10^{18} p_5)(p_1 - p_3)(p_2 - p_3)(p_3 - p_4)} \]
\[ - p_4 e^{-p_4 t} \frac{1.995 \cdot 10^{21} p_1 p_2 p_3^3 p_4 p_5^2 - 6.107 \cdot 10^{20} p_1 p_2 p_3 p_4 p_5 + 6.872 \cdot 10^{18} p_1 p_2 p_3 p_4}{(6.872 \cdot 10^{18} p_4 - 6.872 \cdot 10^{18} p_5)(p_1 - p_4)(p_2 - p_4)(p_3 - p_4)} \]
\[ + p_5 e^{-p_5 t} \frac{1.995 \cdot 10^{21} p_1 p_2^2 p_3^4 p_4 p_5^3 - 6.107 \cdot 10^{20} p_1 p_2 p_3 p_4 p_5 + 6.872 \cdot 10^{18} p_1 p_2 p_3 p_4}{(6.872 \cdot 10^{18} p_5 - 6.872 \cdot 10^{18} p_5)(p_1 - p_5)(p_2 - p_5)(p_3 - p_5)} \]

In the above equation, \( p_1 \ldots p_5 \) is the same notation as \( p_1 \ldots p_5 \).

The dominant pole is selected as \(-0.02\) and the peak of the response as \(1.7\). Due to the acquirement of time domain step response, which saves time in simulation, the search within a limited 4-dimensional space becomes possible. After searching, three sets of poles are selected. Because in equation (39) and (40), poles satisfy \( s + p_1 = 0 \), the set of \( p_1 \ldots p_5 \) are positive.

\[ P_1 = [0.02, 0.31, 2.2, 2.2, 2.3], \; P_2 = [0.02, 0.35, 0.9, 1.3, 1.7], \; P_3 = [0.02, 0.5, 0.6, 0.7, 0.8]. \]

![Figure 5 Zoomed-in time domain step response for pole placement method](image)

According to the zoomed-in figure in Figure 5, three sets of poles have similar time domain behavior and share the same peak 1.7. Time required to reach the peak is 4.004s,
6.014s, and 8.176s respectively. The steady state value and time to reach steady state are the same; steady state value is determined by $TF(s = 0)$ (see (42)), which is independent of $p_1 \ldots p_5$. Settling time is determined by RMDP.

Stability in the presence of delay is assessed by the position of poles of the linear system (Neal-Smith Model combined with reference model) using TRACE-DDE.

<table>
<thead>
<tr>
<th>Set</th>
<th>RMDP</th>
<th>RMNP</th>
<th>Figure for 6 RMPs</th>
<th>Stability</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>-0.0126</td>
<td>-0.2139</td>
<td><img src="image1.png" alt="Diagram 1" /></td>
<td>Stable</td>
</tr>
<tr>
<td>2</td>
<td>-0.0126</td>
<td>-0.0817</td>
<td><img src="image2.png" alt="Diagram 2" /></td>
<td>Stable</td>
</tr>
</tbody>
</table>
According to Table 5, the dominant pole with delay are the same for pole set 1 and 2. For pole set 3, the reference model becomes unstable for a $0.5s$ delay which we consider as the worst case for human dynamics. If the delay decreases to $0.44s$ which is still large than expected from a human, the system becomes stable for pole set 3.
Simulations of the human-in-the-loop MRAC considering three sets of poles for the reference model are given in Figure 6. As what is predicted, simulation with set 3 poles is unstable. Set 1 leads to lower overshoot and faster reach to steady state than with Set 2. It is mainly because Set 1 and Set 2 share the same value of peak but Set 1 leads to faster reach to the peak than Set 2. That is, non-dominant poles of the reference model influence the outcome.

2.4.6 Results

Simulations show the Neal-Smith pilot model response behavior with three different reference models in the MRAC system. The classical LQR shows the compensation between reference model response and energy consumption. The modified LQR shows the effects of rightmost non-dominant poles. And the pole placement method is intended to find the relationship between rightmost non-dominant poles and system performance. Using the
pole placement method, three sets of poles are designed such that they share the same peak value and steady state specifications.

However, the Neal-Smith model is a model useful in the design of human-in-the-loop systems where the human is a skilled pilot. This may not be the same with untrained human subjects. Thus, experiments with real humans are required to distinguish what types of designs contribute to better performance of human.

