SENSORIMOTOR ADAPTATION OF HUMAN CONTROL STRATEGIES: RAMIFICATIONS FOR FUTURE HUMAN-MACHINE INTERFACE DESIGN

by

Corinna E. Lathan

Ph. D. Neuroscience, Massachusetts Institute of Technology (1994)
B. A. Biopsychology and Mathematics, Swarthmore College (1988)

SUBMITTED TO THE DEPARTMENT OF AERONAUTICS AND ASTRONAUTICS IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE DEGREE OF

MASTER OF SCIENCE

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

September, 1995

Signature of Author ______________________________
Department of Aeronautics and Astronautics
August 11, 1995

Certified by ______________________________
Professor Dava Newman
Thesis Supervisor

Accepted by ______________________________
Professor Harold Y. Wachman
Chairperson, Department Graduate Committee

SENSORIMOTOR ADAPTATION OF HUMAN CONTROL STRATEGIES: RAMIFICATIONS FOR FUTURE HUMAN-MACHINE INTERFACE DESIGN

by

Corinna E. Lathan

Submitted to the Department of Aeronautics and Astronautics on August 11, 1995 in partial fulfillment of the requirements for the Degree of Master of Science in Aeronautics and Astronautics.

ABSTRACT

There are many research issues involved with designing, constructing, and applying Human-Machine Interfaces. This thesis addresses three areas of HMI research, 1) operational concerns for multi-task, high workload environments, 2) basic research in sensorimotor and manual control strategies, and 3) computer workstation HMI evaluation for telemedicine applications.

1) The Canadian Astronaut Program Space Unit Life Simulation (CAPSULS) mission offered an ideal opportunity to collect human performance data for individuals in extreme isolation to compliment data collected in microgravity on the International Microgravity Laboratory (IML-1) Space Shuttle mission (STS-42) using similar protocol, hardware, and software. Subjects performed a dual-task experiment to assess cognitive and fine motor control both during isolation alone and when paired with induced sensorimotor adaptation through spaceflight or prism adaptation. Sensorimotor alterations significantly degraded motor but not cognitive performance.

2) Implementing controllers for telerobotic applications is dependent on what we know about the human operator's ability. All controllers induce some sort of sensorimotor transformation so it is important to understand and quantify human performance using different control strategies under different demands of sensorimotor adaptation. Chapter 3 builds on the results from Chapter 2 and tests the hypothesis that performance depends on manual control strategy particularly during induced sensorimotor transformations. In general, rate-control produces slower time constants of adaptation and worse steady-state performance.

3) Telemedicine programs are being designed and developed for health care providers and patients who have limited access to medical facilities due to location or cost. The HMI for telemedicine workstations is of particular concern because of the underlying technical complexities. Chapter 4 describes several potential user scenarios for telemedicine that would benefit from workstation development such as ground-based medical support for astronauts, disaster prevention and relief, and medical education.

Thesis Supervisor: Professor Dava Newman
Title: Assistant Professor, Department of Aeronautics and Astronautics

...ce n'est pas assez d'avoir l'esprit bon, mais le principal est de l'appliquer bien.

...it is not enough to have a good mind, but most importantly, to use it to do good works.

—DESCARTES (1596-1650)

TABLE OF CONTENTS

1.0 Experimental Issues in Human-Machine Interface Design............................

2.0 Memory Processes and Motor Control in Extreme Environments...................

3.0 Sensorimotor Adaptation of Human Control Strategies..............................

CHAPTER 1

EXPERIMENTAL ISSUES IN HUMAN-MACHINE INTERFACE DESIGN

1.1 Introduction

We are on the verge of an exciting new era where technological advances are opening doors in all areas of human performance; The use of virtual environment technologies and the developments in medical informatics are two growing areas. From basic communication to building a space station, these "enabling technologies" require major advances in the Human-Machine Interface (HMI). An HMI is any device that is used to present information, or to sense human actions or responses that enable the user to complete a specified goal. HMIs can be anything from computer keyboards to multimodal, immersive systems, that enable an operator to interact with some structured environment. The research issues involved with designing, constructing, and applying HMI systems are many. This thesis addresses three areas of HMI research, 1) operational concerns for multi-task, high workload environments in Chapter 2, 2) basic research in sensorimotor and manual control strategies in Chapter 3, and 3) computer workstation HMI evaluation for telemedicine applications in Chapter 4.

1) Operational concerns for multi-task, high workload environments: The purpose of our experiments described in Chapter 2 was to assess psychological and motor performance activities for future operations involving human control of partially automated activities (i.e., teleoperation and supervisory control of remote operations). Our data will enhance the knowledge-base on human machine interface technology for human performance in multi-task, high workload environments. For example, the projected operational requirements of the International Space Station will present new challenges for aerospace medicine and physiology and human-machine interface design. Human perceptual and cognitive capabilities as well as the limitations of human performance will be driving factors in the system design.

2) Basic research in sensorimotor and manual control strategies: Design of HMI systems, whether for spaceflight or for rehabilitative purposes assumes a knowledge of human performance and capabilities under "normal" physiological conditions. With the use of any HMI, there will be some alteration of sensorimotor loops, i.e. the integration of

sensorimotor information and the use of that information to perform a motor task. Some sensorimotor alterations may be intended to increase performance such as allowing surgeons to make precise movements through robotics during microsurgery. However, some alterations may be a result of noise or technical limitations such as in full-immersion virtual reality systems which cannot update the visual field fast enough to mimic the real-world. Studying sensorimotor adaptations under controlled circumstances is necessary in order to predict responses of human performance under altered conditions.

3) An application of HMIs to telemedicine systems: Telemedicine refers to the use of telecommunications technology to provide medical care to a patient at a distance from the health care provider. Until recently, most of the research and applications in the field of telemedicine have focused on the technical feasibility and the political infrastructure (e.g., cost-effectiveness, physician and patient acceptance). Increasingly, the user-interface is receiving needed attention. Telemedicine workstations are being specified for the particularly user scenario or application. The interface chosen will have a direct relationship to the health care providers' performance. Assessing human performance during telemedicine activities will be crucial in determining the difference between use and nonuse of a system and the difference between proper diagnosis and misdiagnosis. The first research task in this thesis was to determine the important components of the human-computer interface for each telemedicine application. Each application has different ramifications for the work-station design, the needed computer literacy, and level of technology used. During the final stages of workstation design, experimental methods can be used to quantitatively assess operator performance.

1.2 An Operational Motivation for HMI Experimentation

Chapter 2 presents two experiments which were driven by operational concerns. Human performance under high-workload conditions is assessed through cognitive and motor tasks performed in two extreme environments.

The Mental Workload and Performance Experiment (MWPE) represents the first attempt to explicitly measure astronaut performance in space, at tasks that reflect the kind of computer-interactive tasks that are likely to dominate space-station operations. One of the driving factors of MWPE was the recognition that as missions get longer and as tasks become more automated and computer controlled, the role of the human operator changes. A number of possible causes for a decrease in performance could exist, including possible direct effects of microgravity and microgravity adaptation on the nervous system, reflex adaptations to moving in microgravity, discomfort and mental preoccupation due to space adaptation syndrome (SAS), stress and workload associated with tasks, and noise and other distractions.

The Memory Processes and Motor Control experiment (MEMO) was designed to examine sensorimotor adaptation strategies and performance during a Canadian astronaut program mission simulation, a 7-day isolation experiment with four Canadian astronauts. The goal of MEMO was to look at the performance of a human operator on tasks associated with a high-workload, high stress environment, particularly when sensorimotor transformations were also required within the tasks, but without the effects of microgravity.

For both MEMO and MWPE, we studied short-term memory and fine motor control associated with human/computer interaction. Memory processes were assessed using a Sternberg-like exhaustive memory set containing 1, 2, 4, or 7 letters (Sternberg 1975). Fine motor control was assessed with a Fitts' target acquisition task using velocity-controlled (joystick) and position-controlled (trackball) computer input devices (Fitts and Peterson 1964). Subjects repeated the tasks under two conditions that test perceptual-motor adaptation strategies: 1) During adaptation to the microgravity environment of the space shuttle during MWPE, and 2) While wearing left-right reversing prism goggles during the CAPSULS mission during MEMO. Objective measures of task performance include reaction time, movement time, and overall subjective ratings of workload were recorded.

1.3 A Basic Science Motivation for HMI Experimentation

In Chapter 3, two research topics are addressed that are of interest in evaluating human performance for design of HMIs, namely sensorimotor integration and manual control strategies. Sensorimotor integration in systems physiology refers to the process by which the central nervous system converts sensory inputs into a motor output. Proper integration of sensory inputs is crucial for spatial orientation, the perception of our orientation with respect to the world around us (Welch 1978). The dependence of spatial orientation on different senses has ramifications for normal postural control and locomotion, as well as for pilot and astronaut disorientation, or training in virtual environments. In almost any HMI, there will be some alteration of our normal sensorimotor loops. Some alterations may be intended to increase performance and some may be a result of noise or technical limitations. There is a need therefore, to study sensorimotor loop adaptations in order to predict responses of human performance to alterations.

Manual control refers to the closed-loop control of some system by a human operator through feedback about the desired state of the system. Evaluation of manual control strategies are important for designing HMIs for fully and partially-automated tasks. Both position and rate control devices have been used extensively in teleoperations and in the research laboratory to model human performance (Kim, Tendick et al. 1987; Bejczy, Hannaford et al. 1988; Hannaford, Wood et al. 1991; Das, Zak et al. 1992; Oyama, Tsunemoto et al. 1993; Massimino and Sheridan 1994). How to implement controllers for telerobotic applications is dependent on what we know about the human operator's ability. Therefore, it is important to understand and quantify human performance using different control strategies under different demands of sensorimotor adaptation.

The experiments presented in Chapter 3 compare human performance using position-control or rate-control devices to execute a radial Fitts motor task under induced sensorimotor transformations. Sensorimotor transformations are systematically induced to compare the learning rates, or time constants of adaptation, using each control strategy. Rotational transformations of the visual field are induced from 5 degrees through 60 degrees.

1.4 An Application of HMIs to Telemedicine Systems

Chapter 4 presents an application of HMIs to telemedicine systems. The goals of Chapter 4 are threefold: 1) To provide a review of currently available telemedicine workstations and their associated technologies, 2) To provide a case study of a telemedicine workstation, and 3) To make recommendations for a telemedicine workstation and provide a demonstration of a graphical user interface (GUI). An evaluation of the telemedicine workstations under development is provided, including an assessment of the requirements of an effective system for ground-based support of the astronauts.

Telemedicine programs are being designed and developed for health care providers and patients who have limited access to medical facilities due to location or cost. The National Aeronautics and Space Administration (NASA) has an interest in developing telemedicine workstations for medical ground-support for astronauts during U.S./Russian joint spaceflights and future space station missions. In addition, there is a need for telemedicine systems targeting disaster relief and disaster preparedness and distance learning in the medical community. Versatile or robust telemedicine workstations will prove beneficial in remote under-served locations, and in all areas in need of disaster relief.

Some of the issues that impact workstation design are communication links, training time, and use of technology. A telemedicine workstation may be as simple as a passive computer interface connected to the Internet, to a fully portable, compact unit that registers and evaluates medical input. The bandwidth available may influence the level of technology, but even with low bandwidth, digital imaging capabilities and processing techniques may allow for advanced interfaces. For example, biosensor technology can provide immediate assessment of patient status, or expert-systems may give on-line diagnostic assistance.

Workstations may be connected to medical centers or to specialists in several countries across networks such as the Internet or via satellites. Voice as well as video capabilities may also be available. Defining the communication links, bandwidth requirements, and access to information and expertise are priorities in evaluating the telemedicine application and appropriate workstations, and are also important factors in evaluating cost and dependability.

1.5 References

Bejczy, A. K., B. Hannaford, et al. (1988). Multi-mode manual control in telerobotics. In Proceedings of Romany '88, Udine, Italy,

Das, H., H. Zak, et al. (1992). "Operator performance with alternative manual control modes in teleoperation." Presence **1**: 201-208.

Fitts, P. M. and J. R. Peterson (1964). "Information capacity of discrete motor responses." J. of Exp. Psych. **67**(2):

Hannaford, B. L., L. Wood, et al. (1991). "Performance evaluation of a six-axis generalized force reflecting teleoperator." IEEE Transactions on Systems, Man, and Cybernetics **21**: 620-633.

Kim, W. S., F. Tendick, et al. (1987). "A comparison of position and rate control for telemanipulations with consideration of manipulator system dynamics." IEEE Journal of Robotics and Automation **3**: 426-436.

Massimino, M. J. and T. B. Sheridan (1994). "Teleoperator performance with varying force and visual feedback." Human Factors **36**(1): 145-157.

Oyama, E., N. Tsunemoto, et al. (1993). "Experimental study on remote manipulation using virtual reality." Presence: Teleoperators and Virtual Environments **2**(2): 112-124.

Sternberg, S. (1975). "Memory scanning: New findings and current controversies." Quart. J. of Exp. Psych. **27**: 1-32.

Welch, R. B. (1978). Perceptual Modification: Adapting to altered sensory environments. New York, Academic Press, Inc.

CHAPTER 2

MEMORY PROCESSES AND MOTOR CONTROL IN EXTREME ENVIRONMENTS

The purpose of the experiment described in Chapter 2 was to assess cognitive and motor performance activities in multi-task, high workload environments, particularly those involving human control of partially automated activities (i.e., teleoperation and supervisory control of remote operations). The Canadian Astronaut Program Space Unit Life Simulation (CAPSULS) mission offered an ideal opportunity to collect human performance data for individuals in extreme isolation to compliment data collected in microgravity on the International Microgravity Laboratory (IML-1) Space Shuttle mission (STS-42) using a similar protocol, hardware, and software. The Memory Processes and Motor Control (**MEMO**) experiment was performed during the CAPSULS mission and the Mental Workload and Performance Experiment (**MWPE**) was performed during the IML-1 mission. In both experiments, short-term exhaustive memory and fine motor control associated with human/computer interaction was studied. Memory processes were assessed using a Sternberg-like exhaustive memory set containing 1, 2, 4, or 7 letters. Fine motor control was assessed using velocity-controlled (joystick) and position-controlled (trackball) computer input devices to acquire targets as displayed on a computer screen. Subjects repeated the tasks under two conditions that test perceptual-motor adaptation strategies: 1) During adaptation to the microgravity environment of the space shuttle, and 2) While wearing left-right reversing prism goggles during the CAPSULS mission. Both conditions significantly degraded motor but not cognitive performance. The data collected during both the MEMO and MWPE experiments enhance the knowledge-base of human interface technology for human performance in extreme environments.

2.1 Introduction

Issues in human-computer interactions and the underlying perceptual-motor processes are taking on a new significance with the expansion of virtual environment interfaces and the changing role of the human operator. For example, the projected operational requirements of the International Space Station will necessitate extensive automation and expansion of the supervisory role for its crewmembers. Human perceptual and cognitive capabilities as well as the limitations of human performance are driving factors in system design, particularly in the areas directly affecting human-computer interface design. The first goal of MEMO and MWPE was to quantify short-term memory performance as well as motor control associated with human/computer tasks for planning and commanding in extreme environment that are characterized by high workloads leading to high stress and fatigue. Such environments include outer space, polar region stations, and oil rigs.

The Mental Workload and Performance Experiment (MWPE) was developed in the mid-1980's to look at human performance during spaceflight. MWPE flew on the International Microgravity Laboratory-1 Space Shuttle mission in January of 1992 (Newman and Bussolari 1990; Newman, Lichtenberg et al. 1993). The general motivation for developing MWPE was to investigate the performance of a human operator on tasks associated with high-workload and stress during spaceflight. One of the driving factors was the recognition that as missions get longer and as tasks become more automated and computer controlled, the role of the human operator changes. Perceptual and cognitive capabilities and limitations of humans are becoming driving criteria for system design of the human-machine interface.

Human performance has many aspects that are difficult to measure and distinguish from one another. Space Station operations, however, will put particular emphasis on astronauts' interaction with the Station's many computer control systems. The MWPE experiment was therefore designed to focus on motor and cognitive skills associated with such interactions, principally computer cursor control and short-term memory. Though narrowly focused, the experiment serves as a prototype for further investigations to pursue broader, multidimensional measures of in-space performance. The MWPE performance assessment test is based on the *Fittsberg* task, a combination of Fitts' Law and Sternberg tasks, that combines tests of short-term memory and motor control (Fitts and Peterson 1964; Sternberg 1975; Hartzell, Gopher et al. 1983). This task, and the theory that originally motivated it, is discussed in greater detail below.

The second goal of MEMO and MWPE was to assess human performance while performing interactive computer tasks that require perceptual-motor, or more generally, sensorimotor adaptation. This goal involves de-coupling the workload and stress of microgravity from the alterations in the astronauts normal sensorimotor functioning. In the case of spaceflight, this altered functioning is unwanted and usually degrades performance. However, in many situations, such as in performing a teleoperator task, the sensorimotor transformations are expected to enhance performance. For example, manipulation of a large robot arm would impossible without a control interface that enhanced the forces initiates by the human operator. In either case, knowledge of human performance with normal sensorimotor loops is needed. We begin to address this issue in the MEMO experiment, that provides an initial look at multi-task performance under induced sensorimotor transformation. However, Chapter 3 thoroughly investigates sensorimotor adaptation.

2.2 Experiment 1: Mental Workload and Performance Experiment (MWPE)

Space flight places astronauts in a very stressful working environment, as well as inducing substantial changes in physiological functioning due to microgravity. The Mental Workload and Performance Experiment (MWPE) was designed to assess the influence of the space flight environment on astronaut productivity, particularly for computer-interaction tasks, by measuring cognitive and motor performance during a dual, computer-based memorization and target acquisition task. The MWPE experiment flew on the 8-day, 10-hour First International Microgravity Laboratory (IML-1) Space Shuttle Mission (known as STS-42) launched on 22 January 1992(Newman and Bussolari 1990; Newman, Lichtenberg et al. 1993). Four astronaut test subjects each performed four interactive experiment sessions during the course of the flight along with three sessions both preflight and postflight.

2.3 Experiment 2: Memory Processes and Motor Control (MEMO)

MWPE was enhanced for the Canadian Astronaut Program Space Unit Life Simulation (CAPSULS) mission and was renamed the MEmory processes and MOtor control experiment (MEMO). This mission studied four Canadian astronauts during seven days of isolation at the Defense and Civil Institute of Environmental Engineering (DCIEM), Toronto, Canada. The CAPSULS 7-day Isolation Mission offered an ideal opportunity to collect human performance data for individuals in extreme isolation to compliment the data collected in space on the IML-1 Mission using a similar protocol and ground hardware. The CAPSULS experiment duplicated the mission workload and conditions of isolation with the absence of physiological changes due to exposure to microgravity. The experiment then evaluated operator performance on the same short-term memory and fine motor control tasks that were performed for MWPE. MEMO then examined human performance when a sensorimotor transformation was deliberately induced in order to evaluate the limitations of different human operator control strategies. Specifically, our subjects wore left-right reversing prism goggles for approximately one-third of the trials.

2.4 Background for MWPE and MEMO

The MWPE experiment design was based on two hypotheses regarding astronaut performance in space. The first was that the combined effects of spaceflight such as stress, physiological adaptation, and the direct influence of microgravity might cause a degradation in cognitive performance and particularly short-term memory. The experiment therefore incorporated a sub-task focusing on short-term memory. MWPE's second primary hypothesis was that motor-control adaptation to microgravity, and perhaps other factors such as neurological effects, might result in reduced fine motor control performance, which motivated including a motor-control task. In order to address each of these hypotheses, the MWPE experiment incorporates the dual elements of the "Fittsberg" experimental paradigm, combining Fitts' test of motor performance with Sternberg's test of cognitive performance. Objective measurements of performance on the two tasks are response selection or reaction time and response execution or movement time, where the former represents a cognitive task and the latter a neuromuscular task.

This section presents dual-task methodology background as well as the specific motor and cognitive tasks. Finally, sensorimotor adaptation will be discussed.

2.4.1 *Dual-task versus single-task performance*

The role of the human operator is shifting from that of system control to supervisor of monitor of multiple-tasks or several semi-automated systems (Sheridan 1992). For example, the operator's responsibility is to extract and integrate information and make decisions. Information-processing becomes a crucial factor in addition to sensory-motor skills, and in some cases more important. The more tasks the operator is keeping track of at one time, the more information processing is needed.

A major premise in dual-task evaluation is that performing two tasks is inherently different from performing one task, regardless of the complexity of the tasks (Damos 1991). There are many factors that affect the issue of task combination, contributing to this philosophy. Table 1 summarizes dual-task characteristics of dual tasks and alternative configurations (Damos 1991).

Table 2.1: Characteristics of dual tasks and their alternative configurations (Damos 1991).

Characteristics	Alternative configurations
Number of Stimuli	Two - Physically Separate Two - Superimposed One - Shared
Stimulus Modality	Same Different
Correlation between Stimuli	0.0 (Independent) - 1.0 (Dependent)
Central Processing	Independent Correlated Integrated
Number of Response Channels	Two - Separate One - Shared
Response Modality	Same Different

Our Fittsberg paradigm has two stimuli, letters and a target, however these can be considered as superimposed since they appear right next to each other and the appropriate target corresponds to the memorized letter. Previous experiments with two stimuli, physically separated (Wickens and Gopher 1977) or superimposed (Gopher, Brickner et al. 1982), have shown decrements in dual-task performance when compared with corresponding single-task performance.

The stimulus modality is more complicated; Even though both targets are presented visually, the stimuli access multiple resources in the human information processing systems (Wickens 1991). The memory targets are representative of perceptual/cognitive capabilities whereas the location of the target represents information accessing motor control manipulation. The basic philosophy is that when cognitive resources are divided between two activities, one or both of the task performances will suffer. A simple one-task model is as follows:

Performance (P) = Resources (R) / Difficulty (D)

Decreasing resources (R), decreases performance. The response modality is also complicated since we are using one motor task to reflect access to cognitive information (reaction time) and to motor control abilities (movement time). So although there is one response modality, there are two response channels.

Correlation is easily defined in the computational sense. Our stimuli are uncorrelated since the letters presented do not predict the subsequent direction of the target. Most dual-task studies use uncorrelated stimuli since tasks using correlated stimuli are not comparable to single-task counterparts.

The central processing for the Fittsberg paradigm is considered integrated since the subjects use information from one task (letter recognition) to respond to a subsequent task (target acquisition). Previous experiments using integrated central processing have found a decrement in dual-task performance (Tsang 1986).

2.4.2 *Measurement procedures for dual-task assessment*

A major objective of assessing multi-task performance is to evaluate the workload imposed by a multi-task environment so that the information-processing demand does not exceed the operator's capabilities. Performance-based and subjective assessment of workload are two primary measurement procedures employed in multi-task environments (Eggemeier and Wilson 1991). Performance-based measures depend on the objective measures of the operator's capability to perform the task whereas subjective measures derive estimates of workload through operator reports of workload or expenditure of effort during the task.

Performance-based techniques consist of primary- and secondary- task measurements. Figure 1 shows the hypothetical relationship between operator performance and the level of workload (Damos 1991). In Region 1, the operator has the capability to compensate for increased workload by allocating additional information processing capacity or resources to the task or group of tasks. In Region 2, the operator can no longer compensate for increased workload and Region 3 represents a bottoming out of operator performance. Traditional measures of primary task performance include speed and accuracy measures. Primary-task measures are generally insensitive to workload variations in Region 1 so one of the theories of secondary-task methodology is to shift the total workload to Region 2 where operator performance is expected to reflect variations in workload associated with performance of the task. Typically, a secondary task (e.g., memory search) is used to evaluate changes in the demand associated with variants of a single task (e.g., movement to designated targets. Thus, our Fittsberg paradigm can be considered as a primary movement task and a secondary memory search task.

The Sternberg memory search paradigm (Sternberg 1975) is among the most frequently used secondary-task procedures for multi-task environments, where reaction time is the dependent variable. Schiflett et al. (1982) were able to distinguish different workloads associated with display conditions in an airplane cockpit with a memory search reaction time test when the primary measure, pilot flight performance failed to show systematic differences (Shiflett, Linton et al. 1982).

Subjective ratings are also used to achieve a measure of workload. We use a rating scale technique of overall workload that is a simplified version of the NASA Task Load Index (TLX) (Hart and Staveland 1988). Within the NASA TLX, mental demand is rated according to "How much mental and perceptual activity was required (e.g., thinking, deciding, calculating, remembering, looking, searching, etc.)? Was the task easy or demanding, simple or complex, exacting or forgiving?"

2.4.3 *Sternberg memory task*

Traditionally human memory is studied upon failure, but Sternberg proposed an alternative approach of studying memory; he studied "successful memory" that was almost free of errors (Sternberg 1975). According to his approach, human subjects reveal their memory retrieval mechanisms not by how they fail, but by how much time it takes them to perform a task successfully. The reaction time measurement RT is the period of time from onset of the test stimulus (presentation of memorized letters and targets) to the initiation of cursor movement (the cursor represented by crosshairs on the computer screen). In other words, reaction time is the time it takes the astronaut to recognize the letter from the memory set and begin cursor movement toward acquiring the target. Sternberg's classical model suggests that RT increases approximately linearly with response entropy, H, which is the amount of information needed to uniquely specify the single correct letter out of a memory set of size n, assuming equal probability of any of the memory set letters being the correct choice. The classical response entropy and reaction time models are given by Equations 1 and 2, respectively:

$$H = \log_2(n) \quad (1)$$

$$RT = a + b\log_2(n) \quad (2)$$

where a is a constant associated with a cognitive overhead for mental processing, and the slope of reaction time increases at a rate b with each additional bit of memory-set information content. Sternberg's experiments yielded a slope $b=38$ ms/bit and an overhead of approximately $a=400$ ms (Sternberg 1975).

2.4.4 *Fitts' motor task*

Motor control, as measured by movement time, received experimental emphasis for the connection between the human motor system and information capacity in the early 1960's (Fitts and Peterson 1964). Fitts reasoned (1954) that the distance, D, of a human movement and the width, W, of the target being acquired defined an index of task difficulty, ID, again expressed in bits of information according to the logarithmic representation:

$$ID = \log_2\left(2D/W\right) \qquad (3)$$

Fitts' Law predicts movement time, MT, to be a linear function of index of difficulty. Fitts defines target acquisition movement time according to the time it takes the subject to reach a target, by the following equation:

$$MT = c + d\log_2(2D/W) \qquad (4)$$

where c and d are constants. It seems logical that MT increases if the movement distance increases or target width decreases because under these prescribed conditions the target becomes further away and smaller, respectively.

2.4.5 *Sensorimotor adaptation*

One of the research questions for MWPE and MEMO was to ask how well the astronauts would perform basic cognitive and motor tasks while adapting to the spaceflight environment. Adaptation to alterations in normal sensorimotor loops has ramifications for any task involving a human-computer interface. An example of a sensorimotor transformation is the act of using a horizontal mouse or joystick to control an arrow on a vertical computer screen. In designing systems for teleoperations or telesurgery, the needed sensorimotor adaptations are much more complicated then the simple 90 degree transformation just described. Transformations can be deliberate to improve performance such as enhanced gains for microsurgery procedures. However, sometimes transformations are unwanted such as sensorimotor alterations due to the effects of microgravity. How well we can adapt to sensorimotor transformations and perform the required task is an important area of research for spaceflight as well as ground-based research.

Welch defines adaptation as a "semipermanent change of perception or perceptual-motor coordination that serves to reduce or eliminate a registered discrepancy between or within sensory modalities or the errors in behavior induced by this discrepancy (Welch 1986)." Prisms have been used to study perceptual-motor coordination since 1925 when Helmholtz observed that when reaching for an object, subjects could quickly overcome the errors induced by a displacing a visual field through wedge prisms (for a review see Welch, 1986). Many people have demonstrated that humans are capable of adapting to almost any stable rearrangement of sensory input although the mechanisms for adaptation vary depending on the particular circumstance. Most subject exposures to a perceptual rearrangement are done in a "constrained" fashion as opposed to "unconstrained." Unconstrained implies that the subject is moving about freely whereas the constrained setting is usually in a head-fixed experimental set-up. The experiments presented in the subsequent sections are all constrained perceptual rearrangement paradigms.

2.5 Methods

2.5.1 *The "Fittsberg" task*

We used the "Fittsberg" experimental paradigm(Hartzell, Gopher et al. 1983) that provides independent control and measurement of two tasks: response selection and response execution, where the former represents a cognitive task and the latter, a neuromuscular task. The selection of a (binary) response is based upon the Sternberg memory search task(Sternberg 1975) that requires the subject to determine if a displayed item is a member of a previously memorized set. Fitts' paradigm(Fitts and Peterson 1964) was developed to examine the control and accuracy of movement and was used here to measure response execution. Subjects were required to manually acquire a target of a certain size and distance away from an initial cursor position as quickly and as accurately as possible. The Fittsberg paradigm is illustrated in Figure 2.

From the time the targets appear on the computer screen, to the time it takes the subject to identify the letter is the reaction time (RT), a measure of short-term memory. From the time the subject starts to move the cursor on the screen via the computer device to the time he or she reaches the target is the movement time (MT), a classical measure of motor control. For each memory set, 8 test stimuli were presented, while 12 memory sets were presented for each device.

The specific process of the interactive experiment was as follows: at the beginning of each experimental run, the astronaut subject was directed to prepare to use one of the two computer input devices, the trackball or joystick. When the subject connected the designated device to the computer, she was presented with a "memory set" of one to seven characters to be memorized and used for the upcoming series of tests. Specifically, four memory set sizes comprising 1, 2, 4, and 7 letters were used in the astronaut tests and were chosen to present the astronauts with a wide range of workload.

After memorizing the memory set, the subject pressed the enter key and was then presented with a "target set" of eight characters, or probe items, arranged in a circle (or a clock face pattern). Exactly one of these eight characters matched a member of the memory set: the subject was required to choose that one and to designate it by moving the cursor from the center of the screen to the corresponding target square using the currently selected computer input device. The time elapsed from presentation of the memory set to the beginning of cursor motion was recorded as the reaction time (RT).

The difficulty of selecting the chosen character was varied by changing the size of the target square associated with the target characters, and by changing the diameter of the circle they were arranged in, both of which influence the Fitts' law index of difficulty (ID). The time elapsed from beginning cursor motion (the end of the *RT* period) to settling within the correct target square (remaining at least 400 ms) was recorded as the Fitts movement time (MT). Three possible indices of difficulty ID=2, 3, and 4 bits of information were used (recall Equation 3). The three ID values correspond to small, closely-space targets (W=5 pixels, D=20 pixels, log(8)), large, widely-spaced targets (W=10 pixels, D=30 pixels, log(6)), and small, widely-spaced targets (W=5 pixels, D=30 pixels, log(12)). Distance (D) refers to the distance between the center of the screen and the target location whereas width (W) refers to the number of pixels spanned by a particular target.

