THE HUMAN FACTORS OF COMPUTER INPUT USING A MOUSE DEVICE

DAVID BARKER

Doctor of Philosophy

The University of Aston in Birmingham

September 1993

The Human Factors of Computer

Input Using a Mouse Device

David Barker

Doctor of Philosophy 1993

This thesis investigates how people select items from a computer display using the mouse input device. The term computer mouse refers to a class of input devices which share certain features, but these may have different characteristics which influence the ways in which people use the device. Although task completion time is one of the most commonly used performance measures for input device evaluation, there is no consensus as to its definition. Furthermore most mouse studies fail to provide adequate assurances regarding its correct measurement. Therefore precise and accurate timing software were developed which permitted the recording of movement data which by means of automated analysis yielded the device movements made. Input system gain, an important task parameter, has been poorly defined and misconceptualized in most previous studies. The issue of gain has been clarified and investigated within this thesis. Movement characteristics varied between users and within users, even for the same task conditions. The variables of target size, movement amplitude, and experience exerted significant effects on performance. Subjects consistently undershot the target area. This may be a consequence of the particular task demands. Although task completion times indicated that mouse performance had stabilized after 132 trials the movement traces, even of very experienced users, indicated that there was still considerable room for improvement in performance, as indicated by the proportion of poorly made movements. The mouse input device was suitable for older novice device users, but they took longer to complete the experimental trials. Given the diversity and inconsistency of device movements, even for the same task conditions, caution is urged when interpreting averaged grouped data. Performance was found to be sensitive to; task conditions, device implementations, and experience in ways which are problematic for the theoretical descriptions of device movement, and limit the generalizability of such findings within this thesis.

KEYWORDS
Human factors; Input Devices; Age Differences; Human Computer Interaction;

For Goggy,

who taught me the value of education
and fine whisky

Acknowledgements

I should like to thank Dr. A. Cochrane (formally the Acting Director of I.H.D., Aston University), Mr. J. Kidd (Aston Business School), and Mr J Merton (British Petroleum) for making this Ph.D. possible under the Interdisciplinary Higher Degree Scheme.

I am indebted to my supervisor Dr. R.B. Stammers (Aston University) who stepped into the breech when all seemed lost. Despite very heavy demands on his time he operated an 'open door' policy which facilitated the completion of this work.

I should like to acknowledge the tremendous efforts made by my friend and colleague Dr. Brown (Aston University) who read, and reread the many drafts of this thesis. I Should also like to thank him for providing office space and computer facilities for the production of this thesis.

The data collected for experiments two, three, and four was collected in collaboration with final year students at Aston University, and I should like to express my gratitude to; Peter Wale and Nick Edwards. I should also like to thank Mr. C. Mason (Aston Business School) for his assistance with the measurement of the timing signals described in Chapter Five, and Mr. D. Bates (Aston Business School) for resurrecting my Opus II from the dead.

A long term study made arduous demands on the time of three people in particular, and I should like to thank Diana Johnson, Michael Chung, and Marion Plant, for their goodwill and time.

The study of age differences in relation to device usage would not have been possible without the funding arranged by Dr D.R. Davies (Aston University).

I should like to thank Neil Barker, my brother, for his financial support when the research grant ran out.

This research programme was made possible by funding from the S.E.R.C. and British Petroleum.

TABLE OF CONTENTS

TABLE OF CONTENTS

TABLE OF CONTENTS

Page

TABLE OF FIGURES

TABLE OF FIGURES

TABLE OF FIGURES

LIST OF TABLES

LIST OF TABLES

CHAPTER ONE

INTRODUCTION

This Chapter describes the origins and impetus for this research programme and provides the reader with an overview of the thesis and the associated research activity. The main subject of this thesis is people's use of computer mice. Despite the increasing use of mouse type devices for computer input, relatively little is known about how it is achieved, and in which ways the device movement characteristics differ between users. The structure of this thesis, and its partition into ordered sections and chapters, does not correspond to the iterative nature of the actual research activities.

1.0 Origins of The Research Programme

Introductions are such awkward things. In particular it is often difficult to know where to begin.

> *' "Begin at the beginning," the king said, gravely, "and go till you come to the end; then stop".'*
>
> Lewis Carroll (1865, p12)

This research programme began in response to the question "What are people doing when they use a mouse ?". In particular, what sort of movements are made with the mouse input device by its users? Furthermore, could this knowledge be useful in improving the input process?

The importance of these questions, and hence an additional justification for this research programme, becomes evident when one considers the pervasiveness of interactive information technology within everyday life, and its increasing reliance upon pointing devices (Czaja, 1988; Straayer, 1991), and in particular mouse type devices (Baecker & Buxton, 1987; Price & Cordova, 1983).

The increasing use of computer mice has occurred for a number of reasons. The nature of work itself is in a rapid and ongoing period of change (Argyle, 1989; Bracker & Pearson, 1986), and more generally there is argued to have been a shift from an industrial to an information society (Naisbitt, 1982; Toffler, 1980).

Smith (1980) has identified the growing number of people who have to use a computer as part of their working day, but who have little, or no, typing skill, and are unlikely to acquire it. Such 'direct end users' have generally benefited from a supplementary input device to the keyboard, which has usually been a mouse. Additionally, for many applications, such as those utilizing Windows[1], which place increasing emphasis on the 'direct manipulation' of visual objects, the provision of a mouse has become a prerequisite (Helander, 1988; Computer Shopper, 1991).

---

1 Trade mark of the Microsoft Corporation

Despite the spread of computer mice there may exist a small, but significant, subpopulation of computer users who experience difficulty when using one (Taylor & Hinson, 1988), and others who prefer not to use them at all (Straayer, 1991). With which aspect of the input process do users experience difficulty? More generally, what distinguishes good from poor mouse users?

A review of previous input device studies and the relevant areas of the motor movement literature did not provide useful solutions to the above questions. In an attempt to answer these questions, the research programme investigated the microstructure of device movements made by users.

In order to determine the microstructure of device movements, it was necessary to develop technically sophisticated movement recording apparatus, and to pioneer novel techniques for the description, and subsequent analysis of movement. Consequently, this thesis makes a contribution to the area of computer based measurement, and to the issue of how we conceptualize motor movement.

The microstructure of mouse movement was then investigated, using these techniques through a number of experiments. Having established the methodology and its usefulness, it was generalized to other contemporary devices permitting a comparative study to be made between them.

1.1 Overview of The Thesis

This thesis has been partitioned into five sections comprising eleven chapters. Figure 1.0 shows a pictorial representation of the structure of the thesis. At the risk of being 'recursive', the introduction is taken to be self explanatory.

Diagram of Thesis

A. Introduction

1. Introduction

B. Historic Context

2. HCI (Human Computer Interaction) & Computer Input Devices

3. Motor Theory

C. Method

4. Describing Movement

5. Recording Movement

6. Experimental Task

D. Results

7. Device Movement

8. Age Differences in Movement

9. Skill Acquisition

10. Comparative Device Study

E. Conclusions

11. Discussion & Conclusions

Figure 1.0

The historical review (B) contains two chapters; Chapter Two deals with HCI (Human Computer Interaction), with an emphasis on previous input device studies, and Chapter Three provides a review of the theories of motor movement relevant to input device movement. Chapter Three also contains a review of a selection of investigations which have been concerned with recording and analysing human movement.

The Methodology Section (C) is concerned with the development of the recording and subsequent novel analytical techniques used within this research programme. It contains Chapters Four and Five which are concerned with the innovative descriptions of movement patterns, and the recording of the movements upon which the analysis, pioneered here, was made possible. Chapter Six focuses on the design of the experimental tasks.

This section also describes several experiments which were concerned with determining the precision and accuracy of the experimental apparatus, and the measurement techniques employed. Details of the construction of the analytical techniques, employed to capture the richness of the observed movement behaviour, are also given within this section.

The Results Section (D) contains Chapters Seven through to Ten, which provide details of the four main areas of empirical investigation. In total five experiments were undertaken. Chapter Seven describes the ways in which people moved the mouse, and how this was associated with different task factors and individual differences in previous device experience (experiment 1). Chapter Eight investigates age related differences in device usage (experiment 2). Chapter nine is concerned with the changes in movement patterns associated with increasing task and device experience, as observed during a longitudinal study (experiment 3). Chapter Ten describes a comparative input device study and comprises two experiments (experiments 4 & 5). The experiment 4 investigates three different physical implementations of a conventional mouse device; and the fith experiment compares three contemporary devices (conventional mouse, a hand held thumb operated mouse, and a pen shaped mouse)

The final section (E), Chapter Eleven, integrates the separate experimental findings - across the different studies - within the historical context, and it also addresses the questions posed at the onset of the research programme. Additionally, it discusses the relative advantages and disadvantages of the methodology employed; and the questions raised by the empirical studies.

The structure of the thesis might suggest that the research followed a 'linear' sequence from conception through to conclusion - this was not the case. The recursive flow of research activities, and the entwined sequence of writing, resembled Hofstadter's 'Eternal Golden Braid' (Hofstadter, 1979). Given the iterative nature of the research activities the cause of clarity is best served by a system of forward, and backward, cross referencing throughout the thesis.

CHAPTER TWO

HCI & COMPUTER

INPUT DEVICES

This chapter provides a brief description of the evolution of computer technology and the related emergence of HCI as a distinct area within the Human Factors community. Research into input devices is then identified as an important endeavour within the field of HCI. Despite the large number of input device studies, and the application of cognitive psychology to HCI, there is still the need for experimentation to determine the suitability of a particular device for a particular input task. A recent approach to HCI, the 'task artifact cycle' (Carroll & Campbell, 1989), seeks to develop artifacts by observation of the interaction activities. This is seen as providing a context for the current study which seeks to develop input artifacts (tools) based on the 'natural gestures' made with devices. Finally, a review of previous input device studies is presented.

2.0 Introduction

The proliferation of computer systems and the increase in the number of computer users has been remarkable. In 1947 it was predicted that only six electronic computers would be required to satisfy all of the United States computing needs (Palfreyman & Swade, 1991). However, since the early 1960's, commentators have noted the growth in the number of such systems and their users (Glauthier, 1967; Nickerson, 1962; Bennet, 1972; Smith & Green, 1980; Shackel, 1984, 1991). It has been estimated that there are now over one hundred million personal computers in the world (Palfreyman & Swade, 1991). It is hard to find instances where their use does not impinge to some extent, or other, into the lives of most people in the industrialized world (Smith & Green, 1980).

This increasing accessibility of computer systems to a wider, but not necessarily computer sophisticated public, has brought with it increasing demands upon the user (Whitefield, 1986). The computer can be rigid and very demanding of its users (Smith & Green, 1980). People often experience frustrating and serious difficulties when using such systems (Carroll & Campbell, 1989). In order to use computers effectively people often have to make considerable changes to existing working habits, requiring them to retrain and adapt (Long & Whitefield, 1989; Smith & Green, 1980). These difficulties continue to affect users at all levels of skill and expertise (Shniederman, 1982)

There has been an equally phenomenal growth in the number of people concerned with issues pertaining to human computer interaction (Shackel, 1991; Shniederman, 1987; Carroll & Campbell, 1989; Gaines, 1984; Haller, Mutschler, and Voss, 1985). HCI (Human Computer Interaction) is an interdisciplinary conjunction of several sciences and technologies (Shackel, 1991; Long & Whitefield, 1989), whose practitioners are concerned with extending and improving the quality of HCI (Diaper, 1987; Baecker &

Buxton, 1987; Long & Whitefield, 1989). The importance of HCI issues is well expressed by Monk (1984):

> *'There is an increasing awareness of the case made by HCI specialists that the design of user machine interfaces in any interactive system is crucial for its efficiency and acceptability.'*
> *Monk (1984, p 6)*

Manufacturers have utilized developments in the user interface to gain a competitive advantage (Gaines, 1984; Preece & Keller, 1990; Shneiderman, 1987; Lewis, 1990), with it determining the success or failure of such systems (Baecker & Buxton, 1987). The role of human factors considerations within computer system design has been further acknowledged by the prominence given to HCI issues in the three (Japan, U.K., and Europe) Fifth Generation projects launched in the early 1980's (Moto-Oka, 1982; Alvey, 1982; and ESPRIT, 1983).

Input devices have been identified as one of a number of the key areas of interest within the HCI field (Diaper, 1987), and increasingly efforts are being directed to the problems of input (Foley, Wallace & Chan, 1984; Buxton, 1986). Similarly, prior to the emergence of HCI as a distinct discipline within the human factors area (recently reviewed by Shackel, 1991 and Baecker & Buxton, 1987), input devices have been considered as important contributors towards better human machine interaction (Alvey, 1982; Taylor, 1967; Licklider, 1960; Licklider & Clark, 1962; Johnson, 1967; Nickerson, 1969; Shackel, 1969; Murata, 1991), whilst receiving less attention than output devices (Buxton, 1986; Jacob et al, 1993; Whitfield, Ball, & Bird, 1983).

2.1 Approaches taken to the study of input devices

Researchers have largely been concerned with determining the most appropriate device, or devices, for particular input tasks. Four approaches have been identified: the development of heuristics through experimentation, developing taxonomies of input devices that provide ways of determining a good device task fit, utilizing psychological models of computer input, and the task-artifact approach. These are consistent with the positions outlined by Carroll, Young, and Long (1991).

2.1.1 Experimentally derived heuristics

Our current knowledge of input, like that of HCI in general (Monk & Wright, 1991), has largely been derived from empirical laboratory based studies. Typically, the suitability of particular devices for particular 'types' of task is evaluated with respect to task performance. The findings from these studies have been used as the basis for formulating heuristics, an approach advocated by Shniederman (1982), which can be used to inform input device choice for future systems, without the need for direct experimentation.

Carey (1985) and Haller et al (1985) have attempted to construct such heuristics by considering the findings from numerous device studies. However, both reviews found comparisons between device studies difficult due to the different combinations of devices employed, and the different experimental tasks used. Moreover, the generalizability of any device study findings, to other input situations, has been questioned on the grounds of the user and task specificity of any such study (Monk & Wright, 1991; Greenstein & Arnaut, 1988), and the effects of different physical implementations of the same type of device (Carey, 1985, and Buxton, 1986). Additionally, a system's overall evaluation will often depend upon several performance measures, and other non performance related factors (Whitefield, 1986). Given the multidimensionality of system performance, it is unlikely that a single device would be optimal for all the performance dimensions. Therefore, it is unlikely that one device alone will be the most appropriate for all the tasks to be faced by a system user and the designer's goal will involve selecting the best 'all-round' device or providing a number of appropriate devices (Buxton, 1986).

> *'With respect to tasks, no single input device is well suited to all of the input functions'.*
>
> Whitefield (1986, p103)

2.1.2 Device Taxonomies

Attempts have been made to usefully classify input devices and input tasks so as to match the device to the task. The criteria used to categorise devices should correspond to the task dimensions which influence the suitability of devices for particular task types.

Carey (1985) identified three classes of input device: pointing devices, selection devices, and symbolic devices. Pointing devices operate by identifying a position, or object on a display, and may also communicate positional information (unlike selection devices). Typical devices include: lightpen, mouse, trackerball, joystick and touchscreen. Selection devices provide a fixed set of labels, or categories, from which functions, or objects, can be selected. Typical devices include the data tablet and the fixed labelled keyboard. Symbolic devices operate, via the construction of input sequences, from a fixed set of basic symbols. Typical devices include numeric keypads and alphanumeric keyboards. These categories overlap, in that one input device, from a particular category, can be implemented so as to fall within another category. This reflects the versatile nature of most input systems.

An important distinction has been drawn between direct and indirect pointing devices (Carey, 1985; Whitfield et al, 1983, Whitefield, 1986). An indirect pointing device is one where the movement takes place towards an area which occupies a different location in space, but which is mapped onto the target area, from which the operator receives feedback (Whitefield, 1986). Such devices include: the mouse, trackerball, and joystick. Conversely, for a direct pointing device the movement takes place towards the actual location of the target in space. Such devices include the touch screen and lightpen. The choice of either direct, or indirect, devices should be made having regard to the physical characteristics of the input environment and the input task (Whitfield, et al, 1983; Carey, 1985; Whitefield, 1986; Murata, 1991).

Carey (1985) also identified five generic input operations based upon those of Foley (1980). These were: specify object, specify location, enter numeric value, specify required action, and text entry. These may be mapped onto the earlier categories of input device. Most input devices can be implemented so as to achieve all of the possible input tasks, although some devices, due to their inherent performance advantages, will be better suited to some input tasks than others (Carey, 1985; Greenstein & Arnaut, 1988; Buxton, 1986). However, given a suitable implementation of the input task, and suitably experienced users, it is possible to contrive a previously inappropriate device to an appropriate one (Ewing, Mehrabanzad, Sheck, Ostroff, and Shneiderman, 1987).

Haller et al (1985) and Buxton (1986) have identified the compatibility between the device's operation and that of the system's response as being a key component of the user's resultant performance. Similarly the resolution of a device can affect its suitability for certain tasks (Buxton, 1986; Whitefield, 1986; and Carey, 1985). Thus, device functionality and the type of input task are not sufficient to uniquely identify the most appropriate device. Further analysis, based on the idiosyncrasies of the device, and more specific task details are required (Buxton, 1986; Carey, 1985).

Buxton (1986) proposes a taxonomy of input devices (analogous to a periodic table) based on device properties which might lend themselves to distinguishing the suitability of such devices for different tasks. The categorizing dimensions are; What is sensed (motion, velocity etc), the number of dimensions (1D, 2D, 3D), the motor skills utilized, those that are activated directly by touch or those that require an intermediary. Buxton believes that this taxonomy provides a method to assess device equivalence, and may provide a means of identifying new input devices. However, Buxton then proceeds to introduce other factors which must be considered when making the input device choice; these reduce the usefulness of the device taxonomy in making the device selection.

2.1.3 Cognitive Psychology

One of the most popular approaches to HCI amongst academic researchers has been the application of cognitive psychology to HCI (Lewis, 1990). Of particular relevance to computer input has been the work associated with Card, Moran, and Newell (1980). Their key stroke model for the mouse device, using Fitts' Law, will be discussed later in this Chapter. However, their work has been criticised for its minimal content of psychology and its limited domain of application (Lewis, 1990; Monk & Wright, 1991; Karat, McDonald and Anderson, 1986). More generally the application of cognitive psychology, whilst exciting and promising, has yet to successfully be applied to realistic problems (Monk & Wright, 1991). There appears to be a general consensus that there is a lack of suitable theories, and those that are available are too domain specific and limited in scope.

> *'However, the search for good theories in the field of HCI has so far met with little success. Indeed, there is little agreement about what kind of theories might be appropriate, or what the discipline of HCI might be'*
>
> T. Stewart (1989, p 321)

Similarly,

> *'Formal models have thus far contributed little to our understanding of deeper cognitive issues that manifest themselves in HCI.'*
>
> Baecker & Buxton (1989, p176).

Specifically, when considering input,

> *'Cognitive science has not developed sufficiently to provide detailed models for much of normal cognitive behaviour.'*
>
> Karat et al (1986, p 75)

It is interesting to note that this view was voiced by Bennett over 20 years ago.

> *'There are as yet no valid theoretical models for predicting the effect of interactive facility alternatives on user performance, so there are few, if any, theoretical models for guiding the isolation of experimental variables'*
>
> Bennett (1972, p 176)

When will such theories be available? Buxton believes that our theories will never be sufficiently complete to deal with the complexity of input.

> *'Managing input is so complex that it is unlikely that we will ever totally understand it. No matter how good our theories are, we will probably always have to test design through actual implementations and prototyping.'*
>
> Buxton (1986, p 376).

2.1.4 Task Artifact Cycle

Recently, an alternative view of HCI, associated with the 'task artifact cycle' (Carroll & Campbell, 1989), has been identified by some researchers (Lewis, 1990; Monk & Wright, 1991; Jacob et al, 1993). This approach assumes that the field of HCI exists to provide an understanding of usability, and how to design usable computer artifacts (tools). HCI is seen as the study of an 'ecology of tasks and artifacts'. Artifacts are tools designed to aid people in performing cognitive tasks. Carroll & Campbell advocate a task-artifact design cycle based on observation of the user, when engaged in interaction with the system.

This type of approach is not new to HCI. Katler (1969) advocated a system redevelopment cycle which included analysis, synthesis, and evaluation. The need for empirical validation of systems, in 'practice', was advocated by Whitfield (1964), and the role of user observation as essential, within the area of the ergonomics of the user interface (Edmonds & Green, 1984). However, Carroll & Campbell ascribe a scientific status to the artifacts created by such a cycle. The design of artifacts, as well as their evolution, inherently and inextricably involve psychological issues. Artifacts incorporate psychological assumptions about their usability and suitability for the tasks that users do.

Artifacts are considered, by the authors, as an appropriate medium for expressing and developing theoretical claims regarding usability. A similar claim is made by some 'cognitive scientists' for the status of their computer programs which model cognitive processes (Winston, 1992; Schank & Colby, 1976); Newell & Simon, 1961; Kosslyn, 1980).

Artifacts are not only the system hardware, but also comprise the system's software. In considering the future directions of computer input research, Jacob et al (1993) have adopted the task-user development cycle, and have advocated the search for new input devices and dialogues,

> *'The challenge before us is to design new devices and types of dialogues that better fit and exploit the communication relevant characteristics of humans'.*
>
> Jacob et al (1993, p 1)

They advocate an artifact development which is user 'pushed' rather than technologically 'pulled'. The search for new devices would be assisted, in part, by reference to device taxonomies, such as that proposed by Buxton (1986). However, as we are far from exploiting the full potential of the devices that we already have (Buxton, 1986), a useful approach could seek to implement the input tasks so as to utilize the input gestures made by users with a particular device.

2.2 Early Input Device Studies

It is interesting to note that the years prior to 1969 have been considered as the 'wilderness years' for research into 'man-computer' interaction (Gaines, 1984; Nickerson, 1969), with Shackel's paper (Shackel, 1959) standing in isolation (Gaines, 1984). This view has been challenged by Shackel himself (Shackel, 1991) where he describes the work of the period as sporadic, rather than isolated, with some important foundations having been laid.

The research into what we now term HCI of this decade (1959 - 1969) was considered as a part of, rather than distinct from, the human factors tradition of the day. Many of the studies were published within special interest sections of the electronic engineering, or computer journals, of the period with an emphasis on the technological aspects of the system . Given the technological limits of the time, it is not surprising to find most of the research concerned with the 'machinery' rather than the rather limited interaction processes. However, within this period there were some interesting computer input studies.

At the time of non interactive computer input, Braunstein & Anderson (1961), in anticipating speech input device technology, compared data entry using 'conventional keyboard input' (sic) to that of reading the digits aloud. They used five keyboard naive subjects and found that keyboard entry was slower. However, the subjects reported that keyboard mediated input was easier.

Gurley & Woodward (1959) described a lightpen input device. Interactive computer systems providing line drawing packages were described by Stotz (1963), and Sutherland (1963). They emphasised the computer technology, and in particular the light pen devices used, and how they facilitated input for this type of task. Similarly, we find details of the emergence of touch displays (Johnson, 1967).