Chapter 3: Experiment of Real Human with MRAC system

In previous sections, three methods are implemented to derive a proper $K$ for the reference model. However, since it is a challenge to find the relationship between human performance and various design of MRAC system, instead of an optimal design between performance and energy consumption, the position of poles and the time domain step responses are two main considerations. The classical and modified LQR are not preferred since it is not straightforward to define the positions of poles. The MRAC system used in experiments with real human participants will be based on the three cases defined by pole placement method.

\[
P_1 = [0.02, 0.31, 2.2, 2.3], \quad P_2 = [0.02, 0.35, 0.9, 1.3, 1.7], \quad P_3 = [0.02, 0.5, 0.6, 0.7, 0.8].
\]

3.1 Setup of Experiments

The experiment of real human with MRAC system mentioned below has been approved by Institutional Review Board (IRB) with IRB#: 18-04-05 and Northeastern University generously provided the compensation of recruiting participants.

3.1.1 Recruitment Procedure

In the paper [52] which is based on the United Kingdom Health and Lifestyle Survey for 7130 participants reported by F.A. Huppert in 1987 [53], it shows that human reaction time
(RT) is strongly associated with age. The result suggests that the trajectory of reaction time versus age is relatively flat between the initial decline in 20s and the start of increase around 60. Thus, participants’ ages are selected between 21 to 30 to minimize the differences in reaction time from age to age. Additionally, gender may have effects on reaction time according to [52]. But the difference in gender is relatively small and negligible in simple RT experiments.

Furthermore, [54] suggests that participants’ circadian rhythms and homeostasis have effects on reaction time. [55] shows the RT will be influenced also by caffeine and coffee consumption. In conclusion, human reaction could be influenced by many parameters, including age group, gender, health conditions, caffeine consumption, experience with computer games and how well an individual is trained for the tasks he/she will perform within a study.

In order to carefully record the reaction of human and study it as part of this research, it is extremely important for us to reduce the number of critical parameters that play a role on how human reactions are affected. Due to this reason, we minimized/eliminated the influence of some of the parameters, which are related to age group, health conditions and literacy level.

Although the planned experiments require minimal physical activity (see below), we expect the human subjects to be in good condition of health so that, across subjects, the human reaction will be minimally affected by health conditions. Moreover, for research personnel to clearly explain the experiments, for subjects to be able to thoroughly comprehend the consent form and understand the instructions of study personnel, the subjects will need to have sufficient literacy level in English language.

We recruited 14 participants in total (11 males and 3 females). Since the human subjects would be required to practice and learn the computer game, we recruited subjects who do
not encounter health issues due to playing computer games and who do not have learning disabilities as self-reported on pre-experiment survey.

3.1.2 Human Reaction Time Test

The recruitment procedure cannot eliminate all the influencing parameters in reaction time. Thus, testing the human reaction time is necessary for later analyzing the data and the efficacy of our control algorithms in relation to human reaction times. Participants’ reaction time are measured by the program on website Human Benchmark (https://www.humanbenchmark.com/tests/reactiontime) where the participants will need to click the mouse button in response to certain color changes on the screen. The test is carried on five times and the mean value of the five recordings are taken as the human reaction delay of the participant.

3.1.3 Cursor Control Game

According to Figure 1, the block diagram of human-in-loop model reference adaptive control system, human dynamics block needs to receive two signals and produce one output command signal. In this experiment, system is simulated in Matlab and computer screen and mouse are taken as output and input device separately. The main structure of this Matlab code was developed by Prof. Tansel Yucelen (University of South Florida). Modifications on this code include replacing LQR by pole placement, designing a GUI for human to control in real time, running games in a random order and saving required data automatically.