The following is a summary of parameters measured:

Dependent Variables:

1. Short-term memory is assessed by Reaction Time.
2. Motor performance is measured by Movement Time.

Independent Variables:

1. Four memory set difficulties (1, 2, 4, and 7 letters)
2. Three target acquisition index of difficulties (including target size and distance from the initial cursor position)

3. Graphic input devices (e.g., joystick, trackball,): The trackball provided direct control of cursor position, so that rotation of the trackball corresponded to cursor motion. The joystick, on the other hand, controlled the *rate* of the cursor's motion, making it more challenging to control.
4. Direction of Target Acquisition: The direction (from the center of the screen) of the target to be acquired was recorded to provide information on whether location of a target on the computer screen significantly affects the length of cursor movement. The eight target directions included the cardinal directions North, South, East, and West, and the diagonal directions Northeast, Southeast, Southwest, and Northwest.

As well as these objective performance measures, MWPE and MEMO incorporated an overall subjective rating scale in its experimental paradigm in order to assess mental workload. The rating consisted of a forced-choice scale of workload ranging from very low to very high, for subjects to report their subjective experience of workload. A second rating scale, for the current nausea level, is used in MWPE to assess the effect of space adaptation syndrome on workload and performance associated with the experimental tasks.

2.5.3 *Equipment*

MWPE was conducted using a Payload General Support Computer (PGSC), a GRiD Corporation Model 1530 laptop computer (Fremont, CA) specially modified to be used on the Space Shuttle. The PGSC computer has a specially fitted electro-luminescent display. The PGSC computer is approximately compatible with the IBM-PC desktop computer. The Grid 1530 microcomputer was used to present the experimental paradigm and collect the data for both MWPE and MEMO.

In order to provide the subjects with a variety of computer input devices, a trackball and joystick were purchased from Measurement Systems Corporation (city, CT) by NASA and qualified for flight use. The trackball has no dependence on gravity for normal functioning, and in fact can be used in Earth gravity in any orientation. The trackball provided direct control of cursor position, so that rotation of the trackball corresponded to cursor motion. The joystick, on the other hand, controlled the *rate* of the cursor's motion. (discussed more in chapter 3) For both the MEMO and MWPE task, the subjects used either the position-control device (trackball) or the rate-control device (joystick) to perform the experiment.

The test program was designed to be fully automated, so that the subjects were prompted through the entire testing process once they began the Fittsberg program. This minimized the need for extensive operational training, which is extremely useful when conducting a space flight experiment using the Space Shuttle. The training time that was available was dedicated to learning how to set up and manipulate the experimental workstation, and to achieving skill at the Fittsberg task to minimize training effects during actual data collection.

Finally, only during the MEMO experiment, the subjects repeated the task wearing left-right reversing prisms to induce a sensorimotor transformation. In other words, while wearing the reversing prisms, when the subject moved the joystick to the left, the cursor was seen to move to the right.

2.5.4 *Data collection and analysis*

The experimental results from MWPE and MEMO were delivered to the investigators soon after the mission, on floppy disks copied from the Grid computer. The results were analyzed using ANOVA (ANalysis Of VAriance) tools with the help of the SYSTAT (Evanston, IL) statistical analysis software package. A necessary step in data analysis turned out to be culling out results that reflected very long reaction or movement times, apparently associated with entirely forgetting the memory set. All subjects were trained prior to being tested so the goal was to evaluate steady-state performance and not learning effects.

2.6 Results

2.6.1 *Comparison of extreme environments*

The first set of results are intended to compare the results seen during the 8-day IML-1 spaceflight mission with those obtained from the 7-day CAPSULS mission. One of the major results of MWPE was that all four subjects showed a significant increase in movement time (MT) during spaceflight as shown in Figure 3 by the white bars. In other words, a *decrease* in fine motor control performance was observed in the microgravity environment. In contrast, there was no significant changes in the reaction time of astronauts' performance on the Fittsberg task (Figure 4).

Distinguishing between changes due to sensorimotor adaptation to the microgravity environment and changes due to the fatigue and high stress of spaceflight was not possible during the MWPE experiment. However, performing MEMO during the CAPSULS mission allowed us to distinguish between these two hypotheses by performing the same experiment under a similar workload environment, without the effects of microgravity. The results of the MEMO experiment during the CAPSULS mission are shown in Figure 5. These results show that in fact MT is not affected by isolation per se. Figure 6 shows the average reaction times (RTs) for each session. Again, no significant change is seen across session. However there are significant differences in the devices that will be discussed in the next session.

2.6.2 *Control strategies*

The second set of results designed to look at the effects of sensorimotor adaptation on human-control strategies. Figure 7a-d shows the movement time responses for each of the astronauts using both control devices. The movement times were consistently slower using the trackball device. Performance using both devices decreased (movement times increased) during spaceflight, but returned to normal upon return.

During the MEMO experiment, the subjects performed the Fittsberg task both in the normal viewing condition and again while wearing prisms to induce a left-right reversed sensorimotor transformation. This allowed us to specifically look at the effects of inducing a visuomotor transformation and evaluated operator performance using the trackball or joystick devices. We were also interested in evaluating the responses depending on target direction. It is worth reporting again that all subjects were trained prior to being tested so the goal was to evaluate steady-state performance and not learning effects.

The relative movement times versus target direction are shown in Figure 8. Recall that the subject's task was to remember the letter presented, and move to one of 8 positions depicted in Figures 8 and 9 as target positions 0-7. The results for the joystick device, or rate-control device, in the normal mode are shown on the left side with the results from the prism-adapted state on the right side of Figure 8. Under the normal condition, the movement times to reach the diagonal targets were significantly slower than for the cardinal targets (shown in the hatched area). In the prism conditions the movement time for the cardinal directions was unchanged from the normal condition. An interesting thing to note is that even the performance for the left-right cardinal positions didn't change in the adapted state even though the control movements were reversed. However, in the prism-adapted state, the movement time for all four diagonal targets was increased from the normal state (The black indicates a significant trend at the $p<0.05$ level).

Similarly, results from the trackball device, or position control, are shown in Figure 9. The normal state showed no significant difference in the movement times for the eight directions. However in the prism-adapted state, as for the joystick device, the movement times for the diagonal target directions are significantly longer than for the cardinal directions. Only directions 1 and 5 were significant at the 0.05 level as depicted in black, while directions 3 and 7 had the same trend.

Figure 10 shows the actual movement time data to compare between control devices and adaptation states. There are two main results: 1) In both up-down and left-right cardinal directions, there are no differences in the movement times between devices or adaptation state. 2) In the diagonal target directions, all cases are significantly different from each other ($p<0.05$). Trackball normal is no different from the other conditions and has the best motor performance. There is a significant increase in movement time, denoting a decrease in performance, while using the joystick. A further increase in MT is seen for the trackball prism adapted state, and the longest movement time is using the joystick in the prism adapted state.

2.6.3 *Models of performance*

2.6.3.1 Reaction time (RT)

As previously described, the selection of a response is based upon the Sternberg exhaustive memory search task and the response execution or target acquisition is base on Fitts' paradigm of motor control. It should be noted that the MWPE/MEMO software alters the classical model for reaction time (recall Equation 2) to the linear expression in Equation 5, modeling reaction time as a linear function of the number of letters in the memory set. This enhancement was incorporated into the experimental protocol to reflect recent findings in cognitive performance (Newman 1988).

$$RT = a + b(n) \qquad (2)$$

where a and b are constants associated with a cognitive overhead and the slope of the RT line for mental processing, respectively. The variable n represents the number of letters in the memory set.

Figure 11a shows the linear regression fits to the data obtained for the four control modes during the MEMO experiment and Figure 11b shows the average linear regression. Table 2.2 is a summary of the regression coefficients from Figure 11a.

Table 2.2: Regression Coefficients for MEMO RTs.

Control Mode	A (msec)	B (*Set Size) (msec)
Trackball (Norm)	129	367
Trackball (Prism)	412	367
Joystick (Norm)	0	541
Joystick (Prism)	224	541

2.6.3.2 Movement time (MT)

Recall from Section 2.4.4 that the Fitts task alone results in a response that is proportional to the Index of difficulty. Interestingly, when performed with in a dual-task environment, movement time is no longer defined in a straightforward manner. In fact, movement time was dependent more on the response to the first task, Reaction Time, that the variables associated with the Fitts' task per se, Index of Difficulty and Direction Code. Table 2.3 shows the regression coefficients for each control mode from the following regression equation:

MT = A + (B * Index of Difficulty) + (C * Direction Code) + (D * Reaction Time), where A, B, C, D are regression coefficients, Bolded at $p < .001$.

Table 2.3: Regression Coefficients for MEMO MTs (BOLD is significant at $p < .001$).

Control Mode	A (msec)	B * ID	C * Dir_Code	D * Reaction Time
Trackball (Norm)	**573**	1	11	**230**
Trackball (Prism)	**417**	35	**170**	**223**
Joystick (Norm)	0	**97**	**270**	**140**
Joystick (Prism)	**443**	**116**	**346**	**81**

2.6.3.3 Overall workload

For the MEMO experiment, linear regressions were fit to the variables that could potentially contribute to overall workload -- prism wearing, set size, index of difficulty, and direction of the target. The regression results show that the following equation holds for all subjects with regression coefficients at $p < .0001$. Table 2.4 shows the coefficients for each subject. Figure 12 shows the overall workload as a function of set-size. The average regression fit for both trackball and joystick are similar.

Overall Workload = A + B (Set Size), where A, B, are regression coefficients ($p < .0001$)

Table 2.4 Regression Coefficients for MEMO Overall Workload

Control Device	Subject	A	B (*Set Size)
TRACKBALL	BT	-24	26
	DW	14	10
	JP	-10	22
	MM	20	8
JOYSTICK	BT	-11	13
	DW	14	10
	JP	-7	21
	MM	26	4

2.7 Discussion

2.7.1 *Comparison to spaceflight*

Human performance was quantified using both a short-term memory task and a fine motor control task. The results obtained during an 8-day spaceflight mission were compared to the results obtained during a 7-day isolation mission where conditions matched the workload and environmental parameters of a space mission, but did not include the sensorimotor effects due to microgravity.

No changes in fine motor control were observed over the course of the seven day CAPSULS mission. Therefore, it is likely that the decrease in fine motor control seen during the IML-1 mission was in fact due to changes in sensorimotor loops from exposure to the microgravity environment, rather than workload or fatigue.

No significant changes were seen in cognitive performance during either mission as measured by the short-term memory task. None of the four subjects tested on the IML-1 mission experienced any symptoms of space motion sickness. This may account for their ability to maintain cognitive performance.

2.7.2 *Control strategies*

The second objective of the MEMO experiment was to consider the effects of sensorimotor transformations on the human operator performance using different control strategies. The experimental paradigm helped to answer the following two questions: 1) How is motor control performance affected by using a rate-control (joystick) versus a position-control (trackball) device? 2) Is motor control affected by adaptation state, either a normal or prism-adapted sensorimotor state.

2.7.3.1 Control device

When using the joystick as a rate-controller to acquire the targets, movement time was slower in the diagonal directions in both the normal and prism-adapted state. However, in the position-control mode, using the trackball, movement time was unaffected by target direction in the normal state. Performance decreased (MT increased) in the diagonal directions, only in the adapted state.

Remembering that these were left-right reversing prisms, the motor control performance in the cardinal directions (i.e., up, down, left, right) was unaffected by control mode or adaptation state. However, movement time to the diagonal targets was affected by *both* control mode and adaptation state.

2.7.3.2 Sensorimotor transformation

Since the subjects were trained subjects, steady-state performance was evaluated in both the normal sensorimotor and prism-adapted sensorimotor conditions. One confounding factor was that left-right reversing prisms actually induce different transformations depending on target direction. The north and south targets are not subject to any transformation while the east and west targets are subject to the equivalent of a 180 degree rotational transformation. The diagonal targets are subject to a 90 degree rotational transformation. The differences in the diagonal and cardinal positions may in fact be due to this phenomenon. If this is true, it is interesting to note that subjects can perform just as well with a 180 degree transformation (transformation in one axis) as in the normal unaltered condition. However, a 90 degree rotational transform (transformation in two axes) results in a decrease in performance. In other words, this would imply some transformation threshold that if exceeded, results in a decrease in human operator performance.

2.8 Conclusions

Models of human performance and computer interface design are important for many applications; Some examples are, establishing a human presence in space, teleoperations, and virtual environment training systems. System design for all of these tasks assumes a knowledge of human performance in the microgravity environment and on earth. This may be one of the first attempts to look at position versus rate-control devices in the context of altered sensory motor loops to evaluate human operator performance. One of the most interesting results was that both the position and rate-control devices had similar relative performances in their prism-adapted state as in the normal condition. In other words, both control devices had a decrease in performance only in the diagonal target directions. This predictability may mean that a human operator model developed from results using simple transformations may be applied to human-computer interface design for more complicated tasks.

2.9 References

Damos, D. L. (1991). Dual-task methodology: Some common problems. Multiple-task performance. Washington, DC, Taylor and Francis. 101-119.

Eggemeier, F. T. and G. F. Wilson (1991). Performance-based and subjective assessment of workload in multi-task environments. Multiple-task performance. Washington, DC, Taylor and Francis. 217-278.

Fitts, P. M. and J. R. Peterson (1964). "Information capacity of discrete motor responses." J. of Exp. Psych. **67**(2):

Gopher, D., M. Brickner, et al. (1982). "Different difficulty manipulations interact differently with task emphasis: evidence fore multiple resources." J. Exp. Psych.: Hum. Percept. and Perform. **8**: 146-157.

Hart, S. G. and L. E. Staveland (1988). Development of NASA-TLX: Results of emperical and theoretic research. Human Mental Workload. Amsterdam: North-Holland, 139-183.

Hartzell, E. J., D. Gopher, et al. (1983). The fittsberg law: The joint impact of memory load and movement difficulty. Proceedings of the Human Factors Society 27th Annual Meeting,

Newman, D. J. (1988). Ground-Based Results of the Mental Workload & Performance Experiment (MWPE). Twenty-Third Annual
Conference on Manual Control, Cambridge, Massachusetts,

Newman, D. J. and S. R. Bussolari (1990). Dual-Task Performance on an Interactive Human/Computer Space Shuttle Flight Experiment. Rocky Mountain Bioengineering Symposium, Denver, Colorado, Newman, D. J., B. K. Lichtenberg, et al. (1993). The Mental Workload & Performance Experiment. International Microgravity Laboratory -1 Final Science Review, New Orleans, Louisiana,

Sheridan, T. B. (1992). Telerobotics, Automation, and Human Supervisory Control. Cambridge, MA, MIT Press.

Shiflett, S. G., P. M. Linton, et al. (1982). Evaluation of a pilot workload assessment device to test alternative display formats and control handling qualities. AIAA Workshops on Flight Testing to Identify Pilot Workload and Pilot Dynamics, Edwards Air Force Base, CA,

Sternberg, S. (1975). "Memory scanning: New findings and current controversies." Quart. J. of Exp. Psych. **27**: 1-32.

Tsang, P. (1986). Display/control integrality and time-sharing performance. Human Factors Society, 30th Annual Meeting, Santa Monica, CA,

Welch, R. B. (1986). Chapter 24: Adaptation of space perception. Handbook of human perception and performance, Volume 1: Sensory processes and perception. New York, John Wiley and Sons.

Wickens, C. D. (1991). Processing resources and attention. Multiple-task performance. Washington, DC, Taylor and Francis. 3-34.

Wickens, D. and D. Gopher (1977). "Control theory measures of tracking as indices of attention allocation strageties." Human Factors **19**: 349-365.

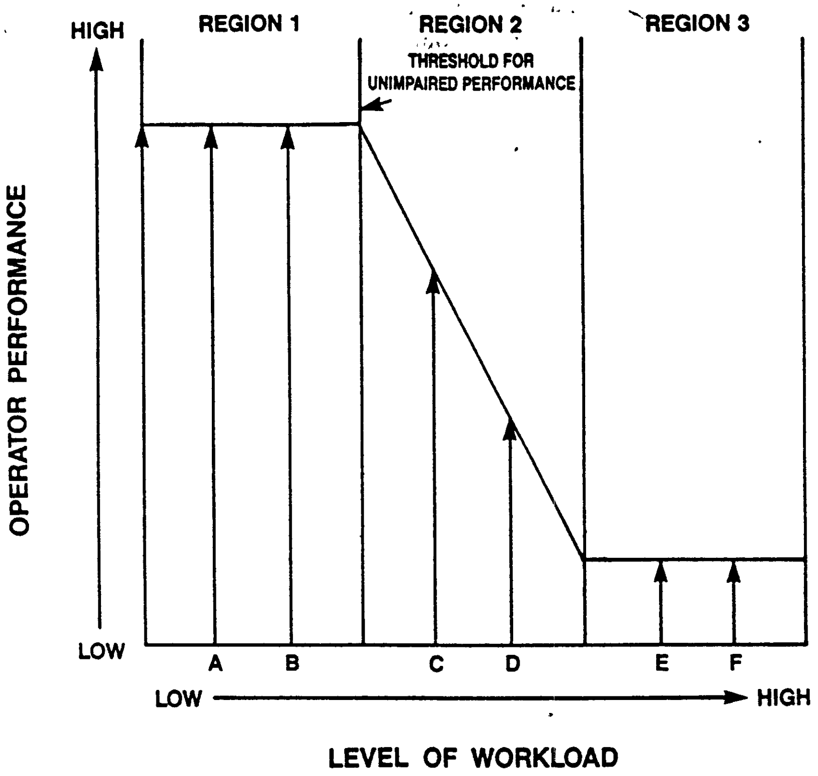


Figure 2.1: Hypothetical relationship between primary task performance and operator workload (Damos, 1990).

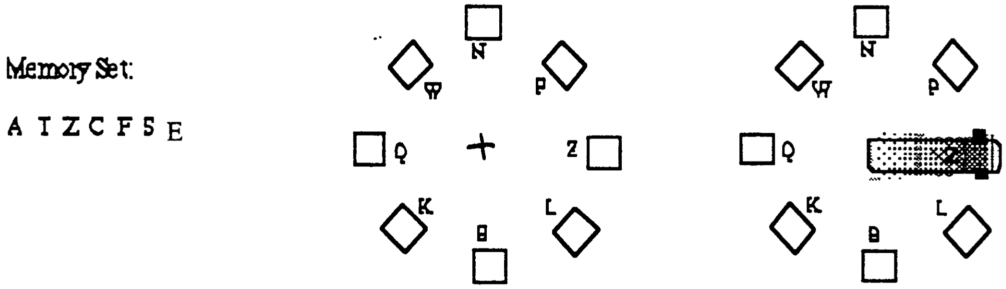


Figure 2.2: The basic paradigm was the same for MWPE and the MEMO experiments. A) The subject was presented with a memory set consisting of 1-7 letters which they were asked to memorize. The subject pressed return on the keyboard to indicate the end of the memorization time. B) The subject was then immediately presented with a test stimulus with a cursor in the center. Only one of the letters from the memory set is presented, the letter Z in this case. C) As soon as the subject spotted the letter, the subject moved the cursor to that location. Once the location is reached, a new test stimulus appears immediately.

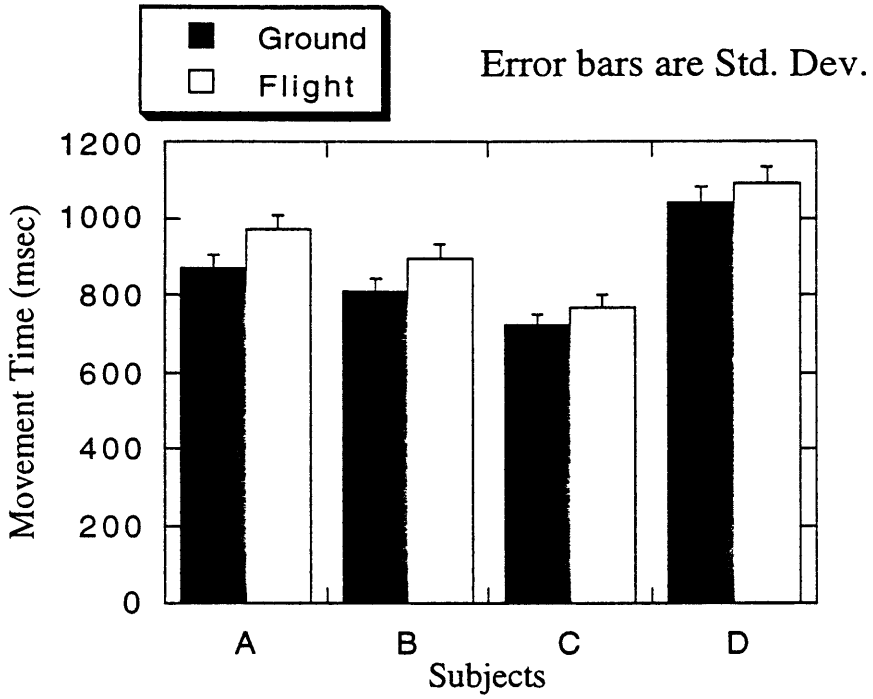


Figure 2.3: MWPE data from ground and spaceflight experimentation. All 4 subjects showed an increase in movement time (p < .001).

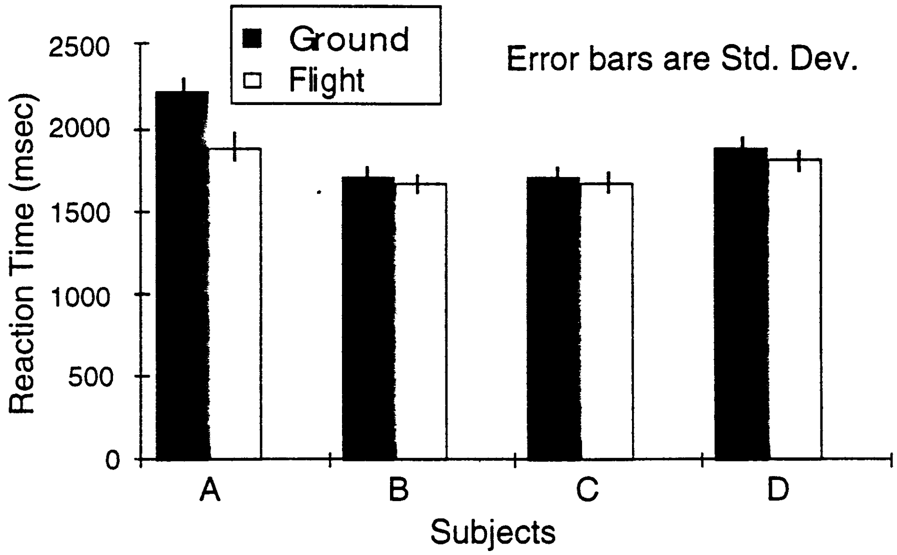


Figure 2.4 Mean reaction times preflight and during spaceflight for each subject. Standard error of the means are shown.

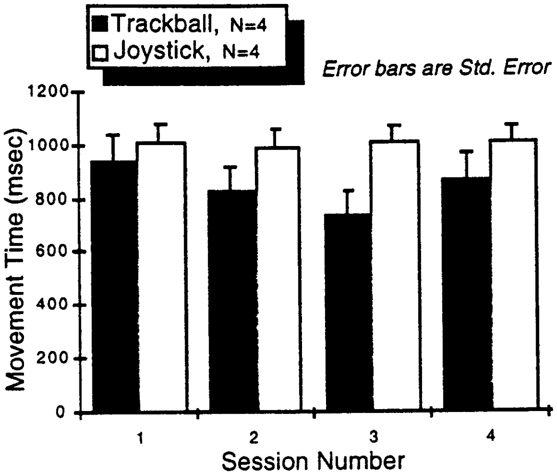


Figure 2.5: Movement times averaged over 4 subjects for each session. Both devices showed no change during the CAPSULS mission.

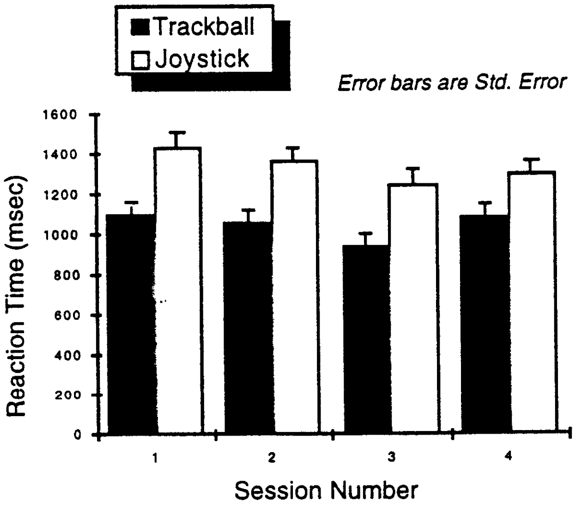


Figure 2.6: Reaction times also showed no changes during the CAPSULS mission for both devices.

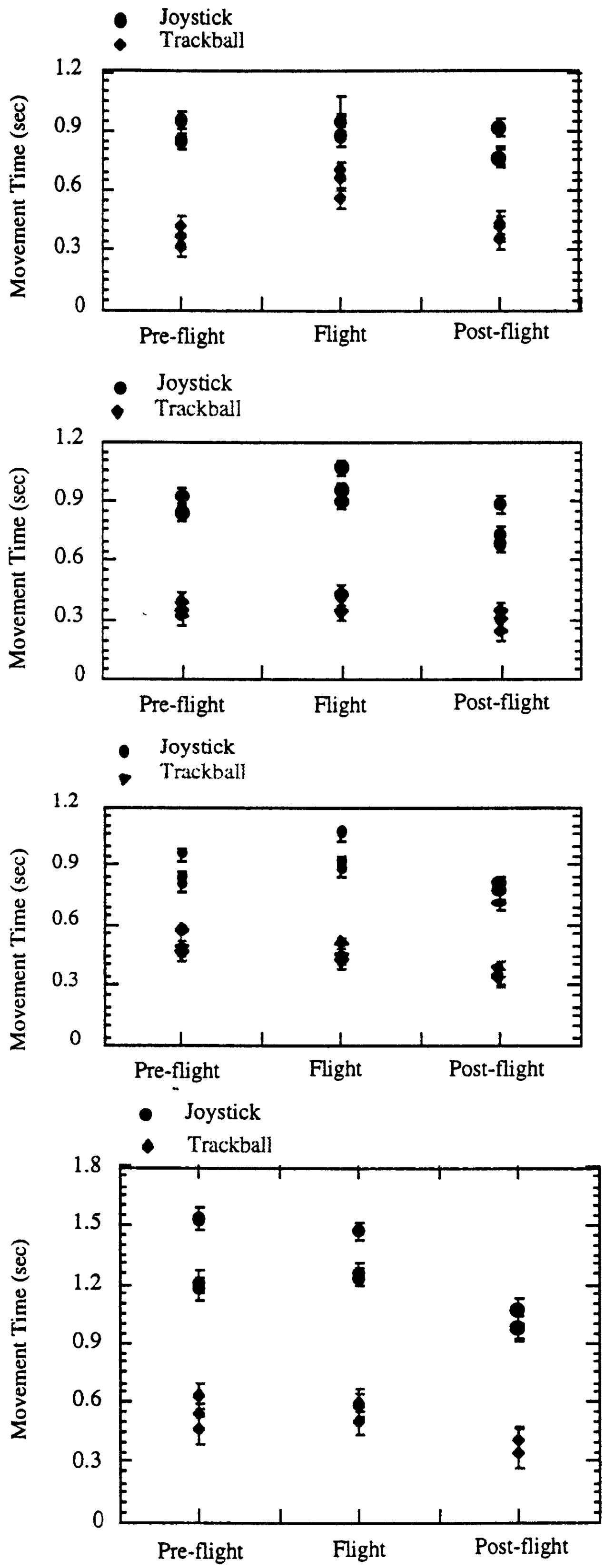
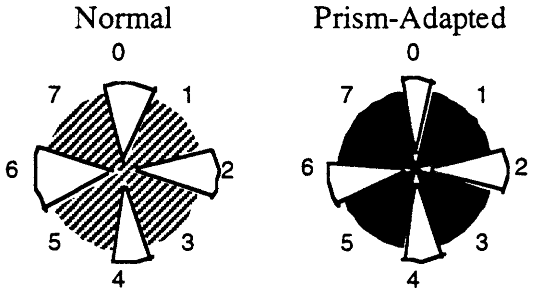


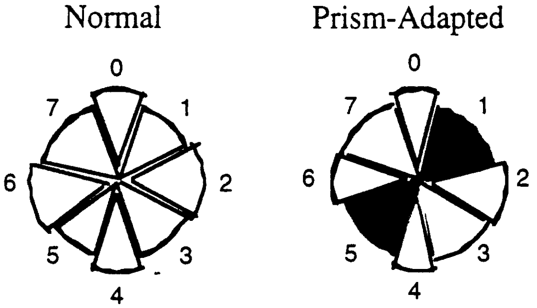
Figure 2.7: MTs for each of the astronauts using both control devices.



	NORMAL	PRISM
Cardinals (0,4):	MT ≈ 700	MT ≈ 1010 msec
Cardinals (2, 6) :	MT ≈ 860	MT ≈ 1140 msec
Diagonals (1,3,5,7) :	MT ≈ 1120	*MT ≈ 2320 msec

* $P<0.05$
Std. Error ≈ 100 msec

Figure 2.8: Joystick. Diagonal targets (hatched areas) resulted in slower MTs relative to the cardinal directions. Only MTs to the diagonal targets were affected by prism-adaptation (filled areas).



	NORMAL	PRISM
Cardinals (0,4):	MT ≈ 710	MT ≈ 820 msec
Cardinals (2, 6) :	MT ≈ 1020	MT ≈ 1090 msec
Diagonals (1,3,5,7) :	MT ≈ 810	*MT ≈ 1450 msec

* $P<0.05$
Std. Error ≈ 100 msec

Figure 2.9: Trackball. Only diagonal targets 1 and 5 showed any significant increase in MTs.