We find a description of a computer console specifically developed to enable the assessment of different techniques of 'Man-Machine Interaction' (Lazovick, Trost, Reickord, & Green, 1961; Lichtenberger & Pirtle, 1967). Similarly, we find the case being made for the empirical validation of computer consoles (Whitfield, 1964).

Input string length and key size were shown to affect the speed and accuracy of keyboard input (Deininger, Billington, & Riesz, 1966). The speed and accuracy of textual retrieval using abbreviated and literal input strings was investigated (Dieninger, Billington, & Michaels, 1968).

The first example of a mouse device can be found in an experimental comparison between it, lightpen, joystick, and a conventional keyboard for text manipulation (English, Engelbart, & Berman, 1967). The mouse device had been constructed by the experimenters.

2.3 The Computer Mouse

The computer mouse is a small hand held device that when moved, over a flat surface, causes movement of a screen cursor (Greenstein & Arnaut, 1988). It provides (x,y) coordinate positional data when it is moved over a plane surface, and provides data entry from a finger operated switch (Ritchie & Turner, 1975; English et al, 1967). The mouse is an example of an indirect pointing device (see section 2.1.2).

Mice have an assortment of physical characteristics; there are mechanical, optical, and 'tailless' mice, in various shapes and sizes, with varying numbers of buttons (Price & Cordova, 1983). The physical implementation of a mouse has been shown to strongly influence its ease of use, and task performance for both short and longterm usage (Abernethy & Hodes, 1987; Hodes & Akaki, 1986; Price & Cardova, 1983; Hill, Gunn, Martin, & Schwartz, 1991). Similarly, the 'feel' of a device is important in determining its effectiveness and acceptance (Baecker & Buxton, 1987).

Mice may be connected to the computer via a specific 'mouse port' or by connection to the existing serial RS232 port. The former being known as a 'bus mouse' and the latter as a 'serial mouse'.

Mouse movement with the serial mouse is usually recorded on two counters within the device. These counters represent the displacement of the mouse in the X and Y planes of movement. An additional counter (register) stores the status of the device buttons. Information taken from these counters is converted into a suitable format for the serial transmission to the computer's serial port. Different Manufacturers provide different data formats for their device, but typically use a transmission rate of 1200 baud.

The bus mouse produces electrical signals (quadrature output) which directly reflect the mouse's movement. This information is 'fed' into the computer where it is processed by the associated 'bus card' which will again cause positional and button status counters to be incremented.

Serial devices have the advantage of requiring no additional hardware other than the provision of a serial port (a standard provision on most computer systems), but their inherent additional signal processing and the relatively slow data transmission rate via the RS232 link make them slower than bus mice with respect to updating the device's spatial and button status. However, this difference, between the two types of device, is rarely apparent to the user due to the relatively slow refresh rate of the display pointer's position.

Contemporary mice typically detect movement in one of three ways; mechanical, opto-mechanical, or optical. In the case of a mechanical mouse, its movement is detected by the corresponding movement of potentiometers, whose change in electrical resistance corresponds to changes in the position of the device. In opto-mechanical mice the device movement is translated into the rotational movement of a disc, whose surface is marked in such a way, that its movement (and hence the movement of the device) can be detected by opto-electronic sensors. An optical mouse detects movement without the need for a mechanical transducer. Movement is detected by the reflections of light from a reflective mouse mat, upon which is superimposed a fine grid. Mechanical, and opto-mechanical mice are mainly of the 'rollerball' type, and they are generally cheaper than optical mice.

The computer program which processes the information from the mouse device is known as a 'device driver'. The device driver not only processes the information sent from the device to the computer port, but also determines and maintains the cursor position on the display. Different device manufacturers provide their own device driver software. However, increasingly devices are being manufactured towards compatibility with either the 'Microsoft' or 'PC Mouse' device driver packages (Computer Shopper, 1991). Some device manufacturers provide the user, by means of a switch on the device, with the option of which of the two driver packages they wish their device to be compatible with.

2.4 Review of recent input device studies

This section is primarily concerned with input studies that have used the mouse device. Rather than consider each study separately this thesis will attempt to discuss the studies with reference to certain issues, which are relevant to this research. However, to begin here is a list of the studies of interest.

Many comparative studies between input devices have been carried out. Devices have typically been compared along the performance variables of positioning time, accuracy, and occasionally user preference. The devices most commonly used have been; mouse, joystick, keyboard, touchscreen, and lightpen. One of the earliest studies to use a mouse was English et al (1967). Some of the later studies are: Card, English, & Burr (1978); Price & Cordova (1983); Haller, Mutschler, and Voss (1985). Karat, McDonald, & Anderson (1986); Ewing et al (1986); Epps (1986); Radwin, Vanderheiden, and Lin (1990); Barker, Carey, and Taylor (1990); Trankle & Deutschmann (1991); Mack & Montaniz (1991); Hill et al (1991); MacKenzie, Sellen, and Buxton (1991); Sears & Schneiderman (1991); Murata (1991); Lin, Radwin & Vanderheiden (1992); Wolf (1992). Studies have mostly employed the mouse device to locate and select areas of the display, rather than focus on other aspects of its use, such as 'dragging' or 'drawing'(Mackenzie et al, 1991).

2.4.1 Variables Studied

The variables investigated in the above studies, apart from the device, include; target size, position of the target, amplitude of movement, direction of movement, gain, user's personality, and subjective usability. No one study has varied all of the above, or otherwise controlled for them all. Many of these variables are common to several studies.

2.4.1.1 Performance criteria

Most studies have judged performance with respect to task completion times and the frequency of errors (Radwin et al, 1990; Trankle & Deutschman, 1991; Haller et al, 1985).

2.4.1.1.1 Task completion time

Given a discrete primary task design, task completion time may be regarded as consisting of the following sequential stages following the presentation of the target:

1) Reaction time
2) Movement time
3) Selection time

ending with the 'acquisition' of target (or error).

This proposed temporal model assumes that after the presentation of the target there is a delay (reaction time), followed by a period in which the cursor is located onto the target (movement time). Having located the cursor on the target (for an error free trial) there is a delay until the selection is indicated (selection time). Although this is not a comprehensive model, applicable to all real world tasks, it applies to most of the experimental studies.

Sears and Shneiderman (1991) and Epps (1986), took task completion time as the period from the presentation of the target until a selection had been indicated (reaction time

+ movement time + selection time). Radwin et al (1990), Lin et al (1992), Trankle and Deutschmann (1991), and Karat et al (1986) (for their practice trials), recorded the time from the first movement of the cursor until a selection had been made (movement time + selection time). Arnaught and Greenstein (1990), using a trackerball and touchscreen distinguished between 'gross' and 'total' movement time (both excluding reaction time), and 'fine adjustment' time. Gross movement time was that time from the first cursor movement until the cursor entered the target. Fine adjustment time was the time from entering the target until the target was selected. Haller et al (1985) distinguished between 'positioning time' (reaction time + movement time) and 'replacing time' (selection time). Wolf (1991) distinguished between 'task time' and 'command time'. Task time was that time from the presentation of the task until task completion(reaction time + movement time + selection time), and command time which removed the initial reaction time and positioning time component from task time (selection time). Card et al (1978) distinguished between 'homing' time and 'positioning' time. Homing time was the delay between the target presentation and the first movement of the device (reaction time). Positioning time was that time from the first cursor movement until target selection (movement time + selection time). Similarly, English et al (1967) distinguished between device access time (time to grasp the device) and motion time (movement time + selection time). Ewing et al (1986), Abernethy & Hodes (1987) and Karat et al (1986) for their main tasks required the users to carry out an array of subtasks resulting in a single completion time measure.

2.4.1.1.2 Accuracy

Most studies of mouse movement include some measure of accuracy. Errors have been taken as the incorrect selection of a screen area, other than the that associated with the target area. In comparative studies this has been used to distinguish between devices. However some studies are able to relate device accuracy to task conditions. Sears and Shneiderman (1991), considering targets of size (32, 16, 4, and 1 pixels) found that errors only occurred for the smallest targets. Trankle and Deutschmann (1990) found that error rates varied between individuals but not across conditions. Card et al (1978) found that the number of errors decreased with increasing target size, and a slight increase in error rate with increasing distance.Hill et al (1991) found that accuracy was a function of task (e.g. drawing or dragging) and is influenced by the required button presses.

2.4.1.1.3 Other Performance Measures

In some comparative studies subjective evaluation measures have been used to assess devices (Haller et al, 1985, Murata, 1991, Abernethy & Hodes, 1987, Wolf, 1991, Sears & Shneiderman, 1991, Karat et al, 1986, Ewing et al, 1986). A consensus from these studies is difficult to arrive at due to the differences between the studies (tasks, subjects, and devices). However, Wolf (1991) found that a higher preference for a device was associated with a greater previous experience with the device. Interestingly, Ewing et al (1986) comparing the mouse and keyboard devices, using a repeated measures design, found that those people who used the mouse device second, had a higher preference for the mouse than those who encountered the mouse device first.

Radwin et al (1990) and Lin et al (1992) used two additional performance measures; Movement path distance and Root Mean Square (RMS) cursor deviation. 'Movement path distance was defined as the sum of periodically sampled cursor displacement magnitudes along the path traversed when acquiring a target' and it gives an approximate measure of the distance actually moved by cursor in acquiring the target. RMS cursor deviation was determined by 'the sum of the squares of the periodically sampled differences in displacement between the actual path the cursor traversed and a straight line drawn between the point at which the cursor first moved outside of the home region to the point at which the cursor crossed the target perimeter'. This gives an indication of how well the movement trajectory corresponded to a straight line.

2.4.1.1.4 Individual Correlates of Performance

Mouse performance, as measured by task completion time, has been found to be associated with cognitive spatial ability (Taylor & Hinson, 1989; Barker et al, 1990), as quantified by the Embedded Figure Test (Witkin, 1950). Additionally, Barker et al (1990) found that mouse performance was correlated with limb positioning skill (as measured by a pegboard task) and the ability to map from the horizontal plane to a corresponding vertical plane. However, Ewing et al (1986) found no relationship between personality, as measured by the Keirsey Temparment Scale (Keirsey & Bates, 1984) and mouse performance.

2.4.1.2 Targets and Their Size

Targets have taken the form of characters (Card et al, 1978; Karat et al, 1986; Ewing et al, 1986; Murata, 1991; English et al, 1967; Haller et al 1985), rectangles (Sears & Shneiderman, 1991; Hill et al, 1991; Trankle & Deutschmann, 1991; Epps, 1986; Price & Cordova, 1983; Wolf, 1991) and circles (Radwin et al, 1990; MacKenzie et al (1991); Lin et al, 1992; Barker et al, 1990). Wolf (1992) required subjects to select spreadsheet cells.

Most studies take target size to be that associated with the displayed target, as opposed to the target dimensions projected onto the device movement area. Sears & Shneiderman (1991), using rectangular targets specified targets of size (mm); 0.4 x 0.6, 1.7 x 2.2, 6.9 x 9.0, and 13.8 x 17.9. Trankle & Deutschmann (1991) used square targets of size (mm); 2.5, 5.0, and 10.0. Epps (1986) used square targets of size (mm); 1.3, 2.7, 5.4, 10.7, and 21.4. Hill et al (1991) used rectangular targets of 12.8mm x 19.2mm. Lin et al, (1992) used circular targets of radius (mm); 2.9, 8.1, and 23.5. Similarly, Radwin et al (1990) used circular targets of radius (mm); 2.7, 8.1, and

24.2; and Barker et al (1990) of radius 2mm. Karat et al (1986), used a rectangular target of 14 x 13 mm. Card et al (1976), used targets of; 1, 2, 4, and 10 characters. English et al (1967) used targets of 1 or 5 characters. MacKenzie et al (1991) using a Fitts' reciprocal tapping task used rectangular targets whose widths ranged from 8 to 64 screen pixels on the Apple Macintosh II computer system.

Comparisons across studies with respect to target size are difficult due to the differing experimental conditions and different target presentations. However, within most studies increasing target size, with all other things being equal, reduced the target acquisition time.

Movement Path Deviation has been shown to reduce for increasing target width (Radwin et al, 1990; Lin et al, 1991).

2.4.1.3 Target Position

Position, in this instance, refers to the location of the screen target area of the display. For example the top left, or bottom right, of the display. Sears & Shneiderman (1991) defined four target positions at the corners of the display. However, position was used as a control measure rather than an experimental variable. Other studies have not controlled for target position.

2.4.1.4 Amplitude of Movement

Movement amplitude (distance to target), as was the case for target size, has largely been studied with reference to the display movement amplitude as opposed to the device movement amplitude. It is identified with the initial distance to be moved rather than the actual distance moved. Movement amplitude, although a continuous variable, has often been categorized for the purpose of statistical analysis.

Radwin et al (1990) used amplitudes of (mm); 24.4 and 110.9. Lin et al (1992) used amplitudes of (mm); 24.3 and

61.7. Card et al (1978) used amplitudes of (mm); 10, 20, 40, 80 and 160. Trankle & Deutshmann (1991) used amplitudes of (mm); 25, 50, 100. Haller et al (1985) used distances of long, medium, and short. Epps (1986) used amplitudes of (mm); 20, 40, 80, and 160. MacKenzie et al (1991) used movement amplitudes of 64, 108, 216, and 432 screen pixels on the Apple Macintosh II computer. In all cases the time to acquire the targets, all other things being equal, increased with increasing movement amplitude.

Movement path deviation has largely been accounted for by movement amplitude and RMS cursor deviation increased with increasing movement amplitude (Radwin et al, 1990; Lin et al, 1992).

2.4.1.5 Direction of Movement

For most purposes direction of movement refers to the direction of a straight line joining the centre of the target to the centre of the start location. This notion of direction is not necessarily the 'line' taken by the user. Most investigators have categorized the movement into horizontal, vertical, and diagonal groups. Radwin et al (1990), Barker et al (1990), and Lin et al (1992) used eight directions of movement; 0, 45, 90, 135, 180, 225, 270, and 315 degrees.

Targets requiring a Horizontal movement have been found to take longer to acquire than those requiring a vertical movement (Radwin et al ,1990 and Trankle & Deutschmann, 1991). Barker et al (1990) found diagonal targets quicker to acquire than horizontal or vertical targets. Others have found no, or a minimal, effect of direction on target acquisition time (Lin et al, 1992 and Card et al, 1978).

Lin et al (1992) and Radwin et al (1990) found that RMS cursor deviation and movement path distance measures (see section 2.4.1.1.3) are greater for diagonal movements.

2.4.1.6 The Mouse-Cursor Transfer Function

The relationship between movement device amplitude and the resultant display movement amplitude has been referred to by a number of investigators; 'mouse-cursor transfer function' (Trankle & Deutschmann, 1991), 'gain' (Arnaut & Greenstein, 1990, Buck, 1980, Carey, 1985), 'control-display gain' (Gibbs, 1962, Lin et al, 1992), and 'control-display ratio' (McCormick & Sanders, 1976). The term mouse-control transfer function is chosen here as it implies less constraint on the relationship between control device movement (mouse) and the resultant display movement (cursor).

In most cases the direction of the mouse (or device) movement is such that a movement of the device away from the body produces a corresponding upward cursor movement on the screen, and left and rightward device movements produce corresponding movements in the same direction on the screen. Although alternative correspondences could be implemented, they are not often found in practice.

The relationship between the control movement (that of the device) and the screen pointer (cursor) movement is usually linear and is often referred to as the gain, where:

$$\text{Gain} = \frac{\text{Display movement}}{\text{Control movement}} \qquad \text{Equation 2.0}$$

and this ratio is held constant for any particular movement (Figure 2.0a).

Linear Gain

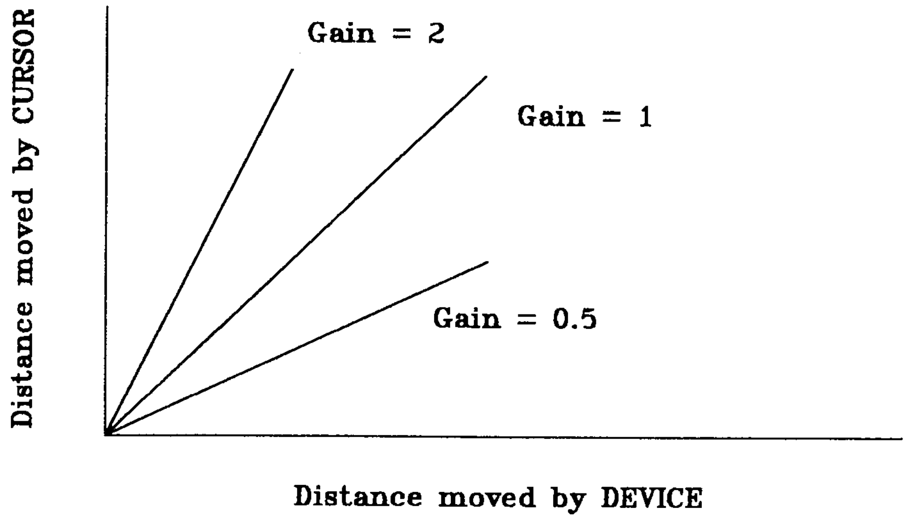


Figure 2.0a

'Accelerator' Gain

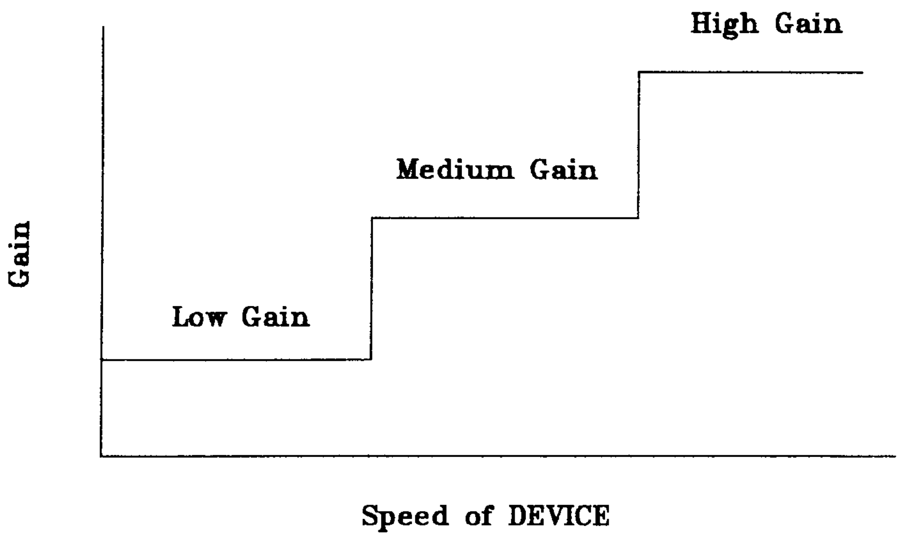


Figure 2.0b

However, with some mouse driver software an 'accelerator' is employed, whereby for different ranges of mouse speed, different linear gains operate (Figure 2.0b)

For the purposes of this study the term gain will be used where the relationship between device control movement and cursor display movement is linear and the ratio of the two is held constant throughout the period of interest.

Lin et al (1992) and Trankle & Deutschmann (1991) provide brief reviews of non-mouse studies which have investigated gain. Trankle (1989) indicates that research into the mouse-cursor transfer function has been largely neglected. Many indirect device studies fail to provide the details of the mouse-cursor transfer function used.

Trankle & Deutschmann (1991) found no difference in task performance using the mouse device for linear gains of 1:1 and 2:1. However, they did report a decline in performance when an accelerator (the linear gain in operation was doubled for device speeds greater than 100 mm/s) was used. This poorer performance persisted even after 1200 trials.

Lin et al (1992) using gains (Control:Display) of; 2:1 (2), 1:1 (1), 1:2 (2), and 1:4 (4), for a mouse device, found that task performance was better for the gain values of 1 and 2. Epps (1986) reported the optimal value of gain for the Mouse Systems Optical mouse was 3. Karat et al (1986) used gains of 1 and 2. Card et al (1978) used a gain of 2.

Performance with smaller targets has been found to be particularly disadvantaged by gains greater than 1:1 (Lin et al, 1992). Similarly, large movement amplitudes are particularly disadvantaged by low gains (Lin et al, 1992). RMS cursor deviation increased with increasing gain (Lin et al, 1992).

Lin et al (1992) concluded that gain was not as important as target width and movement amplitude, as target width and movement amplitude accounted for 74% of the variance of task completion time compared; to only 1.4% for gain. Similarly, Trankle & Deutschmann (1991) concluded that movement amplitude and target width have the strongest influence on cursor positioning times with the gain having little, if any, influence on task completion times.

2.4.1.7 Experimental Tasks

This subsection attempts to draw together the experimental tasks by means of broad task classifications.

2.4.1.7.1 Primary and secondary tasks

Most of the studies so far considered have employed a 'primary task design'. A primary task design is one whereby the subject's task is the operation of the input device, as opposed to a secondary task design, whereby the input device is used to complete some other main task. In primary task designs the use of a device is seen as an end in itself, with minimal additional choice, visual search, and discriminatory processes occurring.

Primary design tasks have typically required subjects to select a target area of the screen. However, MacKenzie et al (1991), using a computer based version of a Fitts' reciprocal tapping task, compared 'dragging' (relocating a screen object) to pointing for the same task conditions. The dragging version of the task was found to take significantly longer than the pointing form of the task. Those studies that have employed a secondary task design are described below.

Karat et al (1986), required subjects to carry out a number of subtasks in order to complete an experimental trial. Each subtask required the user to make a selection from a menu with half of the trials also requiring the input of some text.

Ewing et al (1986), using a database retrieval system required users to select a number of embedded text strings as part of a single trial.

Mack & Montaniz (1991) required users to carry out an assortment of tasks; document comparison, calendar amendment, using a computer based calculator, and spreadsheet editing.

Abernethy & Hodes (1987), in evaluating an ergonomically engineered mouse, used tasks taken from an operable system which required users to make both fine and course device movements and to switch to keyboard entry for some operations.

Haller et al (1985) investigated the lightpen, graphics tablet, mouse, trackerball, voice recogniser, and cursor keys for correcting single character errors in a page of displayed text.

Wolf (1992) compared the mouse, keyboard, and handwriting tablet devices for correcting errors within a spreadsheet. The spreadsheets were taken from two commercially available packages; Microsoft's Excel and Lotus's 123 spreadsheet packages.

Goodwin (1975), studying the lightpen, lightgun, and keyboard employed three types of task. The first required the user to acquire, in numerical order, ten targets (single numerical digits 0 - 9) randomly displayed on the display. The second task used ten identical single character targets ("M") which were randomly positioned on the display and the user had to acquire the targets in a left to right, top to bottom, sequence. The third task required the user to identify, and then acquire and mark, word errors within a paragraph of text. The sequential task was found to be the easiest ("M") and the arbitrary digit location the most difficult.

Secondary tasks may either occur in parallel with, or as an integral part of, the primary task. Parallel tasks are referred to as 'secondary independent tasks' and integral tasks as 'secondary contributing tasks'. All of the above studies have used a secondary contributing task design.

2.4.1.7.2 Discrete and serial tasks

This distinction is particularly relevant to primary task designs. A discrete task is one where the individual's actions are divided into a number of single trials, each consisting of one discrete action, such as pointing to an area of the screen. A sequential task is one where a number of atomic actions are required by the user in order to complete the task.Most studies have employed a primary task design using a discrete selection task. Epps (1986) and Taylor and Hinson (1989) required users to 'chase' around the display acquiring targets. The centre of the previously acquired target furnishing the start position of the next.