In order to simplify the control procedure for participants, both the reference input and uncertain dynamical feedback signals are plotted in the same figure. The command input should be derived from the movement of mouse. Thus, a graphical user interface (GUI) is designed as Figure 7.
Figure 7 Graphical User Interface of Human Experiment

The green line is given as reference input and the red line is given as the reference point in time, after which the human can move the blue cursor by scrolling the slider up and down. To eliminate the time for human to move the mouse and hold the slider due to the sudden occurrence of the game window, three seconds are given to participants for preparation. To notify the end of preparation time, a red vertical line is given and a red dot will occur at 3s. During the experiment for the first four participants, we noticed that after playing for several times, participants tend to move the slider during the three-second preparation time and, as a result, initial input becomes different and non-zero for different trials. Thus, to prevent this, for the first three seconds, mouse is frozen on screen to ensure initial conditions for all sets of games are the same. For Subject No.1 to No.4, the mouse is not frozen and for the rest ten participants, the mouse is frozen during the three-second preparation procedure. In this thesis, experiments with unfrozen initial conditions are considered separately and are not the main purpose of analysis.
Size of entire experiment window is set to be 1200 pixels by 800 pixels and the slider is set to be 30 pixels by 685 pixels. The range of command input varies from +5 (Top) to −5 (Bottom). The design of the slider is based on the mouse with 1000 dots per inch (dpi) and the mouse speed is set to be 7/11 in Windows. Under mentioned settings, the movement of mouse will not be in a very small range which would otherwise lead to the consequence that mouse is merely controlled by participants’ finger nor in a wide range such that participants feel hard to do reaction.

3.2 Experimental Procedure

3.2.1 Preparation

A participant is seated on an adjustable chair in front of a desk with a laptop. Participants sit comfortably in front of the laptop. An input device (mouse) is used to control the blue cursor. A pre and post game assessment of motion dizziness, fatigue and slight short-term discomfort is conducted by using a Simulation Sickness Questionnaire.

3.2.2 Activities

Before the game starts, we measure participants’ reaction times using the website (https://www.humanbenchmark.com/tests/reactiontime) where the participants will need to click the mouse button in respond to certain color changes on the screen. This information will be useful when later analyzing the data and the efficacy of our control algorithms in relation to human reaction times. Study personnel plays the game once for participants to help them understand how the game works.

After signing the consent form, the participant plays next the Cursor Control Game displayed on laptop screen. The sequence of game settings is randomized. The game requires the participants to use the mouse to control the cursor on the screen and to get it as close as possible and as fast as possible to the reference cursor, which remains constant throughout the experiment.
3.2.3 Durations, Number of Sessions, Breaks

The subjects play the Reaction Time Game for about 30 seconds once at the beginning of the experiments and the Cursor Control Game will be demonstrated once for 20 seconds. After that, subjects will play the Cursor Control Game under three conditions corresponding to one set of games, and each set will be repeated five times in a randomized order of conditions (total of 15 games). Each condition of game will take 20 seconds and 5 seconds for rest in between the games within a set. Between each set, a 1 minute rest will be given to the participants. Estimated duration of completing all the games is about 12 minutes.

The total time required for the whole process including preparation will be about 30 minutes.

Chapter 4: Analysis of Experiment Data

4.1 Human Reaction Delay Measurement

Using the program on website Human Benchmark (https://www.humanbenchmark.com/tests/reactiontime) as mentioned above, human reaction time are measured and recorded. Following figure illustrates the average reaction delay time for individuals during the human reaction delay test. Error bar represents the standard deviation of collected data. Due to the absence of participants No.2 and No.10, subject numbers are not successive.
4.2 Performance Index

Using gathered data, plots of average system state output and shaded error bar of standard deviation are plotted. Due to the large number of figures required for three different pole sets and frozen or unfrozen conditions, only the results of pole set one are plotted for all five trials.
Figure 9 Average and error bar of standard deviation for system output and human command input over time

One note here is that for participants with mouse unfrozen, initial condition for human command input is not zero. In analyzing gathered data in experiment, subjects with mouse frozen and subjects with mouse unfrozen are to be plotted separately to see the difference between two groups.
To compare human behaviors and system performance between different designs of MRAC system based on three set of poles, performance indexes are necessary to quantify performance. For confidentiality, Figure 9 is an example result played by researchers for better illustration.

![Example from Researcher](image)

**Figure 10 Illustration of Basic Metrics**

In the above figure, black solid line represents the 3 second line, before which actions are not permitted, and the black line of dashes represents the division of half process, which is used in below for ‘steady state’ analysis. Blue line represents the system state feedback and green line is the reference input which could both be observed by participants during the game. Red line records the filtered command from human.