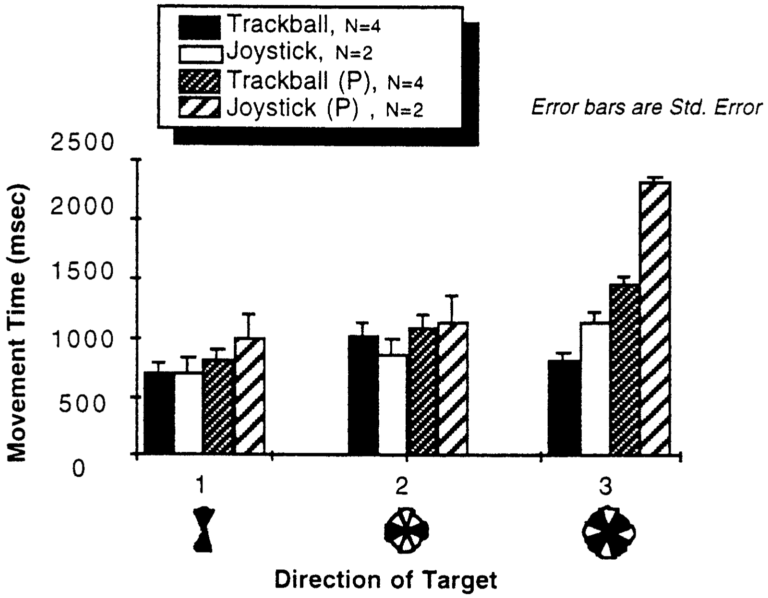


Figure 2.10: There were no significant differences in the cardinal positions for either devices or adaptation state. Only the diagonal target directions showed significant increased in movement times.

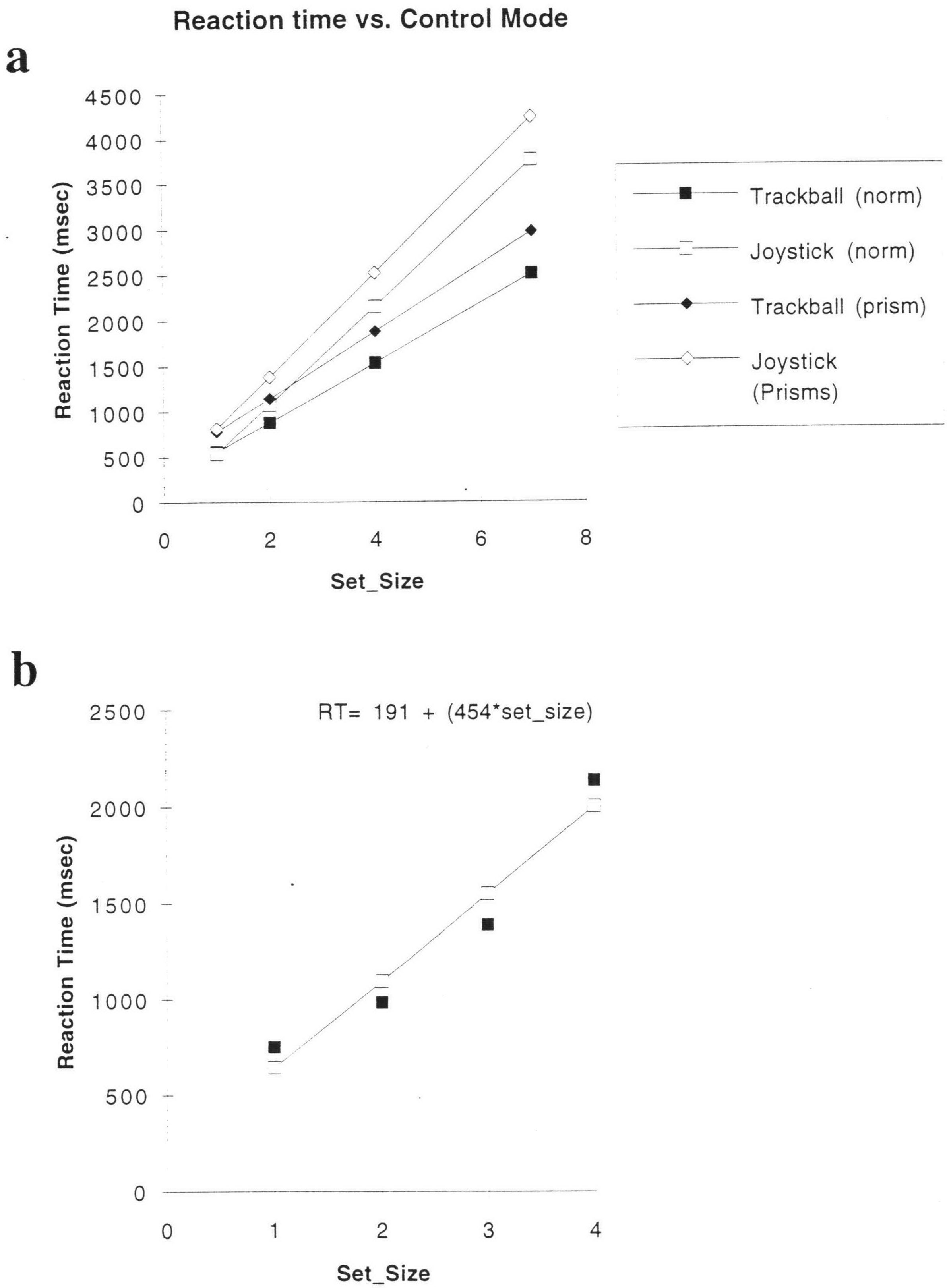


Figure 2.11: a) Linear regression fits to the RT data obtained for the four control modes during the MEMO experiment. b) The average linear regression model.

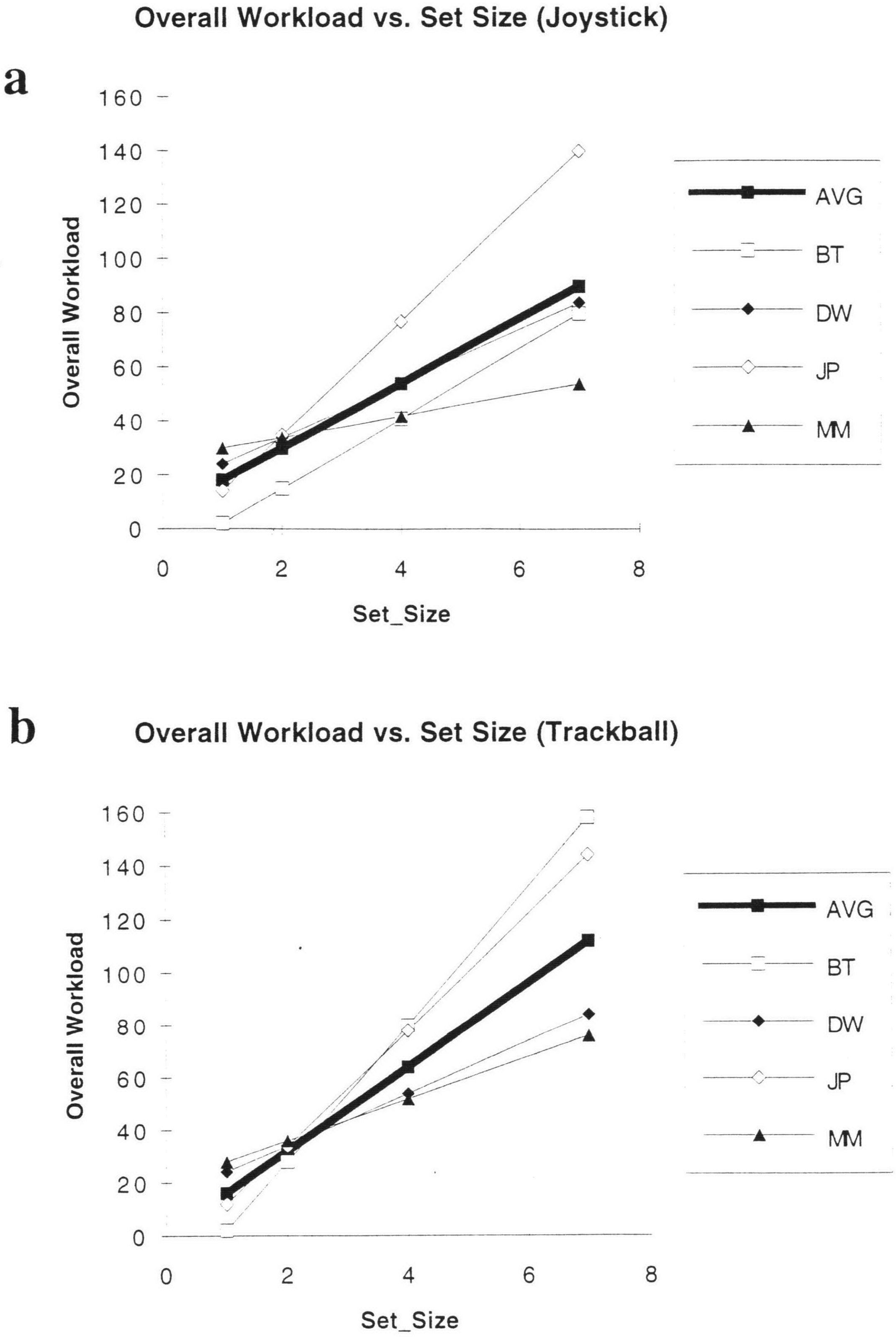


Figure 2.12: The overall workload as a function of set-size for the joystick (a) and the trackball (b).

CHAPTER 3

SENSORIMOTOR ADAPTATION OF HUMAN CONTROL STRATEGIES

In Chapter 3, I address two research topics that are of interest in evaluating human performance for design of human-machine interfaces (HMIs), Sensorimotor adaptation and Manual Control Strategies. Most everyday activities require that people make some sensorimotor transformation between motor activity and visual feedback. Using a horizontal computer mouse to control a cursor on a vertical computer screen requires a 90 degree visuomotor transformations. Even this simple transformation requires some sensorimotor adaptation of the human operator's normal sensorimotor loops. Extreme examples of visuomotor transformations are virtual reality human machine interfaces for computer screens and video games. Other examples include induced transformations for teleoperator activity when human performance must be improved to perform tasks that may be beyond our normal physiological capabilities e.g. microsurgery, or handling of toxic waste (Durlach and Mavor 1995).

Manual control refers to the closed-loop control of some system by a human operator through feedback about the desired state of the system. Evaluation of manual control strategies are important for designing HMIs for fully and partially-automated tasks. How to implement controllers for telerobotic applications is dependent on what we know about the human operator's ability. All of these controllers induce some sort of sensorimotor transformation so it is important to understand and quantify human performance using different control strategies under different demands of sensorimotor adaptation.

These experiments compare human performance using position-control or rate-control to execute a radial Fitts motor task (Fitts and Peterson 1964) under induced sensorimotor transformations. Both position and rate control devices have been used extensively in teleoperations and in the laboratory to model human performance (Kim, Tendick et al. 1987; Bejczy, Hannaford et al. 1988; Das, Zak et al. 1992; Massimino and Sheridan 1994). Position-based teleoperation uses a remote manipulator that is servo-controlled to follow the operator's position commands. Rate control devices interpret human hand displacements as velocity commands. Sensorimotor transformations are systematically induced to compare the learning rates, or time constants of adaptation, using each control strategy. Rotational transformations of the visual field are induced from 5 degrees to 60 degrees. However, first this chapter will provide some background on research issues in sensorimotor adaptation and manual control for human-machine interface applications.

3.1 Sensorimotor Loop Adaptation for HMIs

Sensorimotor integration in systems physiology refers to the process by which the central nervous system converts sensory inputs into a motor output. In almost any HMI, there will be some alteration of our normal sensorimotor loops. Some alterations may be intended to increase performance and some may be a result of noise or technical limitations. There is a need therefore, to study sensorimotor loop adaptations in order to predict responses of human performance to alterations.

Some of the applications of human-machine interfaces such as in telerobotics and virtual environments share common threads. Both systems must have sufficient sensory and reactive capability to successfully translate information and interact with the environment. All of this must occur through mediating technologies that provide sensory feedback and control.

There are some distinctions in that virtual environments are simplifications of the real-environment and as such must deal with complexities of 3-d sensorimotor integration. Simulating real world depth and size perception can be a challenge (Rolland, Gibson et al. 1995); There are effects of computer lag on performance (Ware and Balakrishnan 1994); and our own visual system limitations can cause a problem (McKenna and Zeltzer 1992). Flight simulators are examples of such virtual environment systems that are trying to mimic real-world applications but run into problems know as simulator sickness (Pausch, Crea et al. 1992).

However, both virtual reality and teleoperator systems will induce alterations in the operator's sensorimotor loops and so the perceptual and cognitive capabilities and limitations of human beings must be used as driving criteria for system design. Theoretical models must be developed for characterizing human responses to the alterations associated with use of such systems. Models should predict effect of alterations on subjective response as well as objective task performance and describe how response changes over time. Even the transformations within the x and y dimensions of space may not be independent of each other and so pose a challenge to modeling human performance (Bedford 1989; Ghahramani, Wolpert et al. Submitted). A review of sensorimotor transformation is provided by (Welch 1978), but there are no models predicting performance. A need has been defined for a model that 1) specifies the amount of sensorimotor adaptation that is achievable with different kinds of distortions using different types of training procedures 2) one that addresses adaptation to changes in resolution (i.e.. ability to separate and sense different signals) as well as bias (i.e. deviation b/n mean response and correct response) and 3) address interactions b/n different kinds alterations (distortions, time delays, noise) (Durlach and Mavor 1995).

3.2 Manual Control Issues in HMI Design

With advances in vehicle automation, the role of the human operator has evolved from that of pilot/driver and manual controller to supervisor and decision maker. As discussed in Chapter 2, the projected operational requirements of the International Space Station will necessitate extensive automation and expansion of the supervisory role for its crewmembers. However, manually controlled telerobots remain important for activities which raise issues of astronaut safety in space and or during medical applications which require superhuman performance. Manual control issues are important due to such concerns that a telerobot not damage the space station structure or that a telesurgery device actually improves the surgeons performance and not put the patient under unacceptable risk.

Manual control is simply the receipt of sensory information about the desired state of system and its current state by a human operator and the use of that information by the operator to command inputs to the system through mechanical devices. Again, we experience simple issues in control during everyday activities; For example, you can set the speed of your computer mouse which maps the velocity to screen distance. Telerobotic systems are usually closed-loop in that the system being controlled takes into account the behavior of the human operator and feeds backs it's response which is then acted upon by the operator. The development of human-machine interfaces for telerobotic systems range from simple analog or symbolic controls to highly immersive to provide operator with some sense of sensor feedback (Sheridan 1992). Their purpose is to minimize task completion times and training required to operate remote system.

Teleoperators are systems composed of a human operator, master manipulator (joystick), communication channel, slave manipulator, and the environment (remote task). The simplest teleoperator is a remote manipulator that is servo-controlled to follow the operator's position commands. However, the volume of space comfortable for an operator to maintain hand position for extended periods is small compared with work volume of a human arm; hand controllers cover less area than the robot, so at times, the operator needs to break contact and reposition, changing the offset (called indexing).

Another frequently used form of control of a remote manipulator is resolved rate control (Whitney 1969). Human hand displacements are interpreted as a velocity command in an assigned Cartesian frame. This is used in the space shuttle remote manipulator system (RMS). One three-degree-of-freedom joystick is used for orientation commands. and one for translation. A six-axis hand rate controller has also been used for teleoperations (Bejczy, Hannaford et al. 1988). One important requirement for rate control joysticks is a spring return implemented with hardware or software force commands (Bejczy, Hannaford et al. 1988); this passive force feedback is essential for easily stopping the commanded motion.

A significant issue is which mode is better for teleoperations without force feedback. Position control gave better completion times in simulated teleoperations except for very slow simulated manipulators, for which rate control was slightly better (Kim, Tendick et al. 1987). It is widely thought that for large displacements, rate control can be better because it eliminates the need for repeatedly indexing. However, performance degradation occurs when there are significant rotations between the rate controller's frame and the command frame defined by user's body (Kim, Tso et al. 1993) (e.g. if left-right motion caused end effector motion that is visibly different). This problem is bad for control of orientation when rate commands are referenced to rotation axes fixed to the robot hand although methods have been developed to help account for this such as using a control law that uses a deadband (Parker, Salcudean et al. 1993).

Novel modes of rate control, transitions between rate and position control, and relative performance between rate and position control are unresolved issues that impact application design. Measures of performance evaluation include time of task completion, accuracy, and error (Sheridan 1992).

Kim et al (1987) found that when a workspace is small, position control is better by measures of completion time and accuracy. With larger workspaces or slow manipulators, rate control becomes superior to position control. Mixed results have been found on other tasks with position control superior in some cases (Das, Zak et al. 1992), but on a 6-d task isometric (pure force) rate control was as good as isotonic position control (Zhai and Milgram 1993a).

In order to set and research questions for design of telerobotic systems, we need to know how complicated tasks decompose into more basic tasks, which can then be measured, e.g. when apply rate control vs. position control or how to include force feedback into rate control which has been shown to improve performance (Hannaford, Wood et al. 1991; Das, Zak et al. 1992) depending on stiffness (Jones and Hunter 1990;

Jones and Hunter 1992; Zhai and Milgram 1993a). Isometric and isotonic are extreme ends of continuum of variable-stiffness hand controllers (infinitely stiff versus infinitely pliant).

3.3 Research Facility at the Research Laboratory of Electronics

All testing was done in the facilities of the Sensory Communications group of the Research Laboratory of Electronics which is part of the Department of Electrical Engineering and Computer Science at M.I.T.

Subjects were obtained from the student population through bulletins placed around campus. All experimental protocols were approved by the MIT Committee on the Use of Humans as Experimental Subjects.

The experimental equipment consisted of a video monitor and an in-house developed passive manipulandum arm operated by a subject's hand. A half-silvered mirror covered the subject's hand and the subject looked through a goggle-like window which blocked peripheral vision. The manipulandum was fit with a spring-loaded frame for use during the rate control conditions.

The experiment was executed by software written in C++, developed by students in the sensory communications group. Appendix A shows one script and highlights the crucial sections that were modified for these experiments to implement velocity as well as position control. Data was collected and stored on a Macintosh computer. Each session consisted of several conditions. Each condition consisted of 20 blocks of 10 targets that were completed without a break. Each set of 20 blocks was fit with an exponential and confidence intervals were found for each parameter. The Matlab script developed for such analysis is shown in Appendix B. The equation that was used for parameter fitting was the following:

Response (e.g. reaction time) = K1 (theta) + K2 (theta) * exp (t*K3(theta))

K1 can be thought of as the steady state response and 1/K3 is the time constant of sensorimotor adaptation (although we will just refer to the time constant as K3).

A radial Fitts task (Fitts and Peterson 1964) was used for all three experiments. Subjects were asked to move a cursor, as quickly and accurately as possible, to a target which appeared in one of eight locations, North, Northeast, East, Southeast, South, Southwest, West, and Northwest. After each block of 10 trials, the subjects received feedback through a score which appeared on the screen. Their instructions were to minimize the score which was based on time and accuracy. The subjects were also instructed to hold the manipulandum between their thumbs and forefingers and not to rest their arms on the table.

3.4 Experiment 1: Diagonal vs. Cardinal Target Directions

One issue that arose from the results of the MEMO experiment was whether or not target direction affected steady-state performance as measured by movement time to target (TTT). Figure 2.10 from the MEMO experiment showed two results: 1) Using the joystick for velocity control results in slower movement times in the diagonal directions both under normal and prism-adapted conditions; under prism-adapted conditions, trackball performance also was deteriorated in the diagonal directions. 2) For the cardinal directions, neither control mode or adaptation-state played a role in performance.

Experiment A was designed to test the hypothesis that diagonal and cardinal target locations resulted in different operator performances. Experiment B was designed to test a second hypothesis proposed in Chapter 2 that the differences in diagonal vs. cardinal performance during prism adaptation were due to the fact that the prisms induced a 180 degree rotation for the cardinal directions (E and W), but a 90 degree rotation for the diagonal directions.

Experiment A: This experiment tested movement time performance to diagonal and cardinal targets under steady-state conditions with no sensorimotor transformations. Three subjects were tested under four different conditions:

1) Using velocity-control with diagonal target locations (NW, NE, SW, SE)
2) Using velocity-control with cardinal target locations (N, S, E, W)
3) Using position-control with diagonal target locations (NW, NE, SW, SE)
4) Using position-control with cardinal target locations (N, S, E, W)

Table 1 shows the average and standard deviation of the TTT for each target condition (T), control condition (C), and subject (S). Figure 1 shows the results of this experiment for all three subjects. Two results are of interest. The first is that there were no differences between the responses to cardinal versus diagonal targets. Secondly, movement times during velocity-control are about .75 seconds slower than during position-control. Both of these results are contrary to what was expected given our hypotheses based on the MEMO results that diagonal and cardinal target locations would result in different operator performances, but TTTs to cardinal target locations would be unaffected by control mode.

Table 3.1: Average and standard deviation of the Time to Target (TTT) for each target condition (T), control condition (C), and subject (S) *Experiment 1A.*

T/C/S	TTT AVG.	TTT STDEV
dv1	2.24	0.15
cv1	2.75	0.28
cp1	1.19	0.15
dp1	1.34	0.15
dv2	2.04	0.20
cv2	2.18	0.63
dp2	1.11	0.63
cp2	1.11	0.41
dv3	2.26	0.41
cv3	2.35	0.15
cp3	1.17	0.39
dp3	1.22	0.15

There are several reasons for the differences in the experimental results, primarily related to equipment differences. 1) The experiments in this chapter used the same manipulandum for both control modes instead of using a joystick and trackball as in Chapter 2. 2) The gain of the spring-loaded manipulandum and joystick were most likely not comparable. 3) Finally, the MEMO experiments were done with a powerbook with a screen perpendicular to the floor, whereas the RLE experiments used a planar screen parallel to the floor. These differences were also reflected in the following second experiment.

Experiment B: This experiment tested movement time performance when subjects had to adapt to a 180 degree visuomotor transformation for either diagonal or cardinal target locations. Two subjects were tested for both target directions under velocity-control mode only. Each subject was tested using EITHER diagonal or cardinal targets, and ran 4 conditions of trials in the following order:

1. Normal sensorimotor conditions (0 degrees rotation).
2. 180 degree rotational sensorimotor transformation.
3. 180 degree rotational sensorimotor transformation.
4. Normal sensorimotor conditions (0 degrees rotation).

Table 2 shows the parameters of the exponential fits for each condition for each subject. Figured 2a-d shows the time constant parameter normalized and plotted for each subject. Subject variability played a much larger role than target direction in the learning rates.

Figure 3 shows the steady-state TTT performance. There are no significant difference between cardinal and diagonal movement times when both are subjected to the same 180 rotation. Thus, the hypothesis that the difference in movement times for diagonal vs. cardinal directions during the MEMO experiment may have been due to the difference in sensorimotor transformation induced, is not disproved.

Table 3.2: Parameters of the exponential fits to Time to Target for each subject in *Experiment 1B*. The first two letters under "Subject" are initials and the then c=cardinal target directions or d=diagonal target directions. Each subject was run in 4 conditions of 200 trials. Each condition is labeled in degrees rotation.

		K1		K2		K3	
Subject	cond.	AVG.	stdev	AVG.	stdev	AVG.	stdev
sgd	0	1.74	0.25	0.30	0.57	-0.54	1.27
sgd	180	2.22	0.24	1.35	0.57	-0.52	0.26
sgd	180	1.95	0.23	0.06	0.56	-0.46	4.95
sgd	0	1.70	0.35	1.22	0.55	-1.01	0.68
gsd	0	1.71	0.15	0.57	0.36	-0.09	0.10
gsd	180	1.86	0.15	2.16	0.35	-0.09	0.02
gsd	180	4.51	0.15	-2.45	0.23	0.00	0.01
gsd	0	1.66	0.27	0.84	0.58	-0.63	0.54
bsc	0	*1.86*	0.15	9.76	0.23	*-1.08*	0.00
bsc	180	2.49	0.63	1.74	0.11	*-2.85*	na
bsc	180	1.99	0.47	3.24	0.44	-1.78	0.53
bsc	0	1.71	0.56	0.72	0.30	-2.77	4.54
mgc	0	2.06	0.23	0.19	0.56	-0.47	1.64
mgc	180	2.72	0.20	4.14	0.54	-0.35	0.05
mgc	180	2.44	0.16	0.43	0.48	-0.22	0.29
mgc	0	2.27	0.16	2.43	0.46	-0.19	0.04

3.5 Experiment 2: Position vs. Velocity Control Modes

Using a 60 degree rotational sensorimotor transformation, the adaptation rates and steady-state performance of movements to targets were compared under the two control modes. In addition to time to target measurements, I also measured distance to target (DTT) values during adaptation and during steady-state performance. Five subjects were tested during 6 conditions for BOTH velocity and position control modes.

1. Normal sensorimotor conditions (0 degrees rotation).
2. **60 degree rotational sensorimotor transformation.**
3. 60 degree rotational sensorimotor transformation.
4. 60 degree rotational sensorimotor transformation.
5. Normal sensorimotor conditions (0 degrees rotation).
6. Normal sensorimotor conditions (0 degrees rotation).

Table 3.3: Parameters of the exponential fits to the Time to Target (TTT) responses and Distance to Target (DTT) responses for each subject in *Experiment 2*. All fits are to the first 60 degree rotation adaptation block. For "Control," P=position control mode, V=velocity control mode.

		TTT	TTT	TTT	TTT	TTT	TTT	DTT	DTT	DTT	DTT	DTT	DTT
Subject	Control	K1	stdev	K2	stdev	K3	stdev	K1	stdev	K2	stdev	K3	stdev
1	P	1.22	0.32	1.13	0.56	-0.86	0.60	2.86	0.24	0.41	0.57	-0.52	0.88
2	P	0.62	0.16	0.32	0.47	-0.21	0.37	3.73	0.38	1.38	0.52	-1.21	0.75
3	P	0.79	0.23	0.63	0.56	-0.46	0.50	3.70	0.41	0.92	0.50	-1.37	1.31
4	P	0.63	0.17	0.25	0.49	-0.24	0.52	3.30	0.23	2.44	0.57	-0.48	0.13
5	P	0.62	0.32	0.36	0.57	-0.85	1.84	3.11	0.16	0.90	0.48	-0.23	0.14
1	V	2.60	0.35	1.99	0.55	-1.02	0.42	3.97	0.43	0.83	0.48	-1.49	1.63
2	V	2.10	0.15	1.55	0.38	-0.11	0.04	4.51	0.17	3.33	0.50	-0.26	0.05
3	V	2.33	0.20	1.96	0.55	-0.38	0.12	4.78	0.33	1.79	0.56	-0.91	0.41
5	V	1.77	0.23	1.03	0.56	-0.47	0.30	3.79	0.45	3.40	0.46	-1.65	0.45

Table 3.4: Parameters of the exponential fits to the Time to Target (TTT) responses and Distance to Target (DTT) responses for each subject in *Experiment 3*. All fits are to the first 60 degree rotation adaptation block. For "Control," P=position control mode, V=velocity control mode.

Subject	Control	Order	TTT K1	TTT stdev	TTT K2	TTT stdev	TTT K3	TTT stdev	DTT K1	DTT stdev	DTT K2	DTT stdev	DTT K3	DTT stdev
C1	P	3	1.28	0.38	0.35	0.53	-1.16	2.82	3.46	0.17	3.40	0.49	-0.24	0.04
C2	P	7	1.33	0.24	0.59	0.57	-0.50	0.58	3.81	0.17	1.16	0.50	-0.26	0.13
D1	P	1	1.10	0.58	0.06	0.27	-3.01	58.23	*3.40*	0.15	22.04	0.23	0.00	0.00
D2	P	5	*1.28*	0.15	1.55	0.23	0.00	0.02	3.40	0.54	0.17	0.33	-2.52	16.72
C1	V	7	3.00	0.52	0.81	0.37	-2.23	2.89	4.86	0.63	1.94	0.09	-5.29	6.45
C2	V	3	3.11	0.42	3.40	0.49	-1.43	0.38	5.82	0.36	8.89	0.54	-1.08	0.10
D1	V	5	1.59	0.29	-0.03	0.57	-0.72	16.55	2.19	0.15	1.55	0.24	0.00	0.02
D2	V	1	1.94	0.16	0.27	0.46	-0.19	0.39	3.80	0.16	0.10	0.46	-0.19	1.05

Figure 4 shows the sample responses and curve fits to the TTTs for all 6 conditions. Figure 5 shows the sample responses and curve fits to the DTTs for all 6 conditions. The most interesting condition is the second condition, or the first induced 60 degree rotational sensorimotor transformation. Learning rates and steady-state performances for the velocity versus position control conditions are best studies by focusing on condition 2. Table 3 shows the parameters of the exponential fits for all 5 subjects

Figures 6a and 6b compare the time constants of the TTT fits for the velocity versus the position-control condition. One subject is missing data for the velocity control condition. Three of the 4 subjects with both position and velocity control data have slower adaptation time constants during velocity control than during position control. Figure 7a and 7b compare the time constants for DTT values. Two subjects had slower time constants for velocity control and two had faster time constants of adaptation as compared with position control. None of these differences were significant.

However, steady-state performances for both TTT and DTT were significantly different when comparing position versus velocity control. Figure 8 shows the steady-state TTT and DTT as reflected by the K1 parameter for all subjects. TTTs were slower ($p < .001$) and DTTs longer ($p < .01$) during movements made under velocity control .

3.6 Experiment 3: Modeling Sensorimotor Integration

Experiment 3 was aimed at developing an experimental paradigm for modeling sensorimotor adaptation during different control strategies. The hypothesis was that by systematically adapting to progressive visuomotor rotational transformations (5 degrees - 60 degrees), a set of functions would emerge that would characterize adaptation. For example, for each control strategy, an equation could be formed:

TTT (using position or control) = K1 (theta) + K2 (theta) * exp (t*K3(theta))

or each parameter would be some function (f(theta)) that was control strategy dependent:

$$K_1 = \overline{K}_1 * f(\theta) + \overline{\overline{K_1}}$$

K1 = K1 (e.g. control dependent) * f(theta) + K1 (e.g. subject variability?)

and

K2 = K2 (e.g. control dependent) * f(theta) + K2 (e.g. subject variability?)

So 8 subjects were run on 7 conditions in 2 sessions, 1 for velocity control and 1 using position control.

1. Normal sensorimotor conditions (0 degrees rotation).
2. R1 degree rotational sensorimotor transformation.
3. Normal sensorimotor conditions (0 degrees rotation).
4. R2 degree rotational sensorimotor transformation.
5. Normal sensorimotor conditions (0 degrees rotation).
6. R2 degree rotational sensorimotor transformation.
7. Normal sensorimotor conditions (0 degrees rotation).

where R1, R2, and R3 were one of four possible conditions

1. 5 degrees, 10 degrees, 15 degrees
2. 15 degrees, 10 degrees, 5 degrees
3. 10 degrees, 30 degrees, 60 degrees
4. 60 degrees, 30 degrees, 10 degrees

In order to test if this would be a valid paradigm, we needed to compare the results obtained for the 60 degree rotation during this experiment with Experiment 2 where only a 60 degree rotation was used and there was no potential for learning from the other rotations. Table 4 shows the parameters for the exponential fits for all subjects during the 60 degree rotation trials. This can be compared with Table 3.