2.4.1.8 The effects of experience

A distinction made here is that between task and device experience. Device experience reflects the skills which a user of the device has acquired through previous encounters with the device. Task experience refers to the user's familiarity associated with a particular task.

Most studies, for a particular task, reported initial strong learning effects which reduced over a relatively short number of trials: Radwin et al (1990) and Lin et al (1992), after 720 trials, MacKensie et al (1991) after 160 trials (for the Fitts' reciprocal tapping task), Epps (1991) after only 40 trials, and Card et al (1978), after 1200-1800 trials, reported that performance with regard to completion time stabilised. Sears and Shneiderman (1991) using a mouse experienced user found that, for all target sizes, completion times reduced with task experience. Similarly, Goodwin (1975) and Karat et al (1986) reported strong task learning effects during the experiment.

Learning effects have been assumed to be indicated by a significant reduction in completion time from one block of trials to the next. A lack of a significant difference between adjacent blocks has been taken as indicating the stabilization of performance. Some mouse studies have employed DeJong's equation for learning (DeJong, 1957) to describe the resultant learning curves (Card et al, 1978; Radwin et al, 1990). The equation for learning is given by:

$$T_n = T_1^{-an} \quad \text{Equation 2.1}$$

Where: Tn = Time for nth trial

T1 = constant

a = constant

Wolf (1991) identified experienced mouse users as those who had used a mouse for a period of 1 hour, or more, a week during the preceding three months. The inexperienced mouse users took almost twice as long to complete the tasks, using the mouse, than the experienced device users, and this was largely due to differences in selection and movement times between the two groups. Furthermore, inexperienced mouse users were more likely to prefer the writing tablet to the mouse than the experienced mouse users.However, Mack & Montaniz (1991) reported no main effect of device experience for their four tasks (editing a document, using a calculator, using a calendar, and correction of a spreadsheet), but they did find that experienced mouse users were significantly better than inexperienced users for the spreadsheet task.

2.4.1.9 Subjects

Most studies have used undergraduate students or recruits from employment agencies as subjects. Karat et al (1986) used 26 subjects from an employment agency, most of whom were women. MacKenzie et al (1991) used 12 computer literate students. Mack & Montaniz (1991) used ten subjects from an employment agency. Hill et al (1991) used 40 subjects. Wolf (1991) used 17 subjects, most of whom were recruited from an employment agency. Card et al (1978) used five undergraduates. Sears and Shneiderman (1991), used 36 and then 20 subjects (two experimental parts) drawn from the university's subject pool. Ewing et al (1986), using the same subject pool as above, used 35 subjects. English et al (1967) used 11 subjects in total; 8 of them experienced with the 'on line' system and 3 inexperienced with the system. Presumably, neither group was familiar with the mouse device. Price & Cordova (1983) used forty-two Calma Company employees. Epps (1991) used 12 subjects with equal numbers of men and women. Radwin et al (1990) and Lin et al (1992) used ten subjects. Murata (1991) used 10 undergraduate students. MacKenzie et al (1991) used 12 students 11 of whom were men. Barker et al (1990) used 18 undergraduate students. Taylor and Hinson (1989) used 10 male and 10 female student subjects.

Sears and Shneiderman (1991), Mack and Montaniz (1991), Hill et al (1991), Price & Cordova (1983), Ewing et al (1986), Wolf (1991), Lin et al (1992), and Radwin et al(1990), used subjects whose usage varied in respect of their mouse device experience. Haller et al (1985), Epps (1986), Trankle and Deutschmann (1991), Barker et al (1990), Taylor and Hinson (1989), Karat et al (1986), and Card et al (1978), used subjects with no mouse device experience.

2.4.1.10 Fitts' Law

Fitts' Law (Fitts, 1954) has often been used to describe people's input performance for pointing devices (MacKenzie et al, 1991; Epps et al, 1986; Straayer, 1991). The following section provides details of Fitts' Law as applied to mouse device studies. However, the interested reader might also like to read the relevant section in the following chapter (section 3.2.1) which also discusses Fitts' Law, but within a broader context. Card et al (1978) was one of the first studies to apply Fitts' Law, using the formulation below, to tasks using the mouse input device. The law predicts that the time to move to a target is logarithmically related to the ratio of the movement amplitude to the target width (MacKenzie et al, 1991). More formally:

$$MT = a + b \log_2(A/W + 0.5) \quad \text{Equation 2.2}$$

where: a & b are empirically determined constants

A is movement amplitude

W is target width

MT is movement time

Most device studies have found this equation to provide a 'good fit' to their data (the coefficient of variation r^2 typically being greater than 0.9). The constants a & b have varied between studies (see table 2.0).

<u>Fitts' Law and Mice Studies</u>

Study	Gain	a	b	r^2
Card et al (1978)	2.0	1.03	0.096	0.83
Epps et al (1986)	3.0	0.11	0.39	0.70
Radwin et al (1990)	?	-0.06	0.15	0.88
MacKenzie et al (1991)	?	-0.11	0.22	0.98
Lin et al (1992)	1*	0.13	0.14	0.98

Table 2.0

Typical values for b have been suggested as between 0.2 and 0.9 (MacKenzie et al 1991) and as being between 0.07 and 0.12 (Card et al, 1983). Card et al (1980) suggest the following typical values for the above equation(2.2):

$$MT = 0.8 + 0.1 \log_2(A/W + 0.5) \quad \text{Equation 2.3}$$

This does not include the time to depress the button which adds a constant delay of 0.2 to Equation 2.3, or the reaction time.

2.4.2 Discussion

This discussion will follow a similar structure to the above section on mouse studies. It is intended to provide a critique of the above studies with the aim of developing an experimental programme to meet the aims of the research.

* Optimal gain

2.4.2.1 Previous studies

There has been a general lack of experimentally based investigations concerning pointing devices (Whitefield, 1986). More specifically, despite the increasing popularity of the mouse device, there is a paucity of human factors research concerning the mouse and its use (Price & Cordova, 1983; Hodes & Akagi, 1986). There have been studies concerned with the mouse but of those most have been concerned with comparing a number of input devices along a small range of performance criteria. Relatively few of these studies have sought to optimise the input of information using the mouse device. There has been little or no research investigating the acquisition of device movement skills over a long period of time. Trankle & Deutschmann (1991) indicate that research into the mouse transfer function has to date been largely neglected. The effects of task factors, such as direction of movement and system gain, have not been found to be consistent across studies. Thus, there is a need for research into many aspects of human computer interaction using the mouse input device.

2.4.2.2 Task variables studied

A knowledge of the influence of task variables is important, as it may permit the input task to be optimised, or improved, by incorporating this knowledge into the design of the task. A relatively small number of task variables have been investigated in previous studies. The variables that have received most attention are those of target width and movement amplitude.

2.4.2.2.1 Performance measures

Although the most widely used performance measure is movement time, this has often been defined and measured differently in different studies.

2.4.2.2.1.1 Task completion time

Movement time has largely been implemented as a single measure with some studies confounding movement time with reaction and or selection time.

Many studies have ignored the reaction time, that is the time from the presentation of the target until the onset of device movement occurs. It is possible that differences in task conditions may produce differences in reaction time (see section 3.3). This could be explained by reference to cognitive factors such as preplanning. Therefore, reaction time should not only be included in performance time, but should also be individually measured. Similarly, some studies have included selection time in their performance measure, and others have not. As with reaction time this should be an identifiable separate period in time.

Buck (1980) and Arnaut and Greenstein (1990) categorised the movement phase of the target acquisition activity into two stages. Buck distinguished between acquisition and overshoot times, and Arnaut and Greenstein between gross and fine movements. Both studies based their categorisations on the proximity of the screen cursor to the screen target. This permitted a separate analysis of each of the two movement stages. Both studies reported that the different stages of movement were affected differently by certain aspects of the task.

A classification of submovements based purely on the proximity to the target seems too crude a criterion for such a distinction. However, their findings do suggest that a study using more subtle submovement criteria might prove very useful for predicting the effects of certain task dimensions on the submovements, and hence the overall movement.

Further analysis of the performance time based on the partitioned stages of completion time (reaction time, movement time, and selection time) could be carried out. Such a model of device performance may be regarded as a temporal model. Similarly, a partitioned model of the movement phases would be informative, and could be regarded as a temporal-spatial model of performance. Additionally, measures such as those proposed by Lin et al

(1992) and Radwin et al (1990), movement path distance and RMS cursor deviation, should be applied to the different components, of the temporal-spatial model of performance.

Most studies fail to disclose details of their method of timing, and as Chapter Five will show (sections 5.1 & 5.3), the incorrect implementation of experimental timing is likely to produce relatively large random errors in timing. It is felt that had the complex issues involved in correct timing been addressed the authors would have mentioned this. Therefore, it is most likely that they have not addressed such issues and their timing results may contain relatively large random errors (see section 5.4).

2.4.2.2.1.2 Accuracy ('hitting the target, or not')
The mouse has repeatedly been shown to be a relatively accurate device. Some researchers suspect that this high accuracy is obtained at the expense of speed (Trankle and Deutschmann, 1991). That is, if subjects concentrated less on accuracy they may complete the trials more quickly but would incur more errors as a result. Karat et al (1988), stressed speed and not accuracy when instructing their subjects and obtained a relatively high error rate of 22% compared to typical error rates of 5%. However, this cannot be attributed solely to the experimental directions given to the subjects and the authors suggested that the errors were mostly due to subject difficulties with the mouse buttons.

It is possible that subjects tend to locate the cursor in the centre of the target, when all that is required is the location of the cursor within the target area. This activity might be further encouraged by the shape of the cursor employed. Sears and Shneiderman (1991), Lin et al (1992), Epps (1986), and Radwin et al (1990) employed 'cross hair' shaped cursors, and an arrow was used by Trankle and Deutschmann (1991). This could be investigated by using the notion of the 'effective target width' (Welford, 1968), but most of the previous studies would be unable to execute this analysis, due to the nature of their measurements.

In some of the experiments, the subject's near visual acuity may not have been sufficiently good so as to enable them to meet the visual demands of the task, or it may have influenced their behaviour. Arnaut and Greenstein (1990), using the trackerball and touchscreen devices, were the only experimenters who reported checking their subjects near visual acuity. They ensured that all subjects had a 20/29 near acuity, as measured by the Bausch and Lomb Orthorater.

Accuracy in most of the above studies was measured by the number of target selections which occurred outside the target area, that is, the error rate. This measure was the only one available given the experimental techniques employed. Siddall, Holding, and Draper (1957) distinguished between errors of aim and errors of extent. Errors of aim were said to have occurred when the line of movement deviated from the optimal line/s connecting the movement's start location and the target area. Errors of extent were said to have occurred when the movement was shorter or longer than the minimal movement required to reach the target (see Fig. 2.1). They found that these distinctions in errors were useful in accounting for the types of control processes used in making the movements.

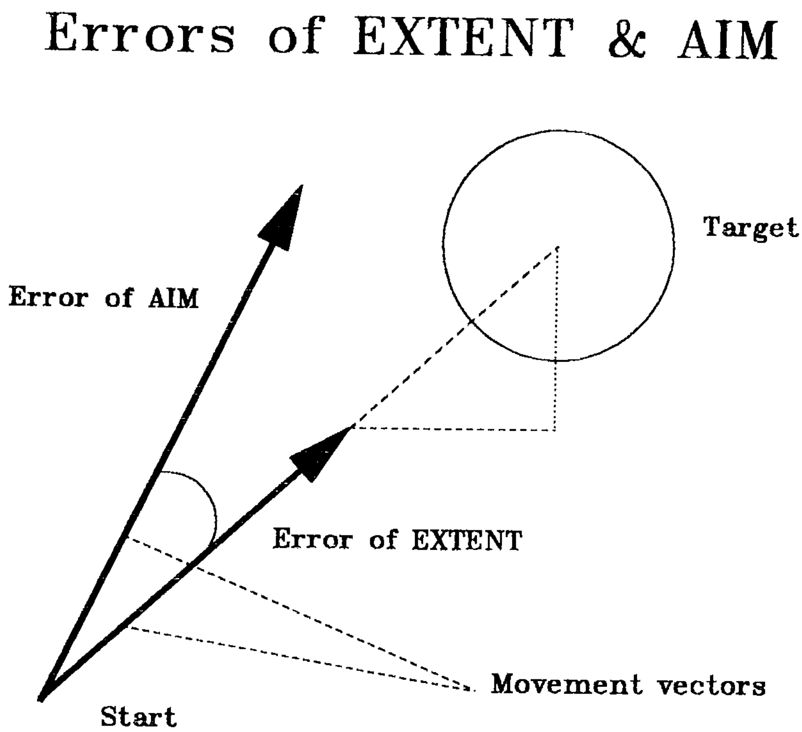


Figure 2.1

2.4.2.2.1.3 Subjective Evaluation

The status of subjective measures within science is controversial. Many researchers believe that science should be supported only to the degree that it is objective (Kosso, 1989), whilst others believe that all knowledge begins with human experience, and they are not separable, and hence subjectivity is inevitable (Muckler & Seven, 1992). Research utilizing subjective measures often seeks to establish the level of subjective/objective consistency, with the areas of inconsistency often yielding useful information (Muckler & Seven, 1992). Subjective measures within input device studies can be useful in supporting findings based on more objective measures, highlighting defects in the experimental design, and can yield information not provided by the more objective measures (Carey, 1985). However, unless used with careful consideration and application, subjective data can produce meaningless or distorted measures of user preferences, and so subjective measures should be confined to particular devices for particular tasks (Carey, 1985).

2.4.2.2.2 Movement amplitude and target width

Relatively few of these studies have distinguished between the movement amplitude required of the mouse (control amplitude), and that required of the cursor (display amplitude). Similarly, many studies have not distinguished between control target width and display target width. In fact most studies neglect the 'virtual target area' on the mouse pad. Most studies have provided the details of cursor movement amplitude and screen target width, and where they have also provided details of the mouse-cursor transfer function, the associated control movement amplitudes and target widths could be calculated. However, some studies provide insufficient details of the transfer function making this impossible. It is indeed possible that many different control movement amplitudes and target widths could result in the same display movement amplitude and display target width, providing that different mouse-cursor transfer functions were used. If the overall movement time was found to be largely determined by control movement amplitude and target width, this could lead to very different subject performances between different studies specifying the same target conditions.

The available range of control movements and target sizes, that can be made without the need to lift the device, is limited by the dimensions of the mouse pad, the system gain, and the resolutions of the input device and screen (See section 5.2). As a consequence of these consideration the system gain would be limited to a minimum of about 2 and a maximum gain of 7 for relatively small targets.

In many cases the movement amplitude is ambiguously defined. It is often unclear whether this distance is from the start location to the target centre, or from the start location to the nearest target boundary. If the target size remains constant, the target is circular, and the amplitude of movement is constant, then there will be no error due to this ambiguity. If however, the measured movement amplitude is taken to the centre of the target (Radwin et al, 1990, Lin et al, 1992) then the minimum required movement amplitude will be less for larger target areas as their boundaries will be closer to the start location. This could lead to a systematic effect (if the users did not locate to the centre of the target area) which would tend to reduce the movement times for larger targets, all other things being equal.

Similarly, as previously mentioned, a target which produces different movement amplitudes from different orientations could be a potential source of bias. The magnitude of this potential error would be increased for 'long thin' targets. Circular targets have the advantage of presenting the same distance to a fixed radial point regardless of its orientation which is not the case for non circular targets. However, in real life tasks, people are not often required to acquire circular targets, but are more typically asked to acquire text or rectangular targets.

Movement amplitude and target width are potentially able to take up a wide range of numeric values within certain limits. However, most studies have only used a few of these potential values and have arbitrarilly labelled them as experimental conditions. In particular very small and large movement amplitudes, and large target areas have been neglected. The categorisation process of target sizes and movement amplitudes has aided the statistical analysis of the experimental data, but how should these categories be decided?

Most studies gave no justification for their selections of movement amplitude or target width. Sears and Shneiderman(1991) claimed their target widths were representative of most values to be found in practice. Similarly, Card et al (1978), presented targets of various character lengths to be representative of different word length targets. This issue is further complicated by the additional variability of the mouse-cursor transfer function. A given display target width and display amplitude will require different control movement amplitudes and control target widths depending upon the mouse-cursor transfer function.

2.4.2.2.3 Target and cursor positions

The starting and finishing locations of a movement might affect performance even when amplitude, gain, and all other factors have been controlled for. It is conceivable that a control movement made from a location nearer to, or further from, the mover's body will be different due to the relative differences in the required positioning of the limbs.

The starting location of the screen cursor in some of the above studies was constant throughout the experiment (Radwin et al, 1990; Sears and Shneiderman, 1991), and for many other studies was not specified. In practice it is unlikely that the starting location of the required movement would always take place from the same screen position. It is most likely that movements which always start from the same location will be better executed than those which start from differing starting locations. Trankle and Deutschmann (1991), and Arnaut and Greenstein (1990) controlled for start position variability by starting trials from one of the four screen corners.

Target positions should also be controlled for in the same way. Sears and Shneiderman (1991) positioned targets at one of the four corners of the screen.

2.4.2.2.4 Direction of movement

The direction of a movement may influence its characteristics by the different demands upon the different components of the system producing the movement.

The reported effects of direction upon movement are equivocal. Some studies have found differences due to the demanded direction of movement, and others have not. Where studies have found a directional effect on movement time (Radwin, 1990; Trankle and Deutschmann, 1991) the horizontal movements were found to take a little longer than vertical movements. Vertical movements with the mouse are associated with the upper arm and horizontal movements with the forearm. Schmidtke and Stier (1961) found that movements predominantly involving the upper arm were quicker than those using the forearm. Alternatively, the anthropometrics of the experimental layout may have favoured upperarm movements to forearm movements. However, Radwin et al (1990) and Lin et al (1992) were the only study to control for individual differences in anthropometrics. Furthermore, the mouse-cursor transfer function used by Radwin favoured the horizontal direction, that is the gain was higher in the horizontal direction than in the vertical direction.

2.4.2.2.5 The mouse-cursor transfer function

Trankle and Deutschmann (1991), reported no difference in user performance for a doubling of the linear transfer function (gain) from 1 to 2, and perhaps surprisingly found that the use of an 'accelerator' in the transfer function worsened users' performance. Lin et al (1992) used gains of 0.5, 1, 2, and 4 and reported an optimal gain in the region of 1 to 2. Similarly, Epps (1986) reported a slightly higher optimal gain of 3. Given these values of optimal gain for the mouse, it is not surprising that Trankle and Deutschmann reported no performance difference between a gain of 1 and a gain of 2.

Given that many of the commercial mouse driver products allow the user to select linear gains from values of 1 to 10, the above range of experimental gains seems inadequate. Similarly, the threshold used for the 'accelerator' could have taken a number of values, but only one (100 mm/s) was used in this study. The default gain of some mouse devices is 5 with an available range of 1 - 10. However an interpretation of this figure is difficult as gain in this sense usually refers to the ratio of device resolution to screen resolution (mouse counts to screen pixels) rather than screen movement to device movement.

In their paper, Trankle and Deutschmann (1990) do not provide information regarding the instructions subjects were given regarding the mouse-cursor transfer function condition employing the 'accelerator'. If the subjects were not instructed, and possibly only slightly practiced prior to the experimental trials, it is possible that they would not choose to use this facility.

Most transfer functions define a linear relationship between the control movement and the display movement. The 'accelerator' provides different linear relationships for different speed thresholds. The possibility of curvilinear relationships has not been explored.

Arnaut and Greenstein (1990) provide a word of caution against gain, and gain alone, being used as an experimental control variable. They found that for identical values of gain, achieved through the use of different ratios of control measures to display measures, different movement times were obtained.

2.4.2.2.6 Experimental tasks

As already indicated most of the previous studies, with the exception of Ewing et al (1986), Wolf (1991), Haller et al (1985), and Karat et al (1986) have employed discrete primary task designs. Of these studies Wolf and Haller provide subtask completion times whereas the completion times given by Karat et al and Ewing et al are difficult to interpret due to the number of subtasks, such as typing in data, included within their measures of task completion time.

Studies employing the discrete primary tasks have largely been repetitive requiring little additional choice, visual search, or discriminatory processes on the part of the user. Such tasks facilitate experimental measurements but they may lead to a false indication of the 'real world' device performance. The subject's performance on such a task, compared to their performance on a 'real world' task might either be improved (due to the lack of additional cognitive demands listed above), or, due to the relatively 'low arousal' of the experimental situation, may result in a poorer performance in the laboratory (see Laycock & Peckham, 1980). However, if more complex 'real world' tasks are employed it becomes more difficult to attribute task performance mostly to device performance.

When investigating the acquisition of device skill through experience on more complex tasks, it becomes more difficult to attribute improvements in performance to device learning, as opposed to task learning.

Goodwin (1975), found that randomly positioned targets were more difficult to acquire than targets whose positions were known in advance, or expected at a particular location. Many of the above studies have employed a random (from the user's perspective) target location for target presentation. In most practical applications involving the mouse input device, experienced application users should be able to anticipate the location of targets which will facilitate their performance.

All of the above studies have required the user to locate a screen cursor onto a target area and then to indicate a selection of that target. This is typical of much of the activity carried out when using a mouse. However, studies have not considered such activities, as 'dragging', and drawing which are also carried out with a mouse. These activities could be conceived as being examples of locating and/or selection (Carey, 1985), but it seems more appropriate to consider them as separate activities to be studied in their own right.

In summary many of the tasks employed to investigate task performance using the mouse input device have borne little resemblance to the tasks found in practice by the user. The experimental tasks employed have placed relatively small cognitive demands on the user; been boring; demanded repetitive device movements in quick succession; and presented targets at largely unpredictable locations.

2.4.2.3 Drawing an analogy

The aim of this section is to show that the task factors of amplitude of movement, target size and gain for computer input using an indirect pointing device, are interdependent. In demonstrating this interdependence between these task conditions it is helpful to draw an analogy between the computer input scenario and that of drawing with the aid of a pantograph.

2.4.2.3.1 The Pantograph/computer task correspondences

The mechanical pantograph provides an insightful analogy to computer input systems employing indirect pointing devices. Figure (2.2) shows a simple mechanical pantograph.