### 4.2.1 Mean Integral Squared Error

To quantify the behavior of human, difference between state of system (blue line in Figure 9) and reference input (green line in Figure 9), which is error are selected. The metric of mean integral squared error (MISE) could be calculated by the following equation
\[ MISE = \frac{\Sigma(y - r)^2}{n} = E[(y - r)^2] \]  

(43)

in which \( y \) is the state of system, \( r \) is the reference input, and \( n \) represents the total number of samples of \( y \). This metric could generally illustrate human performance in tracking tasks. The higher the MISE is, the more error exists in the process, which means the worse human performance is.

As what is observed during the experiment, both the human reaction and system performance in the first ‘approaching’ part is distinctly different from those in the second ‘stable’ steady part. Based on this observation, MISE are calculated both in the whole process (from 3 s to 20 s) for generalized index of human performance and in the second half process (from 11.5 s to 20 s) for ‘steady state’ analysis.

![All subjects in whole process](image1)

![All subjects in second half process](image2)
Figure 11 MISE average and error bar for pole sets (figures are in different range for visualization purpose)

In Figure 11, all subjects (first row) correspond to subject 1 to 16 except the absence of No. 2 and No. 10 which is plotted for the comparison between frozen initial condition and unfrozen initial condition. It seems that the performance with frozen and unfrozen settings are similar in whole process with lower average MISE. This is mainly because of the permission
to move the mouse during the first three seconds, which helps them move faster at start. However, the index of second half process are slightly different between frozen and unfrozen conditions. For subjects with mouse unfrozen, pole set 1 seems to be the best with the least MISE and small standard deviation, but in general results using data from all subjects, the second pole set seems better. This cannot be explained by the reason of frozen or unfrozen initial conditions because the initial condition will not have strong influence in the second half of the experiments. The explanation of this slight difference might be the lack of subjects for experiment with mouse unfrozen and the various character of participants. Due to the difference between frozen and unfrozen conditions in whole process are ignorable, in below comparisons, data gathered in all conditions are considered.

As what is expected, the MISE shows that, in the second half, human performances are much better than the whole process performance. This is mainly because in the first half, subjects are trying to stabilize the system in a relatively smaller range instead of moving the cursor close to the reference from zero. Also, participants’ learning might contribute to their better performance in the second half process.

Apart from the comparison between pole sets, in an individual set, all participants’ training process in MISE are plotted.

![Whole process for pole set 1](image1.png) ![Second half process for pole set 1](image2.png)
In the above figure, average error and standard deviation are plotted in test sequence. X axis represents the trial number for participants and there seems to be a huge increase in human performance after they have their first trial. In pole set 2, the training seems to be relatively small in whole process because participants performed well at their first trial.
4.2.2 Variance of Human Input

To define the human effort, filtered human command input are selected. The following metric is defined for this purpose:

$$\sigma_h^2 = \frac{\sum (c - \bar{c})^2}{n} = E[(c - E[c])^2]$$ (44)

in which $c$ is the filtered human control input, $\bar{c} = E[c]$ is the average of $c$, and $n$ represents the total number of samples of $c$. Because in the game, participants are invited to control the slider by mouse, there is no force feedback. As a result, human effort could be defined by the above variance in equation (44). For better comparison, $\sigma_h^2$ is calculated both in whole process and in the second half process.

![Graph 1](image1.png)

All subjects in whole process

![Graph 2](image2.png)

All subjects in second half process
The above figures show participants’ effort to control the cursor. For pole set one and two, humans take slightly less effort in condition one than that in condition two. The second half process represents energy put in controlling the cursor to be stable on reference input.

Figure 14 is plotted to show the decreasing human control effort.
Figure 14 Average variance of human input and error bar for test sequences (figures are in different range for visualization purpose)
Changes in human control effort show similar patterns as that in Figure 12. But for human effort, the decrease does not fit the curve in Figure 12 along with the training. Especially for set 3, participants struggled in the third trial and put much more effort than that in the second trial.