First, to compare the steady-state performances, the K1 parameter was plotted in Figure 9. In this figure, the TTTs are significantly different depending on the control mode ($p < .05$), but the DTTs are not. When compared with the population of subjects in Figure 8, the TTTp condition is different for the two populations ($p < .01$). This implies that the test conditions are not the same at least for the steady-state parameter.

Secondly, to compare the adaptation rates, Figures 10a-b show the time constant (K3) for each subject's TTT. When compared with Figure 6 in Experiment 2, the main difference is that in Figure 10 shows shorter time constants. In other words, the subjects are learning much faster and in some cases, there is no learning curve. The same can be said when comparing the DTTs in Figures 11a-b with Figure 7 of Experiment 2.

Figure 12 shows the average TTT time constant for each rotation condition. The time constants for position control are shorter or equal to velocity control. There isn't a simple function of time constant dependent on rotation and control mode, but part of this may be because subject variability isn't taken into account. A pattern is even less obvious in Figure 13 which shows the average DTT time constants.

Figures 14a-b show the steady-state performance as seen through the K1 parameter. The velocity control mode TTT steady-state performance is consistently slower than the position performance, but interestingly, not dependent on degree of rotation. The DTT steady-state values also do not depend on degree of rotation.

3.7 Discussion

Defining an object's location in space in intrinsic or body-centered coordinates for reaching requires an integration of sensory and motor information and a formation of internal maps. When an unusual Sensorimotor transformation is induced, there are many ways that a new map could be encoded. Understanding how the original maps are formed is a crucial step in understanding how new maps could be formed.

Making an arm movement to a target in space, involves many complicated steps. The first involves a transformation from extrinsic coordinates which define an object's location in space relative to the individual to an intrinsic or body-centered frame of reference. The second transformation is one from a given kinematic map to the necessary dynamics for the generation of movement. Finally, the required dynamics must be translated into actual muscle activations and executed. Two questions that arise are, what kind of interactions or transformations are needed between different sensory modalities; particularly, how does vision and kinesthetic senses interact to form a common reference frame? And secondly, what are the anatomical and physiological building blocks for visuomotor sensorimotor integrations.

Human psychophysical studies have tried to answer the question of what coordinate framework reaching movements are coded in and what are the transformations between sensory information and motor commands (Soechting and Ross 1984; Soechting and Flanders 1989a; Flanders and Soechting 1990; Soechting, Tillery et al. 1990). Soechting and Flanders (1989a) studied the errors subjects made in pointing to remembered locations of a visually-specified target and tried to determine the underlying sensorimotor process. They reported that subjects could accurately use a visual representation OR kinesthetic representation to locate the direction of a target in space. However, there were large errors in the amplitude of movements. In further examination of these errors (Soechting and Flanders 1989b), they observed a linear relationship between the orientation angles of the arm at its final position and the extrinsic coordinates of the target during inaccurate movements and a nonlinear relationship during accurate movements. They concluded that subjects implement a linear approximation to the transformation from visual to kinesthetic coordinates in the process of deriving muscle patterns.

Tillery et all (1991) looked at movements to kinesthetically defined locations by movement a subject's unseen finger (Tillery, Flanders et al. 1991). The spatial location of the hand was given by kinesthetic inputs which is quite difference from a visually coded target location. One might expect that since kinesthetic information is direction related to the orientation of the hand, this information could then be used to map the current position of the hand to a coordinate frame of extrapersonal space that would then be matched to the desired hand location in extrapersonal space (which was specified by purely kinesthetic cues) and the error signal generated.

An alternative would be that kinesthetic information is transformed to a reference frame more similar to the initial visual representation of the target. In this case, the spatial location of the target could be predicted using only kinesthetic cues. This would be reasonable since kinesthetic information is directly related to the orientation of the arm and this information could then be used to map the end position of the arm to a coordinate frame of extrapersonal space that would then be matched to the target location in extrapersonal space and the error signal generated. This hypothesis might predict that subjects in our experiments would never adapt to our rotations in the visuomotor field since kinesthetic cues would always be generating an error signal. This result is clearly not the case. In fact it seems that adaptation happens very quickly which would indicate that the kinesthetic cues are ignored.

The results from Tillery et al, also indicate that kinesthetic information alone cannot accurately specify the target location in space as the subjects could not reliably use a pointer to indicate where their unseen hand had been passively displaced. The subject could use a pointer to indicate a visually specified target and could reproduce the target location with active movements of their arm. So the errors resulted because the subjects were not good at utilizing kinesthetic information to specify a spatial representation of the hand (or target). Thus they conclude that visually derived information is put into a kinesthetic reference frame and this common frame gives rise to the motor error signal specifying the desired arm movement direction. This is consistent with the physiological evidence of motor cortex cells coding for the direction of arm movement in space (Georgopoulos, Schwartz et al. 1986; Georgopoulos and Grillner 1989).

3.8 References

Bedford, F. (1989). “Constraints on learning new mappings between perceptual dimensions.” J. Exp. Psych. **15**(2): 232-248.

Bejczy, A. K., B. Hannaford, et al. (1988). Multi-mode manual control in telerobotics. In Proceedings of Romany '88, Udine, Italy,

Das, H., H. Zak, et al. (1992). "Operator performance with alternative manual control modes in teleoperation." Presence **1**: 201-208.

Durlach, N. I. and A. S. Mavor, Ed. (1995). Virtual Reality: Scientific and Technological Challenges. Washington, D.C., National Academy Press.

Fitts, P. M. and J. R. Peterson (1964). "Information capacity of discrete motor responses." J. of Exp. Psych. **67**(2):

Flanders, M. and J. F. Soechting (1990). "Arm muscle activation for static forces in three-dimensional space." J. Neurophys. **64**(6): 1818-1837.

Georgopoulos, A. P. and S. Grillner (1989). "Visuomotor coordination in reaching and locomotion." Science **15 Sept.**: 1209-1210.

Georgopoulos, A. P., A. B. Schwartz, et al. (1986). "Neuronal population coding of movement direction." Science **233**: 1416-1419.

Ghahramani, Z., D. Wolpert, et al. (Submitted). "Representation of the visuomotor coordinate transformation: Patterns of generalization to local and contextual remappings." :

Hannaford, B. L., L. Wood, et al. (1991). "Performance evaluation of a six-axis generalized force reflecting teleoperator." IEEE Transactions on Systems, Man, and Cybernetics **21**: 620-633.

Jones, L. A. and I. W. Hunter (1990). "A perceptual analysis of stiffness." Exp. Brain Res. **79**: 150-156.

Jones, L. A. and I. W. Hunter (1992). Human operator perception of mechanical variables and their effects on tracking performance. ASME Winter Annual Meeting: Issues in the Development of Kinesthetic Displays for Teleoperation and Virtual Environments, Anaheim, CA,

Kim, W. S., F. Tendick, et al. (1987). "A comparison of position and rate control for telemanipulations with consideration of manipulator system dynamics." IEEE Journal of Robotics and Automation **3**: 426-436.

Kim, W. S., K. S. Tso, et al. (1993). An operator interface design for a telerobotic inspection system. AIAA Aerospace Design Conference, Irvine, CA,

Massimino, M. J. and T. B. Sheridan (1994). "Teleoperator performance with varying force and visual feedback." Human Factors **36**(1): 145-157.

McKenna, M. and D. Zeltzer (1992). "Three dimensional visual display systems for virtual environments." Presence **1**(4): 421-458.

Parker, N. R., S. E. Salcudean, et al. (1993). Application of Force Feedback to Heave Duty Machines. Proceedings of IEEE International Conference on Robotics and Automation, Pausch, R., T. Crea, et al. (1992). "A literature survey for virtual environments: Military flight simulator visual systems and simulator sickness." Presence **1**(3): 344-363.

Rolland, J. P., W. Gibson, et al. (1995). "Towards quantifying depth and size perception in virtual environments." Presence **4**(1): 24-49.

Sheridan, T. B. (1992). Telerobotics, Automation, and Human Supervisory Control. Cambridge, MA, MIT Press.

Soechting, J. F. and M. Flanders (1989a). "Sensorimotor representations for pointing to targets in three-dimensional space." J. Neurophys. **62**(2): 582-594.

Soechting, J. F. and M. Flanders (1989b). "Errors in pointing are due to approximations in sensorimotor transformations." J. Neurophys. **62**(2): 595-608.

Soechting, J. F. and B. Ross (1984). "Psychophysical determination of coordinate representation of human arm orientation." Neuroscience **13**(2): 595-604.

Soechting, J. F., S. I. H. Tillery, et al. (1990). "Transformation from head- to shoulder-centered representation of target direction in arm movements." J. Cog. Neuro. **2**(1): 32-43.

Tillery, S. I. H., M. Flanders, et al. (1991). "A coordinate system for the synthesis of visual and kinesthetic information." J. Neuroscience **11**(3): 770-778.

Ware, C. and R. Balakrishnan (1994). "Reaching for objects in VR displays: Lag and frame rate." ACM Trans. on Computer-Human Interaction **1**(4): 331-356.

Welch, R. B. (1978). Perceptual Modification: Adapting to altered sensory environments. New York, Academic Press, Inc.

Whitney, D. (1969). "Resolved motion rate control of manipulators and human prosthesis." IEEE Transactions on Man-Machine Systems **MMS-10**: 47-53.

Zhai, S. and P. Milgram (1993a). Human performance evaluation of manipulation schemes in virtual environments. Proceedings of the IEEE Virtual Reality Annual International Symposium, Seattle, WA,

APPENDIX A

IMPLEMENTING VELOCITY CONTROL THROUGH SOFTWARE

```
/* Modified by Corrie Lathan, 1995 to run in rate-control mode
/*
 *  FAST Session v1.1 - (Flat Alternative Sensimotor Testbed Session)
 *  (c) 1994 Evan Wies; Massachusetts Institute Of Technology
 *
 *  File: RadialFitts.cp
 *
 *  Description:  Code that executes one set of the Radial Fitts Task.
 *                Also contains code to write RFT Data to the current response file.
 *
 */

#include "FAST.Script.h"
#include "RadialFitts.h"
#include "FOOGS.h"
#include "XYTS_LIB.h"
#include <math.h>


/*******************************************************************
*******************/
// Global Data for Radial Fitts Task
//
grFilledCircle  *newManip, *oldManip;
grFilledCircle  *TaskDest;
Timer*                  timer;

rftData                 *TaskResults;
rftDescriptor *TaskParms;

double* xBuffy;                         // Octosample buffer;
double* yBuffy;
double  XSPOT,  YSPOT;                  //for  corrie's  DrawVelocityManip

const int MISampleRate = 5;             // 5ms per sample rate
const int SkipsPerOctosample = 5;       // save every n samples for octopus
const int maxtrl = 20;
const int maxxy = 5000;                         // Octosample buffer size specifiers


/*******************************************************************
*******************/
// Function Prototypes
//
/* Called from outside this source file. */
rftData* RadialFittsTask( int numTrials, rftDescriptor* taskInfoPTR );
void    WriteRadialFittsData( int setNumber, int numSets, rftData* myData );
```

```
/* Support for above functions. */
void InitRFT( int numTrials, rftDescriptor* taskInfoPTR );
void CloseRFT( void );
void WaitForHome( void );
void GoToTarget( int trial, int target, int fBlank );
void GoToTarget_Flash( int trial, int target, int prevTarget, int fBlank );
void DrawManip( XYTSvalRecPTR destXYTSptr, double* pathLength, int hideFlag );
double DrawVelocityManip( XYTSvalRecPTR destXYTSptr, double*
pathLength, int hideFlag ); //for corrie
void FlashTarget( int frameNum, int targetNum );
Boolean StoppedAtTarget( int target, XYTSvalRecPTR destXYTSptr );
extern void ManipRun( CursorTransInfo* CTinfo, XYTSvalRecPTR destXYTSptr );

/* Routines for copying RFT data structures... */
void CopyTargetInfo( TargetInfo* src, TargetInfo* dest );
void CopyHomeInfo( HomeInfo* src, HomeInfo* dest );
void CopyCursorInfo( CursorInfo* src, CursorInfo* dest );
void CopyRFTAngles( RFTargetAngles* src, RFTargetAngles* dest );
void CopyRFTdescriptor( rftDescriptor* src, rftDescriptor* dest );


/**********************************************************************
*******************/
// RadialFittsTask:    Takes the number of desired trials, and three structures each of
//                                  which describes the parameters for the
Target, the Home, and the
//                                  Cursor.  It returns a pointer to the results of
the session.
//

/**********************************************************************
*******************/
// WriteRadialFittsData:  Takes a pointer to rftData and the set number, and writes
//                                  formatted data to the current response file
(determined by
//                                  global values.
//

/**********************************************************************
*******************/
// InitRFT:    Initializes the Radial Fitts Task.  This procedure creates the manipulandum,
//                   cursor, and home objects according to the caller's
parameters.  It also
//                   initializes the temporal alterations and timers.
//

/**********************************************************************
*******************/
// WaitForHome:     Prepares the subject to run the experiment.  Has subject go
//                         to lower corner and then to home.
//
void WaitForHome( void )
{
	XYTSvalRec   XYTSvals;
```

```
        int                 wl, wt;
        double              XYTSscrnX, XYTSscrnY;
        Rect         myRect;
        double              nothing;
        Boolean             done;

/* Move To Lower Right Corner */
        PenNormal();
        TextMode( srcXor );
        TextSize( 24 );
        MoveTo( 400, 625 );
        DrawFRString( "\pMove To Lower Right Corner!" );

        wl = testbedWPtr->portRect.left;
        wt = testbedWPtr->portRect.top;
        XYTSscrnX = XYTSscrnY = 1000;
/*      SetRect( &myRect, wl, wt, wl+40, wt+40 );           */
        SetRect( &myRect, wl, wt, wl+225, wt+225 );           //for corrie
        done = false;
        do {
                if ( MInt_Flag ) {
                        ManipRun( &(TaskResults->CTinfo), &XYTSvals );
                        sxfNull( XYTSvals.Xval, XYTSvals.Yval,
                                        smX_Offset, smY_Offset, PPIi, PPIj,
                                        &XYTSscrnX, &XYTSscrnY );
/*                      done = ( (XYTSscrnX < (wl+40)) && (XYTSscrnY < (wt+40)) );
        */
                        done = ( (XYTSscrnX < (wl+225)) && (XYTSscrnY <
(wt+225)) );            //for corrie
                        MInt_Flag = false;
                }
                if ( VBL_Flag ) {
                        PenNormal();
                        FillRect( &myRect, &white );
                        DrawManip( &XYTSvals, &nothing, false );
                        VBL_Flag = false;
                }
        } while ( ! done );
        FillRect( &myRect, &black );
        PenNormal();
        TextMode( notSrcCopy );
        TextSize( 24 );
        MoveTo( 400, 625 );
        DrawFRString( "\p                    " );

/* Move To Home */
        TextMode( srcXor );
        MoveTo( 400, 625 );
        DrawFRString( "\pMove To Home Target!" );
        done = false;
        do {
                if ( MInt_Flag ) {
                        ManipRun( &(TaskResults->CTinfo), &XYTSvals );
                        done = StoppedAtTarget(0, &XYTSvals);
```

```
                        MInt_Flag = false;
                }
                if ( VBL_Flag ) {
                        TaskDest[0].Draw2( &black, &white );
                        DrawManip( &XYTSvals, &nothing, false );
                        VBL_Flag = false;
                }
        } while ( ! done );
        if ( TaskParms->RFTinfo.fThump ) ThumpThumper();
        if ( TaskParms->RFTinfo.fSpeak ) SpeakSpeaker();
        if ( TaskParms->RFTinfo.fFlash ) TaskDest[0].Draw2( &white, &white );

        FillRect( &(testbedWPtr->portRect), &black );
}


/***********************************************************************
*******************/
// GoToTarget:  Draws the the 'target'th target and then waits till the subject
//                              gets there... it then records the time this took to
//                              TaskResults->Data[trial].  Alternate the parameter 'target'
between 0 and
//                              'trial' to do a FittsTask.
//
void GoToTarget( int trial, int target, int fBlank )
{
        long            TaskTime;
        long            numxy, countxy;
        int             xytrial;
        double                  pathLength;
        XYTSvalRec      XYTSvals;
        Boolean                 done;

/* Clear Home and pathLength and Start Timer */
        TaskDest[0].Draw( &black );
        XSPOT = CENTER_X;               // for Corrie
        YSPOT = CENTER_Y;               // for Corrie
        pathLength = 0;
        StartTimer( timer );

/* Set Octosampling variables... */
        numxy   = 0;  countxy = 0;
        xytrial = (trial<maxtrl)?trial:(maxtrl-1);

/* Run Task Until Destination Reached */
        done = false;
        do {
                if ( MInt_Flag ) {
                        ManipRun( &(TaskResults->CTinfo), &XYTSvals );
                        numxy++;
                        if ( (numxy < maxxy) && (target) && !(numxy %
SkipsPerOctosample) ) {
                                countxy++;
```

```
/*				xBuffy[(maxxy*xytrial)+(countxy)] = XYTSvals.Xval;
	*/
/*				yBuffy[(maxxy*xytrial)+(countxy)] = XYTSvals.Yval;
	*/
				xBuffy[(maxxy*xytrial)+(countxy)]  =  XSPOT;
		// for Corrie
				yBuffy[(maxxy*xytrial)+(countxy)]  =  YSPOT;
			// for Corrie
			}
			done = StoppedAtTarget(target, &XYTSvals);
			MInt_Flag = false;
		}
		if ( VBL_Flag )  {
			TaskDest[target].Draw2( &black, &white );
/*			DrawManip( &XYTSvals, &pathLength, (fBlank && target) );
	*/
			DrawVelocityManip(  &XYTSvals,  &pathLength,  (fBlank
&& target) );			// for Corrie
			VBL_Flag = false;
		}
	} while ( ! done );
	TaskDest[target].Draw( &black );

/* Find out how long that took... */
	TaskTime = StopTimer( timer );

/* Record how big our OctoBuffer is... */
	if (target)  {
		xBuffy[maxxy*xytrial] = countxy;
		yBuffy[maxxy*xytrial] = (double)TaskTime;
	}

/* Record Data! */
	TaskResults->Data[trial].trialNum = trial;
	if ( target == 0 ) {
		TaskResults->Data[trial].subjHX = XYTSvals.Xval;
		TaskResults->Data[trial].subjHY = XYTSvals.Yval;
		TaskResults->Data[trial].dtoh = pathLength;
		TaskResults->Data[trial].ttoh = TaskTime;
	}
	else {
		TaskResults->Data[trial].subjTX = XYTSvals.Xval;
		TaskResults->Data[trial].subjTY = XYTSvals.Yval;
		TaskResults->Data[trial].dtot = pathLength;
		TaskResults->Data[trial].ttot = TaskTime;
	}
}


/*****************************************************************
*******************/
// GoToTarget_Flash:  Draws the the 'target'th target and then waits till the subject
//					gets there... it then records the time this took to
```

```
//                                  TaskResults->Data[trial].  Alternate the parameter
'target' between
//                                  0 and 'trial' to do a FittsTask.
//                                  This routine flashes the target...


/******************************************************************
******************/
// DrawManip:  Updates the manipulanum location and draws it.
//

/*int prevFlashed = 0;   */        //for corrie
/******************************************************************
******************/
// DrawVelocityManip:  Updates the manipulanum location and draws it.
//for corrie
//
double DrawVelocityManip( XYTSvalRecPTR destXYTSptr, double*
pathLength, int hideFlag )
{
        double rad, xdiff, ydiff;
        double junkint;
        newManip->SetCenter( XSPOT,YSPOT );

                XSPOT = 10.00 * modf((XSPOT + 0.035*((destXYTSptr-
>Xval) - CENTER_X))/10,&junkint);//destXYTSptr->Xval
                YSPOT = 10.00 * modf((YSPOT + 0.035*((destXYTSptr-
>Yval) - CENTER_Y))/10,&junkint);

        xdiff = ((XSPOT) - CENTER_X);
        ydiff = ((YSPOT) - CENTER_Y);

        oldManip->Draw( &black );

        // if (! hideFlag )
                newManip->Draw( &white );
        // else
        //      newManip->Draw( &black );

        (*pathLength) += newManip->DistToCenter( oldManip );

        oldManip->Become( newManip );
        return(rad);
}

int prevFlashed = 0;
/******************************************************************
******************/
// FlashTarget:  Puts the passed target in the proper state with respect to a frame count.
//

/******************************************************************
******************/
// StoppedAtTarget:  Calculates whether subject has stopped at the target or not...
```

```
//                                    depends on trial parameters and state of trial.
//
Boolean StoppedAtTarget( int target, XYTSvalRecPTR destXYTSptr )
{
        Boolean cond, cond1, cond2;
/*      double          rad, konst;          // for corrie*/
        double          rad, xdiff, ydiff, konst;
        double          CurVelocity;
        int             crit;

        if ( target == 0 ) {
                crit  = TaskParms->RFTinfo.Hcrit;
                konst = TaskParms->RFTinfo.Hconst;
        }
        else {
                crit  = TaskParms->RFTinfo.Tcrit;
                konst = TaskParms->RFTinfo.Tconst;
        }

        switch ( crit )  {

                // Touching Time Period OR EXCEED RADIUS     //for corrie
                case 3:
                        cond1 = false;  cond2 = false;
                        if (target > 0)  {
                                // Calculate radial position of cursor...
                                xdiff = ((XSPOT) - CENTER_X);           //for
corrie
                                ydiff = ((YSPOT) - CENTER_Y);           //for
corrie
                                xdiff *= xdiff; ydiff *= ydiff;
                                rad = sqrt(xdiff + ydiff);

                                // Set flag if cursor is beyond target distance by 2
inches...
                                cond1 = ( rad > (TaskResults-
>Data[target].distance + 2.5) );
                        //      break;
                        }

                        if (TaskDest[target].CheckCollision(newManip) )
                                contact++;
                        else contact = 0;
                        cond2 = (contact > (konst/5));

                        cond = cond1 || cond2;
                        break;
        }
        return cond;
}
```

APPENDIX B

EXPONENTIAL FITS AND CONFIDENCE INTERVALS

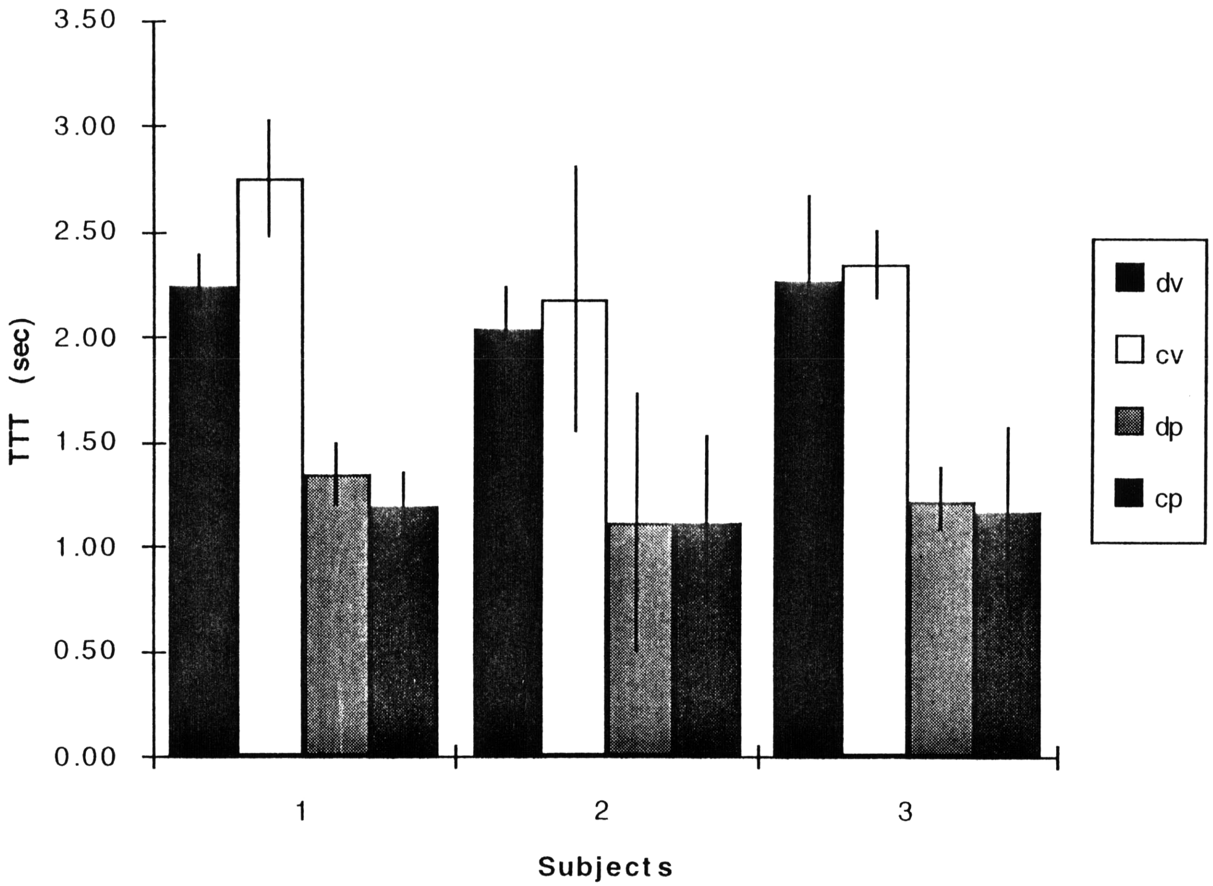
```
% Radial_Anal created by Corrie Lathan, June 14, 1995

%Loads radial fitts data saved as text in the format
%'subject_modeblock#, eg. 'CEL_VEL1'
%Plots each of 20 sets (10 trials each)
%Fits exp to TTT, DTT, TTH, DTH (and 95% conf. intervals)

%_______________________
%SET FLAGS

PLOT_HOME = 'n';  %for tom's data plot home, corrie's doesn't
PSINE = 'y';           %for printing plots of data
folder = 'Transfer:jrp:';  %data path
%folder = 'Sloth_510:Rotation_data:velocity:d2_vel:';  %data path
%folder = 'Sloth_510:Friday_data:diag_180:gs_diag:';  %data path %for control data
prefix = 'jrp_';   %subject:mode

%_______________________

%takes all the blocks for one subject
            first_code = input(['Enter first code  ']);
            if (isempty(first_code))
            first_code = run_code;
            end
            last_code = input(['Enter final code  ']);
            if (isempty(last_code))
            last_code = first_code;
            end

for eta = first_code:last_code
            fprintf('Eta = %2.2f\n',eta);
            eta_code = sprintf('%g',eta);

%Loads radial fitts data saved as text

            data_path = [folder,prefix]
            eval(['load ',data_path,eta_code]);
            aves = eval([prefix, eta_code]);

%Plots each of 20 sets (10 trials each)
% set axes and plot screen
if (PLOT_HOME == 'n')|(PLOT_HOME == 'N')
            %time_scale = 1e6;
            time_scale = 7;
            %time_scale = 7e6;
            dist_scale = 10;
            %dist_scale = 15;
      clg;
```

```
        subplot(211)
        Y = [aves(:,2)/1e6];
        GenLinFitCI   %calls exponential fit program.
        %pause;
        CISIG=[sigmaa; CI]
        %V=[0 20 1e6 time_scale];
        V=[0 20 0 time_scale];          %for control data
        axis(V)
        %eval(['save ','Sloth_510:Rotation_data:stats:',prefix,eta_code, '.TTT CISIG
/ascii /tabs ']);
        eval(['save ','Transfer:friday:',prefix,eta_code, '.TTT CISIG /ascii /tabs ']);
                %for control data
        title_name=[prefix,eta_code,' Time To Target'];
        text(mean(X),time_scale-1, title_name)
        clear Y CI CISIG sigmaa

%pause;

        subplot(212)
        Y = [aves(:,3)];
        GenLinFitCI   %calls exponential fit program.
        CISIG=[sigmaa; CI]
        V=[0 20 1 dist_scale];
        axis(V)
        %eval(['save ','Sloth_510:Rotation_data:stats:',prefix,eta_code, '.DTT CISIG
/ascii /tabs ']);
        eval(['save ','Transfer:friday:',prefix,eta_code, '.DTT CISIG /ascii /tabs ']);%for
control data
        title_name=[prefix,eta_code,' Distance To Target'];
        text(mean(X),dist_scale-3, title_name)
        %text(mean(X),max(Y)-1, title_name)
        clear Y CI CISIG sigmaa
        %pause;
        else

end

                if (PSINE == 'y')|(PSINE == 'Y')
                        orient portrait
                print;
                        fprintf('printing graph\n');
                end

end                     %end eta loop

%_______________________

%General Linear Fit with Confidence Intervals
%raw form of equation to be fit:
%                                               y(x) = k1 + k2 * exp(-1* k3
*x)
%this equation is represented by the following m-files:
%               y(x;k)                          expo(x,k)
%               d(y(x;k))/d(k(n))               dexpo(x,k,n)
```