A Simple Pantograph

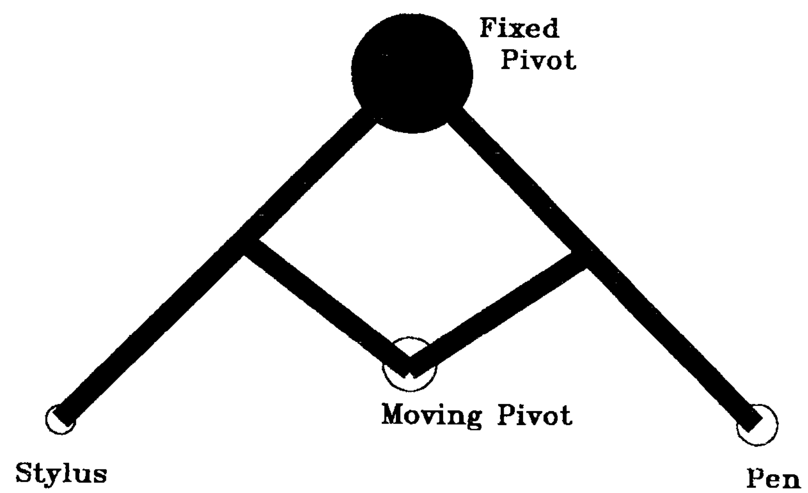


Figure 2.2

The movement of the stylus corresponds to the movement of the input device. The movement of the pen corresponds to the movement of the cursor. The display target corresponds to a target area drawn onto a sheet of paper beneath the pantograph. The corresponding device target corresponds to the area that would have to be traced by the stylus in order to exactly redraw the target area. The 'gain' of the input system corresponds to the mechanical advantage of the pantograph system.

With the above pantograph it is conceivable that tasks could be designed that would correspond to the acquisition of a target by a computer input device. Employing a suitable performance measure, it would be possible to investigate the pantograph task conditions with respect to the performance criteria. It is likely that task conditions, such as stylus/pen movement amplitudes, and stylus/pen target sizes, would influence the performance of the task. That is, we could carry out experiments similar to those anticipated with the computer input system.

2.4.2.3.2 Experimental difficulties

The pantograph system graphically illustrates one of the main difficulties encountered in attempting to identify the individual effects of task factors on performance for such systems. Namely that a change in Gain cannot occur independently of changes in control/display amplitude or target size. This may be demonstrated more formally as follows:

Let D_1 be the distance moved by the stylus, D_2 be the distance moved by the pen, T_2 the target area, and T_1 the corresponding control target area at the stylus. Then:

$$D_1/T_1 = D_2/T_2 = \text{constant} \qquad \text{Eqn (2.4)}$$

A change in any of the above parameters will require a corresponding change in one of the other parameters. It is impossible to independently vary any of the above parameters without having a corresponding effect on the value of one, or more, of the other parameters.

This lack of independence between these important task factors has to be considered when investigating their effects on performance. In attempting to investigate the effects of variation in one factor there must also be corresponding changes in at least one other factor, and therefore any resulting differences in performance cannot be attributed solely to the variation in one of the factors. A similar situation arises within physics for the equation of state for an ideal gas which relates the temperature, pressure, and volume of an ideal gas (Ramsden, 1985).

$$P_1V_1/T_1 = P_2V_2/T_2 \quad \text{Equation 2.5}$$

Where: P = pressure

V = Volume

T = Temperature

2.4.2.3.3 Implications for previous and current studies

Experimental findings which claim to have identified influences due to amplitude movement, or target dimensions, will have confounded at least two of the component variables underlying the amplitude of movement or target size. Thus their findings, will only be applicable to those situations where both the identified factor, and its confounded counterparts, are held constant. The identification of an effect due to variations in these variables, by means of an ANOVA test, does not meaningfully identify the variables' effects, as the ANOVA model assumes that such variables are independent from each other. Therefore, we cannot meaningfully talk of an effect of gain but instead we can describe the influences of target size and movement amplitude for a particular value of gain.

2.4.2.4 The effects of device experience

Many of the above input device studies have failed to control for device experience amongst their subjects (Sears and Shneiderman, 1991; Radwin et al, 1990; Arnaut and Greenstein. 1990; and Ewing et al, 1986). It is assumed that in general, with all other things being equal, that a person experienced with an input device would tend to perform better than a less experienced person (Wolf, 1991; Karrat et al, 1986; Sears and

Shneiderman, 1991). A lack of control for device experience in studies will probably have less of an affect on factors employing repeated measures as opposed to factors measured between subjects. However, there are indications that previous device experience, or the lack of it, can interact with task performance (Karrat et al, 1986; Wolf, 1991).

Most studies have only analysed subjects' performance data when their performance has stabilized. In practice this has been said to have occurred when a subject's performance has not significantly improved from one block of trials to another. However, there are reasons for doubting this level of performance as being the optimal level. With the exception of Sears and Shneiderman (1991), who considered one 'expert' user, none of the above studies have contrasted the attained subject performances with highly skilled device users. There is evidence that even subjects highly practiced in a motor skill (10 million trials over seven years), still show improvements in performance (Crossman, 1959).

2.4.2.5 Task experience

Strong learning effects have been demonstrated even for discrete primary tasks. Such learning effects may be considered as comprising two, conceptually separable kinds of learning; task and device learning. Most of the above studies have confounded device learning with task learning. For primary discrete tasks it is assumed, given that the tasks are relatively simple and undemanding, that task learning occurs very rapidly after a relatively small number of trials. In the case of inexperienced device users, it is conceivable that complex task learning could interact with device skill acquisition so as to reduce the rate of device skill acquisition, and to reduce task performance.

A method of separating the influences of task and device learning on performance would be to have the subjects carry out the same set of high level tasks (Eg select an item from a menu) but employing a different input device.

2.4.2.6 Fitts' Law

Fitts' Law has been successfully applied to input tasks utilizing the mouse device (as measured by r^2 the coefficient of variation). However, invariably the particular equations resulting from the application of Fitts' Law have derived the equation coefficients a and b (see equation 2.2) from the experimental data. These values have been shown to vary between studies which has been largely attributed to the task differences between them. Therefore, if Fitts' Law were to be used in a predictive fashion, what values would be selected for these coefficients? Card et al (1983) suggest that values based upon the average figures derived from previous studies should be used. However, it is believed that the resultant predictability of such an equation would be much lower than the previously very high accountability, as measured by the r^2 scores (see table 2.0).

A number of alternative equations have been suggested to account for device movement time data. MacKenzie et al (1991) propose a modification to equation 2.1 so as to ensure that the coefficient (b) can only take positive values; as a negative value is considered to be theoretically undesirable (see equation 2.5).

$$MT = a + b \log_2(A/W + 1) \qquad \text{Equation 2.6}$$

Epps (1986) found that the equation suggested by Kvalseth (1981):

$$\log_2(MT) = \log_2(a) + b \log_2(A) + c \log_2(W) \qquad \text{Equation 2.7}$$

where: a,b,c are constants
A = movement amplitude
W = target width

was a better description than Fitts' Law of movement time for the mouse device.

Fitts' Law takes into account movement amplitude and target size which have been found to be important determiners of movement time. However, it does not consider the system gain which has been shown to affect movement time (Lin et al, 1992) so as to produce different solutions for the equation coefficients for identical values of movement amplitude and target size. Similarly, Radwin et al (1990) suggest that Fitts' law should be modified to incorporate a directional component.

Given the different ways in which researchers have defined task completion time, the derived constants for the Fitts' Law equation would be expected to be different based on this fact alone. To meaningfully average the empirically derived constants, the constituents of completion time (reaction time, movement time, and selection time), would have to be known.

Fitts' Law only yields information regarding the movement times of a pointing task. The earlier discussion on the suitability of task completion time as a description of pointing behaviour suggested that this was inadequate for many purposes. This criticism can be equally applied to Fitts' Law. This Law tells us very little about the types of movements that are made with the mouse device.

> '*Performance must be viewed in a wider perspective than just Fitts' Law*'.
>
> Straayer (1991, p 119).

2.5 Conclusions

Very little is known about mouse device usage and the factors influencing users' behaviour.

Comparisons across studies are problematic due to the varied operationalisation and implementation of experimental variables, and differences between tasks and subjects. This is further problematised due to the uncertainty regarding influencing factors which have not been adequately described.

Studies have shown that differences in mouse construction and implementation influence people's task performance. Despite this, many researchers still refer to the mouse device as if there were only one mainfestation of this device. Accordingly, there has been little research into differences between different types of mice.

Despite evidence demonstrating the importance of previous device experience when using an input device, studies have often ignored, or failed to control for, such subject experience. Furthermore, there has been very little research into device skill acquisition over a relatively long period of time.

Device behaviour has been considered within too narrow a context. Although performance, as measured by task completion time, may be important, it tells us little about the nature of input behaviour, and has limited scope for improving the input task. Much of the knowledge gained from performance time studies has to be considered within the experimental context from which it came. Studies have mostly used contrived tasks and unrepresentative samples.

As a consequence of the above conclusions the research programme described within this thesis was embarked upon.

In particular it is concerned with describing device performance within the broader context of movement phases and temporal movement stages. It will also examine individual, and task factors within this broader behavioural context.

Such movement descriptions will provide a general framework within which movements, made with different devices, can be understood in a coherent way.

CHAPTER THREE

MOTOR THEORY

This chapter attempts to place device movement within a psychomotor context. It focuses on the motor program perspective for the control of motor movements. The device movements, made within this study, are considered to fall within the class of acts termed 'discrete motor movements'. In particular the following theories of discrete aimed movements; Fitts' Law (Fitts, 1954), the iterative corrections model (Crossman & Goodeve, 1963), the two phase model (Woodworth, 1899), the single correction model (Howarth, Beggs, & Bowden, 1971), the impulse timing model (Schmidt, Zelaznik, Hawkins, Frank, & Quinn, 1979), and the stochastic optimized submovement model (Meyer, Smith, & Wright, 1982) are discussed with reference to their predicted spatial and temporal output patterns. However, such theories tell us very little about the characteristics of the submovements made, and how they relate to the task conditions. Finally, a number of kinematic studies are described which inform subsequent debates within this thesis.

3.0 Introduction

Movements are the only means by which we can act and interact with our environment (Kelso, 1982). Most computer input devices rely, to some extent, on the development of high speed motor skills for their operation, and within this context, the speed and accuracy of human motor actions can be very important determinants of overall input performance (Carey, 1985). As the literature on motor skills is vast and diverse this chapter will focus on those areas of motor theory dealing with those motor acts associated with mouse device use. A discrete motor movement is one which involves a single reaching movement to acquire a stationary target, such as pointing (Glencross & Barrett, 1989). Investigations of such deceptively 'simple' movements have revealed that they arise from very complex processes (Sheridan, 1984; Glencross & Barrett, 1989; Stelmach, 1982; Greer, 1984; Holding, 1989). Target acquisition using a mouse device is considered to be an example of a discrete motor act.

Studies of motor skill have mostly been concerned with determining and describing the processes underlying the control of movement (Kelso, 1982; Summers, 1989; Colley, 1989), and those of skill acquisition. Understanding, within this context, has sought to conceptualize movement by constructing models of the processes of movement production and control:

> *'... understanding movement means not only recognizing the [end] product of skilled behaviour - whether a particular goal was achieved accurately and efficiently - but also questioning how it is that such movements are controlled and coordinated.'*
>
> Kelso (1982, p1)

and in particular:

> *'In understanding the nature of movement organization, and the processes involved in it we should look to the movement itself.'*
>
> Kelso (1982, p2)

Further, Kelso indicates that the important aspects of a movement are those indicated by kinematic descriptions for that movement.

> '*If we are to understand the manner in which that outcome was attained, we need to analyse the kinematics of the movement.*'
>
> Kelso (1982, p12)

The need to consider the movements themselves, and their associated kinematic patterns in particular, is fully embraced within this thesis. However, it is not believed, here, that our understanding is necessarily enhanced by reducing these movements to hypothetical neuromuscular processes.

Within this thesis the movement patterns, in their own right, within a task context, are considered to form a basis for the understanding of movement, without further recourse to underlying process mechanisms. These descriptions are considered as being emergent properties (Checkland, 1976) from the interaction of the motor system and its environment (Kelso, 1982). Such descriptions of movement behaviour do not need to be justified with reference to the underlying physical processes (Valentine, 1982).

The primary concern of this thesis is describing mouse movement and not evaluating, testing, or refining theories of motor movement. However, theories of motor control can provide a context for the discussion of device movement, and attempts are made, within this thesis, to relate this investigation's empirical findings to theories of motor control. As Rosenbaum (a motor control theorist) indicates:

> '*Psychologists do not usually deny physical causes of behaviour; in fact, they are usually pleased if their models find physiological support. However, the explanations that psychologists pursue do not require one-to-one mappings of identified biological mechanisms to behavioural or mental phenomena.*'
>
> Rosenbaum (1991, p5)

An understanding of device movement derived from the movement patterns, within a task context, might facilitate beneficial changes to the task environment, whereas, an understanding based upon the hypothetical underlying neuromuscular processes is unlikely to lead to such changes.

3.1 Perspectives on motor control

Kelso (1982) suggests that motor theorists have mostly been concerned with discerning, and understanding, law like regularities in people's motor activities. In their endeavour to achieve these goals they have proposed hypothetical constructs, such as the motor program and perceptual trace (discussed shortly) as a way of understanding motor acts. Traditionally two theoretical positions have been adopted to explain the control of practiced movements; explanations have either been based on open-loop or closed-loop models of movement (Summers, 1989; Genter, 1987; Sheridan, 1984; Stelmach, 1982; Schmidt, 1982; Magil, 1985).

An open-loop system is one where the system's output is unaltered by external information during its execution. Conversely, a closed-loop system is one whose output execution can be amended on the basis of external information. The work of Adams (Adams, 1971) is that identified most closely with closed-loop models of movement production, although a concession is made to an initial motor program (Summers, 1989); and the work of Schmidt and colleagues (Schmidt, 1975; Schmidt, 1976; Schmidt, 1982; Schmidt et al, 1979) is associated with open-loop theories, in the form of the Generalized Motor Program.

It is generally agreed, amongst motor theorists, that positions based on either a pure closed-loop model or open-loop model of movement are untenable and that credible models of movement need to integrate both open-loop and closed-loop mechanisms to produce a hybrid control model (Winstein & Schmidt, 1989; Kelso & Stelmach, 1976; Holding, 1989).

More recent approaches to the study of motor behaviour (Turvey, Fitch, & Tuller, 1982; Saltzman & Kelso, 1987) have focused on the motor system's ability to self organize, especially at the lower levels of movement execution, and how it addresses the 'degrees of freedom problem'. The flexibility of the motor system is such that any given movement may be accomplished in a number of different ways (Rosenbaum, 1991; Kelso, 1982; Tuller et al, 1982). Understanding how the motor system executes a particular movement, apparently without too much difficulty, given the large number of alternative ways of effecting the movement, is the solution to the degrees of freedom problem. Summers (1989) has difficulty in accommodating these later approaches within the motor program concept.

The emphasis of the following sections on the motor program view of movement control is due to its relative importance over the past twenty years within motor theory. Further, many of those theories of motor control which are able to prescribe the types of movement that might be made with the mouse device (explored later in this chapter) are derived from the motor program concept.

3.1.1 Adams's closed loop theory

The publication of Adams's closed loop theory of motor control and skill acquisition (Adams, 1971) has been identified as an important landmark within motor theory (Kelso, 1982; Magil, 1985; Stelmach, 1982; Summers, 1989; Newell, 1991). Although his theory of movement production is mostly rejected, the original work was a catalyst for many subsequent studies (Stelmach, 1982; Kelso, 1982). The learning aspects of his theory will be discussed in a later chapter.

Adams considered that motor performance was a consequence of a comparison between a memory of movement (perceptual trace) and the feedback arising from the movement act, with any discrepancies between the two serving as the stimulus for the subsequent corrective response. The perceptual trace is formed from the consolidation of feedback from a person's past experiences with correct responses. A memory trace was proposed to account for the movement's selection and its initiation. This trace was also consolidated through experience, and was acknowledged by Adams, as a modest motor program.

A number of objections have been raised against this theory, which have contributed to the demise of this view of motor control (see Stelmach, 1982, for a more detailed discussion). These objections were: the emphasis on reinforcement theory, the accomplishment of novel movements, accounting for the uniqueness of motor acts (no two movements are identical in how they are achieved), the lack of cognitive economy, and skill acquisition in the absence of knowledge of results. In general theories of movement have shifted from a one-to-one to a one-to-many relation between what is represented and the movement sequence that is produced (Newell, 1991).

3.1.2 The motor program

The idea of a motor program is considered to have its origins in the work of Lashley (1917) who observed the leg movements of a patient made in the absence of feedback (Summers, 1989, Sheridan, 1984). Motor programs have been defined in many different ways (Summers, 1989) and have undergone several revisions in the light of empirical findings. The motor program was intended to account for the findings that suggested that some movements were made in the absence of concurrent feedback. An early definition was provided by Keele:

> *'a set of muscle commands that are structured before a movement begins, and that allows the sequence to be carried out uninfluenced by peripheral feedback.'*
>
> Keele (1968, p387)

This view of the motor program is regarded as a 'strict' interpretation of the motor program concept (Sheridan, 1984), as it appears to imply that any movement utilizing peripheral feedback cannot be considered. Objections have been raised on the grounds that 'much of the grace and subtlety of movement that is present when feedback is available deteriorates when feedback is withdrawn' (Rosenbaum, 1991). Additionally the existence of rapid feedback loops within the movement system have challenged this view of the motor program. Furthermore, researchers have argued that the motor program must operate at a more abstract level than at the level of muscle commands (Tuller, Turvey, & Fitch, 1982). It has been suggested that if one considers the role of the word 'allow' within Keele's definition it does permit movements utilizing feedback to be considered as originating from a motor program (Sheridan, 1984; Rosenbaum, 1991). The role of feedback in motor acts and the need for a more abstract description of the motor program lead Schmidt to reformulate the motor program definition:

> *'A motor program is an abstract structure in memory that is prepared in advance of the movement, when it is executed the result is the contraction and relaxation of muscles causing movement to occur without the involvement of feedback leading to corrections for errors of selection.'*
>
> Schmidt (1982, p205)

This redefinition of the motor program allows for the possibility of rapidly processed amendments to the ongoing movement allowing it to reach its goal. Thus feedback can be used to assist a current movement but not to initiate a new movement.

The production of motor activity is seen as a constructional, or generative process, in which abstract descriptions of an action, at the higher levels, are transformed into specific patterns of movement at the lower levels. It should be noted that this hierarchical view of the motor program was shared by Keele in his later deliberations regarding the motor program concept (Keele, 1982).

Rather than have separate movement programs for every motor act, the motor program is considered to represent a class of possible movement acts with movement program parameters determining the final movement act. The correct parameters are selected through the recall schema, and the anticipated sensory consequences of the movement are generated by means of the recognition schema. Research efforts have been directed to establishing candidates for the movement parameters and those invariant program features.

Investigators have found the relative timing between movements to be an invariant feature of the program, with force and duration being program parameters (The Impulse Timing Model, Schmidt et al, 1979). However reanalysis of the data supporting the relative temporal invariance of complex movements has questioned the earlier findings (Genter, 1987). Additionally there is increasing evidence to suggest that the invariant features of movement may depend upon the task and the stage of skill of acquisition (Summers, 1989). For example, it has been suggested that for simple movements, the movement endpoints may be represented in the motor program (Bizzi, Polit, & Morasso, 1976; Kelso & Holt, 1980; Laabs & Simmons, 1981; Schmidt & McGowan, 1980). To account for these findings the Mass Spring Model has been suggested (Feldman, 1986). However, studies of more complex arm movements suggest that limb trajectory is controlled in addition to endpoint location (Bizzi & Abend, 1983; Atkeson & Hollerbach, 1985). Overall, these studies suggest that the muscle properties controlled, in order to accomplish a movement, are dependent upon the task to be performed (Stein, 1982).

The notion of a hierarchical 'top-down' driven motor program has been questioned. There is now considerable evidence for the existence of relatively autonomous units of activity which constitute the building blocks of more complex motor acts (Summers, 1989; Rosenbaum, 1991). These findings suggest that the higher levels of the motor system do not determine the micro-details of movement execution, but, that instead, the fine structuring of a movement is left to autonomous subsystems. Summers suggests that the current view is:

> *'The motor program only sets up the correct patterns of interaction among the submovements of a particular motor task. Through such a dynamic system the same intended motor outcome can be achieved in a variety of different ways.'*
>
> Summers (1989, p65)

The mechanisms by which the higher centres organize the lower centres for the anticipated movement has been termed tuning (Greene, 1972; Turvey, 1977; Easton, 1978; Arbib, 1980; Kelso & Tuller, 1981). This view of a distributed motor control system is considered to be problematic for the motor program view. It is difficult to identify any part of the motor control system with the traditional motor program concept (Summers, 1989). The lower level self organized coordinated motor units probably come closest to the motor program concept. However, as Pew has indicated:

> *'We should not be searching for a unitary integrative concept of a motor program: It may have different representations at different levels in the motor system.'*
>
> Pew (1984, p26)

New approaches to the study of motor movement (direct perception, synergetics, ecological, and network modeling) are discussed by Rosenbaum (Rosenbaum, 1991). However, a full discussion of these perspectives is considered to be beyond the scope of this thesis. Further more, it is unclear what predictions, with respect to the movement patterns, would be made for device movement.

3.2 Theories of discrete motor control

The following theories of discrete motor movements have been selected for review, based on their prominence in the motor literature, and the specific nature of their predictions regarding such movements.

3.2.1 Fitts' Law

Fitts (1954) proposed one of the most influential formulations, Fitts' Law (Keele, 1968), within the area of psychomotor skill. Many authors have commented on its applicability to a wide range of motor tasks (Rosenbaum, 1991; Sheridan, 1982; Glencross & Barrett, 1989; Keele, 1982; Wallace & Newell, 1983; Gan & Hoffmann, 1988; Meyer et al, 1982; Schmidt, Zelaznik, & Frank, 1978; Andres & Hartung, 1989; Langolf, Chaffin & Foulke, 1976; Meyer et al, 1988).

Fitts derived his equation by analogy from the seminal work of Shannon & Weaver (1949) concerning the quantification of information. The now famous equation, shown below, was taken from Fitts & Peterson (1964).

$$MT = a + b \log_2(2A/W) \qquad \text{Equation 3.0}$$

Where: MT is movement time

a and b are constants

A is movement amplitude

W is target width

$\log_2(2A/W)$ is the Index of Difficulty

Despite the widespread acceptance of Fitts' Law a number of problems, some theoretical, and others empirical, have been identified with it.

In his original paper (Fitts, 1954) Fitts reasoned that for a highly repetitive, over-learnt motor response, the time to execute the movement, if made at the fastest speed possible whilst maintaining accuracy, would be determined largely by the information processing capacity of the motor system involved in generating that movement. Many tasks, to which Fitts' Law has been applied, cannot be considered as over rehearsed repetitive motor acts and would violate this assumption. That is, Fitts' Law has been applied to non Fitts' tasks, where a Fitts' task is one which meets the original task assumptions of Fitts. Additionally, given the degrees of freedom identified with the human motor system it is likely that different individuals will be using different configurations of the motor system to effect the same movements which, even if the task were a true Fitts task, would give rise to a different associated information processing capacity (Langolf, Chaffin, & Foulke, 1976).

The information theoretical origins of Fitts' Law have been questioned. Crossman & Goodeve (1963) were unable to derive the equation from the original writings of Shannon & Weaver (1949); whilst Kvalseth (1979) and MacKenzie (1989) believe that the analogical derivation of the law by Fitts was inappropriate.