4.2.3 Duration of States within the State of Range

The total length of time in which system state stays in a certain range of reference input are measured to show the performance in tracking. The longer the state stays in range, the better the tracking performance is. The range is selected to be ±5%, which, according to the design of GUI in Figure 7, equals to the range of 8 pixels on screen.

```
All subjects
```
Due to the reason illustrated in section 4.2.1, three different pole sets are compared based on results from all subjects. For pole set three, the system state stays in ±5% for about 2 seconds, which is less than 3 seconds from pole set two and 3.5 seconds from pole set one. It seems that pole set one have the best performance. However, the small length of time is not consistent. This is mainly because of the relatively small range selected here. For participants, it is hard to recognize four pixels on each side and still have motivation in continuing their control. Considering this fact, the range is enlarged to ±20% which is 32 pixels on screen. New figures are as below.
Based on the above figure, pole set three is the worst one in steady state tracking. The rest of two pole sets shares relatively close length of time when the system state stays in the ±20% range of the reference.

### 4.2.4 Number of Time Human Input Changes Direction

Human effort could also be analyzed by the number of times human input changes the direction by the direction of mouse movement. To eliminate the undesired oscillation of hand,
the minimum peak prominence is selected to be 0.1 and minimum peak distance is 20 samples, which is 0.1 second.

![Bar charts showing the average number of times human input changes direction for different pole sets and subject conditions.](image)

Figure 17 Average number of times human input changes direction (figures are in different range for visualization purpose)

For pole set one, participants change the direction of mouse for about 10 times, which is more than pole set two and three. This is possibly because of the fastest response of pole set one among others.
4.2.5 Number of Time System Output Changes Direction

In different design of systems, the ‘Speed’ of system varies a lot. To illustrate how many chances for human to realize and control the system, the number of system direction change is counted. The change of direction should have at least 0.1 peak prominence and 0.1 second interval between.

Due to the various system response speed, system direction change times are counted.

Figure 18 Average number of times system output changes direction (figures are in different range for visualization purpose)
Number of times system output changes direction shares the same pattern as with the number of times system human input changes direction. And with less system direction change times, the system responds slower and, as a result, participants have less chance to change their control input, which might cause more effort and poor performance for pole set 3. Additionally, the less number of times system output changes direction does not mean the system is stabilized and do not need to move. Based on observation, the reason for less number of times is the limitation of time. On the average sense, it appears that human input direction changes more than the system output direction changes. This is mainly because the shift in the direction of mouse movement will not always result in the system output direction change.

4.3 Analysis of Performance versus Control Effort

According to the above figures, in human behavior performance, pole set three is the worst and the result of MISE of set two is slightly better than set one. The comparison on human effort shows that in playing the game with pole set three, participants tend to devote more effort in controlling and results in worse performance. The main difference between three different pole sets are the second right most root and the peak time in step response in the reference model.

For better analyzing the result, a threshold is determined to count all the cases in which performance is better than the threshold index. In plotting the number of MISE, because the smaller MISE is, the better performance it represents, inverse of MISE is selected for better visualization. Higher inverse of MISE represents better performance of human.
In Figure 19, x axis represents the threshold and y axis is the number of cases within the range defined by the threshold. For inverse MISE, cases in which inverse MISE are higher than the threshold are counted. In other words, cases with better performance than the threshold index are counted. For $\sigma_h^2$, case is counted when its $\sigma_h^2$ are lower than the threshold which means less human control effort are made in this case. Thus, in Figure 19, number of cases with better results, either better human performance or less human control effort, are counted and plotted.
Based on the above figures, it could be proved that pole set three is the worst design among three different sets. In inverse MISE, set two have better results than set one in lower threshold values. But set one performs better after the threshold increase over 60 for whole process. Generally speaking, participants in set one put in less effort and achieve better inverse MISE result, which means closer to the reference input during the process. And conclusion could be made that pole set one is the best design among three different design using pole placement method both in human effort and human behavior.

Chapter 5: Human Dynamics System Identification

5.1 Settings of System Identification

The figure above represents the input and output for human dynamics. As is shown in the Figure 20, for a real human, reference and system feedback are input to the system. Based on the observation of two inputs, human will make decision and give command to the system as command output with reaction delay. Because the collected data in Cursor Control Game include reference input, system feedback input, and command output, which are input
and output for human dynamics, it is possible for us to analyze the behavior of a real human to see if the linear part of human performance matches a certain transfer function with delay. In this section, System Identification Toolbox in Matlab is being used based on the user’s guide book from Matlab [56].

In System Identification Toolbox, input is the system state feedback from 3 second to the end, output of the block is the human control command, delays are set to be the average latency for individuals in Figure 8, and two poles and two zeros system are selected. Due to the different design of the Cursor Control Game, initial condition is set to be zero for those ten participants with mouse frozen and first four participants with mouse unfrozen are not taken into consideration in system identification.