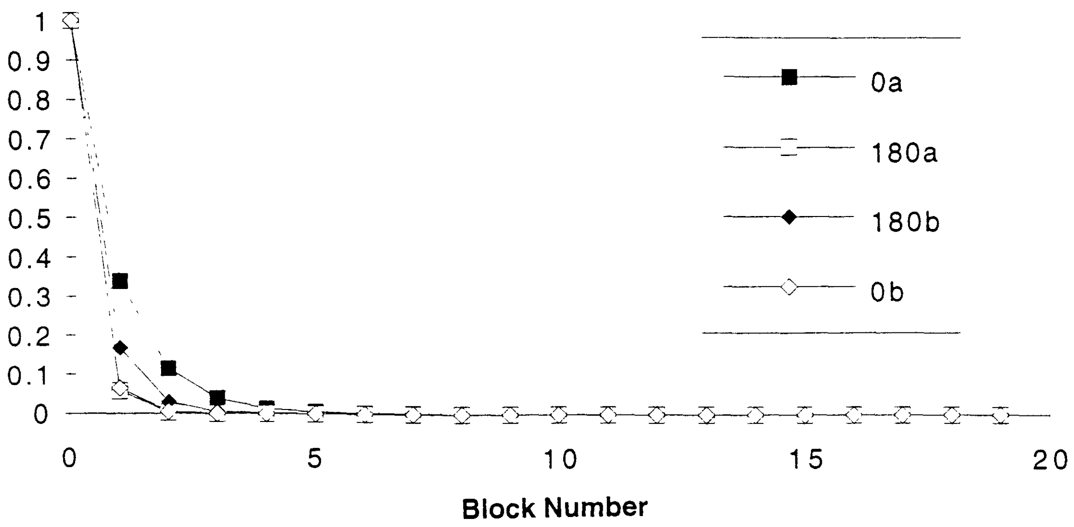
```
%               chisq(y(X;k),Y)                         chiexpo(k,X,Y)
%       with  X,Y = dataset to be fitted
%clear;
%clg;
%note, trial number should be set to trial #-1
%load realDatd2p;X= realDatd2p(:,1); Y= realDatd2p(:,2); %load in the test data
%load fake1.mat;X= fake_dat(:,1)*2; Y= fake_dat(:,2); %load in the test data
X= [1:1:20]' -1;
%Y= 1 + 3 * exp(-0.5*(X)); Y= Y+randn(size(X))/10;

%plot(X,Y,'ro');drawnow; hold on;
plot(X,Y);drawnow; hold on;

k=[1.1, 3.1, -0.45];%initial guess of parameters
%options(14)=1500;
K=fmins('chiexpo',k,[],[],X,Y);
%K=fmins('chiexpo',k,options,[],X,Y);
plot(X,expo(X,K),'w+');

%here we calculate the CIs of the Ks
nn = 10;% ten trials per set
alpha=zeros(3);
for j=(1:3), for i=(1:3),
        alpha(i,j)=nn*sum(dexpo(X,K,i) .* dexpo(X,K,j));
 end;end;

covara=inv(alpha);

dchisq=[ 1.00  2.71  4.00  6.63  9.00  15.51;...
      68.3  90.0  95.4  99.0  99.73 99.99];cc=3;
sigmaa=(dchisq(1,cc)/2)*(abs(diag(sqrt(covara))))'
Khi=(K + sigmaa);
Klo=(K - sigmaa);

title([...
'k1:',num2str(K(1)),',  ±',num2str(sigmaa(1)),';  ',...
'k2:',num2str(K(2)),',  ±',num2str(sigmaa(2)),';  ',...
'k3:',num2str(K(3)),',  ±',num2str(sigmaa(3)),';  ',...
'  @',num2str(dchisq(2,cc)),'%CI'])

CI=[Khi; K; Klo]
plot(X,expo(X,K),'w-','LineWidth',2);
Highs = ...
max([expo(X,[CI(2,1),CI(2,2),CI(1,3)]),...
        expo(X,[CI(2,1),CI(2,2),CI(3,3)]),...
        expo(X,[CI(1,1),CI(2,2),CI(2,3)]),...
        expo(X,[CI(3,1),CI(2,2),CI(2,3)]),...
        expo(X,[CI(2,1),CI(1,2),CI(2,3)]),...
        expo(X,[CI(2,1),CI(3,2),CI(2,3)])]')';
Lows = ...
min([expo(X,[CI(2,1),CI(2,2),CI(1,3)]),...
        expo(X,[CI(2,1),CI(2,2),CI(3,3)]),...
        expo(X,[CI(1,1),CI(2,2),CI(2,3)]),...
        expo(X,[CI(3,1),CI(2,2),CI(2,3)]),...
```

```
         expo(X,[CI(2,1),CI(1,2),CI(2,3)]),...
         expo(X,[CI(2,1),CI(3,2),CI(2,3)])]')';
%plot(X,Highs, 'g:')
plot(X,Highs, ':')

%pause;
%plot(X,Highs,'w-','LineWidth',2);

plot(X,Lows, ':')
%plot(X,Lows, 'g:')
%plot(X,expo(X,Khi), 'y--')
%plot(X,expo(X,Klo), 'y--')
%plot(X,expo(X,[CI(2,1),CI(2,2),CI(1,3)]), 'y:')
%plot(X,expo(X,[CI(2,1),CI(2,2),CI(3,3)]), 'y:')
%plot(X,expo(X,[CI(1,1),CI(2,2),CI(2,3)]), 'y:')
%plot(X,expo(X,[CI(3,1),CI(2,2),CI(2,3)]), 'y:')
%plot(X,expo(X,[CI(2,1),CI(1,2),CI(2,3)]), 'y:')
%plot(X,expo(X,[CI(2,1),CI(3,2),CI(2,3)]), 'y:')

%______________________

%expo.m:  y(x;k) = expo(x,k)
function y = expo(x,k),
y=      k(1) + k(2) * exp(k(3)*x);

%______________________

%dexpo.m:  d(y(x;k))/d(k(n))=dexpo(x,k,n)
function dy = dexpo(x,k,n),
if (n==1),
        dy= 1;
elseif (n==2),
        dy= exp(x*k(3));
elseif (n==3),
        dy= k(2)*x.*exp(x*k(3));
else,
        dy= NaN;
end;

%______________________

%chiexpo.m:  chisq(y(X;k), Y) = chiexpo(k, X, Y)
function chi = chiexpo(k,X,Y),
chi= ...
sum( ...
        ( ...
                k(1) + k(2) * exp(k(3)*X)               ...
                                        - Y ...
        ).^2 ...
 );
```

Figure 3.1: The Time to Target (TTT) response for three subjects. In the figure legend, d=diagonal target directions, c=cardinal target directions, v=velocity control mode, and p=position control mode. Standard deviations are shown for each condition which consists of 200 trials.



a. Subject BS (Cardinal)



b. Subject MG (Cardinal)

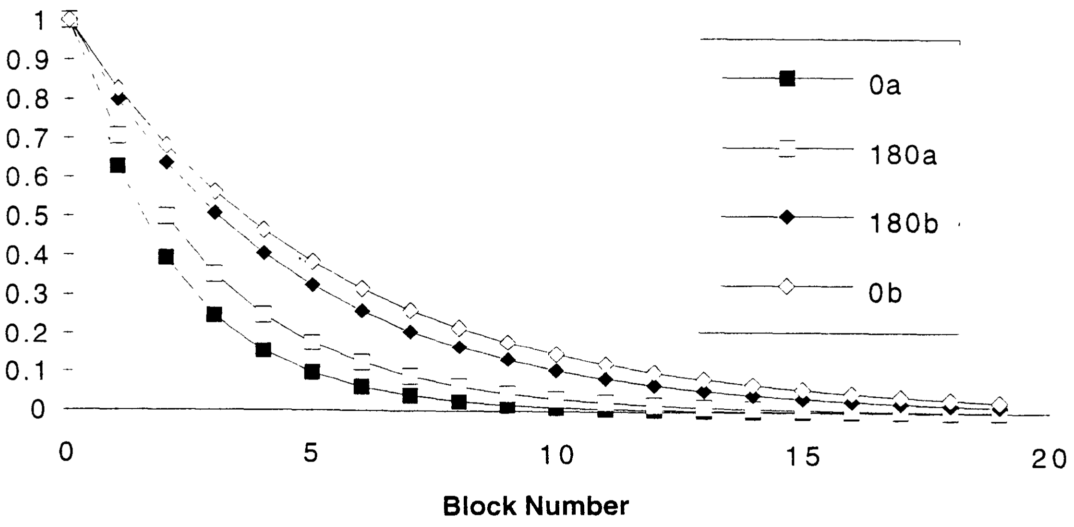
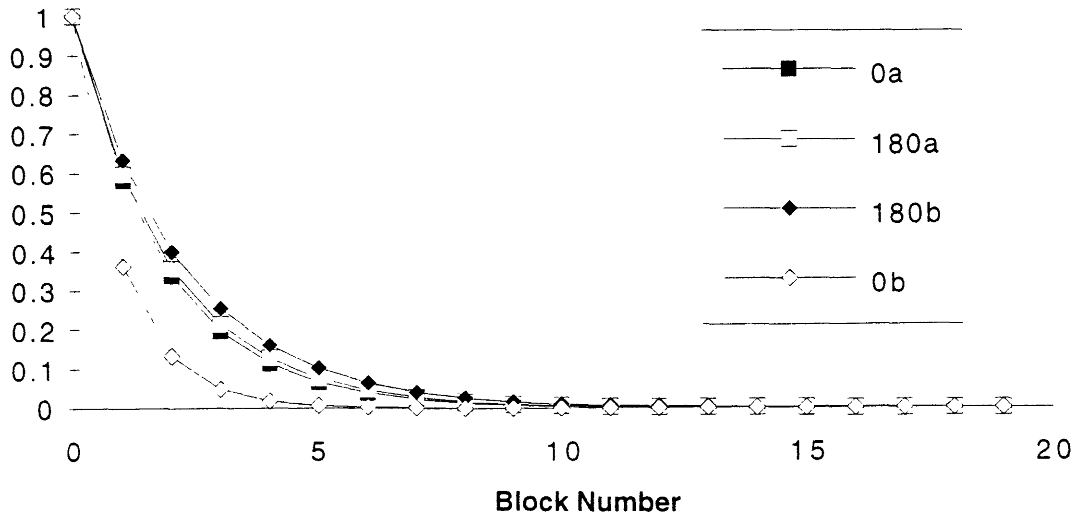
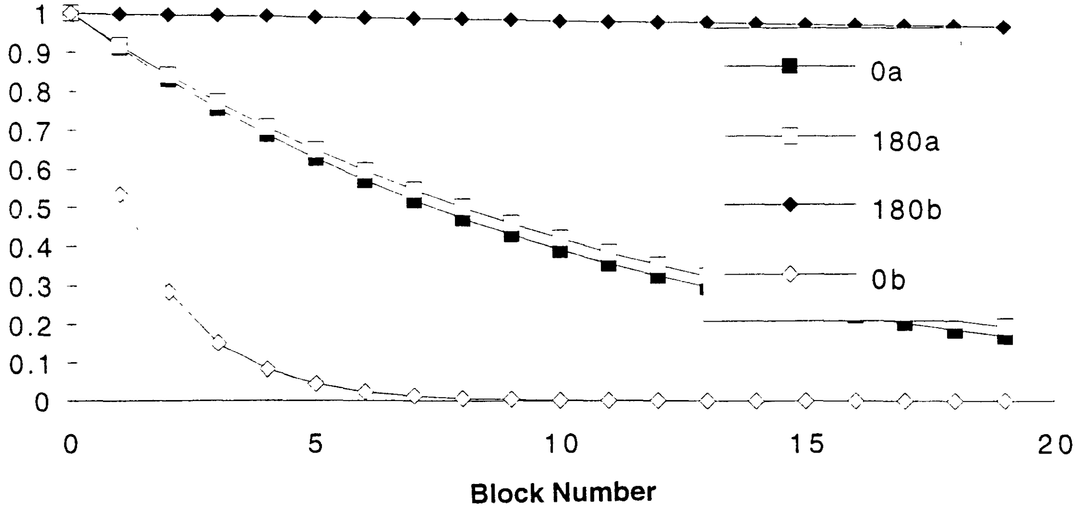


Figure 2a-b: Normalized time constant versus block number for Subject BS (a) and Subject MG (b). Both subjects were run in four conditions with targets at the cardinal locations (N, S, E, and W).



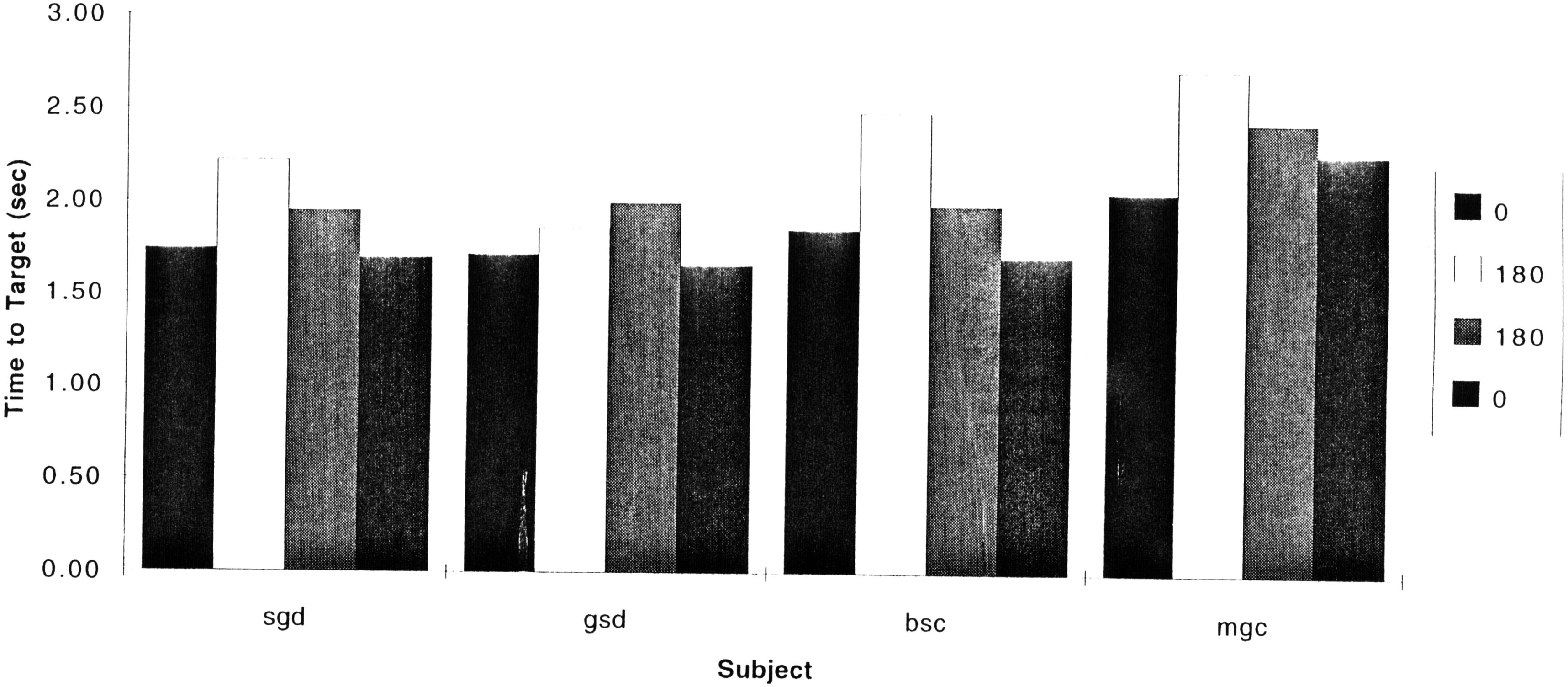
c. Subject SG (Diagonal)



d. Subject GS (Diagonal)

Figure 2c-d: Normalized time constant versus block number for Subject SG (c) and Subject GS (d). Both subjects were run in four conditions with targets at the diagonal locations (NE, NW, SE, and SW).

Figure 3.3: Steady State Time to Target, Parameter K1. Subject initials are followed by the direction of the targets presented, c=Cardinal, d=diagonal



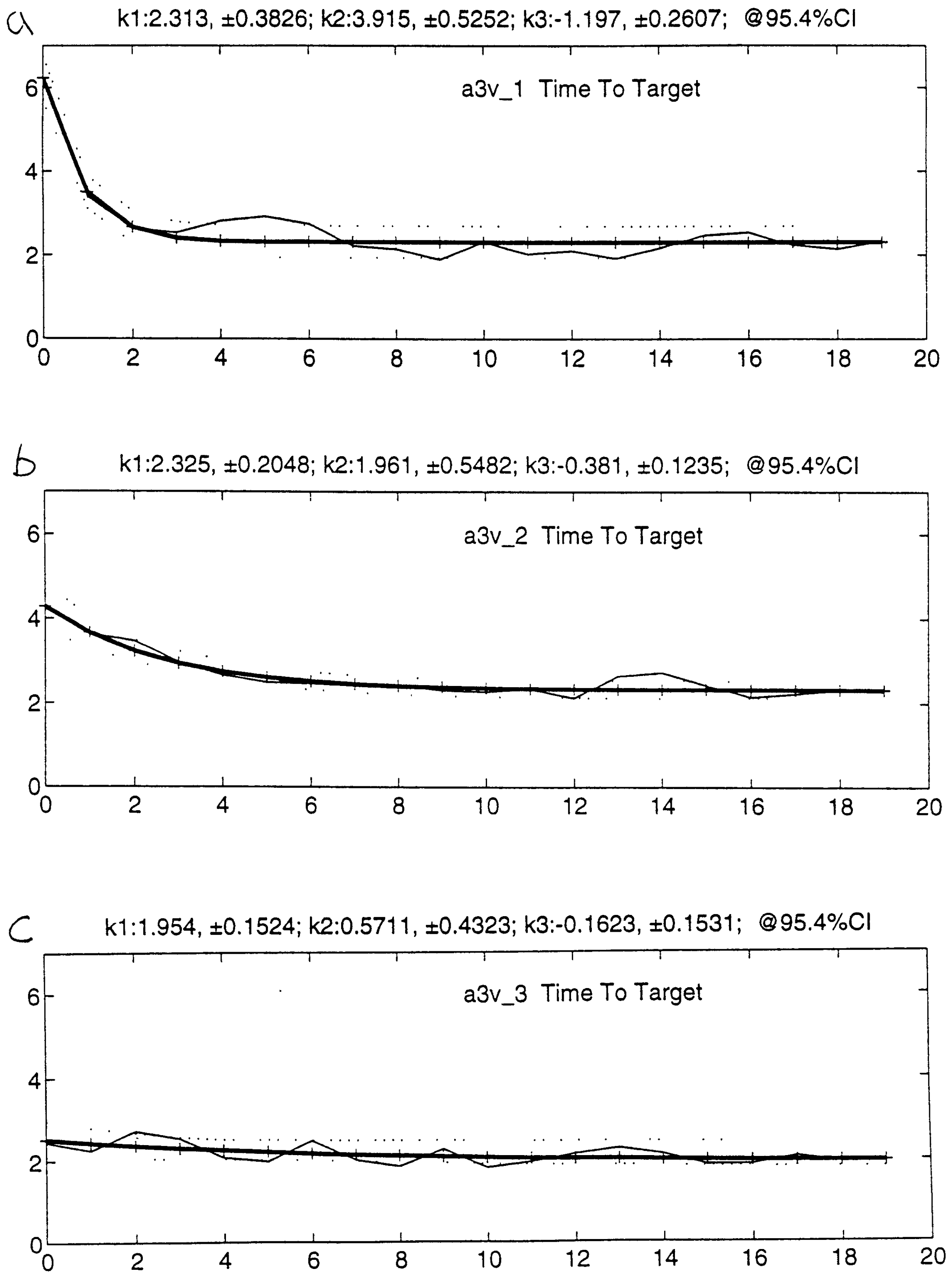


Figure 4a-c: Sample TTT responses using velocity control mode, Conditions 1-3.

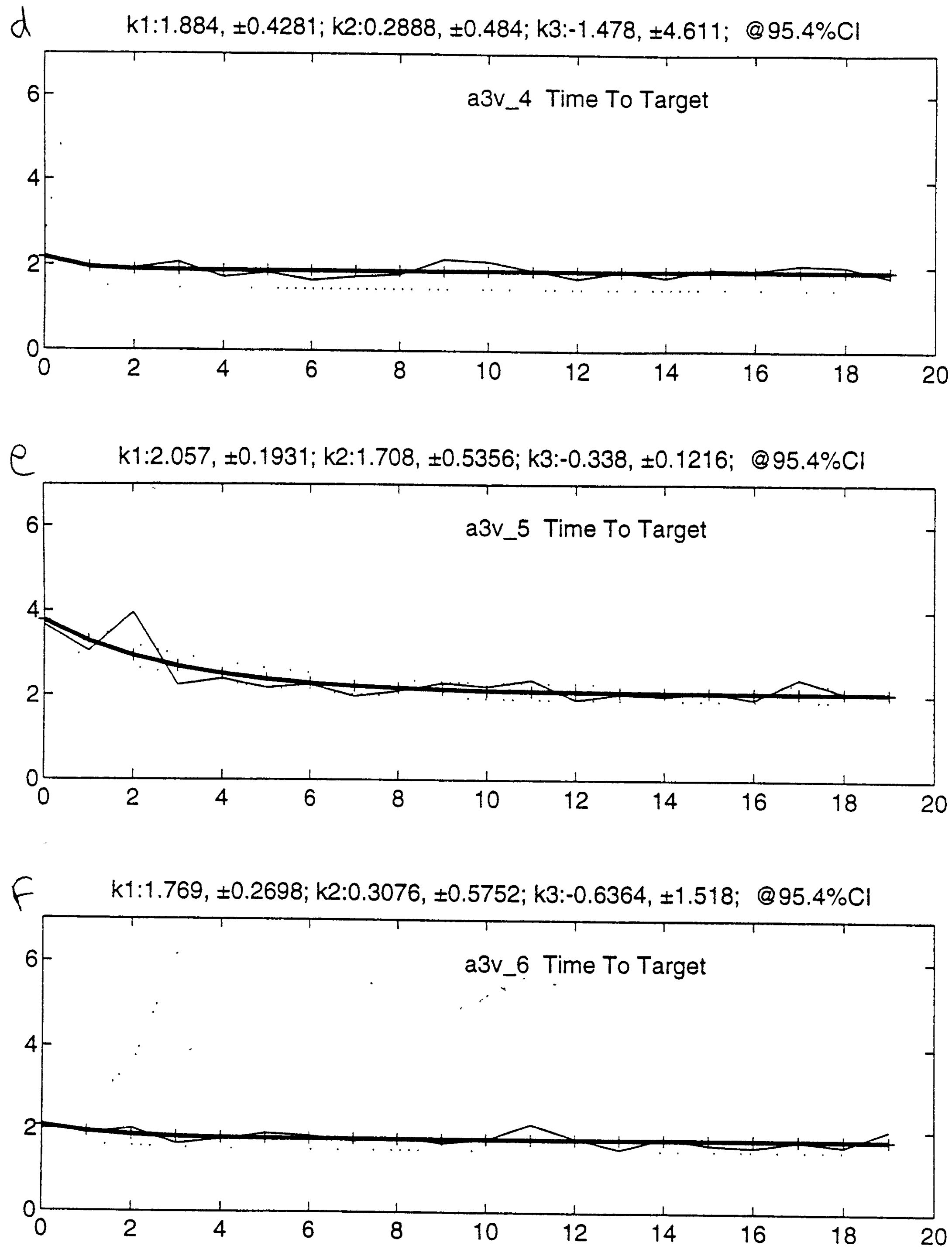
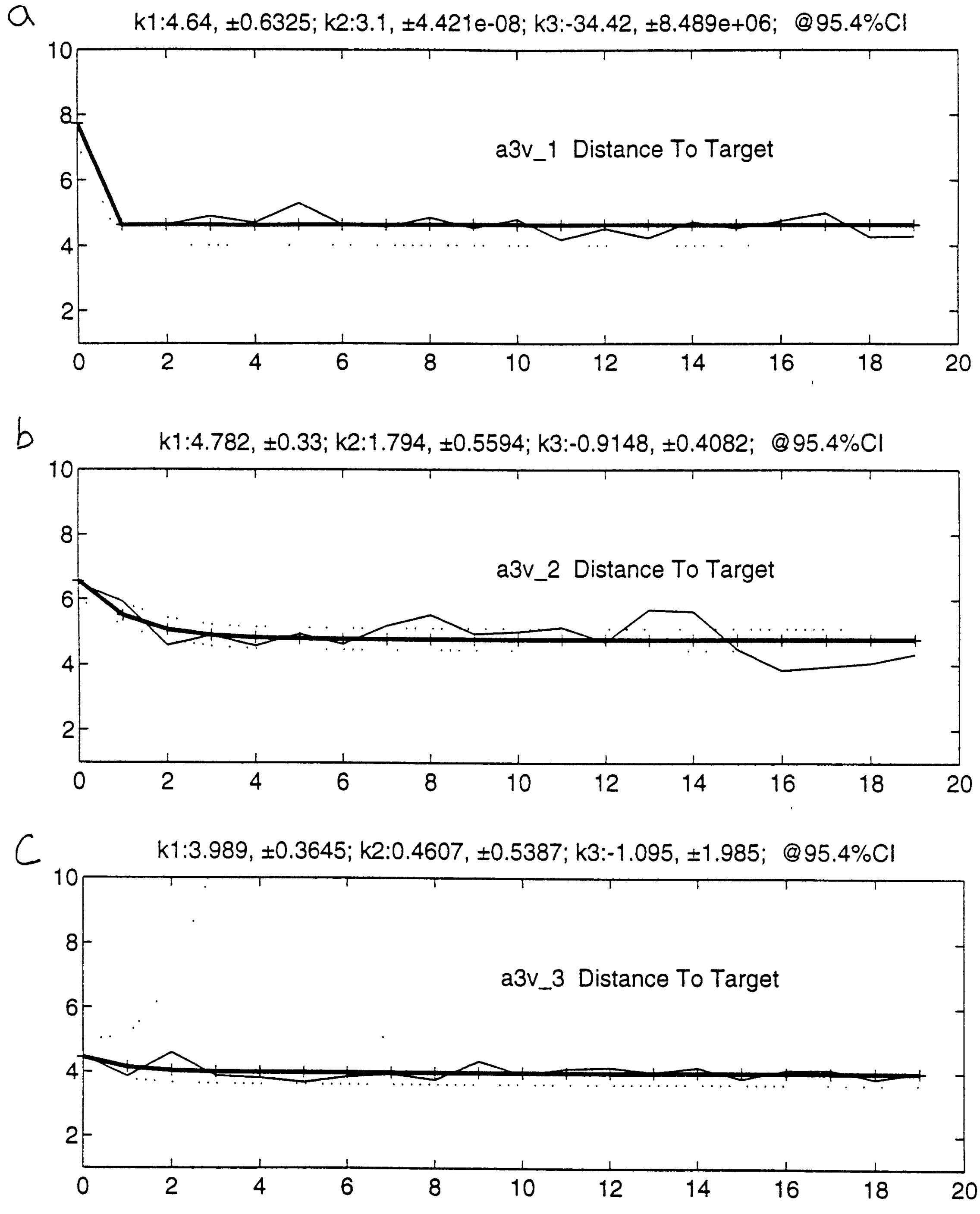


Figure 4d-f: Sample TTT responses using velocity control mode, Conditions 4-6.



a k1:4.64, ±0.6325; k2:3.1, ±4.421e-08; k3:-34.42, ±8.489e+06; @95.4%CI

b k1:4.782, ±0.33; k2:1.794, ±0.5594; k3:-0.9148, ±0.4082; @95.4%CI

c k1:3.989, ±0.3645; k2:0.4607, ±0.5387; k3:-1.095, ±1.985; @95.4%CI

Figure 5a-c: Sample DTT responses using velocity control mode, Conditions 1-3.

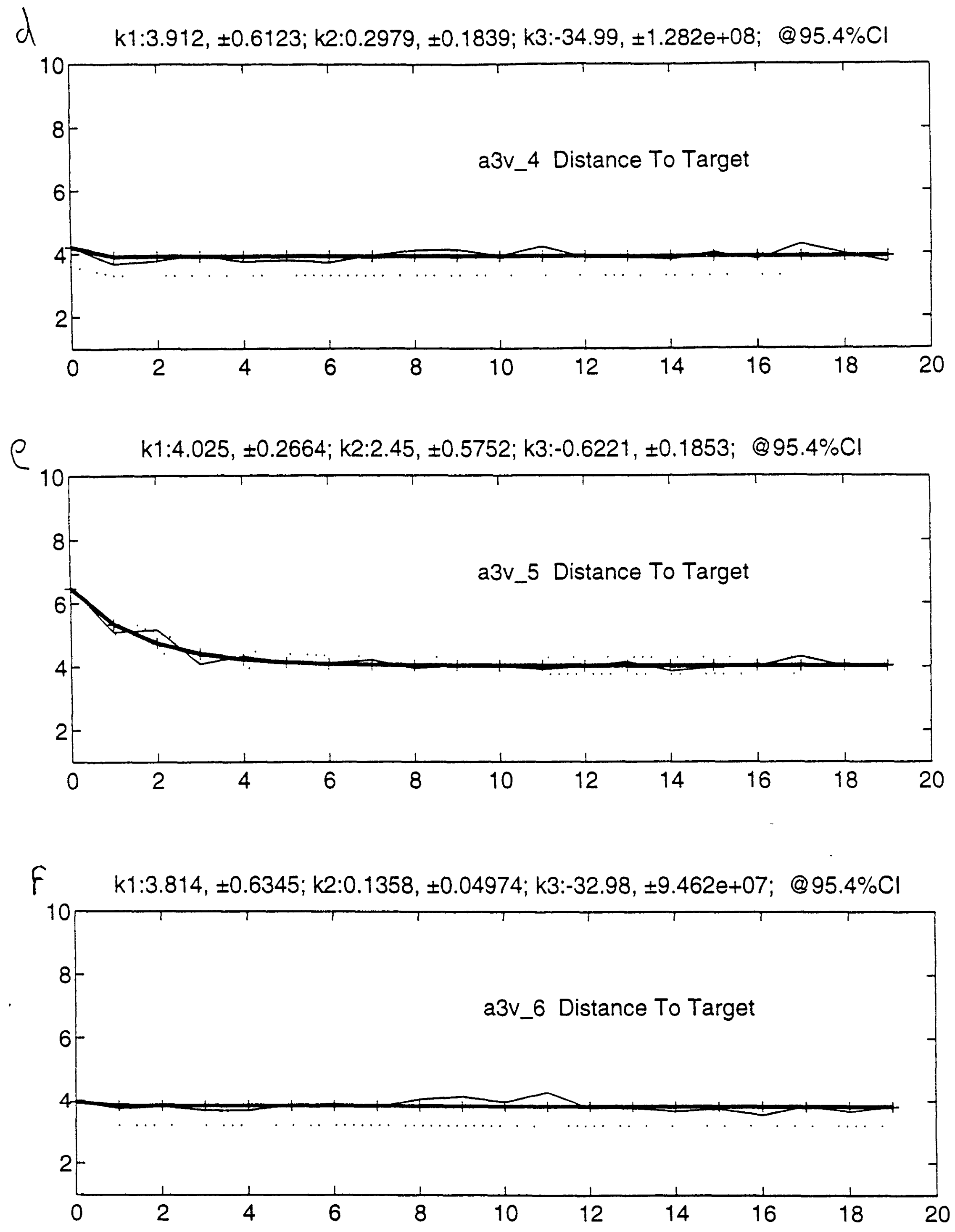


Figure 5d-f: Sample DTT responses using velocity control mode, Conditions 4-6.