We are left with the justification of Fitts' Law resting on its ability to describe empirical data. However, a number of researchers have reported instances where different formulations have given a better account of their data. Welford (1968) amended the original formulation so that:

$$MT = b \log_2(A/W + 0.5) \quad \text{Equation 3.1}$$

Where : MT is movement time

b is a constant

A is movement amplitude

W is target width

MacKenzie (1989) by direct analogy with Shannon & Weavers analysis advocates

$$MT = a + b \log_2(A/W + 1) \quad \text{Equation 3.2}$$

Kvalseth (1980) advocates a power model where:

$$MT = a\ A^b\ W^c \qquad \text{Equation 3.3}$$

Where : MT is movement time

a, b, and c are constants

A is movement amplitude

W is target width

A linear tradeoff, rather than a logarithmic one has been proposed by some researchers (Howarth et al, 1971; Schmidt et al, 1979; Meyer et al, 1982). Meyer et al (1982) suggest that a linear relationship is more likely to be observed for temporally constrained tasks rather than a spatially constrained one. It has been suggested that for tasks where the target width is greater than the movement amplitude Fitts' Law does not hold (Klapp, 1985; Klapp & Greim, 1979). Crossman & Goodeve (1963) found that for rapid movements (less than 300ms duration) the endpoint variability is less than would have been suggested by Fitts' Law.

The equivalence of movement amplitude and target width, as indicated by their ratio in determining the task Index of Difficulty (ID), must also be questioned. Differences in movement time have been found for different combinations of target width and movement amplitude which have yielded identical task IDs (Meyer et al, 1988; Sheridan, 1979).

Given the complexity of the human psychomotor system, the varied tasks that it has to accomplish, and the differing environmental conditions in which such acts are performed, it is unlikely that such motor behaviour can be described in such simple formulations as those presented by Fitts' Law. As Sheridan indicates:

'The notion that the overall performance of the human motor system plus associated feedback mechanisms can be described by any simple formulation assuming constant weighting factors over a variety of tasks, such as proposed by Fitts' Law, seems to be misfounded.'

Sheridan (1979, p185)

3.2.2 Iterative corrections model

Crossman & Goodeve (1963, 1983) and Keele (1968) propose a movement model utilizing discrete sampled feedback which predicts that the movement to a target consists of a series of discrete submovements of constant duration and relative accuracy, which increasingly reduce the distance to the target. Keele (1968), using estimates of 190ms and 260ms for the correction time derived from visual feedback, estimated the values for the coefficient (b) in Fitts' Law (see equation 3.0). He found values similar to previously empirically determined values. However, despite the lack of visual feedback, the logarithmic speed accuracy trade off has still been observed (Abrahams et al, 1983; Wallace & Newall, 1983).

Crossman & Goodeve's model drew upon engineering control principles and was expressed in mathematical terms. Their main concern was to demonstrate how such a model could account for the relationship described by Fitts' Law. However, a number of problems have been identified with this movement model.

Evidence of submovements has been found but when such submovements are distinguishable they are not found to have constant durations or cover constant proportions to the target (see section 3.5.4). The time required to respond to visual feedback has been shown to be much less than the estimates used by Keele (see section 3.2.6). The duration and accuracy of the first movement is assumed to be independent of the task demands yet some findings suggest otherwise (Jagacinski et al, 1980; Langolf et al, 1976). For a particular task ID (Index of Difficulty) there should either always be a submovement, or never be a submovement, but in practice their occurrence is probablistically related to task ID (Meyer et al, 1980; Carlton, 1980; Jagacinski et al, 1980). Further more, submovements occur more often than would be expected from the iterative corrections model. It has been shown that there is considerable endpoint variability for submovements (Meyer et al, 1988).

3.2.3 Single correction model

Howarth et al (1971) and Beggs & Howarth (1972a, 1972b) propose that the terminating accuracy of a movement is determined by a movement correction occurring approximately 250ms before the termination of the movement. The accuracy of this correction is largely determined by the remaining distance to the target at the time of this correction, such that:

$$E^2 = a + bd^2 \qquad \text{Equation 3.4}$$

Where: a & b are constants

E^2 is mean square error

d^2 is distance to target

Their model was developed for relatively slow, temporally constrained, movements. It should be noted that whilst the originators of this model refer to the importance of the 'final corrective movement' subsequent reviewers (Glencross & Barret, 1989; Sheridan, 1982) use the phrase 'single corrective movement'.

The final uncorrected movement distance was related to time for movement by:

$$d = A\ (t/T)^B \qquad \text{Equation 3.5}$$

Where A & B are constants

t is visual corrective reaction time

T is the total movement time

By reducing the time available to complete a movement the resultant terminating accuracy would be reduced as the distance over which the uncontrolled movement (d) would occur increased.

Bullock & Grossberg (1988) and Rosenbaum (1991) claim that the work of Howarth et al (1971) and Beggs & Howarth (1972a, 1972b) provided evidence of bell shaped velocity profiles for discrete movements. This was not the case. Howarth et al (1971) and Beggs & Howarth (1972a, 1972b) did not record or derive any individual movement data. Their approach curves were derived from aggregated movement data across several trials (see fig 3.0). This describes a phenomenon apparent when movements are considered collectively, and it is overly ambitious to conclude from this data that individual movements show a bell-shaped velocity curve.

Keele (1981) argues that this model of movement is incomplete as it does not take into account the influence of movement amplitude on terminal accuracy.

Distance to Impact and The Proportion of Time Left (Howarth & Beggs)

3.2.4 Impulse timing model

Schmidt et al (1979) proposed that rapid arm movements are effected by 'flinging' the limb toward a target via a neuromotor impulse delivered to the arm muscles which comes to rest in accordance with the situational dynamics. This impulse causes the muscles to exert a burst of force for the first half of the movement. Inherent variability in the force impulse was considered to directly contribute to endpoint spatial variability. Impulse variability was considered to originate from variability in the impulse's duration and or magnitude.

The standard deviation of the force is assumed to be proportional to the magnitude of the force, and the standard deviation of the impulse duration is assumed to be proportional to the impulse duration.

Thus, if more force is required to propel the limb a greater distance, then the endpoint variability will increase, and if more time is taken in propelling the limb more time variability will result. The subject's task is to optimise the time/force, tradeoff.

Giving:

$$MT = k\ (D/W) \quad \text{Equation 3.6}$$

Where: MT is movement time

k is a constant

D is amplitude of movement

W is the end point variability

This relationship was supported by their experimental findings. Subjects were asked to move a stylus, as accurately as possible, towards a target area for movements of fixed durations (140 - 200 ms). However, Meyer et al (1982) have shown that Schmidt et al's (1979) formulation was flawed, and should have yielded an equation of the type:

$$W = k\ D \quad \text{Equation 3.7}$$

Which should have made endpoint variability independent of time. Given, the inadequacies of Schmidt et al's (1979) analysis Meyer et al (1982) developed a revised impulse variability model.

Meyer et al (1982) made some similar assumptions to those of Schmidt et al. They assumed that a movement is generated by a force pulse whose magnitude and duration depend upon a force parameter and a time parameter. These two parameters are programmed so as to achieve the desired distance/time characteristics of the movement. These parameters are assumed to be inherently noisy, and their standard deviations are proportional to the magnitudes of the force and duration. They propose that the associated force-time curves produce symmetrical acceleration and deceleration phases. Figure 3.1 shows the predicted waveforms for movement displacement, velocity, and acceleration.

Meyer et al (1982) suggest that a sequence of overlapping miniature impulses, essentially ballistic in nature, might account for tasks where Fitts' Law applies. They argue that such a sequence will produce a single smooth movement trajectory rather than several discrete submovements. However, this has not always been found to be the case (see section 3.5.4)

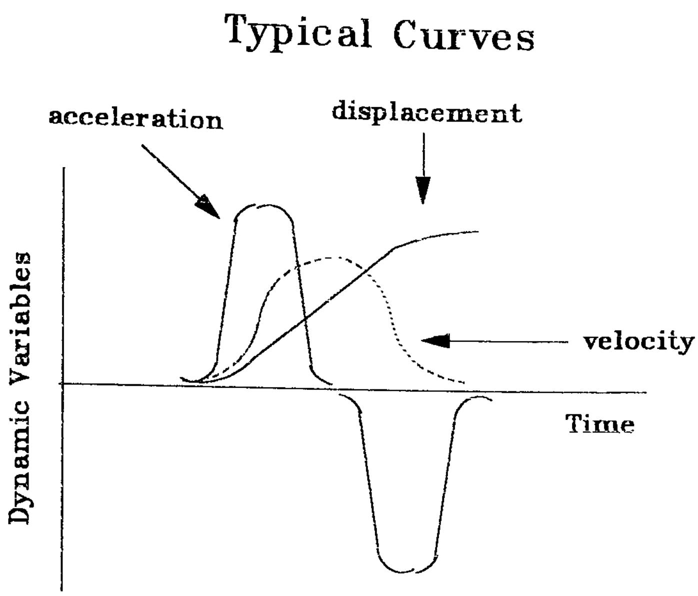


Figure 3.1

3.2.5 Stochastic optimal submovement model

Meyer, Abrams, Kornblum, Wright & Smith (1988) present a hybrid of the iterative corrections model and the impulse variability model. The model makes the following assumptions:

- Each movement consists of one or two submovements
- The existence of noise in the neuromotor system
- Submovement end points are normally distributed
- The minimisation of movement times
- The processing of information prior to movement

The model assumes that a movement will consist of one, or two submovements. These movements will be directed towards the target centre but the end points of such movements will be affected by neuromotor noise. As a consequence of such noise, submovement end points will be normally distributed about the target centre. In some cases the accuracy of the first submovement will be such so as to eliminate the need for a secondary submovement. However, if not sufficiently accurate with the first submovement a secondary submovement will be made on the basis of visual feedback. These assumptions are defended by the model proposers by reference to previous work.

The endpoint deviation is given by a rearrangement of equation 3.6 giving:

$$W = k\ (D/MT) \qquad \text{Equation 3.8}$$

Where: MT is movement time

k is a constant

D is amplitude of movement

W is the end point variability

The model assumes that the average velocities of submovements is such so as to optimise the total movement time. That is, the average velocity of the primary movement is that which will lead to a minimal completion time when the secondary movement has been made.

Information for use with the primary movement is assumed to occur prior to the movement's initiation. Secondary movements make use of information processed on the 'fly' with there being no 'dead' time between primary and secondary submovements. The total time is given by the addition of the two submovement durations.

These assumptions, and some accompanying mathematical approximations, predict that the total movement time will be approximately given by:

$$MT = a + b\ (D/W)^{0.5} \qquad \text{Equation 3.9}$$

Where: MT is movement time

a and b are constants

D is movement distance

W is target width

The model predicts the following effects of increasing D/W:

- square root increase in MT
- square root increase in first submovement duration
- square root increase in second submovement duration
- increase in endpoint variability of first submovement
- increase in relative number of secondary submovements
- increase in number of tertiary submovements

3.2.6 Two phase model

Woodworth (1899) was one of the first to investigate the distinction between slow and fast movements, made with and without vision (Sheridan, 1984). From his studies he concluded that there were two phases to the control of slow voluntary movements; an initial impulse phase followed by a period of current control. The amount of time spent on the current control phase increased with increased target difficulty. Further, he concluded that the accuracy of current control phase was dependent upon the presence of visual information but that the accuracy of the initial impulse phase was independent of vision.

The initial rapid impulse phase has often been regarded as a ballistic movement and the current control phase has been regarded as comprising of visually controlled movements (Woodworth, 1899; Welford, 1968; Paillard, 1980). Welford (1968) reformulated Fitts' Law so that it consisted of two mathematical terms corresponding to the two phases of movement. This distinction between ballistic and controlled movements is considered to be one of the most fundamental distinctions in motor theory (Sheridan, 1984).

Ballistic & controlled phases of movement have been identified with different electromyographic activity. Rapid movements are associated with distinct agonist and antagonist bursts of activity, whereas controlled movements are associated with the agonist and antagonist muscles 'continually playing against each other' (Stetson & Bouman, 1935). Similarly, slow movements have been associated with agonists and antagonist activity throughout, whereas rapid movements display an initial burst of the agonist followed by an antagonistic burst of activity, and then a final burst of agonist and antagonist activity coinciding with the movement's termination (Wacholder & Altenburger, 1926; Angel, 1974; Lestienne, 1979; Wadman et al, 1979). Edgerton, Hewitt, & Smith (1972) found that different types of muscle fibre were associated with slow and fast movements; they identified 'phasic' units as those being activated for fast movements and 'tonic' units which were activated for slow movements. However, Brooks (1979) urges caution when attempting to distinguish between slow and fast movements on the basis of their muscle activation, as such activity patterns can arise through the particular demands of the motor task.

The velocity profile associated with a ballistic movement is a symmetrical bell shape (Gan & Hoffman, 1981; Megaw, 1974; Vince, 1948; Crossman & Goodeve, 1963). Sidall, Holding, & Draper (1957) found that errors of extent, as opposed to errors of aim, were associated with ballistic movements, and that these errors of extent were due to target overshoots rather than undershoots. Some researchers have reported that movement times, for a ballistic phase, are proportional to the square root of the movement's amplitude (Gan & Hoffman, 1988; Jagacinski et al, 1980).

Ballistic movements are considered to be initiated as a whole, and to run their course without the possibility of modification (Welford, 1968; Poulton, 1974). However, there is evidence that fast acting feedback loops (of the order of 30ms - 80ms) permit rapid corrections to perturbed limb trajectories (Marsden, Merton, & Morton, 1972; Carlton, 1983), and thus, making the distinction of a true ballistic movement phase, a difficult one. A less problematic definition would consider ballistic movements to be characterized by rapid movements made independently of visual 'feedback' (Schmidt & white, 1972; Glencross, 1977; Brooks, 1979).

Studies attempting to determine the processing time for visual feedback, originating from a movement, have arrived at time intervals of between 260ms and 135ms. Typical values that have been found are: 190ms - 260ms (Keele & Posner, 1968), 135ms (Carlton, 1981b), 290ms (Beggs & Howarth, 1970), 200ms (Wallace & Newell, 1983), 165ms (Young & Zelaznik, 1992). These times also include the time required to act upon such feedback. Despite the evidence that movements of a duration greater than about 200ms can benefit from visually based corrections, Wallace & Newall (1983) showed that, for movements requiring a relatively low degree of precision, visually based corrections were not in evidence.

The distinction between ballistic and controlled movements is problematic (Meyer et al, 1988; Sheridan, 1979).

> *'The distinction made between controlled & ballistic movements is not as simple as often presented. There is some doubt about what constitutes a ballistic movement; also there is difficulty in that controlled movements may have both a ballistic & control phase.'*
>
> Sheridan (1979, p 181)

3.3 Reaction time

Studies of movement reaction time have largely been considered within the context of the motor program. It was argued that prior to a movements execution its motor program had to be assembled, and it was considered that reaction time, or movement latency, could be taken as an indication of the complexity of that motor program (Summers, 1989) with longer reaction times being associated with increasing movement complexity (Klapp, 1980). Glencross & Barrett (1989), in reviewing the reaction time paradigm, considered it as having been an unfruitful line of investigation. They attributed this to confusion over the experimental techniques employed, and the elusive nature of motor complexity.

In particular they indicate that researchers confounded their findings by a failure to isolate that phase of movement that was preprogrammed from the subsequent controlled phase. They also suggest that choice reaction time is a better index of complexity than simple reaction time. They indicate that choice reaction time should increase for movements requiring a high degree of precision to be programmed in advance. Summers (1989) concluded that reaction time studies do lend support to the concept of preprogramming for motor movements.

3.4 Errors

Schmidt (1982) distinguished between errors in the execution of the correct motor program and errors of motor program selection. Similarly, Glencross & Barrett (1989) distinguish between 'ballpark correct' and 'ballpark incorrect' responses. In other cases the term directional errors has been adopted to describe such movements (Megaw, 1972, Glenncross & Barret, 1989).

Directional errors have been defined as movements whose initial acceleration takes the cursor or limb away from the target (Angel & Higgins, 1969; Higgins & Angel, 1970), or towards an alternative target (Megaw, 1972).

There is evidence that the movement characteristics associated with ballpark incorrect responses are different to ballpark correct responses. Incorrect responses have been shown to have shorter movement durations (Angel & Higgins, 1969; Megaw, 1972), smaller movement amplitudes and peak accelerations (Megaw, 1972), and longer reaction times (Gibbs, 1965; Angle & Higgins, 1969). However Megaw (1972) reported error reaction times which were shorter than the corresponding reaction times of correct movements.

The time to respond to an incorrect movement, the Error Correction Reaction Time (ECRT), has been defined differently by different researchers. It has been taken as the duration of the incorrect movement (Angel & Higgins, 1969; Gibbs, 1965), as the time for the incorrect movement's acceleration to become negative from the onset of movement (Higgins & Angel, 1970), and as the period from the movement's onset to its peak acceleration (Megaw, 1972).

Megaw (1972) indicates the appropriateness of an ECRT measure depends on whether there is evidence in the erroneous movement's characteristics, which suggest that its deceleration phase represents the onset of the movement's correction, or just the 'normal' movement termination. Furthermore, there is no certainty that the onset of the erroneous movement coincides with the stimulus presentation for the ECRT.

Comparisons of the values of ECRT are confounded by the different operational definitions used by researchers, however, for a movement task, ECRT periods as small as 63.5ms have been found (Megaw, 1972). These corrections were assumed to be too quick for visually based amendment, and Higgins & Angel (1970) found ECRT (98ms) to be quicker than the time required for proprioceptive feedback based corrections (136ms). This suggests that the movements were corrected independently of visual or proprioceptive feedback. However, as mentioned earlier (section 3.2.6) there is evidence that fast acting feedback loops exist in the motor system. There is some evidence to suggest that corrective muscular activity occurs prior to the instigation of incorrect movements (Cooke & Diggles, 1984; Morris, 1981).

Higgins & Angel (1970) suggested a centrally based error correction mechanism by which subjects are able to monitor their own actions internally, and make comparisons between the actual motor commands and an internal reference value. A number of criticisms of this viewpoint have been raised (see Glenncross & Barrett, 1989). It has been suggested that directional movement errors can occur through incorrect response selection arising from a premature onset of movement (Megaw, 1972).

A more complete discussion of the theoretical models of directional error is beyond the scope of this thesis.

3.5 Kinematic Studies

A number of studies have attempted to investigate the nature of aimed responses with reference to their microstructure. In order to elicit the microstructure of such movements specific techniques have been developed. The tasks, measurement techniques, aims, and equipment employed by these studies have varied between the studies but useful comparisons amongst them can be made.

3.5.1 Aims

Most studies, considered within this section, have been concerned with the evaluation of movement theories with respect to their predictions regarding the nature of submovements. Some researchers have been concerned with establishing group differences in movement (Murrel & Entwisel, 1960; Hay, 1979). Others, such as Fleischer and Lange (1983) and Bullock and Harley (1972) have been concerned with the evaluation of suitable techniques for the measurement of human motion.

3.5.2 Tasks

Carlton (1980) investigated three types of task: a Fitt's reciprocal tapping task, a discrete aimed response (location to a single target), and a peg transfer task. Each task had two conditions which resulted in task IDs of 4.65 and 7.00. In each case the movement amplitude was 203mm, and the variation in task ID was achieved through the variation of target width.

In a second experiment the discrete task was repeated, but with a higher task ID of 8.74 (obtained from a movement amplitude of 640mm and target width of 3mm). The discrete task always demanded an initial right to left movement. The tapping task required the user to reciprocally tap two plates by moving between them as fast as possible. The subject was required achieve as many 'taps' as possible within a 15 second period. The peg location task was similar to the tapping task in that the subject was required to move a peg from one hole to another and then repeat the movement.

For all three tasks they found evidence of corrective movements in some of the trials, but not for every trial. It was also found that for the higher task ID, in the second experiment, that the number of trials exhibiting corrective movements increased.

Hay (1979), investigating childrens' hand movements, required subjects to point to one of three targets situated in front of them. The central target was directly in front, the remaining two were located 20^{o} either side of the central target.

Murrell & Entwisle (1960), investigating age differences in target acquisition, required subjects to indicate, by pointing to, the required one of four targets.

Langolf et al (1976) used a peg locating tapping task to investigate very small movements made under a microscope, and a more traditional Fitts' reciprocal tapping task for investigating larger movements. They found evidence of discrete movement corrections for the microscope task but not for the tapping task.

Crossman & Goodeve (1963) employed a Fitts' reciprocal tapping task using movement amplitudes of 8 and 16 inches, and target widths of 0.5 and 0.125 inches. Subjects were also asked to make very rapid responses to relatively large target areas.

In a second experiment Crossman & Goodeve employed a reciprocal angular wrist positioning task. Target widths of 2.5° and 10° and movement amplitudes of 20° and 80° were used. In both experiments submovements were associated with the more difficult tasks.

Megaw (1972), investigating directional errors of movement, required subjects to align a pointer, using a pivoted control lever, with one of five possible target 'neons'. The control lever and pointer were pivoted which resulted in the control movement producing pointer movements in the opposite direction. Each target neon was illuminated for 80ms as a target stimulus. He found that typically movements gave rise to a symmetrical acceleration phase followed by an initially smooth deceleration phase. However, after the peak value of deceleration an irregular acceleration profile was observed, with a series of accelerations and decelerations. Such accelerations would be associated with an asymmetrical velocity profile, with the later section of the initial deceleration phase, showing evidence of submovements.

Angel & Higgins (1969) required their subjects to position an oscilloscope cursor to either the left or right of centre using a joystick where the control display directional correspondence could be reversed. Higgins & Angel (1970) required subjects to move an oscilloscope cursor either to a target above the cursor, or below the cursor. The experimenters were able to reverse the correspondence between the joystick movement and the cursor movement. Asymmetric bell shaped velocities were obtained, with evidence of submovements during the deceleration phase.

Annett et al (1958) employed a peg location task, which used the same size pegs but varied the diameter of the holes. This gave task IDs of 4, 6, 8, and 10. Subjects were required to move eight pegs from their holders and locate them in a similar set of holders nearby. When all of the pegs had been successfully transferred, subjects then had to relocate them back into the original holders. It was found that movements consisted of two phases; an initial primary movement followed by a secondary movement superimposed upon the first.

Vince (1948) required subjects to reposition a spring loaded pointer between two lines 2.5 cm apart. The lines were drawn on a rotating drum and the subject controlled the pointer by means of an attached cord. The subjects were required to carry out this experiment at different rates (20, 40, 60, 100, 120, 200 strokes per minute, and at the fastest rate the subject could manage). The experiment was conducted with and without subject vision. Visually based movement corrections were evident for movement durations of 500ms or more.

Fleischer and Lange (1983) required subjects to move a pile of 'chips' from one point to another, and when the pile of chips was completely relocated the process began in reverse. The distance between stacks was 300mm and the stack width was 22mm. The stacking cylinders, upon which the chips were placed, were 17mm in diameter. It was found that the hand movements became more dispersed about the optimal line of travel, between the two locations, with increasing time.

Hay, Bard, Fleury & Teasdale (1991), investigated children's aimed movements, requiring different amplitudes and or directions, with or without vision. Subjects were required to position a lever, that was fixed by a universal joint to the floor, which rose to approximately chin height, to the various target locations.

Abend, Bizzi, & Morasso (1982) required subjects to manoeuvre rod in the horizontal plane so as to be aligned with one of six targets. The rod consisted of two linkages fixed to a joint in front of the user. The targets were red light emitting diodes of 4mm diameter and they were arranged at equal distances from the resting location. The movements of between 20cm and 40cm were made in darkness.