5.2 Comparison Between Real Human and Identified System

Using System Identification Toolbox, the transfer function for an individual case could be derived based on data collected in the cursor control game. To compare the difference between the behavior of real human and identified system, transfer functions of the identified systems are implemented in the Cursor Control Game to substitute the position of a real human.

Based on the FitPercent provided by the System Identification Toolbox, which is defined as

\[
FitPercent = 100 \left( 1 - \frac{\| y_{measured} - y_{model} \|}{\| y_{measured} - y_{measured} \|} \right)
\]

where \( y_{measured} \) is the measured output data, \( y_{model} \) is the simulated or predicted response of the model, the best fit case for each pole set is taken as an example.

To maintain the confidentiality of data, only the difference between system state and simulation results are plotted for individuals. In simulating control command input, the input to identified system is the step reference input and state feedback recorded in the experiment. In simulating system output, the identified system functions as human dynamics block in
For the pole set one, the best fit result is from the participant No. 6 in the second trial, for which we obtain

\[ TF(s) = e^{-0.355s} \cdot \frac{0.3356s^2 + 0.5247s + 0.5936}{s^2 + 2.391s + 0.04738} \]

\[ \text{FitPercent} = 57.9159\% \]

For the pole set two, the best fit result is from participant No. 6 in the first trial, for which we have:

\[ TF(s) = e^{-0.355s} \cdot \frac{0.4637s^2 + 0.3301s + 0.1299}{s^2 + 0.08924s + 0.002732} \]

\[ \text{FitPercent} = 70.1110\% \]

For the pole set three, the best fit result is from participant No. 6 in the second trial, for which we find:

\[ TF(s) = e^{-0.355s} \cdot \frac{0.3883s^2 + 0.2396s + 0.05864}{s^2 + 0.03761s + 0.003296} \]

\[ \text{FitPercent} = 77.5464\% \]
To help visualization, y-axis for figures in the first column is the scaled human input error, in which the magnitude of error is normalized by the peak-to-peak human input for individual cases. Figures in the second column are normalized by the reference input, which is 1 crad. It could be observed that the error of the system state is lower than 15% for pole set one and pole set three.
5.3 Distribution of Poles and Zeros of Human Model

To better understand the generalized characteristic of transfer function derived from System Identification Toolbox, the distribution of poles and zeros are examined. According to [57], human gesticulation is hard to oscillate over 4.5 Hz. For this reason, identified system are supposed to have poles whose real part are on the right side of $-9\pi$. In Figure 22, cases are counted when both poles and zeros have their real part bigger than the threshold (at right side of threshold).

![Number of Cases at right side of Threshold](image1)

Numbers of cases within threshold

![Pole set 1 poles and zeros distribution](image2)

Pole set 1 poles and zeros distribution

![Pole set 2 poles and zeros distribution](image3)

Pole set 2 poles and zeros distribution

![Pole set 3 poles and zeros distribution](image4)

Pole set 3 poles and zeros distribution

Figure 22 Number of cases within threshold
According to Figure 22, 80% of poles are on the right side of threshold 15, which represents hand movement frequency at about 2.5 Hz. For better visualization, the threshold is set to be 10 in plotting the distribution of poles and zeros. When threshold equals to 10, 78% of cases are counted with pole set one, 76% of cases with pole set two are covered, and 80% in set three are included. Cases whose poles and zeros fit this range are plotted and analyzed. Average and standard deviation of real part of poles and zeros in previous selected cases are calculated.

Real part of center of poles or zeros could be expressed in following equation.

\[
Re(P_o) = \frac{\sum_{i=1}^{n} Re(P_i)}{n}
\]

\[
Re(Z_o) = \frac{\sum_{i=1}^{n} Re(Z_i)}{n}
\]

in which \(Re(\cdot)\) represents the real part of a complex number, \(P_i\) and \(Z_i\) are poles and zeros separately, and \(n\) is the total number of poles or zeros in the range.