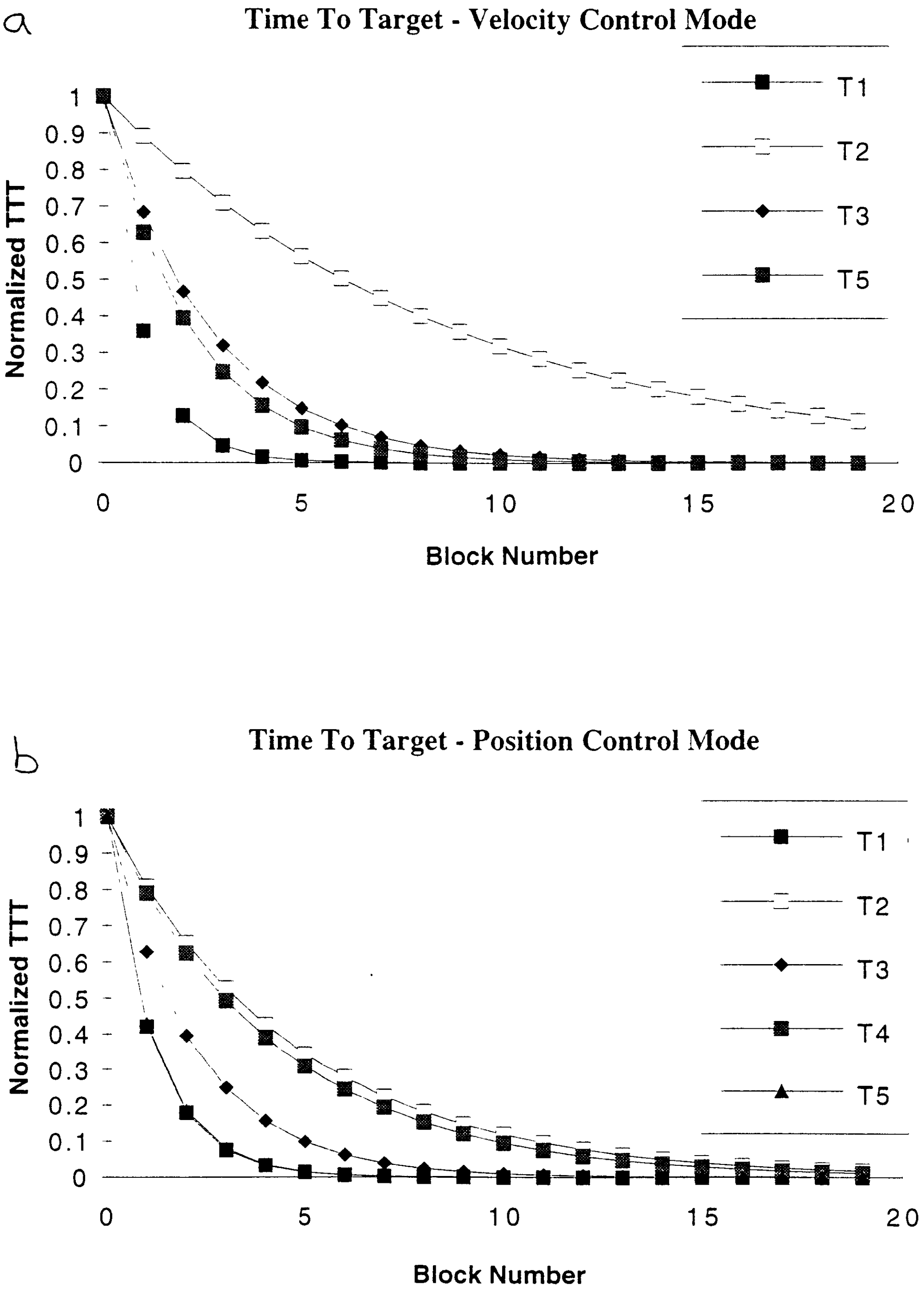
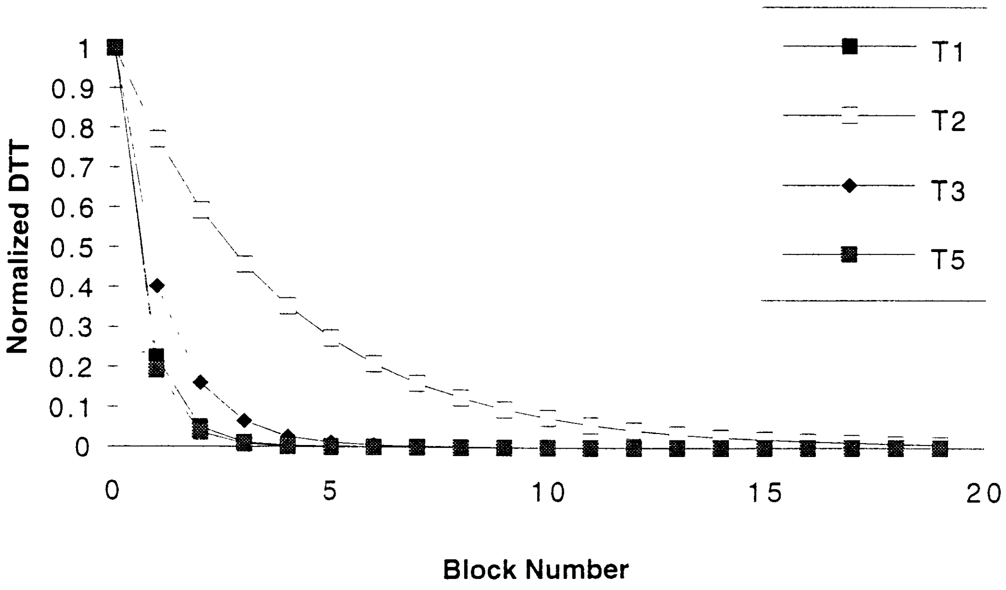


Figure 6: Normalized time constants (K3) for the TTT responses for velocity control mode (a) and position control mode (b) for *Experiment 2.*

a

Distance To Target - Velocity Control Mode



b

Distance To Target - Position Control Mode

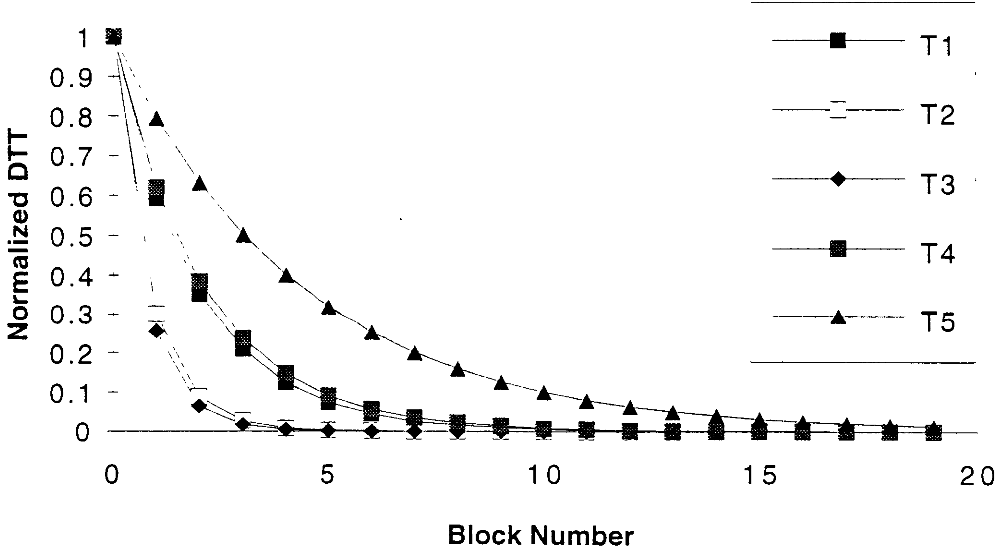


Figure 7: Normalized time constants (K3) for the DTT responses for velocity control mode (a) and position control mode (b) for *Experiment 2*.

Figure 3.8: Steady-state performance during the *Experiment 2* 60 degree rotation adaptation block. TTT=Time to Target, DTT=Distance to Target, v=velocity control mode, and p=position control mode. The scale (in seconds) for the Time to Target responses is on the left and the scale (in inches) for the Distance to Target responses is on the right.

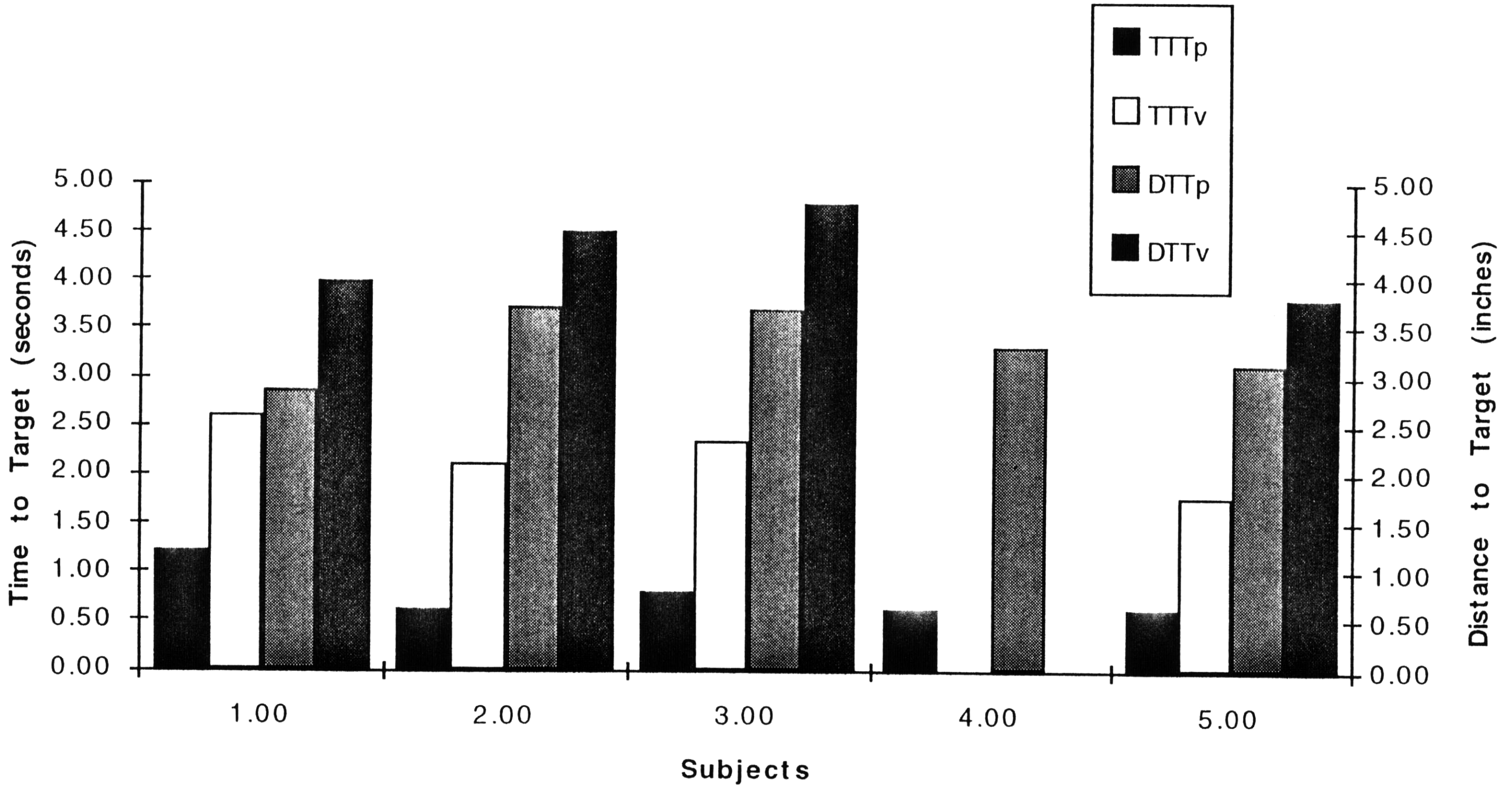
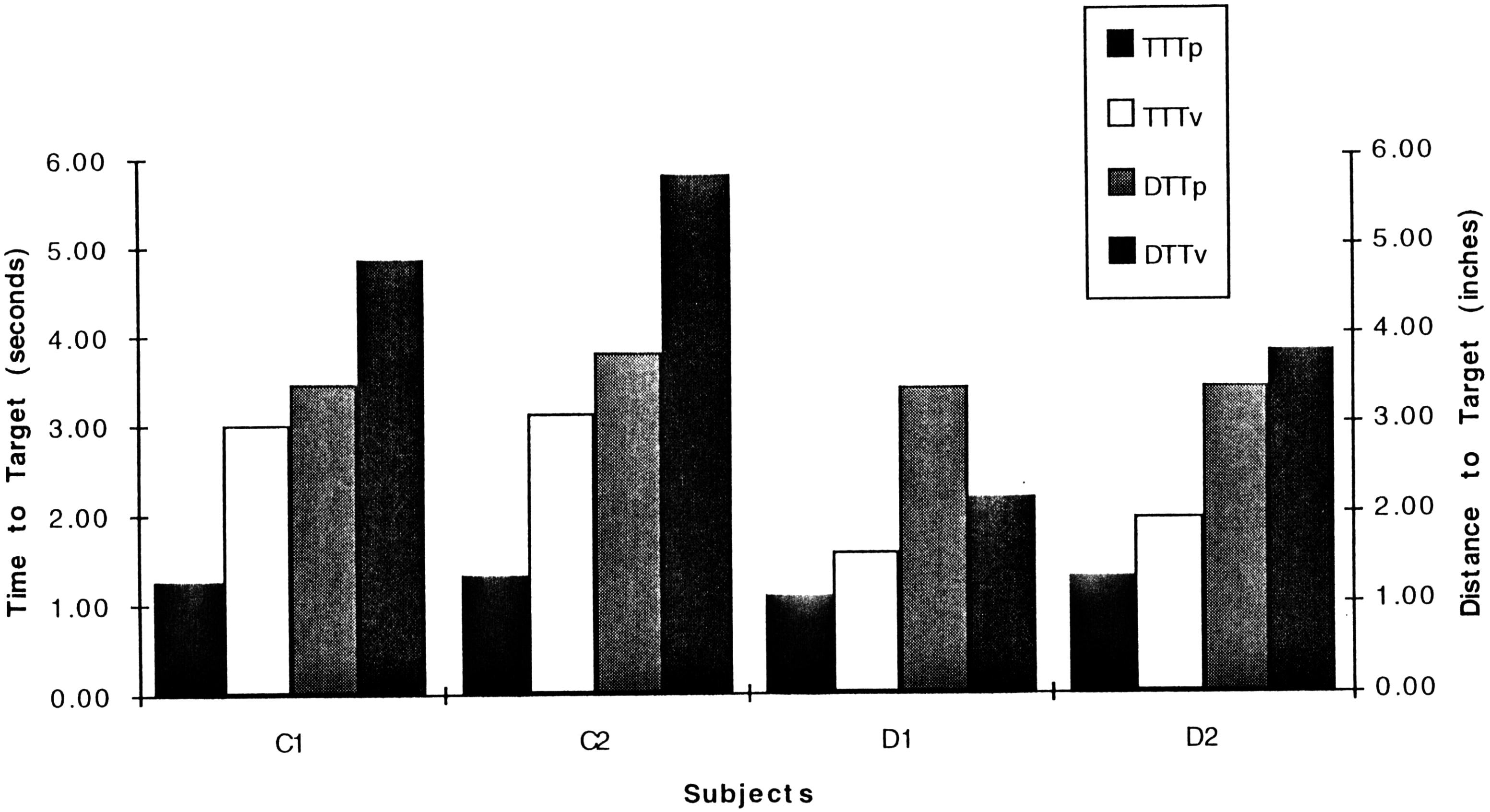


Figure 3.9: Steady-state performance during the *Experiment 3* 60 degree rotation adaptation block. TTT=Time to Target, DTT=Distance to Target, v=velocity control mode, and p=position control mode. The scale (in seconds) for the Time to Target responses is on the left and the scale (in inches) for the Distance to Target responses is on the right.



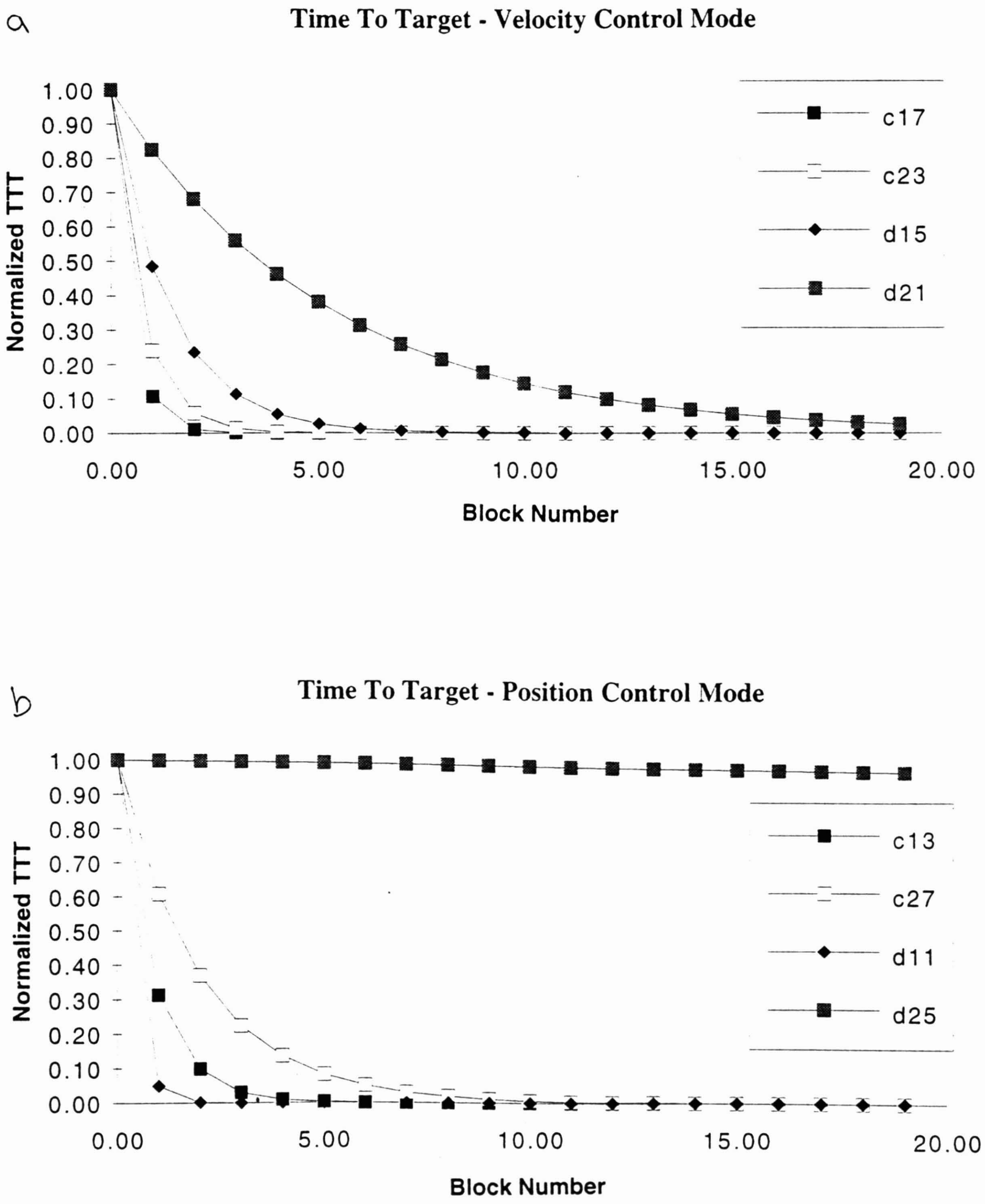
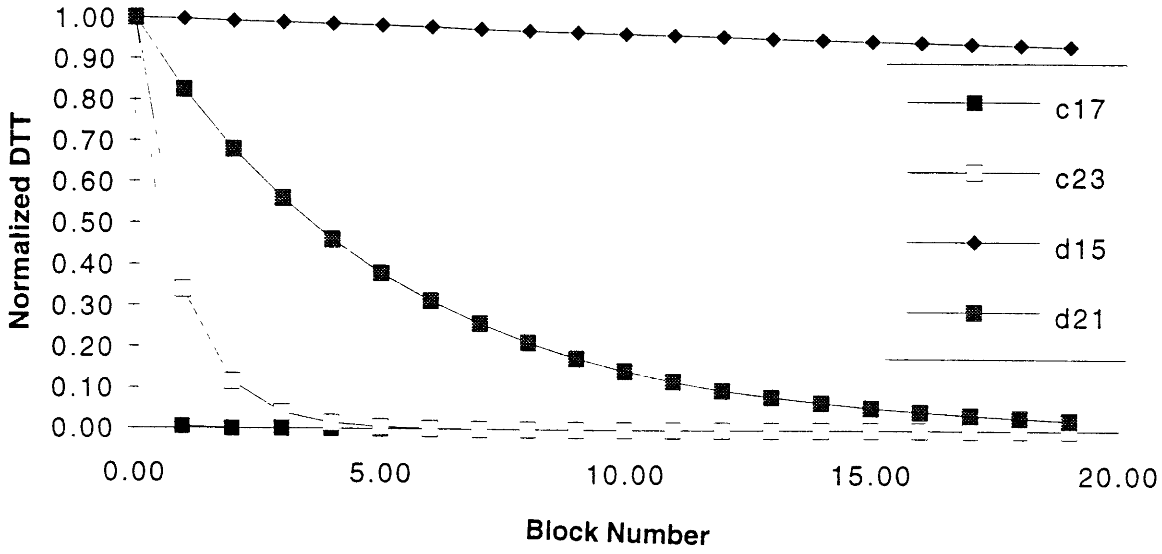


Figure 10: Normalized time constants (K3) for the TTT responses for velocity control mode (a) and position control mode (b) for *Experiment 3*.

a

Distance To Target - Velocity Control Mode



b

Distance To Target - Position Control Mode

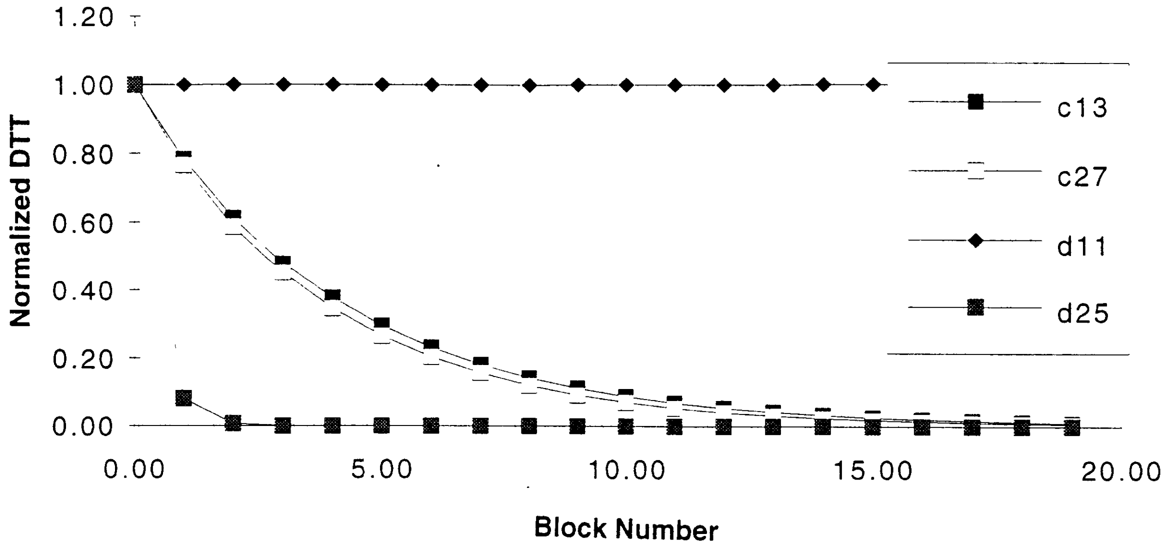


Figure 11: Normalized time constants (K3) for the DTT responses for velocity control mode (a) and position control mode (b) for *Experiment 3.*

Figure 3.12: Average Time to Target (TTT) time constants for each rotation condition. In the figure legend, p=position control mode, v=velocity control mode.

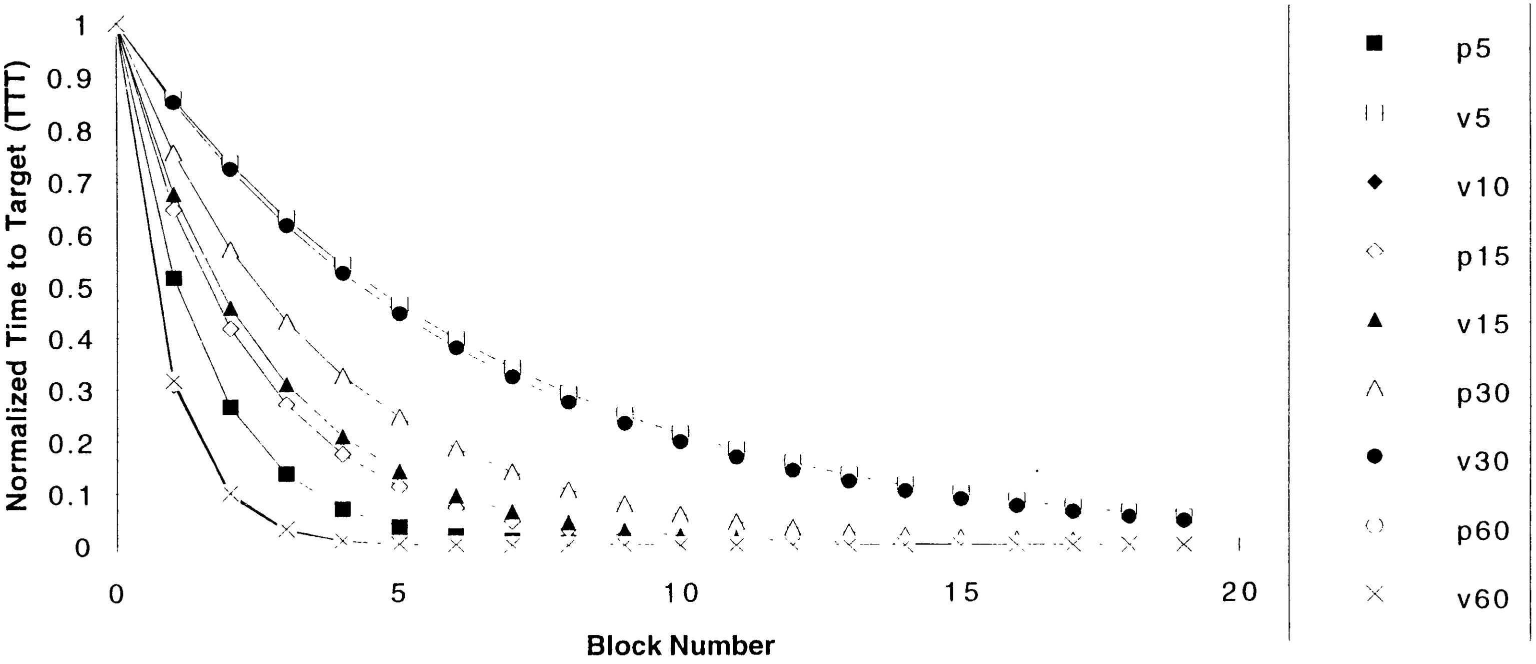
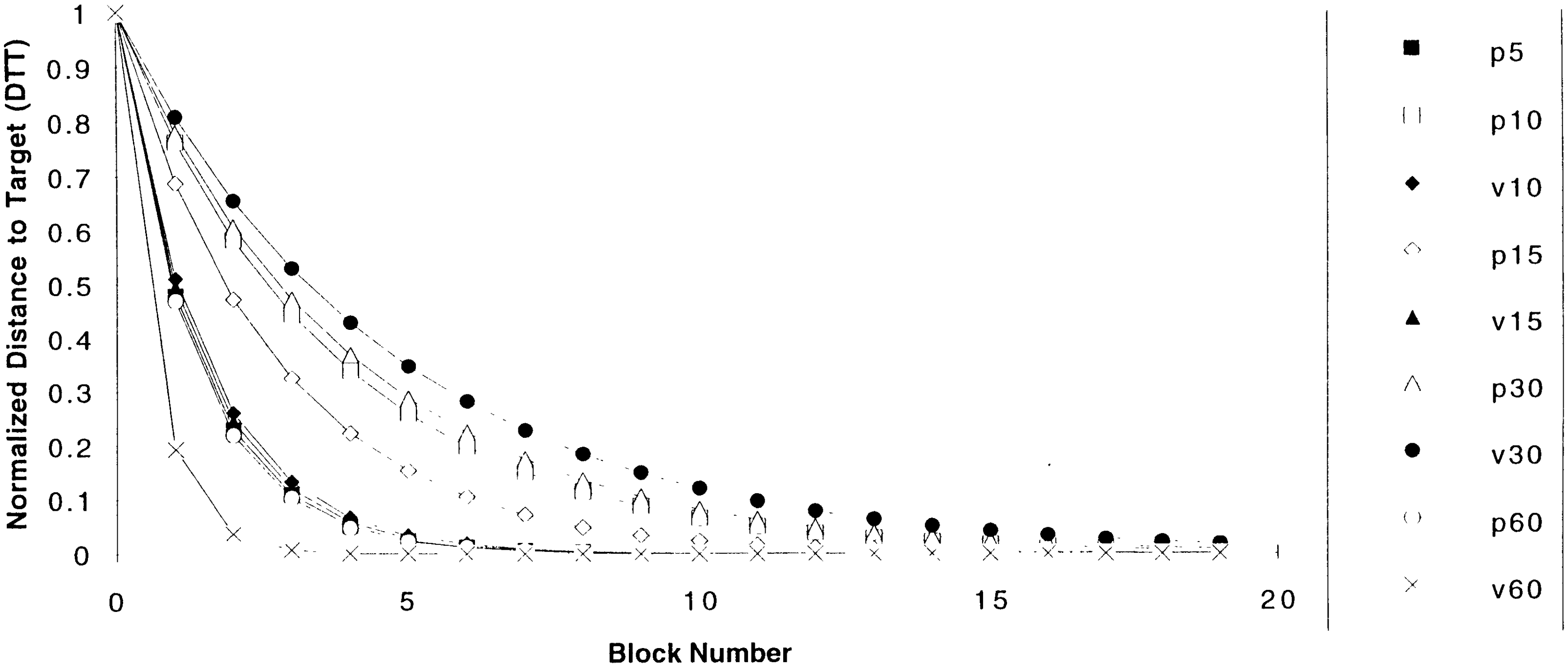


Figure 3.13: Average Distance to Target (DTT) time constants for each rotation condition. In the figure legend, p=position control mode, v=velocity control mode.