The resultant movement trajectories approximated to a straight line.

Jagacinski et al (1980) required subjects to acquire a target presented on an oscilloscope, by use of a spring loaded joystick. The joystick had two modes of operation, either as a position indicator, or as a rate controller. The cursor was a vertical bar, and the target two spatially separated vertical bars. The subject's task was to locate the cursor within the two vertical bars of the target which would appear either to the left or the right of the cursor. The selection of the target was indicated by remaining within the target region for a duration of 350 ms. Three target widths (separations between bars) and three movement amplitudes, gave rise to nine task I.D's. It was found that movements consisted of a number of movement phases(submovements), which exhibited large individual differences between subjects. Subjects were approximately one metre from the screen.

Meyer et al (1988) required users to rotate a handle so as to align a triangular cursor between two vertical bars. The cursor displacement was proportional to the angular displacement of the control. Target widths ranged from 1.61° to 6.34° (degrees of wrist rotation) and target distances from 15.81° to 39.52° (degrees of wrist rotation). This gave task IDs in the range 2.32 to 4.96. The user was required to locate the cursor on a fixation dot and hold it there for 800 ms. There then followed a 600ms period for which the fixation dot changed to a '+' (plus) sign. At the end of this foreperiod the subject heard 4 tones of 50ms duration, separated by pauses of 260ms. The final tone was of a higher frequency than the previous three, and served to prepare the user to begin the location task.

All targets appeared to the right of the fixation dot, and therefore required all wrist rotations to be (initially) in a clockwise direction. Subjects were approximately 43 cm from the screen. The findings were evaluated in the light of the Stochastic Optimized Submovement Model, and were taken to support the model. They found that the D/W ratio had a strong effect on movement time, but that the distance (D) had a greater effect than width (W).

3.5.3 Measurement techniques

Annett et al (1958), employed two techniques to measure the same peg transfer task, in order to allow comparisons between the two methods. The first technique employed electrical contacts at the peg holder and peg target sites. Thus, when an unobtrusive electrical current was passed through the subject, it was possible to determine when a peg had been grasped and removed from its holder, and when it had been located and released into the target. The second technique involved the use of film to record the hand movements made during the experimental task. This permitted the subject's movements to be recorded on film at a sampling rate of 20.8 ms. Due to the vast quantity of data generated, by the second method, only the last 40 task cycles (40 x 8 x 2 peg locations) were analysed. In both methods four submovements/phases were identified: grasp, movement loaded, position, movement unloaded.

Vince (1948) had a pointer scribe the subject's movements onto the surface of a rotating smoked drum. The pointer, via a cam, was linked to a handle, or cord, which could be moved by the subject.

Murrell & Entwisle (1960) used film techniques to record hand movements. A lamp, flickering at a rate of 10ms, was placed on the subject's hand which permitted velocities and accelerations to be calculated.

Crossman & Goodeve (1963), for their first experiment, attached a sliding potentiometer to the movement stylus, which, via electrical differentiation techniques, yielded the velocity of the stylus. In their second experiment the positional and velocity patterns associated with a subject's wrist rotation were determined by means of a potentiometer and tachometer.

Hay (1979) used a film technique to record hand position.

Abend et al (1982) measured hand location by means of a hand position transducer, which provided a spatial resolution of 1mm and a sampling rate of 10ms. Having determined the spatial and temporal coordinates they were able to calculate the associated hand velocities and accelerations.

Higgins & Angel (1970) and Angel & Higgins (1969) used a potentiometer and tachometer to record the position and velocity of a joystick's motion. Outputs from these transducers were recorded onto magnetic tape along with task details.

Megaw (1972), using a potentiometer and accelerometer, recorded control lever displacement, and filtered acceleration, onto paper using ultra violet light.

Langolf et al (1976) used two recording techniques. For the microscopic task, a piece of 1.1 mm diameter piano wire was attached to a handle which the subjects grasped. The movement of the wire was detected by means of a bar attached to the wire, which was also connected to a linear differential transformer, which in turn, produced a voltage proportional to the piano wire's displacement. These signal voltages were digitized using a 20ms sampling rate. For the Fitts reciprocal tapping task the spatial coordinates of the stylus were determined at 33ms intervals by means of cinematography.

Hey et al (1991) measured arm displacement by means of two potentiometers whose output was connected to a computer. The potentiometers were sampled at approximately 3ms intervals, with positional data being constructed through trigonometric analysis, and velocity and acceleration data being calculated by a cubic spline method.

Carlton (1980) also used a film technique to record and analyse subjects' movements. The film provided a sampling rate of 4.5 ms, with a spatial accuracy of +/- 0.2 mm. A

Vanguard Movement Analyser was used to determine the X and Y coordinates, velocities, and accelerations of the movements. This was achieved by differentiating the smoothed displacement values using the cubic spline method. The resultant graphical plots were used to identify the presence of submovements. These all occurred near the target. That is, after an initial movement.

Taylor & Birmingham (1948) required subjects to relocate a displaced oscilloscope cursor by means of a joystick. Subjects viewed the screen from a distance through stereoscopic binoculars. The subjects had to realign the cursor using the cross hairs within the binoculars. Three displacement amplitudes were used.

Fleischer and Lange (1983) using the ultrasonic detectors to monitor the subject's hand movements, during the moving of 'chips' from one stack to another, used a sampling rate of 20 ms with a positional accuracy of +/- 1 mm. They investigated the resultant movements by constructing movement traces, and then finding the 'best fit' straight line through the traces. By contrasting the earlier trials with the later trials they were able to demonstrate that the movements had become more dispersed with increasing time. They also constructed graphs which showed the duration of stay at a given location.

Jagacinski et al (1980) were able to determine the position and velocity of a joystick every 5ms, during trials on their one dimensional location task. Acceleration was determined by smoothing the velocity data using a least squares quadratic technique over its past 110 ms history, and then differentiating this smoothed velocity to obtain the acceleration. These resultant data were again smoothed. An algorithm was developed which determined the beginning and ending of a submovement's duration and movement accuracy.

The beginning of the first submovement was taken to indicate the end of the reaction time period, and the beginning of any subsequent movement was taken as the endpoint of any previous movement. Movements lasting less than 70ms or deemed counter-productive were not considered as submovements so as to exclude system noise.

Meyer et al (1988), used a film recording technique with a sampling rate of 1 ms and a spatial accuracy of +/- 0.05mm. Positional data was smoothed and differentiated in order to determine velocities and accelerations. Initially, the positional data were passed through a low pass 30 Hz filter so as to remove spurious electrical noise. The filtered data were then passed through a differentiating digital filter so as to give associated velocities and accelerations. Finally, these data were passed through a stringent 0-7 Hz low pass filter so as to remove unwanted hand tremor. A parsing algorithm was then applied to the differentiated filtered data so as to determine submovements according to the following criteria:

1) A primary movement was indicated by the velocity being greater than 4^{o}/s for over 20ms.

The end of a primary movement was indicated by:

i) A change in direction

ii) The acceleration going from a positive to a negative

iii) The rate of deceleration increasing

The end of the movement was indicated by the stop and hold criteria, which required the movement to be in the range +/- 12^{o}/s for at least 160ms. The overall time was measured from the start of the movement until the beginning of the 'stop and hold' period. If, during the period from the end of the first movement to the beginning of the 'stop and hold', the velocity was greater than 4^{o}/s for more than 60ms, and a distance of at least 1^{o} was moved by the handle, then a secondary movement was said to have occurred.

3.5.4 Evaluation of findings

The findings by these kinematic studies must be considered within their experimental contexts. They have employed tasks which have had a relatively poor ecological validity. Tasks have typically required subjects to make well practiced, highly constrained, and experimentally contrived movements, which have often been made in the presence of highly obtrusive instrumentation.

3.5.4.1 Evidence of submovements.

Most of the motor studies above found evidence of discrete submovements within a movement. However, submovements were not always present, but their likelihood of occurring increased with increasing target difficulty (Crossman & Goodeve, 1963; Carlton, 1980). Vince (1948) found submovements ceased to occur for movement rates of more than two movements a second. When corrections were evident they were noticeably slower than the preceding movement and occured near the target.

Langolf et al (1976) found evidence of discrete submovements during their microscopic task but not for their Fitts' reciprocal tapping task. When discrete corrections occured they were observed on average 200ms after the beginning of the first movement.

Symmetrical bell shaped velocity patterns were associated with rapid movements (Crossman & Goodeve, 1963; Vince, 1948), and for movements made without visual feedback (Abend et al, 1982). Asymmetrical bell shaped velocity patterns (due to prolonged deceleration phases) have been found (Crossman & Goodeve, 1963; Vince, 1948; Franklin et al, 1948; Hay et al, 1991) and have been associated with more difficult task conditions (Crossman & Goodeve, 1963) and slower movements (Vince, 1948).

Hay (1979), investigating children's movement, identified three types of movement; one very rapid movement producing a symmetrical bell shape velocity curve, a second type was similar to the first, but showed prolonged deceleration, and a third type which showed several accelerations, and/or slow speed. It was the second type which became more likely with increasing child age.

Murrell & Entwisle (1960) reported finding movements of a complex nature (see Figure 3.2). This 'typical' movement pattern reveals a continuous movement comprised of two submovements, which themselves contain trajectory corrections. Similarly, Megaw (1974) reported complex movements, which consisted of an initial symmetrical acceleration, followed by irregular accelerations and decelerations, after the peak value of deceleration in the initial phase of movement. Typically the irregular acceleration patterns occured after 350ms.

Jagacinski et al (1980), found that first submovements were slower and more accurate than subsequent submovements.

Movement Pattern
(Murrell & Entwisle)

Figure 3.2

Abend et al (1982) found that the movement trajectories approximated a straight line rather than a curve.

3.5.4.2 The effect of task conditions

The studies that have focused on the affects of task conditions on movement have investigated target size and movement amplitude.

3.5.4.2.1 Target width

Annett et al (1958), found that for reduced target width the task completion time increased. This increase was largely due to the increased time spent in locating the peg in the hole, and not in travelling to the peg. Similarly, for reduced target width, Carlton (1980) and Meyer (1988), found that the likelihood of a corrective movement increased, and so too did the overall movement time.

Meyer et al (1988) attributed the increased time, for reduced target width, to the effects of width upon subsequent submovements, and not the initial submovement. However, Langolf et al (1976) found that increasing target width, for the Fitts' reciprocal tapping task, slowed the whole movement down, rather than being associated with more time being spent near the target.

Carlton (1980) also found that for increasing task difficulty the nature of the resultant submovements altered. For very difficult tasks the submovements not only exhibited corrective movements, but the submovements appeared to have at least three phases of movement; an initial distance covering phase, an initial correction phase, and a final correction phase. Jagacinski et al (1980) reports the effect of target width as being complex, probably due to the nature of the ending conditions and device conditions.

3.5.4.2.2 Movement amplitude

Jagacinski et al (1980), found that increased movement amplitude resulted in slower, more accurate, first movements. They found the movement time of the first movement was proportional to the square of movement amplitude, but the duration of the second movement was independent of amplitude. Meyer et al (1988) found that increasing the movement amplitude increased the overall movement time, which was due to increases in both initial and subsequent submovements. Meyer et al (1988) also found that the likelihood of secondary submovements increased with increasing movement amplitude.

Meyer et al (1988) also found that increases in the movement amplitude had slightly greater effect on movement time than did a proportionally similar reduction in target width.

Frankin et al (1948) found that increasing the movement distance increased the peak acceleration and deceleration values, and the time to complete the movement.

3.5.5 Discussion of kinematic studies

The following discussion considers the applicability of the findings, from these studies of motor movement, to tasks employing a mouse device. It then discusses the analysis of submovements made in some of these studies. The discussion ends by considering the use made of filtering and automated parsing.

3.5.5.1 Task

The tasks associated with the above motor studies are very different to those normally associated with the use of the mouse. It is difficult to envisage an everyday task using a mouse in the way prescribed by a reciprocal Fitt's tapping task, or in an analogous manner to the peg transfer task. However, the discrete aiming task, such as that employed by Carlton (1980), has some similarities with the user's task of target acquisition using a mouse.

A stronger case for the dissimilarities between the way in which a mouse is likely to be used and the motor tasks is the highly repetitive, and unvarying, nature of the motor tasks. A number of task elements were held constant, such as direction, starting and ending positions, etc. Other task factors, such as visual search, choice of target, and additional task demands (often found in mouse tasks) were absent from the motor studies. Further, the subjects can be considered highly skilled due to their high degree of experience over the narrow range of movements demanded. Thus, even the discrete aiming tasks become less like mouse tasks on closer inspection.

Certain tasks, such as those requiring wrist rotations, have yielded more fine-grained kinematic data than other types of limb movements (Crossman & Goodeve, 1963). It has been suggested that this arises through the dynamics of the limb systems involved in effecting the movement.

Given the above considerations it is doubtful whether the findings of the motor studies considered above would be applicable to the mouse task. However, the techniques employed in certain motor studies could be of some use to the research programme.

3.5.5.2 Submovements

Carlton (1980) found that for difficult tasks the likelihood of three or more submovements increased. Additionally, some of the studies have found evidence of complex movements comprising several submovements (Murrell & Entwisle, 1960; Megaw, 1974). However, Carlton (1980) and Meyer et al (1988) typically found two submovements in their analysis. In both studies the subjects were highly practiced on the largely unvarying experimental tasks; suggesting that an analysis of submovements, for difficult tasks, or inexperienced task subjects, should consider the possibility of a number of submovements.

Jagacinski et al (1980) presented, for each submovement, information regarding the submovement's accuracy, duration, and start and stop times. Similarly, Megaw (1972) described a number of parameters which applied to each movement. Meyer et al (1988) and Carlton (1980) do not present such information. This information, and additional submovement performance measures, may provide useful data, with respect to the effect of task variables, and assist in the identification of individual differences in movement, and differences between submovements.

Some of the above studies (Jagacinski et al, 1980; Carlton, 1980; Meyer et al, 1988; Crossman & Goodeve, 1963) acknowledged the existence of periods of non movement during the completion of some trials. However, no detailed analysis was presented. Such periods of non movement may be important when investigating task effects, individual differences, and task experience. They should therefore also be included in a detailed analysis of submovements.

3.5.5.3 Filtering

Many of the motor studies discussed which have derived velocity and acceleration patterns for an associated movement have made extensive use of filtering. The filtering process has been justified by the need to remove the effects of hand tremor and electrical noise. The velocity and acceleration curves produced without such filtering would be 'noisy'. Similarly, where curve fitting techniques have also been employed, during the processing of these movement graphs, smoothing of the data will probably have taken place. In either case the effects on the data are those of a low pass filter. The issue is whether such noise effects arise as measurement artifacts, or actually reflect the subject's motor output. If the latter is true can they safely be ignored when analysing submovements?

Meyer et al (1988) identify hand tremor and electrical noise as the main sources of noise. However, an equally likely source of noise must be the measurement techniques used. Measurements of location and time cannot be error free. These errors will be reflected as deviations about the true velocity and acceleration values. These errors would produce the same effects as would relatively high frequency noise on the plots of velocity and acceleration, and could be largely removed through the use of filtering employing a low pass filter.

Filtering is also important in assisting an automated process of submovement analysis. An automated movement parser would need to identify the turning points in the velocity and acceleration curves. A noisy curve makes this identification more difficult.

3.5.5.4 Automated movement parsing

Many studies, concerned with eliciting the microstructure of movement, have commented on the large volume of data collected and their associated processing. Most studies that have not employed an automated movement parser have had to limit themselves to the analysis of a small proportion of their collected data. Meyer et al (1988) and Jagacinski et al (1980) made use of automated movement parsers. Automated parsing of a movement, into its submovements, has the disadvantage that in being limited by certain rules it will not identify occurrences where the application of such rules would be inappropriate. However, it has the advantage that the rules used for movement parsing are explicit, objective and available for scrutiny.

3.6 Conclusions

Kinematic based studies of motor movement have contributed to a better understanding of motor acts. However, as such studies have primarily sought to contribute to an understanding of movement with reference to the underlying control processes, such as determining the presence or absence of submovements, and they tell us very little about the submovement characteristics and how they relate to task conditions. This was echoed by Kelso, 'at present there is little information on the space time configuration of a movement, or how movement organization changes under various environmental conditions' (Kelso, p.12, 1982).

It should be noted that many of the tasks used for the basis of kinematic movement study bear little resemblance to task of moving a computer mouse. The influence of task factors on the observed movement patterns has been equivocal and cannot easily be generalized to the mouse task. However, kinematic studies have often identified complex movements, those containing submovements, whose complexity increases with increasing task difficulty. Accordingly it is expected that such movement complexity, and its relationship to task difficulty, will be observed for movements made with the mouse device. Furthermore, the criteria, used in some of previous kinematic studies to identify submovements, are able to serve as the basis for submovement identification within this thesis. Additionally, motor theory provides a theoretical basis for the subsequent movement analysis. It is proposed, within this thesis, that an understanding of device movement (derived from a kinematic based analysis) can be gained by considering these detailed movement characteristics (such as duration, accuracy, speed etc.) within the input selection task context, using mouse type devices. Although this research is not intended as an evaluation of certain theories of motor control, a number of movement theories are able to make predictions regarding the resultant device movement characteristics. These predictions can be used as part of the basis for determining how the device movements should be analysed. Further more, they provide a context for the discussion of device movements.

Previous kinematic studies have largely ignored differences in movement patterns arising from group and individual differences, despite there being evidence of large individual differences for such kinematic patterns (Jajacinski et al, 1980; Hay, 1979). Such differences would serve as the basis for suggestions in altering the input task so as to facilitate the user's actions. There is also need for an automated movement analysis process that will facilitate the processing of large quantities of movement data and provide an objective, reliable description of people's movements.

CHAPTER FOUR

DESCRIPTIONS OF PEOPLE'S POINTING BEHAVIOUR

This chapter proposes a descriptive framework for the behavioural aspects of indirect pointing. Two models of behaviour are proposed, one emphasising the temporal aspects of movement, and the other focusing on discrete submovements. Identifying submovements was facilitated by the analysis of the kinematic patterns associated with the movement, and by drawing an analogy, between language grammar and movement. Accordingly, the reader is also introduced to kinesology and syntactic parsing techniques. This analogy facilitates examination of movement data, based on different movement grammars. By analysing actual movement data, this chapter shows that the conclusions reached, regarding a movement, are influenced by the way we arbitrarily choose to define movement. In particular, it shows that the movement criteria used by previous researchers may be insufficient to capture some important aspects of the movement.

4.0 Introduction

> *'Although the imagination is a marvellous human faculty, a thorough understanding of movement events must rest on firmer foundations.'*

Kelley (1971, p 3)

The observation of movement may be considered as comprising two activities; the recording of the activity, and the subsequent analysis applied to those measurements. In deciding what information to record one must first consider the requirements of the analysis to which the movement data will be subjected (Mitchelson, 1975). This chapter is concerned with determining the analytical requirements demanded by the research programme, and the following chapter deals with the issues of recording movement.

> *'If we are going to understand movement the first thing we have to do is describe it.'*

Smyth & Wing (1984, p. 2)

Attempts to classify an action into component parts are to some extent arbitrary processes (Annett et al, 1958). More generally, for a finite quantity of data, any one of an infinite number of possible descriptions could be constructed. As Medawar has indicated:

> *'In all sensation we pick and choose, interpret, seek and impose order, and devise and test hypotheses about what we witness.'*

Medawar (1967, P. 133)

However, for a particular behaviour, some descriptions are considered to be better than others. Annett et al (1958), has suggested that, for human movement, the partition of a movement cycle into component parts should reflect the corresponding changes in the control of the movement, or be made so as to facilitate the measurement process. These criteria are shared here (see section 3.0), but the analysis of device, and or pointer movement, should also contribute to the building of a 'Rich Picture' (Checkland, 1976) of the complex movement behaviour. One discipline of particular relevance to the description of movement is that of kinesiology (see section 3.0).

Kinesiology has been defined as the study of motion, which is characterized by the movement of human beings, and those other objects, which are influenced directly by humans (Kelley, 1971). It is primarily concerned with the recording and description of the motion itself, rather than with the control processes organizing those movements. However, such kinematic descriptions may inform the types of control processes conjectured to underlie the motion (Kelso, 1982). The following sections focus on temporal and kinematic descriptions of movement.

4.1 Temporal analysis of device movement

The singularly most used temporal characteristic of device movement has been task completion time (see section 2.4.1.1). How such a measurement is made, or partitioned, has been shown to affect the conclusions reached (Annett et al, 1958).

In Chapter Two, task completion time was partitioned into three stages:

1) Reaction Time
2) Movement Time
3) Time on Target

The importance of reaction time in motor movement has already been discussed in Chapter Three. However, the remaining two stages are considered to be inadequate to usefully describe device movement. It is believed that these stages reflect, what Annett (1958) has termed, 'taking measurements at convenient points' rather than meaningful units of movement which reflect a change in control.

The Movement Time stage fails to distinguish between what are believed to be different control phases of movement (see Chapter Three). Similarly, Kinematic studies of human movement have suggested that the movement stage can contain submovements, and or, movement corrections (see Chapter Three). Further, this research has found many instances of periods of non movement during the above movement stage (Chapter Seven).

The Time on Target fails to take into account the possibility that the device may be moving, or stationary, within the target, and might leave the target area to reenter at a later time. Evidence for both of these movement characteristics were found in this research programme (see Chapter Seven).

A more appropriate temporal model of device movement is proposed below.

1) Reaction Time

2) Movement Time

   2a) Time spent in movement

   2b) Time spent in non movement

3) Time on Target

   3a) Time spent in movement

   3b) Time spent in non movement

However, even this 'fine grained' temporal model is subject to some of the previous criticisms, and an analysis based on the submovements themselves would be more useful (see Chapters Two & Three for a more detailed discussion).

4.2 Kinematic Description

The first considerations in motion description are space and time. The kinematic approach is one where the motion is described by reference to its observed space and time characteristics (Kelley, 1971), as opposed to the kinetic approach which is additionally concerned with the causes of motion. However, useful descriptions of movement have been obtained with reference to time (Chapters Two and Three), or space, alone. Analyses based upon spatial movement traces have produced insights into hand positioning tasks (Fleischer & Becker, 1986; Fleischer & Lange; 1983). Kinematic descriptions on the other hand include both spatial and temporal information.

4.2.1 Movement Parameters

Given a moving object, whose location with respect to time is known, it is often possible to derive additional useful movement parameters. These derivatives are listed in Table 4.0 below.

(Position = x)

<u>Parameter</u>		<u>derivative</u>
Velocity	=	dx/dt
Acceleration	=	d^2x/dt^2
Jerk	=	d^3x/dt^3

Table 4.0 Movement Parameters

The 'dx/dt' term is a mathematical symbol which denotes the differentiation of the variable x with respect to the variable t. The application of the differentiation process produces the derivative of x with respect to t. When x denotes the position of an object, and t the associated time, the derivative will give the velocity of the object. That is, the rate of change in position with respect to time.

The term, 'd^2x/dt^2', denotes the second derivative of x with respect to t, and given that x is position, and t is time, then the second derivative provides the object's associated acceleration.