Figure 23 Center of poles and zeros with standard deviation on real axis
5.4 Results

In this section, System Identification Toolbox is utilized to analyze human dynamics for better understanding how human behaves in the Cursor Control Game. Identified systems are simulated given the record of state feedback and reference input from experiment to see if the transfer function derived by System Identification Toolbox would behave similar to the real human under same circumstances. Then, the identified transfer function is implemented in the human-in-loop MRAC system to see if the identified system would behave close to the specific participant. The result in Figure 21 shows that the nonlinearity of human decision-making leads to relatively big difference peaks at about 35% in pole set one. However, generally speaking, the performances of identified system and real human with same input are similar, with the error in the range of ±10%. For the system state output, the simulation results are close to the real human with the maximum error at 22% in simulation for pole set two. Additionally, errors of system state output are not diverging, which could support that the identified transfer function have the potential to simulate the real human behavior and be a tool for stability analysis.

In Figure 22, it appears that 80% of poles and zeros are on the right side of −15, which shows a lower frequency in using mouse for the Cursor Control Game than the human gesticulation frequency in [57]. The distribution of poles shows that almost all the poles and zeros are gathered in a relatively small range around the real axis. We however note that the nonlinearity and robustness of real human dynamics system must also be assessed in future studies.

According to Figure 23, in average sense, center of poles on real axis for set three are closer to imaginary axis than other two sets of poles. However, large standard deviation shows that this might be an inconclusive result because of the lack of sufficiently many experiments. Due to the lack of sufficiently many experiment and uncertainty of human, no evident results in predicting human behavior are acquired. However, despite some unstable
or negative fit percent identified system, in some cases, participants’ behavior does match a
two-poles-two-zeros transfer function and differences are mainly because of the
nonlinearity of human reaction and the limitation of mouse which may prevent humans from
giving instructions smoothly.

Chapter 6: Conclusions

6.1 Designs and Experiment of Human-in-Loop MRAC system

In Chapter 2, LQR method, modified LQR method, and pole placement method are
implemented and discussed in the circumstance of designing MRAC system inspired by [49].
Due to the main purpose of design is experiment based on real-time simulation, power
consumption is not concerned in designing. The pole placement method is selected for its
high flexibility in designing poles with certain time domain step input response. The design
shows same feature in time domain response except the length of time system reaches the
peak. In this way, three different MRAC systems are designed with different response speed.
Because human dynamics models could only serve for stability analysis, real human is
required to control the MRAC system for the comparison of three designs.

In Chapter 3, a Cursor Control Game is designed based on the previous three designs
of human-in-loop MRAC systems. Participants are recruited to play the game and their
performance will be recorded and analyzed to determine which system is the most suitable
and friendly for real human users. According to the result discussed in Chapter 4, the design
with pole set one which represents the system with the fastest response speed among three
designs is the best design for users, both in performance and human effort during the game.

Then, with the record of system feedback, reference input, and human command input,
System Identification Toolbox in Matlab is used in analyzing human dynamics. The fitness
varies a lot due to the relatively short length of playing time and the uncertainty of
participants’ decision-making during the control process. Three cases in different designs are taken as examples. The high FitPercent in several cases shows that, in those cases, human behavior does fit the identified transfer function within a satisfactory level with less than 20% error. The error between simulation of human command given to the MRAC system and the recorded command form real human shows the good fitness and the potential in analyzing real human behavior on MRAC system using transfer function with human reaction delay. Poles and zeros of identified transfer functions are relatively intensely distributed around the imaginary axis with real part in the range from $-2$ to $0$.

6.2 Further Works and Potential Improvements

Due to the limitation of time and effort, the research in this thesis is just a small step in understanding how human reacts to the MRAC system and whether some certain kinds of designs could improve human performance with less effort from users. To get a more conclusive and convincing conclusion on the relationship of pole placement method design and human performance, more design of systems should be tried to determine if the faster system response is beneficial to human behavior. Additionally, only 14 participants recruited here who are from 21 to 30 years old with no health issues. For further research, more volunteers could be recruited to generalize the results and analyze the identified systems.

The Cursor Control Game could also be used in understanding how humans learn using methods in Figure 12 and Figure 14. The implementation of communication delays between human and MRAC system to simulate the remote-control case could also be conducted.

For system identification, nonlinear elements could be implemented in the transfer function to approach the behavior of a real human in playing Cursor Control Game.

The above plans demonstrate some of the potentials of the research in human-in-the-loop MRAC system. Our group will continue our research in understanding how to design user friendly MRAC system and human behavior.
References