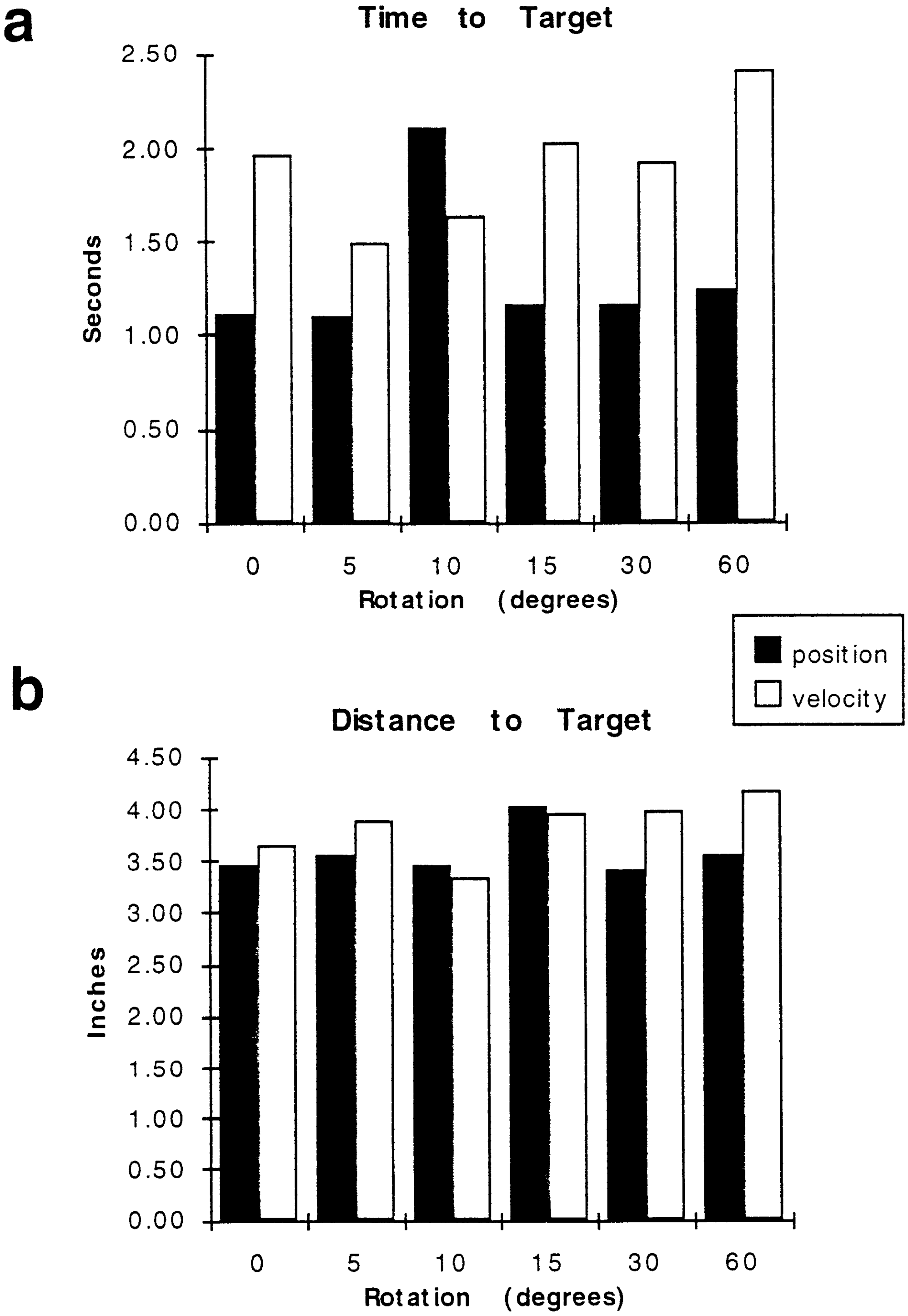


Figure 3.14: Steady-state Time to Target (a) and Distance to Target (b) performance for all rotation conditions. The responses using position control mode are shown with white bars and the velocity control mode responses in black bars.

CHAPTER 4

TELEMEDICINE WORKSTATION, EVALUATION AND TECHNOLOGY ASSESSMENT*

Telemedicine programs are being designed and developed for health care providers and patients who have limited access to medical facilities due to location or cost (Arthur D. Little 1992; Grigsby, Sandberg et al. 1994; DeBakey 1995). NASA has an interest in developing telemedicine workstations for medical ground-support for astronauts during U.S./Russian joint spaceflights and future space station activities. In addition, there is a need for telemedicine systems targeting disaster relief and disaster preparedness, and distance learning in the medical community. Section 4.1 defines telemedicine and gives the motivation for studying the user interface. The user scenarios are described in Section 4.2 and the general telemedicine system architectures are shown. Section 4.3 describes NASA Johnson Space Center's Telemedicine Instrumentation Pack (TIP) which is being developed as the medical workstation to be used by the astronauts. Section 4.4 introduces workstation evaluation and Section 4.5 presents a case-study of The New England Medical Center Telemedicine Program (Patterson 1995). The technical background materials are provided in Appendix A.

4.1 Human-machine interface design for telemedicine applications

Telemedicine refers to the use of telecommunications technology to provide medical care to a patient at a distance from the health care provider. The goal of telemedicine is to improve clinical diagnosis, care and efficiency of treatment whether through satellite communications, ISDN links, or hospital-based Picture Archiving and Communication Systems (PACS) (Kohli 1989; Mukhedkar, Laxminarayan et al. 1990; McClelland, Adamson et al. 1995). Telemedicine systems can deliver care to patients anywhere in the world by combining communications technology with medical expertise.

Telemedicine has been shown to improve medical treatment particularly where specific medium to high risk patients can be identified. For example, pre-selected households were installed with ECG equipment (Thorborg and Sjoqvist 1990). Training in the use of the equipment for clinical staff and home users can present a difficulty, but is

---

* See accompanying demonstration video and web page.

made up for in a reduction in training times for the first responders since there is an expert monitoring the system remotely.

Until recently, most of the research and applications in the field of telemedicine have focused on the technical feasibility and the political infrastructure (e.g., cost-effectiveness, physician and patient acceptance) (Grigsby, Kaehny et al. 1993; Scott and Neuberger 1995). Increasingly, the user-interface is receiving needed attention as more and more telemedicine workstations are being developed for particular user scenarios or applications (Smith and Mosier 1984; Nakano, Nagai et al. 1990). Study of the human-computer interface can be defined as "the discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them" (McClelland, Adamson et al. 1995).

Human-machine interface design is now considered part of the development process of telemedicine systems. Most interfaces are multimedia-based front-ends to complex networks. Videoconferencing is an example of a commonly integrated media (Turner, Brick et al. 1995). The interface is critical because many users lack significant experience with the technologies involved. For example, network protocols are complex can require many set-up parameters. The clinician should be isolated from these hardware requirements. Software designers must adopt a user-centered approach which is compatibility with the user's expectations, provides the user with flexibility and control, explicit structure, continuous and informative feedback, error prevention and correction, and on-line help (McClelland, Adamson et al. 1995).

Previous evaluations of telemedicine systems have focuses on general characteristics and not the user-interface specifically (Grigsby, Sandberg et al. 1994; Bashshur 1995). However, lessons learned from past evaluations are applicable here. In his review of telemedicine evaluation, Bashshur (1995) summarized the past telemedicine projects up to the 1970s with three points. 1) The systems were underused and did not fully exploiting the technical capabilities they had. 2) The projects had narrowly defined function and targeted special groups. and 3) Few conclusions were able to be drawn concerning role or effect of telemedicine in the healthcare delivery system. He also observed that client and provider acceptance were high and increased with experience and familiarity of the system.

Bashshur suggests that three conditions be met before an evaluation. 1) The appropriate environments and specific healthcare needs are identified that would be best met through a telemedicine system. 2) The specification of informational requirements necessary for remote diagnosis, treatment, and follow-up as well as for education. and 3)

An attempt to exploit to the extent possible the technological and system capabilities that are in place. He then suggests that evaluations must consider these three conditions and evaluate the appropriate contexts, optimal system configurations, and the full range of effects of the telemedicine system whether immediate or delayed, intended or unintended, and direct or indirect.

4.2 Defining the User-Scenarios

The telemedicine system architectures for the user-scenarios are shown in Figures 1-3. There is a continuum in both user medical expertise and system requirements.

4.2.1 *The Primary User Scenario*

To provide medical care for astronauts in support of U.S./Russian joint spaceflights and future space station activities. Figure 1 shows a conceptual model of this user scenario which includes a trained flight surgeon at the ground-based telemedicine workstation and a trained astronaut "patient" in space. The astronaut will be using NASA Johnson Space Center's Telemedicine Instrumentation Pack (TIP). TIP is a portable, small suitcase, containing medical instrumentation and support equipment. The specifications of TIP will be covered in more detail in Section 4.4. The third node of the user scenarios would provide access to medical databases and medical consultants or experts.

4.2.2 *Secondary User Scenarios*

To provide assistance in disaster relief and disaster preparedness (see Figure 2) and to provide opportunities for distance learning and retraining in the medical community (Figure 3).

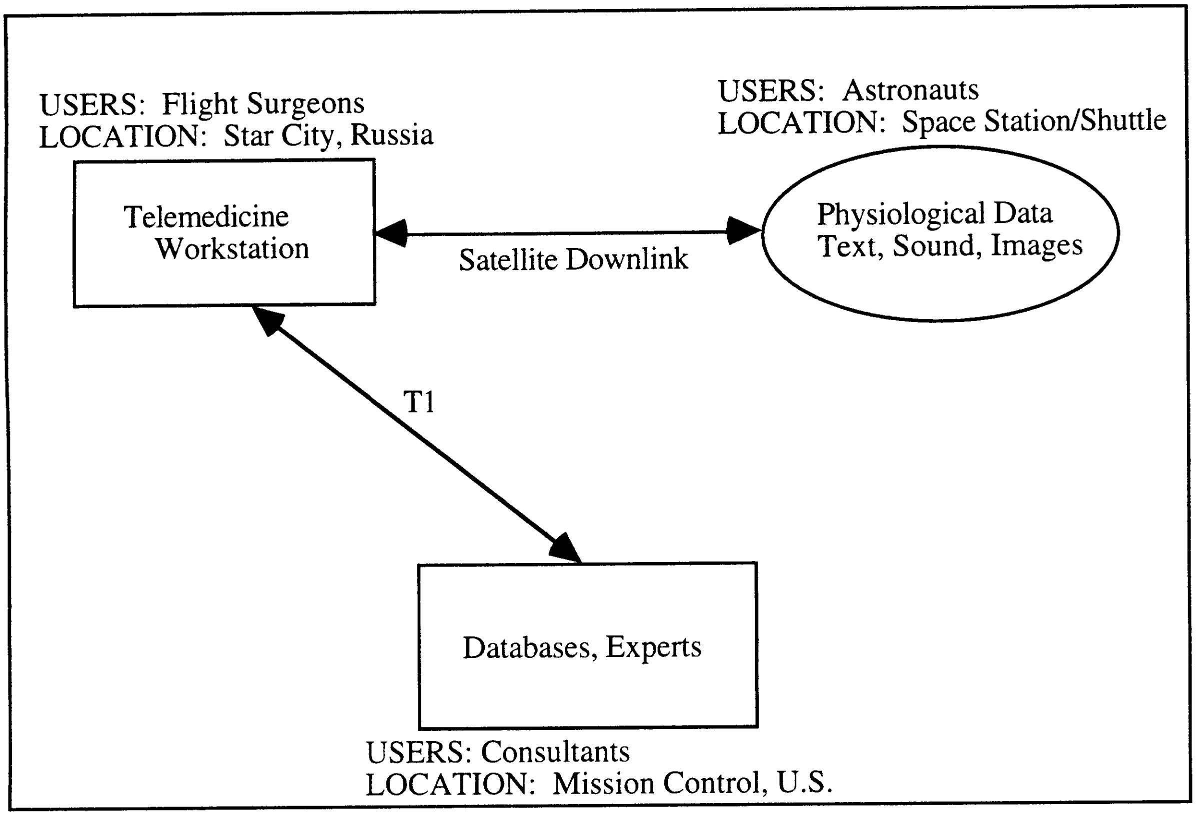


Figure 1: Ground-based support for astronauts

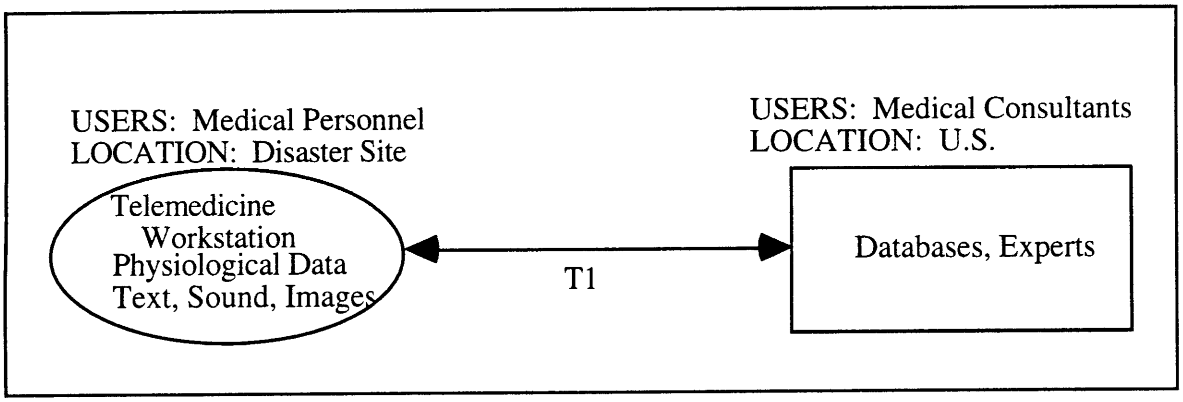


Figure 2: Disaster preparedness and relief.

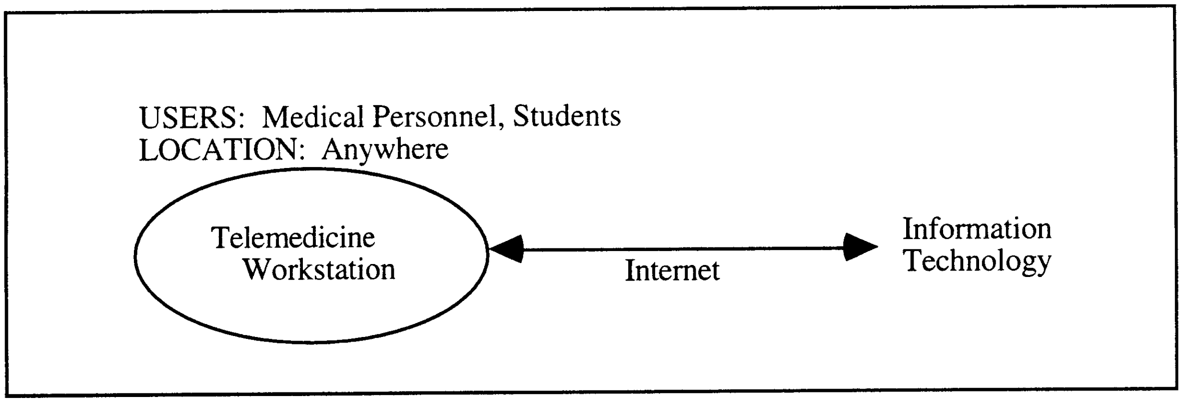


Figure 3: Medical education.

4.3 Telemedicine Instrumentation Pack (TIP)

The Telemedicine Instrumentation Pack (TIP) is being developed by KRUG Life Sciences, a contractor for NASA Johnson Space Center. TIP's purpose is to provide medical instrumentation and support equipment in a portable pack for use by the astronauts on future spaceflights. Video capabilities include eye, ear-nose-throat, skin, and general macro-imaging. Data capabilities include electrocardiogram (ECG) waveforms, heart rate, and blood oxygen saturation. TIP provides audio of the heart, lungs, and bowels. All data obtained can be accessed at serial output ports for transmission to earth. Support equipment includes a flat panel liquid crystal display (LCD), a remote head CCD camera, a light source, and a power supply. The following is a summary of TIP capabilities:

VIDEO: An otoscope and opthalmoscope provide general exams of the eyes and ears respectively. A macro/zoom lens is used for dermatology examination. A rhino-laryngoscope is used for high quality imaging through a fiber optic cable to image nasal, sinus, and vocal chords. A fundus camera is used for video retinal imaging.

AUDIO: An electronic stethoscope is used for the transmission of heart, lung, and bowel sounds. (A piezoelectric crystal located distally from the earplugs converts sound waves to an electrical signal. The audio out is digitized to get a broader bandwidth than is possible with analog signals).

DATA SUBSYSTEMS: A pulse oximeter is used to determine oxygen saturation and heart rate. An ECG monitor is capable of recording 3- 12 lead electrical activity. Blood pressure is automated. An onboard computer multiplexes the data streams for transmission.

DEVELOPMENT: Potential developments include replaceable instrumentation modules and advanced HMIs. For example, a heads-up display could augment or replace the current flat panel LCD. Technologies that would allow a "hands free" operation are also being evaluated.

TIP will have to handle a variety of medical scenarios. The more preparation that can be done before launch, the better the care of the astronauts. Potential medical scenarios are as follows:

Medical Scenarios

1. Daily physicals/General check-ups
2. Floating particles in the eye

3. Decompression sickness
4. Cuts/abrasions
5. Gastrointestinal disorders (constipation, diarrhea)
6. Toxic spills
7. Heart attack

Each of the medical scenarios will involve varying levels of intervention from a ground-based flight surgeon. For example, data from daily physicals would be continually down-loaded for evaluation. A small particle in the eye could be dealt with completely independent from ground-based intervention whereas a heart attack would involve two-way data transmissions of physiological data and intervention.

4.4 Workstation Evaluation

Telemedicine is being conducted in hospitals and universities all around the world, using a wide variety of equipment. This reports concentrates primarily upon general workstations, suitable to fill many different roles. Technical background for the components of a telemedicine workstation is provided in Appendix A*; Platform, display, input devices, and networking components are summarized. The role a general workstation can fill is three-fold, with each tier increasing in complexity:

Tele-education/Reference: Essentially the system provides access to clinical data and images, as well as hypertext images and diagnostic aids. The system may be connected, either by modem or network to a LAN or the Internet, to facilitate communication, but this aspect of the project is secondary. Such a system has little to distinguish it from a conventional workstation, save its dedicated purpose.

Portable Service Provider: This system, adds teleconferencing capability to the previous workstation. Images and clinical data may be sent from a hospital, local or remote to the station, and the reverse. Such a system is equipped with some fairly simple image capture hardware, and provides for either voice or text communication with a remote sight.

Full Workstation: This is a system designed to be used at a stationary site. Separate hardware is available for videoconferencing and image capture, links are provided for a variety of different modes of communication, and the system provides integrated control and access to remote cameras, instruments, and patient records.

---

* Thanks to UROP Michael Metzger for assistance with compiling the technical overview and video demonstration.

Table 1 shows a compiled list of existing vendors and universities whose products and projects are being used.

Table 1: List of Vendors and Universities

3M Medical Imaging Systems	Dupont Medical Products
A&S Communication	Telemedicine Alliance of Health Organizations
AAC	Boise State University Center for Health Policy
AC&E Limited	University of Arkansas
Acuity Imaging Inc.	National Laboratory for the Study of Rural Telemedicine
ADCOM Electronics	DeJarnette
Agfa Division Technical Imaging Sys	DR Systems
American Telecare, Inc.	Corabi
Analogic	
Andries Tek	University of Washington, ICSL
Apollo Software	Lockheed Martin
CAE-Link	Texas TelemedicineProject
CPI/MicroAge	University of Texas Medical Branch
DataView Imaging	Gammex RMI
EMED	GE Medical Systems
HealthCom	Medical Image Management, Inc.
Heraeus Surgical	Los Alamos NL (Sunrise)
ICON Medical Systems	ICE Communications, Inc.
Loral Medical Imaging Systems	J.D. Technical Services/RediVu
md/tv	Applied Science Associates, Inc.
Medweb	Cyberspace Telemedical Office
Olicon Imaging Systems	Line Imaging Systems
Peirce-Phelps	New England College of Medicine
Radman	Magnetic Research, Inc.
Roche Image Analysis Systems	AT&T (Picasso)
Rstar	CompuMed
Scottcare Corp	University of Pisa (IMPHONE)
Shure	Mountaineer Doctor Television
Specialized Home Care, Inc.	BioPac
Stryker Endoscopy	Georgetown Medical Center
Turcan-Wingard Associates	PictureTel
United Medical Network	High Plains Rural Health Network, and
Video Dynamics	Decisions Systems Group (Brigham and Women's Hospital and Harvard Medical School)

The best workstation depends on the user's needs. All of the workstations evaluated will be entered into a database that will be part of on on-line search engine. The user defines her telemedicine workstation needs via a checklist interface. The system will search for the workstations that fulfill those needs. Each system will be described in a standardized format, shown as follows:

System Name:	
Vendor:	
Contact Information:	
Pricing:	May not be applicable for academic systems
Processor Type:	Will also include benchmarks to evaluate the relative capabilities of different families of processor.
Display:	Resolution, color depth, and screen size.
Communications:	Communication protocols which are supported transparently, as well as the possible data transmission rates, and any requirements for the link between computers.
Interface:	Both hardware (keyboard, touch screen, voice) and software (access to remote databases, authoring tools, Graphical interfaces, etc.)
Video capabilities:	Full motion video (and sound).
Expansion:	Options which may be purchased, either through the vendor or, for open architecture systems, supported third parties.
Role:	The role the system was designed to satisfy. Possibilities include image transmission and archiving, teleradiology, videoconferencing, general, and others.

Finally, there will be a short paragraph describing any other relevant or unusual features of the workstation in question. For workstations which have been studied or had their capabilities analyzed in the literature, the relevant references will also be reported.

Three workstations evaluations are presented here as representative types of workstations in use that will be incorporated into the database. Workstation A, the Lockheed Martin Telemedicine Workstation, is a general purpose workstation whereas Workstation B, the MediaStation 5000, is a special-purpose workstation utilizing third-party technology. Workstation C, the CADx, is a high-end, pre-packaged, special purpose workstation.

A. General Purpose Workstation

System Name: Telemedicine Workstation
Vendor: Lockheed Martin
Contact Information: James P. McCormick
Project Manager, Medical Integration
Lockheed Martin Information Systems
12506 Lake Underhill Road, MP830
Orlando, FL 32825-6401
Phone: 407 826-6401
Fax: 407 826-6539
Pricing:
Processor Type: Pentium, 66 MHz
Display: 17" 1280x1024 color monitor
Communications: Codec 384 Kbps video channel, V.35, 802.3, T1, ISDN
LAN: Ethernet, Super Ethernet, ATM FDDI
Interface: Touch screen, keyboard, mouse,
Video capabilities: 384 Kbps video
Expansion: Expanded video capabilities, remotely controlled cameras, NTSC instrumentation package (Otoscope, Opthalmoscope, Dermascope, etc.), Super VHS VCR, Full Motion Video display, capture, store and forward.

Role: Integrated, general purpose workstation.

The Lockheed Martin Telemedicine Workstation is a flexible, open architecture general purpose workstation. It is designed to allow point to point videoconferencing across a 384 Kbps phone link, and can take advantage of the higher bandwidth of a dedicated T1 line, or a satellite link with additional hardware. The software is the company md/tv's HouseCallTM, a Microsoft Windows based package. The system allows access and manipulation of 4GL multimedia databases, control of remote cameras, electronic record keeping, integrated multimedia and videoconferencing capabilities, and annotation of images. The primary interface is icon driven, by use of the touchscreen or the mouse.

B. Special Purpose Workstation: Using 3rd Party Technology

System Name:	MediaStation 5000
Vendor:	University of Washington Image Computing Systems Laboratory
Contact Information:	
Pricing:	
Processor Type:	Texas InstrumentsTMS320C80 2 billion operations per second
Display:	1280x1024 capable
Communications:	NA
Interface:	NA
Video capabilities:	MPEG-1 compression of 352x240 pixel video @ 30 Hz
Expansion:	NA
Role:	Advanced Digital Signal Processing

The MediaStation 5000 is a plug in board for a Local bus or PCI PC which greatly enhances its signal processing capabilities. The board can perform real time MPEG-1 compression and decompression at 30 framed a second on SIF video, with 16 bit 44 kHz audio. The board is programmable and can also be used for compression, two and three dimensional image processing and graphics, and other DSP intensive tasks. Projects incorporating the board include the GSP9 a general purpose telemedicine workstation (using the ITU H.320 standard for videoconferencing), the GSP7 a real-time optical imaging workstation, and the GSP8 a tool for visualization and analysis of three dimensional ultrasonic angiographs.

C. Special Purpose Worksation: High-End Pre-Packaged

System Name: CADx

Vendor: Georgetown Medical Center (ISIS Center), Cray Research
Contact Information:

Pricing:
Processor Type: Cray J916
Display: 2048 x 2048
Communications: Still under development, to support HIPPI, ATM and other high speed communications standards
Interface: Non-standard
Video capabilities: none
Expansion: under development
Role: Filmless radiology

The CADx software and system is an attempt to bring about filmless radiology. The initial goal is to achieve this within the hospital itself, while designing the system for easy transmission and receipt of images to allow a transition to a telemedicine framework. The system has two primary functions. First it will receive, store and catalog all X-rays replacing conventional film with direct digital capture and computer display. The design goal is for the system to display the image in a fraction of a second, rather than the several seconds a high end workstation would require. Second, the system aids with the analysis of the image. Currently, the analysis is restricted to finding microcalcifications in mammography data in order to detect breast cancer. The system can detect microcalcifications in the range of 100-200 microns, about the width of a human hair.

The next phase of the project is to expand analysis to other ailments, and areas of the body. In addition, Georgetown Medical Center is being connected with area hospitals, allowing them to transmit their mammography data for automatic analysis.

4.5 Case Study: The New England Medical Center Telemedicine Program

As a technology designed and developed to provide a tool to health care providers in unique situations, the human/machine interfaces should also be evaluated. The interface chosen will have a direct relationship to the health care providers' performance. Human performance during telemedicine activities will be crucial in determining the difference between use and nonuse of a system and the difference between diagnosis and misdiagnosis. Determining the important components of the human-computer interface for each telemedicine system will have ramifications for the work-station design, the needed computer literacy, and level of technology used. A related component to operator performance is the amount and type of training needed for effective use of the telemedicine workstation. Therefore, training duration and learning-curve effects should also be considered in workstation design.

The telemedicine system chosen for our case study is that developed by the Medical Image Management Company for the New England Medical Center Telemedicine Program. John Patterson, Vice President and CIO, outlined the NEMC program at the Telemedicine 2000 conference in Lake Tahoe, CA, in June, 1995. He summarized the NEMC Telemedicine Program as being clinically driven, affordable, requiring low bandwidth (less than anticipated), and using third party product development, and an "open" architecture environment (i.e., the hardware is independent of the server). A summary of the workstation is shown on the following page as Workstation D.

Amidst national focus on downsizing health care systems, there is a drive to increase the efficiency of health delivery systems, to achieve high quality, low cost care. Some of the driving forces behind the development of the NEMC telemedicine applications are 1) the maintenance and expansion of traditional referral relationships, 2) the drive to expand services; provide national and international diagnosis and consultation services, 3) reduce the cost of health care 4) increase the access and quality of medical care for rural, distant, and underserved populations

NEMC has several deployed telemedicine programs in Massachusetts. Telepsychiatry was initiated for nursing home geriatric populations with medical rounds using 384 Kb links. Second opinion consultations on fetal ultrasound monitoring and care management were obtained using store and forward data transmitted on 256 Kb lines. Ophthalmologic consultation, diagnosis and care management and angiogram consultation and care management cases were also observed.

D. Open Architecture Workstation Used By NEMC Telemedicine Program

System Name: N/A (see On-Call interface software)

Vendor: Medical Image Management
Contact Information.: 815 Montgomery Street
Fall River MA, 02720
Phone: 508-672-2931
Fax: 508-672-5008
Pricing: $15,000-$20,000
Processor: Intel 486
Display: any (1024x768x256 and above likely)
Communications: ISDN, 10baseT Ethernet, fiber optic capability
Interface: Windows 3.1 based,
Video capabilities: PictureTel Full motion video, PCS-100
Expansion: Cardiac angiogram, ophthalmology, obstetric ultrasound, home health consultation, pediatric cardiology ultrasound, comprehensive patient consultation, echo cardiology, medical image acquisition, management, archiving

Role: General Purpose

Medical Image Management has produced a workstation based almost on open standards. The system software and hardware can be added to an existing PC running Windows or a Macintosh, or a PC can be purchased along with the system. Using On-Call software by The Method Factory, the workstation allows access to on-line databases, lets the physician develop customized forms for use by the patient or secondary care giver, and allows for easy control of remote cameras and instrumentation.

Two major telemedicine systems are currently being deployed, one in Latin America and one in support of Pediatric Leukemia patients in Massachusetts. The Latin America project is a comprehensive telemedicine system for a 400,000 member managed care population in Buenos Aires. This system is a SmallTalk shell around the client. This system architecture is shown in Figure 4. The second system is a comprehensive home health system incorporating videoconferencing, medical appliances, bi-directional home to hospital results messaging, expert system which provides diagnosis and educational services. One of the goals of this system is to prevent rehospitalization which can be costly as well as traumatic. Dr. Larry Wolf at the NEMC Floating Hospital for Children is testing this system with Pediatric Leukemia patients.

Future plans of the NEMC Telemedicine Program include expanding the applications of these comprehensive clinical "workstations" to nursing homes, community heath centers, correctional populations, and psychiatric hospitals. These workstations would also include care management, outcomes and productivity improvements.

4.5.1 *Communication Links: ISDN*

Integrated Services Digital Network (ISDN) is a set of communication standards which allow a single wire or optical fiber to carry voice, digital network services, and video. ISDN is intended to replace the current Public Switched Telephone Network, and supports both old copper wire and switching systems and newer fiber optic systems. ISDN accomplishes this by defining several different types of channels. ISDN also provides additional flexibility by providing users with access to more than one channel at a time for data transmission.

B Channel: This is a 64 kbps channel which carries customer information, or provides a connection based or connectionless link.

D Channel: This is an access channel carrying control or signaling information, or providing a data link.

Basic Rate Interface: The BRI consists of two B and one D channels. This channel can carry two simultaneous voice and one "data" communication, or provide a communication link with and bandwidth of 128 kbps.

Primary Rate Interface: In the United States, this consists of 23 B channels and on 64 Kbps D channel, or a 23B+D connection. This provides a total bandwidth of 1.544 MBps, which corresponds to that of a T-1 trunk, and provides service equivalent to DS-1.