Similarly, the term, 'd^3x/dx^3', denotes the third derivative of x with respect to t, and is referred to as 'jerk', the rate of change of acceleration. Figure 4.0 illustrates these movement parameters for an arbitrary data set.

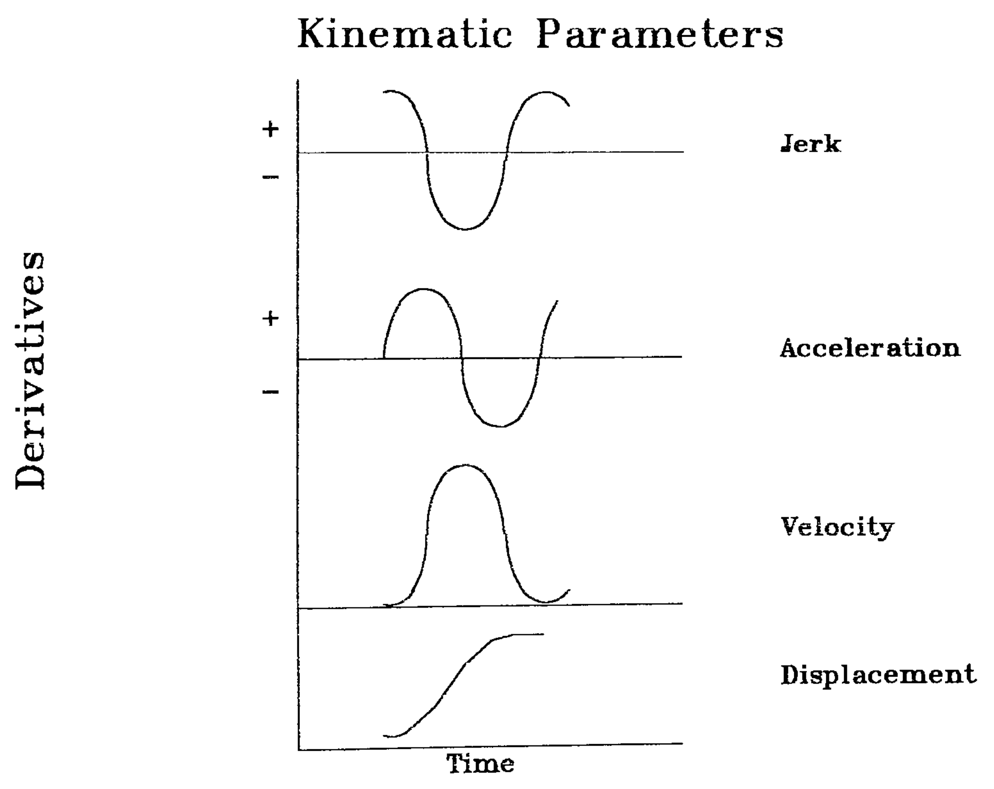


Figure 4.0

The derivative of a mathematical function indicates the rate of change in one variable with respect to another. In some instances the rate of change will be positive, in others negative, and in some cases zero. Given that one variable is changing with respect to another, those instances where no change is occurring have a special practical significance. These are known as the 'turning points' of a function and occur when dx/dt = 0. Turning points are of three types:

1) Maximums

2) Minimums

3) Points of inflection

These turning points are distinguished by considering the sign (+/-) of the rate of change in the instances immediately before and after the zero point of the rate of change. Figure 4.1 illustrates these distinctions between turning points and Table 4.1 lists the distinguishing criteria.

Turning Point Type	Sign of Rate of Change Before	At	After
Maximum	Positive	Zero	Negative
Minimum	Negative	Zero	Positive
Inflection	Positive	Zero	Positive
Inflection	Negative	Zero	Negative

Table 4.1 Turning Points

Given that x is a function of t, then the names given to the turning points are indicative of the relative values of x at a particular value of t. That is, the 'local' peak in x occurs at a maximum turning point, and a 'local' trough in x occurs at a minimum turning point. The term local is used to indicate that there may exist an absolute value of x that is greater than, or less than, those values of x associated with the turning points.

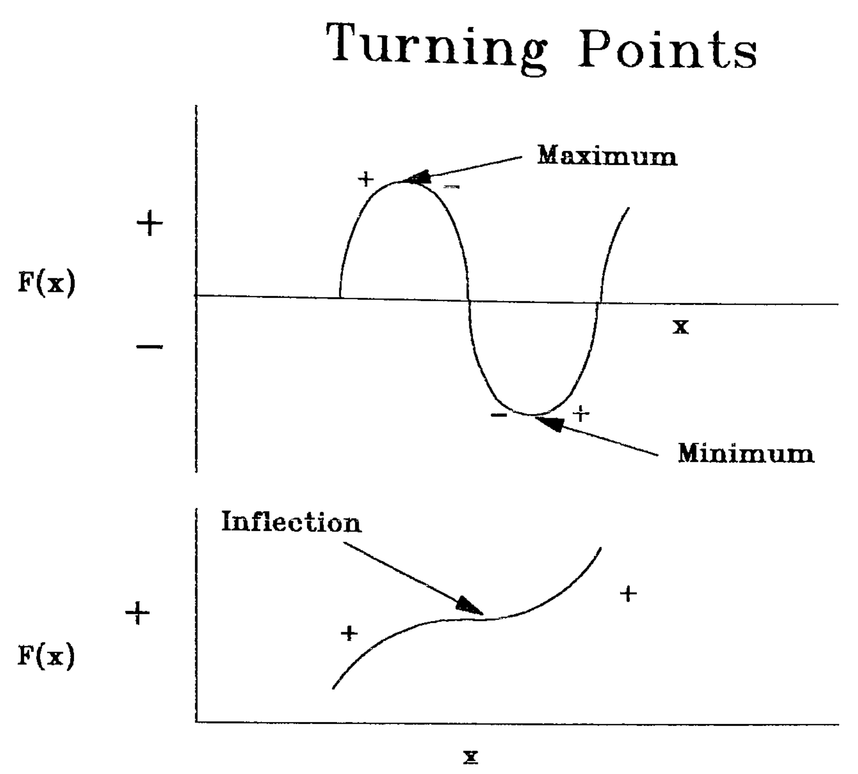


Figure 4.1

4.2.2 Determination of movement parameters

Given a set of data points, linking an object's spatial location to particular instances in time, how can the above movement parameters be determined? Broadly speaking, there are two approaches to solving the above problem, both of which provide several alternative numerical methods of solution.

The first group of solutions require that the data points have been collected at equal intervals of time. Given that this is the case, such methods as the Newton Forward, Stirling and Newton Backward difference formulae are applicable for the calculation the other movement parameters (Burden et al, 1981; Lanczos, 1967; Scheid, 1968).

Alternatively, if the data have not been sampled at equal intervals in time then an equation (linear or polynomial) describing the data points is determined which in turn is algebraically differentiated with respect to time.

Numerical time values are then substituted into the derived equation which, when solved, provides values of the derivative. For example, consider the data points shown in Figure 4.2. The equation, 'of best fit' is:

$$\boxed{X = t^2.}$$

Differentiating X with respect to t yields:

$$\boxed{dx/dt = 2t}$$

Thus, when t = 3, dx/dt = 2x3 = 6.

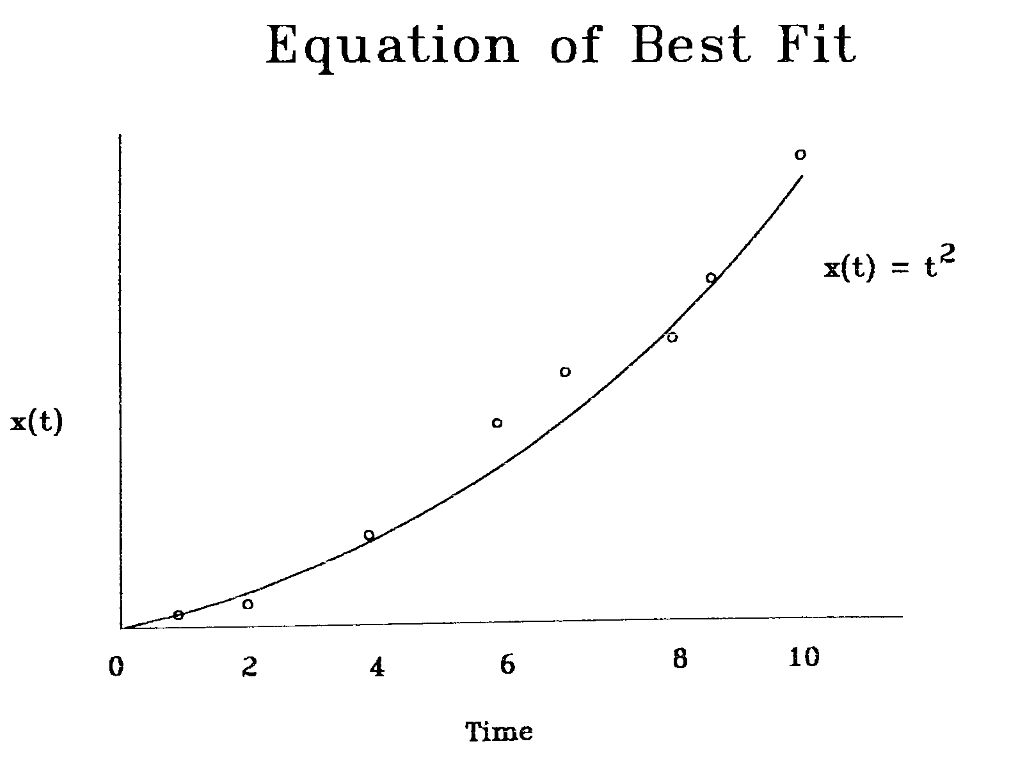


Figure 4.2

There are many methods for calculating the line of best fit through a set of data points (Lanczos, 1967). Two techniques which have been used in human motion studies are the Cubic Spline (Morasso et al, 1983; Carlton, 1980 & 1981; Whiting & Zernicke, 1982; Hay et al, 1991;), and the Quadratic Least Squares (Jagacinski et al, 1980; Abend et al, 1982) methods. The quadratic least squares method was preferred to the cubic spline method, following the inspection of some typical movement data, mainly due to the latter's ease of application. The algorithm used was derived from that proposed by Burden, Faires, & Reynolds (1981), and is listed in Appendix A.

In order to determine the overall velocity of the device and/or pointer movement the x and y directional component velocities were obtained separately and then combined by vector addition (Spiegel, 1974) to give the resultant velocity (see Appendix A). The acceleration of the device was obtained by redifferentiation of the velocity pattern rather than being derived directly from the positional data (Lanczos, 1967; Smith, 1975). Having calculated the movement parameters the next task was to provide an analytical framework for their description. Task details of the important kinematic studies can be found in Chapter Three.

4.3 Kinematic Analysis of Human Movement

Kinematic based movement descriptions have largely utilized the velocity and acceleration patterns derived from those movements. Researchers have commented upon the shape of these curves with respect to other aspects of the movement and task (see Chapter Three). Other studies have characterised the movements by use of a more detailed systematic analysis; in studies of handwriting (Morasso et al, 1983; Hulstijn & Galen, 1983; Galen & Teulings, 1983); investigations into issues of motor control (Carlton, 1980; Meyer et al, 1988; Carlton, 1981; Langolf et al, 1976; Jagacinski et al, 1980; Megaw, 1972); and movement in children (Hay et al, 1991; Hay, 1979).

In the above studies the partition of the movement sequences, based on kinematic data, has been influenced, in part, by the task design. However, in most cases, some commonality amongst the different partition criteria can be observed. Although not explicitly stated, these descriptions of the movement have rested on the assumptions of Newtonian mechanics; in particular, Newton's Second Law which asserts that a body will continue in a state of rest, or uniform motion, unless a force is acting upon the body (Whelan & Hodgson, 1978). We find that most of the partitioning criteria are concerned with identifying periods of acceleration, deceleration, and changes in movement direction. Variations in the accelerometric curve correspond to variations in motor and brake torque and thus provide a sensitive index of the organization of movement (Bouisset & Lestienne, 1974).

Those studies which have provided a detailed systematic analysis of movement, based on kinematic data, have primarily sought to determine the presence of movement corrections,(Carlton, 1980 & 1981) or submovements (Jagacinski et al, 1980; Meyer et al, 1988). Although the authors differ in their labelling of these movement events, corrections and submovements denote the same events within their studies. The criteria to determine the presence of a submovement were similar in each case (Meyer et al, 1988) and are summarized below (also see Figure 4.3):

i) shift from deceleration to acceleration (new movement)

ii) an increase in deceleration following the local minimum (movement correction)

iii) changes in movement direction

Jagacinski et al (1980) omitted criterion (ii) in their identification of the onset of a submovement. The primary submovement, in each case, was taken as the first transition from non movement to movement. The change in direction (iii) was appropriate, due to the unidimensional nature of the movement task and would not prove useful for two dimensional movements. The above criteria whilst useful in partitioning the movement stage into phases of movement is considered inadequate for the following reasons:

- The case of constant, or near constant, velocity
- The case of irregular acceleration patterns
- The case of non movement

These are illustrated in Figure 4.3. Phases of Constant, or near constant velocity, are in practice difficult to determine. Their detection will depend upon the resolution of the measurements taken and the nature of movement's velocity profile. Edwards (1965) failed to find phases of uniform velocity for limb movements. Similarly, visual inspection of a large random sample (over 1000 trials) of movement patterns, within this research programme, failed to reveal any periods of constant velocity.

Earlier studies have taken irregular deceleration patterns as indicators of movement corrections. However, they have only considered the final part of deceleration waveform for such irregularities. Such irregular accelerations could occur in any one, or more, of the accelerative quadrants (See Figure 4.10). A visual inspection of a number of movement curves (see above) revealed that such acceleration irregularities occurred in parts of the acceleration curve other than the final decelerative phase.

In numerous instances, a visual inspection of the movement curve revealed periods of non movement.

Considering the above, the following taxonomy of movement phases was proposed:

1) Acceleration phase

2) Deceleration phase

3) Irregular acceleration

   3a) Irregular rising acceleration

   3b) Irregular decending acceleration

4) Irregular deceleration

   4a) Irregular rising deceleration

   4b) Irregular decending deceleration

5) Non movement phase

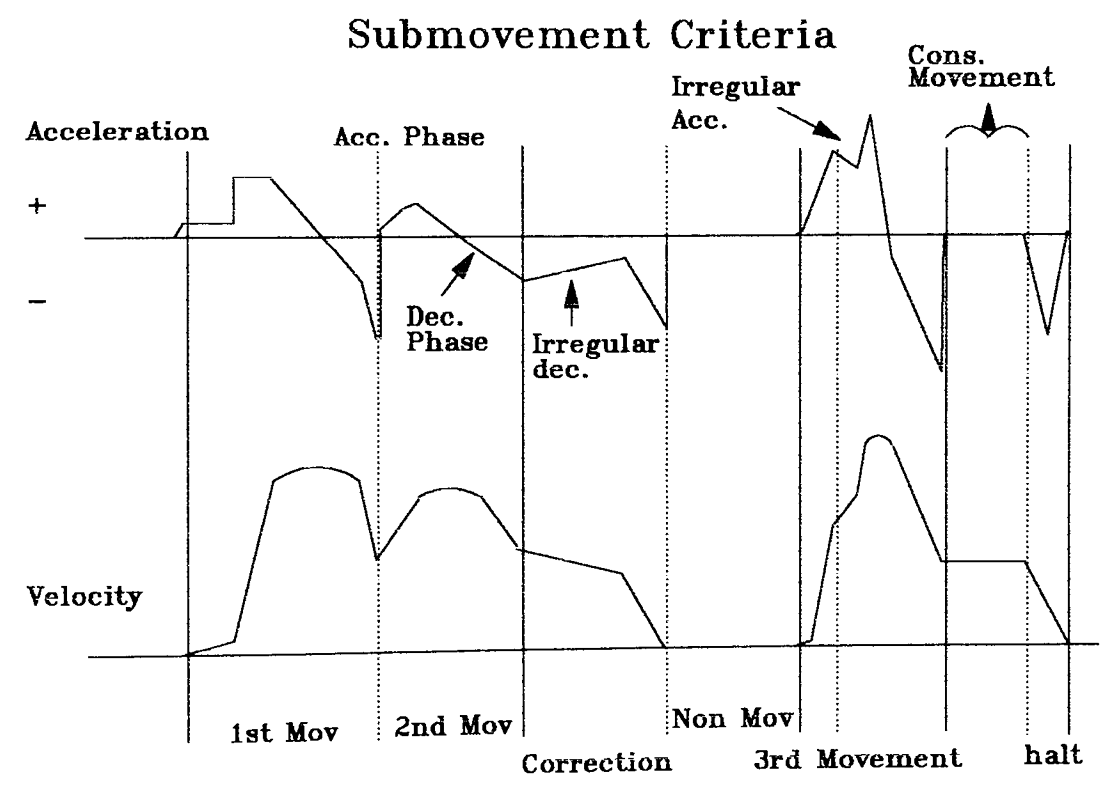


Figure 4.3

These movement phases are determined by considering the turning points associated with the velocity and acceleration movement patterns (discussed shortly). Having identified the phases of movement the next task was one of partitioning the movement stage into discrete phases of movement.

Previously, it was indicated that different movement descriptions could lead to different conclusions being reached from the same movement data. Bearing this in mind, and the need for an automated analysis of movement data (see Chapter 3), a description of movement analogous to linguistic grammar has been adopted within this thesis. This facilitates both the discussion and analysis of data using alternative movement descriptions (grammars), and an automated analysis of movement (syntactic parser). In order to appreciate this analogy it will be necessary to introduce the reader to a popular grammatical notation.

However, before this discussion proceeds, it would be useful to draw a distinction between the temporal model and the model based on discrete movements. The partitioned elements within the temporal model were referred to as 'stages' which were comprised of ordered intervals of time. Within the discrete movement model the partitions correspond to ordered submovements which in turn will be comprised of movement phases.

4.4 Movement and Language: An Analogy

The process of describing movement by partitioning movement elements is held to be analogous with the syntactic description of a sentence within linguistics. Meyer et al (1988) hinted at such an analogy when they referred to their computer based analysis as 'movement parsing'. However, this was the limit of their analogy, and it is used to a greater extent within this research programme.

A syntactical analysis is one concerned with the organization of meaningful elements within a sentence (Atkinson et al, 1982). Here, the sentence becomes the experimental trial, phrases become discrete movements, words become movement phases, and letters become turning points. Before presenting the movement partitioning criteria it is useful to introduce some concepts associated with the representation of grammar.

The most common way of describing the syntax of a language is that of the Backus-Naur Form (Naur, 1960), BNF, (Bartle, 1985; Brown, 1979). BNF consists of a set of production rules, or rewrite rules, an example of which is given below.

<Sentence> ::= <Verb Phrase> <Noun Phrase>

<Noun> ::= [cat | dog| house|]

The productions comprise two parts, the left and right hand sides, separated by the ::= symbol. The left hand side is termed the pattern, and the right hand side the expansion. Patterns and expansions consist of strings of symbols which are either terminal symbols (surrounded by [ ]) or non terminal symbols (surrounded by < >). The ::= symbol implies that any non terminal symbol in the pattern may be replaced with the expansion symbols. The * operator after a symbol implies that symbol may be repeated any number of times. The ? operator after a symbol implies that the symbol's presence is optional. The representation of a hypothetical noun phrase using BNF is shown below.

<Noun Phrase> ::= <determiner>?
<adjective> <noun>

<determiner> ::= [the]

<adjective> ::= [red | fast | cold]

<noun> ::= [cat | dog | car]

Often, rather than using the BNF notation to represent grammatical rules, a graphical representation, known as a 'transition network', is employed. A transition network is a structure which conveniently captures a variety of facts about syntax (Winston, 1979). A transitional network representing the above noun phrase is shown in Figure 4.4. This shows a Finite State Transition Network (FSTN) representation of the above grammatical rule. The nodes are known as states and the arcs determine the permissible routes between nodes. To parse the statement you begin at the start node, and take each statement element in turn, and attempt to transverse the network by following permissible arcs. If it is not possible to arrive at the end state then the parse is said to be incomplete, and the statement is taken to be ungrammatical.

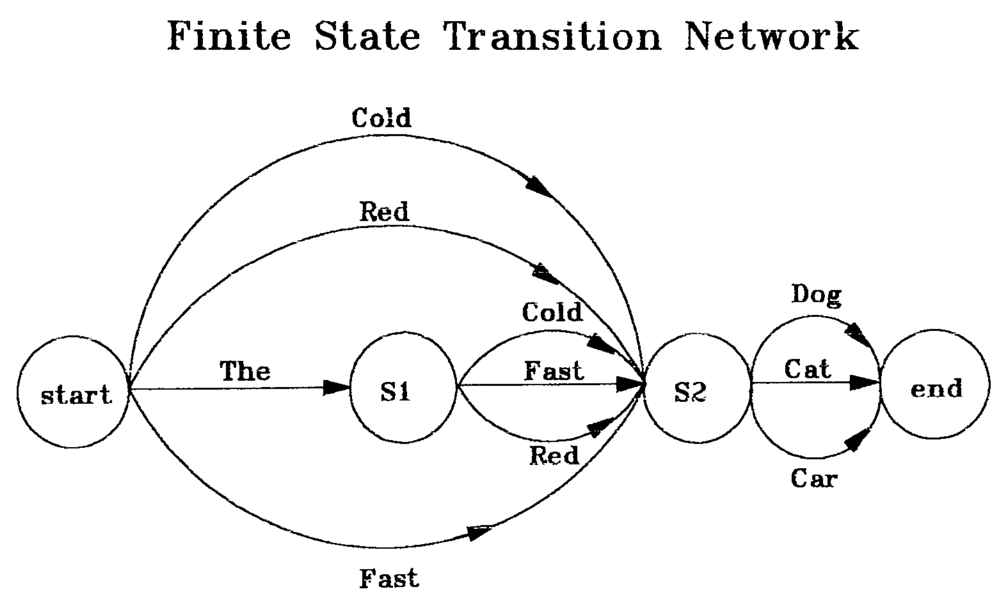


Figure 4.4

Chomsky (1957) distinguished between four complexities of grammar (0 through to 3). Different complexities of grammar require different types of transition network. FSTNs are only able to deal with grammars of type 3 complexity. Recursive Transition Network (RTN), unlike FSTNs are ones in which the arcs are not restricted to terminals, and are able to handle grammars of types 2 & 3. The above noun phrase is shown using an RTN in Figure 4.5. In transversing the network it may be necessary to 'jump' to another RTN, defined by the label indicated by the arc, and when this RTN has been transversed to return to the previous RTN and continue with the parse.