Table 4.2: NEMC Telemedicine System Cost

Telemedicine System Costs $ (Installed & Trained) for Comprehensive Remote Diagnosis and Consultation System

System Element	Cost	Total Cost
Baseline system configured to support basic video conferencing, ultrasound and echocardiology consultations	<$14,000	<$14,000
Add cardiac catherization digitization and transmission capability	$8,000	$22,500
Add high performance camera & lightbox for advanced patient consultations and transmission of slides, EKGs and initial static medical image review	$5,000	$27,500
Add medical appliances(1) for patient interview consultation	$1-10,000	$28,500-37,500
Add static diagnostic image digital acquisition/review capability	<$21,500	$50,000-59,000

(1) The exact complement of medical appliances would be selected by the clinical users as required.

BroadBand standards for ISDN have also been designed, which will rely upon implementing ISDN on top of the SONET protocol. Channel classification of the different services is provided below.

{T-0}	64 Kbps
{T-1}	1.544 Mbps
{T-1C}	3.15 Mbps
{T-2}	6.31 Mbps
{T-3}	44.736 Mbps
{T-4}	274.176 Mbps

At present, not all areas of the United States have access to ISDN services, and those areas that do often only have T-0 and T-1 capability.

4.5.2 *User Interface*

User interfaces employing full-motion video and other such multimedia information require broadband networks as well as advanced human interface design. Broadband networks such as ISDN have been used for telemedicine applications (McClelland, Adamson et al. 1995) as well as Picture Archiving Communication Systems (PACS) (Kohli 1989). With increased bandwidth and graphical user interfaces, more attention has been paid to the interface design and in fact, an ISDN terminal adapter was developed (Nakano, Nagai et al. 1990) with a friendly interface based on proposed guidelines (Smith and Mosier 1984).

The NEMC Telemedicine Program in collaboration with The Method Factory, Inc. have developed a user interface, "On-Call". On call is an object-oriented software developed to provide a flexible interface for health care providers and their patients. On-call has two modes. One mode is the author's mode in which a health care provider could construct a multimedia checklist. A library of multimedia interface buttons are clearly marked on the screen which can be used to create a hierarchical patient instruction software. Depending on patient answers, a pre-taped instructional video may be triggered, further written information may be requested, or a video-conference with the doctor may be initiated. The author of the checklist can set the severity index of all answers so that if a certain threshold is reached a videoconferencing call may also automatically be initiated. A "constellation," linking symptoms can also be created that can then elicit a pre-programmed response. In the patient mode, a seamless program appears that explains and instructs while keeping track of all the responses. All patient records are updated both on the On-call workstation as well as at the hospital. On-call will be evaluated during the Pediatric leukemia project as well as the Argentina project.

4.5.3 *Cost*

The NEMC Telemedicine Program believes that it can reduce costs and liability while improving access to and quality of care for state and federally funded mental health, community health center, nursing home and correctional populations. One of the advantages of the NEMC Telemedicine Program's approach is that individual doctors or health-care providers can make affordable commitments to a telemedicine system without an entire institution committing costly resources to developing a program. The commitment to telemedicine services is a segmented decision process that may be only $700/month to support. Bandwidth costs are also low since ISDN can be provided at $42/month.

The original investment of a baseline system supporting videoconferencing, ultrasound, and echocardiography is less than 14,000 as shown in Table 2.

4.6 References

Arthur D. Little, I. (1992). Telecommunications: Can it help solve America's health care problems? Arthur D. Little, Inc., Cambridge, MA.

Bashshur, R. L. (1995). “On the definition and evaluation of telemedicine.” Telemedicine Journal **1**(1): 19-30.

DeBakey, M. E. (1995). “Telemedicine has now come of age.” Telemedicine Journal **1**(1): 3-4.

Grigsby, J., M. M. Kaehny, et al. (1993). Analysis of expansion of access to care through use of telemedicine and mobile health services. Center for Health Policy Research, Denver, CO.

Grigsby, J., E. J. Sandberg, et al. (1994). Analysis of expansion of access to care through use of telemedicine and mobile health services. Center for Health Policy Research, Denver, CO.

Kohli, J. (1989). “Medical imaging applications of emerging broadband networks.” IEEE Community Magazine **27**(12): 8-16.

McClelland, I., K. Adamson, et al. (1995). “Information issues in telemedicine systems.” J. of Telemedicine and Telecare **1**(1): 7-12.

Mukhedkar, D., S. Laxminarayan, et al. (1990). “Global telemedicine using INMARSAT satellite system.” IEEE, SPIE **1355**: 19-24.

Nakano, S., N. Nagai, et al. (1990). “Network control method and human machine interface design for ISDN multimedia terminal.” IEEE J. on Selected Areas in Comm. **8**: 3.

Patterson, J. (1995). The New England Medical Center TeleMedicine Program. Telemedicine 2000, Lake Tahoe, CA,

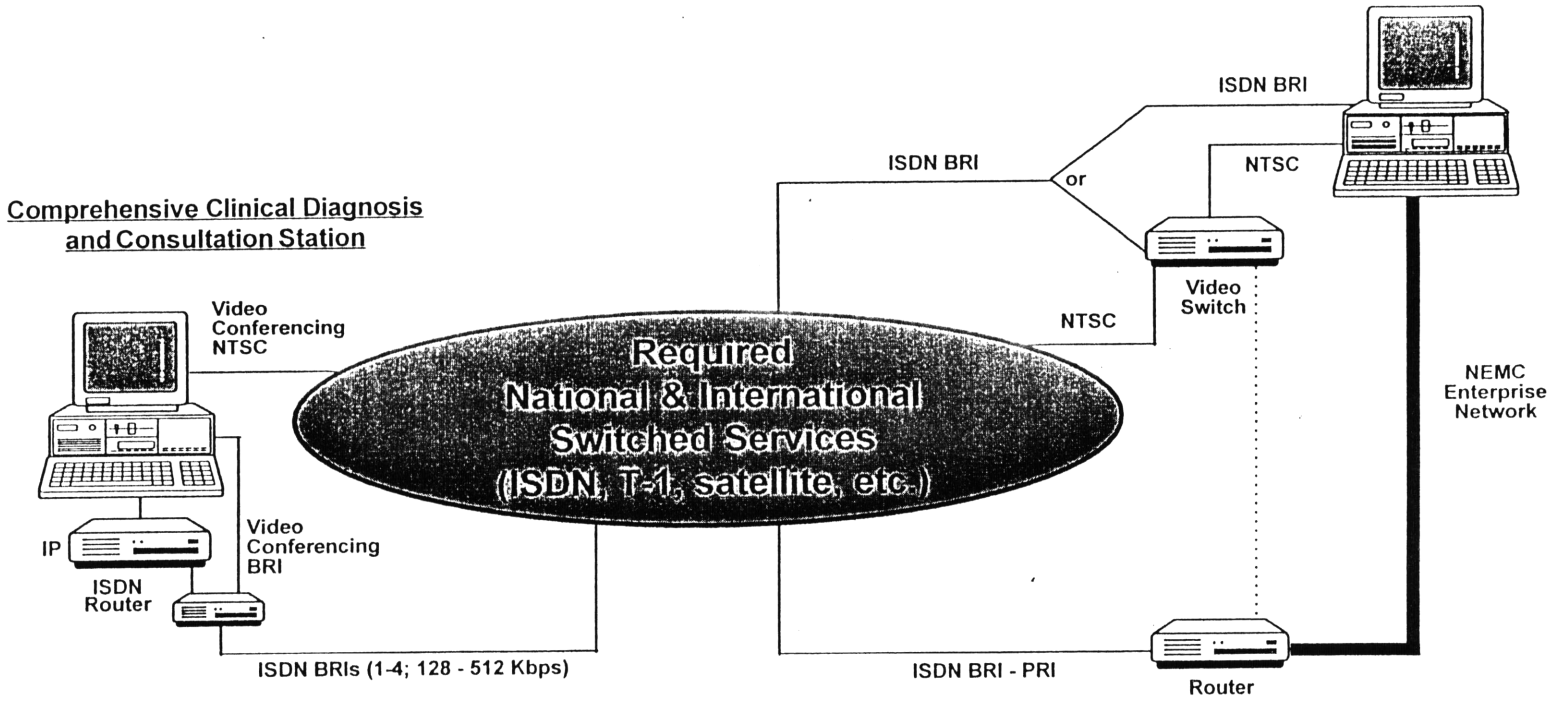
Scott, J. and N. Neuberger, Ed. (1995). The Telemedicine Monitor: a guide to telemedicine policy, programs, and opportunities. AJ Publishing, Inc.

Smith, S. M. and J. N. Mosier (1984). Design guidelines for user system interface software. The Mitre Corp, Bedford, MA.

Thorborg, S. and B. S. Sjoqvist (1990). "Mobimed -- a telemedicine system for mobile monitoring of physiological parameters." IEEE, SPIE **1355**: 32-35.

Turner, J., J. Brick, et al. (1995). "MDTV telemedicine project: Technical considerations in videoconferencing for medical applications." Telemedicine Journal **1**(1): 67-72.

Figure 4.4: NEMC Telemedicine System Architecture



APPENDIX A

TECHNICAL BACKGROUND

1.0 Telecommunication-Based Levels of Telemedicine

Just as there are many different aspects of treating a patient, from surgical intervention to accurate record keeping, so there are many activities which fall under the umbrella of telemedicine. These can be broken down roughly in terms of the degree of technical sophistication required to perform each activity.

Text Data Transmission : This is the simplest level of telemedicine. It simply involves passing a (secure) message from one computer to another, something which happens millions of times each day. This capability can be very easily and inexpensively achieved.

Image Transmission : Image transmission require greater communications bandwidth and processing power. The image must be compressed, transmitted, decompressed, and displayed at the remote location. This requires a processor of sufficient power to handle the compression and decompression on each end, a link quick enough so that the image can be transmitted in a reasonable amount of time, and a screen with sufficient resolution to display the resultant image.

Tele-Consultation: In Tele-Consultation, there is real-time interaction between the doctor and the patient. This could involve transmission of telemetry, a videoconferencing link, rapid transmission of images, or all of the above. The connection is not sufficient for the doctor to serve as the primary diagnostician, but may be used for second opinions, and to screen patients whose condition is not time-critical.

Tele-Presence: Tele-Presence requires a higher degree of control than Tele-consultation. Images and data is sufficient for the physician to be the primary diagnostician, and medical imaging, visual data and audio data as well as instrument telemetry are all available. Skilled assistance on the other end of the link need not be that of a physician.

Remote Intervention : This involves physical interaction with the remote patient. A robust, very high bandwidth link, specialized three dimensional pointing devices, teleoperated waldos, and a great deal of computational power are all required to bring this about. For surgery, a full virtual reality environment, complete with tactile feedback would be required. Although some research is being conducted in these areas, a general affordable workstation with these capabilities is still many years away.

2.0 Choice of Platform

There are three main platforms to choose from in the U.S, A UNIX RISC system, an Intel system, or a Apple Macintosh based system. Each system has its own advantages and disadvantages, especially when speaking of the Intel and UNIX systems which may vary considerably from vendor to vendor. The UNIX system has some advantages in robustness, the DOS in cost, the Macintosh in ease of maintenance and use, but all can perform the same basic tasks. The designer of the particular system must decide which aspects of the system are to be most important, and decide accordingly.

UNIX systems: In general, UNIX systems are the most technically sophisticated of the three choices. Versions of the UNIX operating system exist for many high end RISC processors. Between UNIX, the operating system, and Xwindows, the interface which users generally see, the details of the underlying hardware is abstracted away. As a result it is generally much easier to port a program from one UNIX system to another than it would be to port that same program to a Windows system or a Macintosh. Ideally, this prevents dependency on one system vendor for hardware. In actuality, the function of the software in question will settle portability issues. For example, a program which is primarily devoted to image transmission across a network, or file manipulation is likely to be very portable. However, a program which relies upon a digital image capture board which is only produced by one particular manufacturer will still require the board no matter how portable the rest of the system is.

In addition to portability, cutting edge UNIX systems tend to be faster than cutting edge Intel or Macintosh systems. This is largely because of the portable nature of UNIX. Since UNIX code is generally distributed in the form of source code, each time a new processor is developed, the programs used can be recompiled and re-optimized to handle even radical redesign. Intel's x086 series, by contrast must maintain backward compatibility with chips produced over a decade ago. In a sense, each processor must be able to do everything its predecessors could, as well as the new tricks that make it particularly speedy.

UNIX as an operating system also tends to be less susceptible to system crashes than the Macintosh or the DOS/Windows platform. It is an operating system that networks easily, as most of the theory of modern networking was worked out on UNIX platforms of one kind or another.

However, UNIX platforms tend to be expensive, requiring large amounts of RAM (16MB+) and sizable hard drives to deliver acceptable performance. In order to keep systems up and running, experienced system administrators are required, and adding hardware or software is also a job for an expert. UNIX systems tend to be less accessible to a novice, and software for them tends to be more in keeping with an academic environment than a business one.

SUMMARY

PROS	CONS
Good portability	Expensive
Robust, crash resistance systems	Not a large business market
High performance processors	Often require technical sophistication

INTEL/DOS/WINDOWS: The greatest single advantage the Intel platform has is its enormous size. Far more x086 based computers are sold than all other kinds combined. This keeps hardware and software costs low, and provides for rapid turnover in the marketplace. The development tools for this platform are both prolific and excellent, and there are a huge number of choices for any particular piece of hardware or software. This keeps prices low, and gives the user a great deal of flexibility, but also means that a great deal of information must often be sifted through to obtain the needed system.

There are essentially four operating systems available for an Intel PC. The first is DOS. A barebones DOS system is almost certainly inadequate for any workstation that must deal with images, as DOS programs are effectively limited to using only the lower 640Kb of system memory. This is obviously impractical when dealing with images that may be many times this size. Windows provides a uniform graphic interface, access to a system's full memory, and a level of hardware abstraction. Windows is widespread, and supports a huge amount of hardware and software, but it is technically less sophisticated than UNIX, and more prone to crash. Then there are both OS/2 and Windows NT. Both systems are on a par with UNIX for technical sophistication, robustness, and crash protection. Unfortunately both operating systems are comparatively new, and have a relatively small base of supported hardware and software.

SUMMARY

PROS	CONS
Inexpensive	Operating system unsophisticated
Well-known	Possibility hardware/software
Excellent development tools	Unsupported
Wide choice of vendors	Knowledge needed to separate good vendors from bad

Apple Macintosh: The Macintosh's great advantage, is that at present, it is a single vendor computer. This means that software and hardware setup and configuration is very simple. The Macintosh also pioneered the Graphic User Interface, and Macintosh software tends to possess a consistent look and feel.

The new Mac's are based on the PowerPc series of microchips, the result of a collaboration between Apple, IBM, and Motorola. These are Risc chips, and in general provide the best computational power at a given cost of any of the system mentioned. For a complete computer however, the competition and larger sales of Intel based systems tend to make them slightly cheaper.

The Macintosh operating system is in a state of flux at present. The current version is probably on par with a DOS/Windows system -- more prone to crash than a UNIX system, but still sophisticated enough for intense use. The next version should be more powerful, and on a level of sophistication equal to that of a UNIX system, but as of yet it does not exist.

SUMMARY

PROS	CONS
Easy set up, maintenance, use	Small user base
# of vendors	
Strong development environment	

3.0 Display

Monitors : Most computer monitors today can display a wide range of resolutions and color depths from 640x480 pixel 4 bit (16 color) VGA to 1024x768 24 bit (true color). The difficulty arises again in the wide range of uses a generalized workstation must be put to. Ordinary television is of much lower resolution than virtually all computer monitors, but a television program achieves much greater apparent resolution by being able to zoom in and zoom out at will. The difficulty lies in the fact that there are two conflicting standards at work. A doctor involved in a face to face consultation with a patient can easily accept television quality video, but she wants the image to be large, not confined to a tiny corner of the computer screen, in order to foster the illusion that the patient is talking to a doctor, not a computer. By contrast, a doctor who must examine an X-ray or the results of an MRI scan will want a display with the finest resolution. Currently existing display technology is sufficient for X-ray images (2.5k by 2k by 12 bit gray scale) but other applications such as mammography require a greater resolution than has been attained by any commercially available monitor.

For applications in which a 1280x1024 true color image is sufficient, monitor size also becomes an issue. Such images can be displayed on a monitor as small as 14 inches (with a still smaller view area) or as large as 21 inches (or larger). In general, a greater image size means less eye-strain, and easier use over the long run, but adds greatly to the cost and weight of the system. In addition, large monitors are far more prone to exhibit minor imperfections which distort the image in one way or another, and as a result must be chosen far more carefully.

Image Capture: Every image that is viewed on a computer screen must be digitized and entered into the computer somehow. There are three main ways of accomplishing this.

Flatbed Scanners: Flatbed scanners are dedicated devices which scan a document or a radiology film at a specified resolution. Inexpensive scanners with image sizes of 1280x1024 and 300 dot per inch resolution are fairly common. More powerful and expensive scanners also exist which are sufficient to input 2.5k by 2k radiology film at 8 bit per pixel gray scale.

Video Cameras: A video camera may display a field of view, or it may be mounted above a light board in order to read documents or films. The best video camera are limited to a resolution of approximately 1280x1024 pixels.

Direct Digital Capture: For certain forms of medical imaging, such as MRI scans and CT scans, the image may never be recorded on a film, but may be directly recorded as digital data. In this case, displaying the data on a computer is largely a matter of knowing how the information is coded. many forms of radiology are moving toward direct digital captures, and away from films.

Real time video: Real time, NTSC quality video requires a communication bandwidth of 90 Mbs/second. There are a large number of compression schemes which deliver video at various bandwidths, as well as more general compression applicable to any series of images, such as MPEG. Real time is generally defined as 30 frames a second.

3-d Models: As telemedicine grows more sophisticated, and 3 dimensional images are transmitted, display becomes quite a different prospect. While the previous problems of resolution and bandwidth still exist, computational resources become equally important. A great deal of computational power is required to display, at an arbitrary angle, the a two dimensional representation of a three dimensional image. The are a number of systems (notably SGI) and add on boards which can perform these calculations, thus removing that burden from the CPU. These will become increasingly important as more complex information is translated. It is unlikely, however, that they will play any role in a general workstation in the immediate future.

Stereoscopic Display: A stereoscopic display is one which presents a slightly different image to each eye, creating a sensation of depth. These displays fall into to broad categories. The first involves a conventional computer monitor, and a set of dedicated glasses. By opaquing each eye on the glasses many times each second and carefully controlling the image on the monitor, the illusion of depth results. The second method of achieving this effect is to place two, very small monitors in a set of glasses, and have each image differ slightly. At present, stereoscopic displays are not cost-effective for most applications.

4.0 Input Devices

Keyboard: While the keyboard is the usual choice for entering large quantities of textual or numeric data, it does have several disadvantages. An inexperienced user can be extremely slow, and its use is in no way intuitive in a graphical environment. Nevertheless it is probably still the best, and certainly the most popular way to input text into a computer, and an indispensable part of most systems.

Mouse/Trackball: What the keyboard is to text, the mouse is to a display screen. Mice combined with a uniform display allow a user to easily select from a broad list of options, designate areas of interest, and navigate an onscreen "pushbutton" interface. Mice generally need a fairly large (1.5' x 1.5') flat area in order to be used, and a fair amount of elbow room. Trackballs, which require less room are preferred by some, and can be made an integral element of the keyboard.

Pen based system : A pen based computer generally looks much like a child's Etch a Sketch, minus the knobs. An imperceptible grid built into the display surface picks up the pressure of the stylus, and translates the motions of the pen into actions. Such systems tend to be very portable, and extremely appropriate when used as checklists, or for other purposes not involving large amount of data entry. Most pen based systems also have strong handwriting recognition capabilities, although these of course require that the person write legibly. Although a complete novice could use a pen based system more quickly than a keyboard system, after only a few hours of practice, most people can type more quickly and accurately than they could write with such a system. The portability is perhaps the most important aspect of these systems.

Touch Screen: Similar technology to the pen based system, only incorporating a stand alone monitor, rather than a pad and stylus. Touchscreens can quickly used with no training, but they tend to be very limited in their ability to input information. Good for choosing among a limited number of options, but probably not a viable choice for a user of any experience. One obvious downside to the touchscreen is the inevitable fingerprints which collect on the display area as it is used.

Speech Recognition : Speech recognition in a fairly limited form is already broadly available. On the positive side, it provides a very natural, hands-free way to communicate with the computer. On the down side, current systems can tend to be finicky about certain users, or the level of background noise. They require moderate computational power, and require users to speak in a fairly small vocabulary of clearly dictated commands. While attempts to decode both continuous speech and full sentences are underway, both place unreasonable demands on the processing power of the computer, and even so are not fully satisfactory. Speech recognition could be ideal for manipulation of images during surgery, or when the hands are otherwise occupied, and the command set is small (larger, smaller, bright, next, etc.). A system which could accept the command "bring up Mrs. McGillicuddy's records" might even be possible, although the backup keyboard system would probably be more efficient.

5.0 Networking

One of the central issues in designing a telemedicine workstation is determining how it will transmit and receive the necessary information. The speed of the link between the two computers determines absolutely the amount of information that can pass between them.

In order for the information to be stored in a computer, and then transmitted, it must be converted from analog to digital form. The way this is done depends on the type of data involved, but follows the same general principles. Essentially, the analog data is sampled, its values are tested at regular intervals and recorded. The more accurately the values can be stored, and the more frequent the sampling, the more closely the digital information reproduces the analog original. If a low sampling rate and value range is chosen, information is likely to be lost. If the sampling rate and accuracy equals or exceeds the precision of the instrument which captured the data originally, then practically no information is lost.

A telemedicine workstation has four basic classes of data it must be able to transmit and receive. These can be thought of as static, one, two and three dimensional data files.

Static data: Files quite simply do not change. These would include information such as the patient's records, treatment notes, and other essentially textual data. These are unlikely to pose a significant burden on even the most primitive communication systems.

One dimensional data: Files are data sequences, such as audio, or instrument telemetry. It may be desirable to store these files, and forward them when they are complete, but it is more likely they will be transmitted in real time. Uncompressed CD quality audio consists of 16 bit measurements taken approximately 44,100 times a second. This requires a bandwidth of approximately 88.2 Kbytes/sec. Instrument telemetry should fall in a similar range, or may be distributed as a graphic file.

Two dimensional data: Data Consists of still images. Computer image are stored as grids, in which the squares (called pixels) each can be a different color. The size of the image can be determined by multiplying its width in pixels times its height in pixels, times the number of different colors a pixel can possess. 24 bit color is generally assumed to be photorealistic, or sufficiently subtle to display the smallest differences the human eye can perceive. 16 and 8 bit color are also common (displaying 32,768 and 256 different colors, respectively.) Some common image sizes are listed below:

Magnetic Resonance Study: 256x256 pixel images, 8 bit color, 20-40 frames per study ~ 2MB per study.
CT study: 512x512 pixel images, 8 bit color, 20-40 frames per study ~8 MB per study.
Radiogram: 2048x2048, 12 grayscale ~ 6 MB

Three dimensional data: Three dimensional data is an image which changes in time, a movie. Again, these must be sampled at a certain time rate. 60 fps, or frames per second is generally considered to provide for full-motion video (indistinguishable from life). High quality video, depending on the exact parameters involved, requires anything from tens to over a hundred Mbytes of storage per second of video. If this video is to be transmitted in real time, then the communications link must be capable of transmitting many megabytes of data per second.

5.1 *Hardware Links*

Cables: Cables are the connection along which the signal actual travels. There are many different sorts of cable currently in use.

Coaxial Cable: This is a cable with a solid central conductor, surrounded by an insulating shell, which is in turn shielded by a wire mesh. It is often used to carry high frequency signals such as video or radio.

Twisted Pair: This is cable where two conductors are twisted together. This cable may be either shielded or unshielded.

Optical Fiber: Optical Fiber is made from plastic or glass, and information is transmitted along it in the form of pules of a laser. It is cheaper, and less susceptible to external noise that copper wire, but more difficult to connect.

Routers: A router is a device which forwards information within or between networks.

Concentrator: A concentrator is a device which takes many inputs, and combines them into a single, higher bandwidth output.

Modems: A modem is a device which can transmit and receive digital information over a conventional analog phone line. It accomplishes this by encoding the data as a series of tones, which are decoded at the other end. The speed of modems is generally expressed in baud, or bits per second. Early modems were able to transmit data at between 300 and 1200 baud. Newer modems transmit at 2400, 9600, 14400 and 28800 baud. A connection between two modems is limited to the speed of the slower modem, and may be further limited by the amount of noise on the telephone line.

5.2 *Network Types*

Local Area Network (LAN): The Local in LAN generally refers to an area ranging from the interior of an office, to one encompassing a city block. LANs are generally built using connections explicitly laid for that purpose, and provide a high speed, low cost way to share information.

Common commercial LANs are Ethernet and Token Ring, which provide transfer rates of 10Mbps and 16 Mbps respectively. Higher throughput can be achieved by using multiple wires, or fiber optic cable. There are currently several competing standards for 100 Mbps capable LANs.

Wide Area Network (WAN): Wide Area Networks are of special interest in telemedicine, since they generally cover whole continents, or even span the globe. These networks form the basis of the global telephone system. Unlike Local Area Networks, modifying or adding to the hardware of a WAN is a monumental task, simply because of the scale of the changes which must be made.

WANs can easily be broken down in terms of the bandwidth they deliver. X.25 is an existing low bandwidth network, as is N-ISDN. B-ISDN and systems implemented on top of SONET would represent broad band services.

The Internet: The Internet is a logical network which spans many physical networks. It grew up around the backbone networks of ARPAnet and NSFnet, networks laid to facilitate scientific collaboration, and MILNET, a military network. Tens of thousands of local networks have attached to the Internet, providing a network which spans the globe, as well as many different communication and networking protocols.

5.3 *Network Protocols*

TCP/IP: TCP/IP is in fact not one communications protocol, but two. Together, the protocols form the network and transport layer for Ethernet. The TCP/IP protocol is used by many UNIX systems. Because TCP/IP is includes the IP protocol, which is best effort, it is poorly suited to transmitting real-time sound and video data, since it cannot guarantee a constant bandwidth between any two points in the network.

TCP: Transmission Control Protocol forms the transport layer of the TCP/IP standard. It ensures a full duplex, multi-plexing, connection oriented link. In addition, the link is reliable. In this context that means that the packets sent along the link are complete, uncorrupted, and in the same order as they were sent. TCP is found in most UNIX systems. Those systems that do not use TCP generally use UDP.

IP: Internet Protocol is the network level standard on which both TCP and UDP are built. IP is connectionless, and based on a best effort method of routing. This means that a fixed bandwidth between any two points on the network cannot be guaranteed, but will instead be dependent on the load on the network at any given time.

UDP: User Datagram Protocol is a common alternative to TCP. It is a connectionless protocol which includes a check-sum, but does not correct for packets which arrive out of order, nor does it guarantee delivery. As a result it is a very simple protocol, since all error-processing and retransmission must be taken care of by the application program. UDP provides network layer, transport layer and session layer protocols. A datagram is simply a packet in a connectionless protocol.

Thick Coaxial Ethernet, 10Base5
Thin Coaxial Ethernet, 10Base2
Twisted-Pair Ethernet, 10Base-T
Fiber Optic Ethernet, 10Base-F

In addition, a standard for 100 Mbps exists. This bandwidth may be achieved using fiber optic cable, and two or four pair twisted pair cable.

ATM: ATM or Asynchronous Transfer Mode is a standard designed to provide multi-media service over a network ranging in size from a LAN to a WAN. ATM does not fit well into the OSI network hierarchy, as it provides services from both the Data Link and the Network Level.

ATM is an attempt to improve on STM, or synchronous transfer mode. In a connection based synchronous link, a certain bandwidth is guarantees between two points. Under normal conditions, much of this bandwidth may go unused, although the full bandwidth is necessary to ensure real-time delivery of a "bursty" signal. ATM attempts to utilize these gaps to provide more efficient user of the total resources of the network, while still ensuring that users will have access to the desired bandwidth on demand.

ATM does this by splitting all communications into 53 byte long packets. By ensuring small, fixed size packets, and rapid error recovery and routing, ATM attempts to gain the efficiency of packet switching wile still providing the "bandwidth on demand" of a connection based link.

CHAPTER 5

SUMMARY

In Chapter 4, the study of the human-machine interface (HMI) was defined as "the discipline concerned with the design, evaluation and implementation of interactive computing systems for human use and with the study of major phenomena surrounding them" (McClelland, Adamson et al. 1995). An integral research component to HMI design is the understanding of sensorimotor transformations and human sensorimotor abilities at a basic level. Throughout normal physiological development, we are constantly adapting to new sensorimotor inputs as we learn to, for example, ride a bike, fly a kite, or use a computer. Trying to understand some of the limits of our adaptive capabilities was one of the goals of this thesis. What sort of transformations are we capable of adapting to and what does the learning curve look like?

Chapters 2 and 3 bridged the gap between basic sensorimotor research and advanced HMI design by asking the questions; how do rate-control and position control modes compare when a sensorimotor adaptation is induced? What are the learning curves and what is the steady-state performance? This may be one of the first attempts to look at position versus rate-control devices in the context of altered sensory motor loops to evaluate human operator performance. One interesting results was that for both Chapter 2 and Chapter 3 experiments, although rate-control resulted in slower TTTs and longer DTTs than for position control, learning rates for both control modes were not significantly different. The main ramification of this is that training times may be similar for both control modes. This is important when considering telerobotic control for building the space station, for example.

The mechanisms for sensorimotor adaptation seem to be primarily visuomotor in nature. One hypothesis is that we implement a linear approximation to the transformation from visual to kinesthetic coordinates in the process of deriving muscle patterns (Soechting and Flanders 1989b). Kinesthetic information alone seems to be unable to give an accurate spatial location of an object in space (Tillery et al. 1991).

Models of human performance are also important for HMI design. Another significant finding reported in this thesis was that executing multi-tasks changes performance and therefore the models predicting performance on one task alone. In Chapter 2, the model of movement time changed due to the requirements of a dual-task. However, it was the secondary performance measure, reaction time, which correlated with the measure of overall workload. The implications are that objectively modeling performance on a particular HMI and subjective evaluations will only correlate if the proper testing parameters are found.

References

McClelland, I., K. Adamson, et al. (1995). "Information issues in telemedicine systems." J. of Telemedicine and Telecare **1**(1): 7-12.

Soechting, J. F. and M. Flanders (1989b). "Errors in pointing are due to approximations in sensorimotor transformations." J. Neurophys. **62**(2): 595-608.

Tillery, S. I. H., M. Flanders, et al. (1991). "A coordinate system for the synthesis of visual and kinesthetic information." J. Neuroscience **11**(3): 770-778.