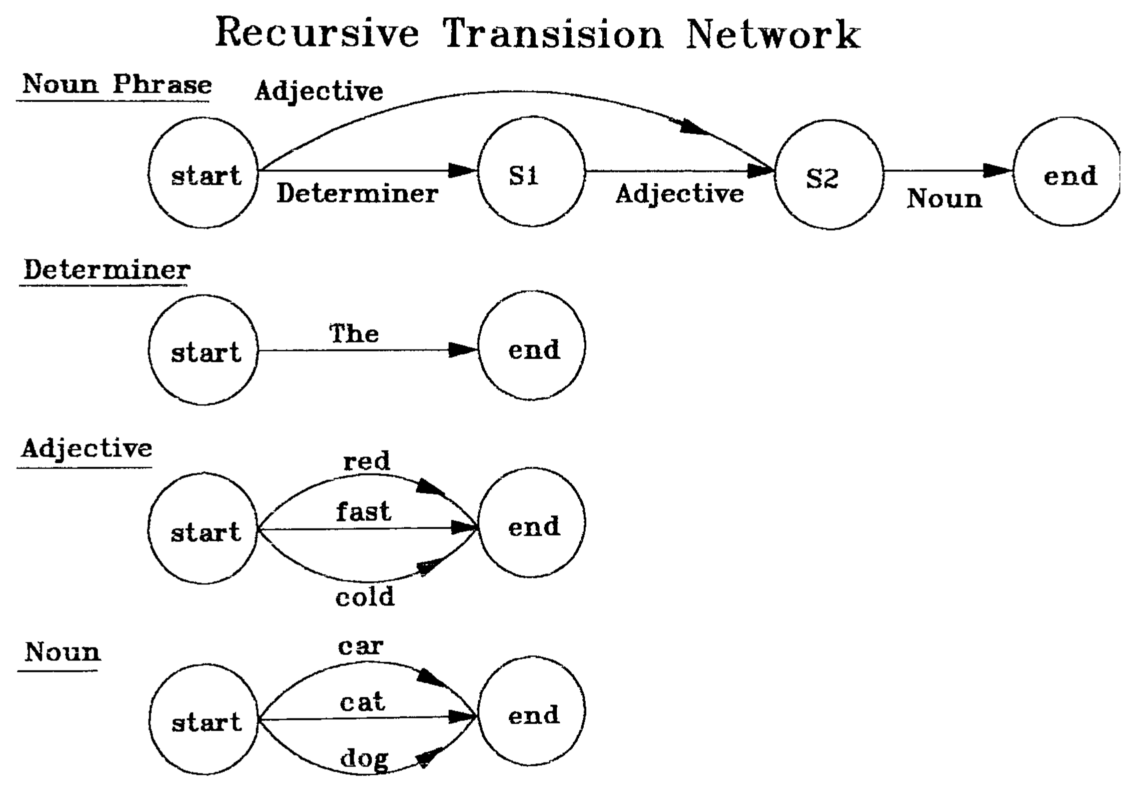


Figure 4.5

A further elaboration of transition networks are Augmented Transition Networks (Woods, 1970). They are recursive like RTNs but they have additional features, such as; the transitions being dependent upon future or past statement elements, registers to facilitate back tracking, and the reassignment of statement elements , allowing them to describe the most complex of grammars (level 0).

4.5 Towards a syntax of movement

Consider a 'simple' movement, one possessing a symmetrical bell shaped velocity curve, in terms of the previously suggested movement phases. It would be described by a syntactic rule as represented in Figure 4.6. There would be a period of non movement (start delay) followed by a period of acceleration, then a period of deceleration, and finally a period of non movement.

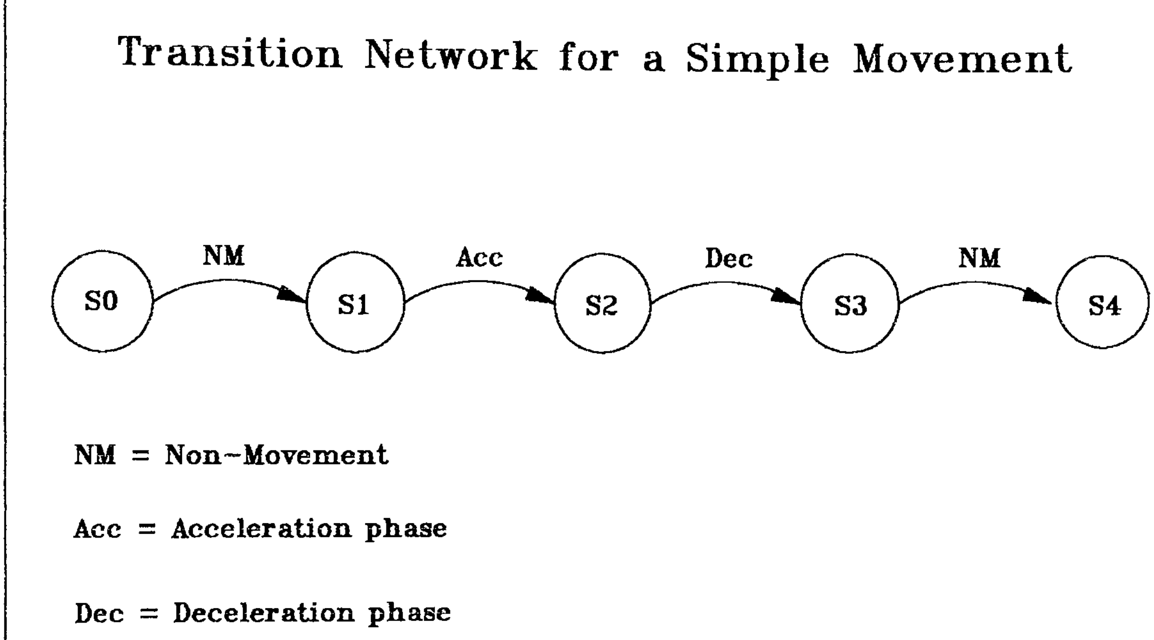
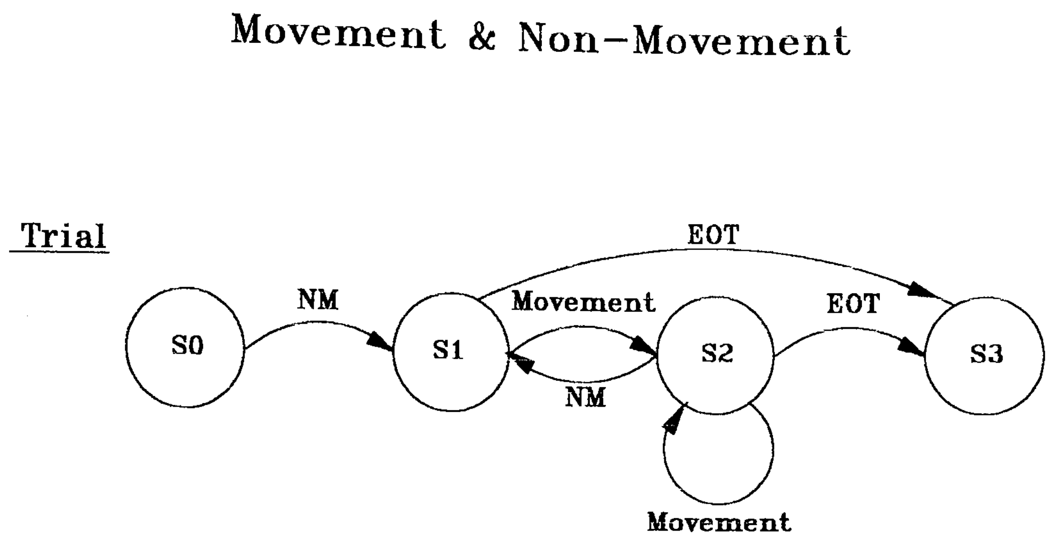


Figure 4.6



EOT = End of Trial : NM = Non-Movement

Figure 4.7

More generally, each correct movement trial would consist of a non-movement, followed by a movement which may be followed by either a non-movement or another movement phase. This is captured in Figure 4.7. However, it is the classification of submovements which is more problematic.

Consider the case of a 'simple' movement whose deceleration phase contained a corrective, or irregular deceleration pattern (see Figure 4.8). Do we consider this as one 'simple' submovement with a correction, or as two submovements with one being an acceleration phase and the other being an irregular deceleration? Certainly, we need to be aware that a correction has been made. Furthermore, if the irregular phase of movement is uniquely identified, we can determine its associated movement characteristics (e.g. accuracy, duration etc.).

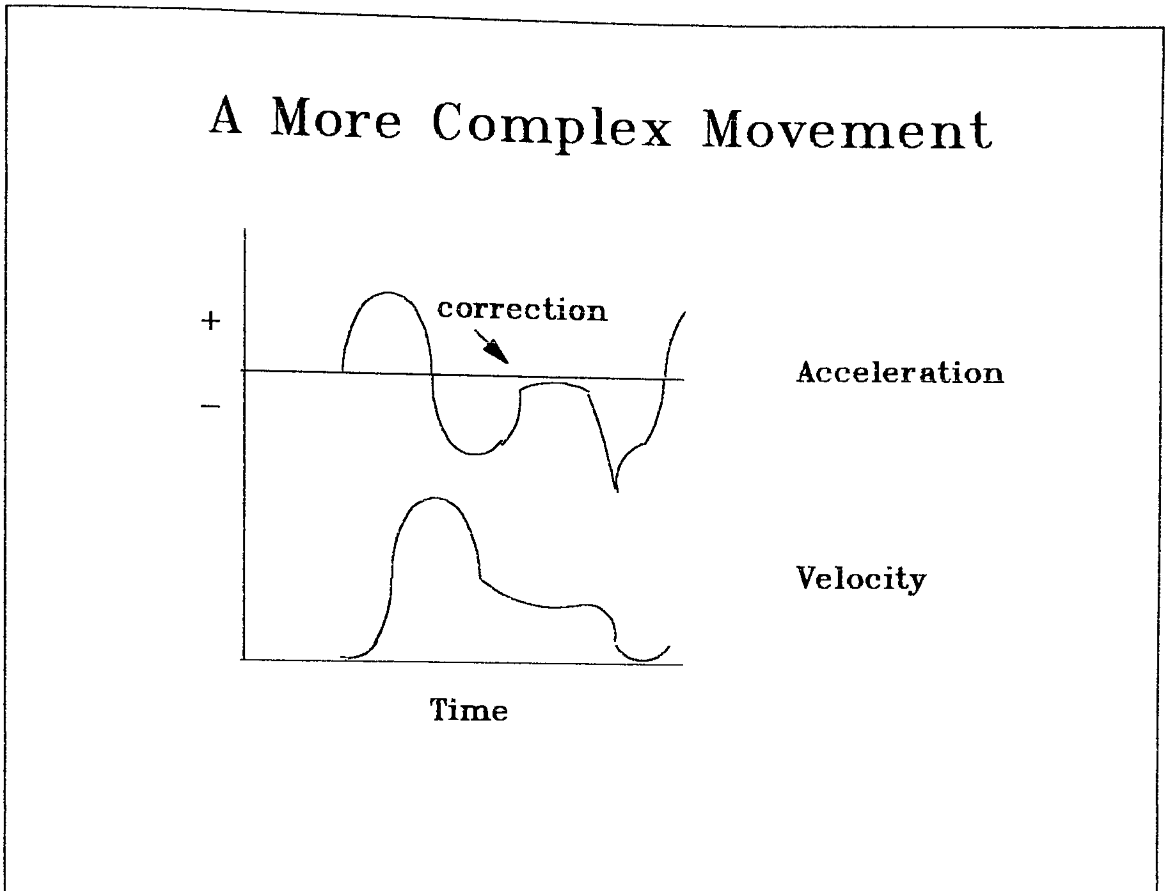
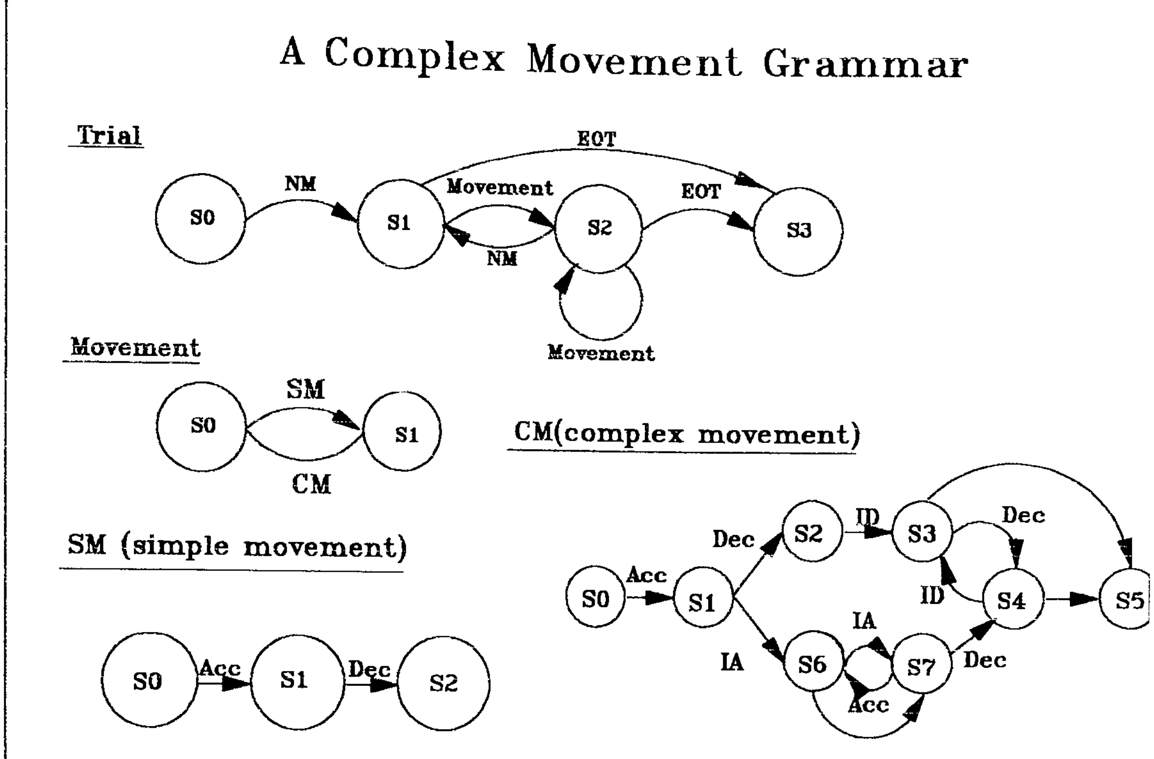


Figure 4.8

This suggests that a distinction be drawn between simple movements (those consisting of an acceleration followed by deceleration), complex movements (those containing corrections), and non movements. Complex movements can be further subdivided into irregular accelerations, irregular decelerations, which in turn can be further subdivided into those occurring before and after the associated turning point. Figure 4.9 illustrates a movement grammar based on these distinctions.



EOT = End of Trail : IA = Irregular Acceleration
ID = Irregular Deceleration
Figure 4.9

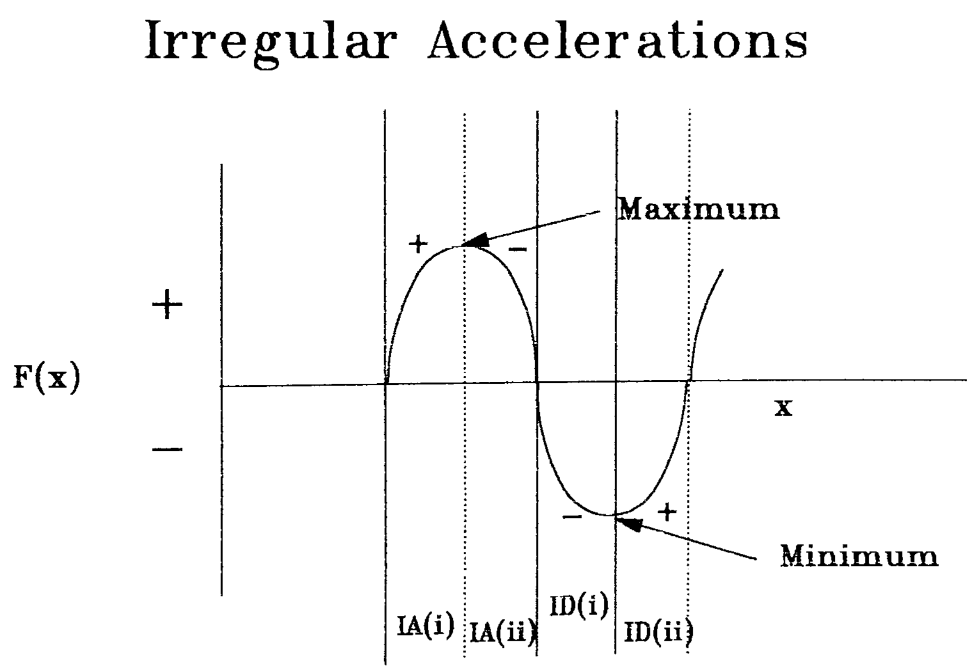
4.5.1 Details of the movement parser

In actuality two movement parsers were developed for the analysis of movement. The first may be considered as a 'lexical' parser which determined the 'critical points' from the kinematic curves. These points correspond to the 'atomic' movement phases (as indicated above, Figure 4.9) and are the 'lowest levels' of submovement that can be distinguished. Having identified the low-level phases of movement, the higher level descriptions of movement are then constructed by the second parser.

The second parser, the 'movement parser', permitted an analysis of discrete movements based on alternative movement grammars which facilitated movement descriptions at different levels of analysis. For example, we could be interested in the presence, or absence, of corrective movement phases, without the need to consider the different types of corrective movement. In selecting an appropriate syntactical rule this could be achieved without too much difficulty.

4.5.1.1 Lexical parser

The input stream to the lexical parser was the sampled velocity and acceleration values associated with a movement trial. The parser required two passes through this input stream. The first pass identified three movement phases; acceleration, deceleration, and non-movement. A second pass was made to determine if any irregular acceleration, or deceleration, phases occurred. This was achieved by determining the local maximum and minimum turning points associated with either a deceleration or acceleration phase, and comparing the acceleration data points for that particular acceleration/deceleration phase with the movement turning points (See Figure 4.10). Thus, the kinematic input stream is converted to an output stream of atomic movement phases.



The type of irregular phase is determined by its relative position within the sequence of the movment and its associated maximum and minimum turning points

Figure 4.10

Due to this lexical definition, the role of accelerative and decelerative phases depends upon the context in which they are encountered. For example an accelerative phase may be taken as the start of a new movement, or may follow on from an irregular accelerative phase. Only by considering the previous, or future, movement elements can this ambiguity be resolved. That is, the movement grammar requires an augmented transition network in order to parse these movements. However, if irregular, the accelerative phases are removed from the lexicon, this requirement may be lifted. The computer program is listed in Appendix A.

4.5.1.2 Movement Parser

The movement parser converts the input stream of atomic movement phases into an output stream of discrete submovements. The criteria for movement partition is represented by the syntactical rules used to parse the input stream.

Lastly, a movement analyser calculates various parameters of a submovement, for example its duration and distance covered which is then attached to the parsed movement output. These parameters and their calculation are discussed later within this chapter.

4.5.2 An example

I would like to illustrate how different descriptions of a movement can influence the conclusions drawn from the same movement data. I should like to consider some alternative descriptions of the movement data collected from an experienced subject during an experiment (described in a later chapter). The subject's task was to locate a screen pointer, using a computer mouse, within a circular target area and then to indicate task completion by pressing a mouse button.

The trial completion time was 1.021s which is not very informative in itself. Temporal partitioning of the movement time gave a Reaction Time of 0.392s, a Movement Time of 0.471s, and finally, a Selection Time 0.158s.

Further temporal partitioning revealed the Movement Time had no periods of non-movement and that the time on target consisted solely of a non-movement. This might suggest that the trial consisted of a single accurate movement preceded by a reaction time interval, and followed by a small period of time on target.

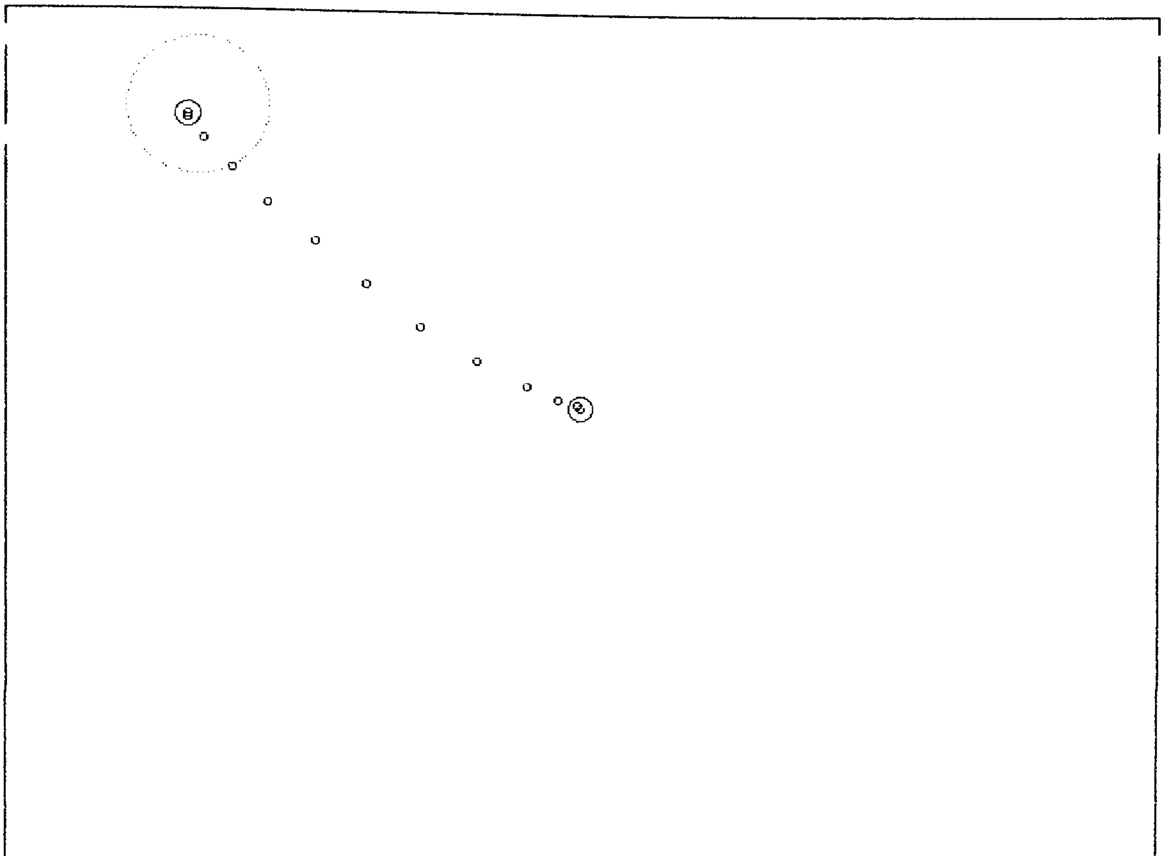


Figure 4.11a

The movement trace (Figure 4.11a) appears to support this view of the movement. The small circles represent the sampled trajectory points, and the very large circle the target area. An encircled trajectory point indicates that the point represents a non-movement. The distance between sampled points gives some indication of the movement speed with the faster movements having a greater spatial separation than the slower movements. The trajectory of this single movement is a curve rather than a straight line. Distinguishing between phases of non-movement, and movement, using the syntactical rule illustrated in Figure 4.11b, yielded the three movement phases described below.

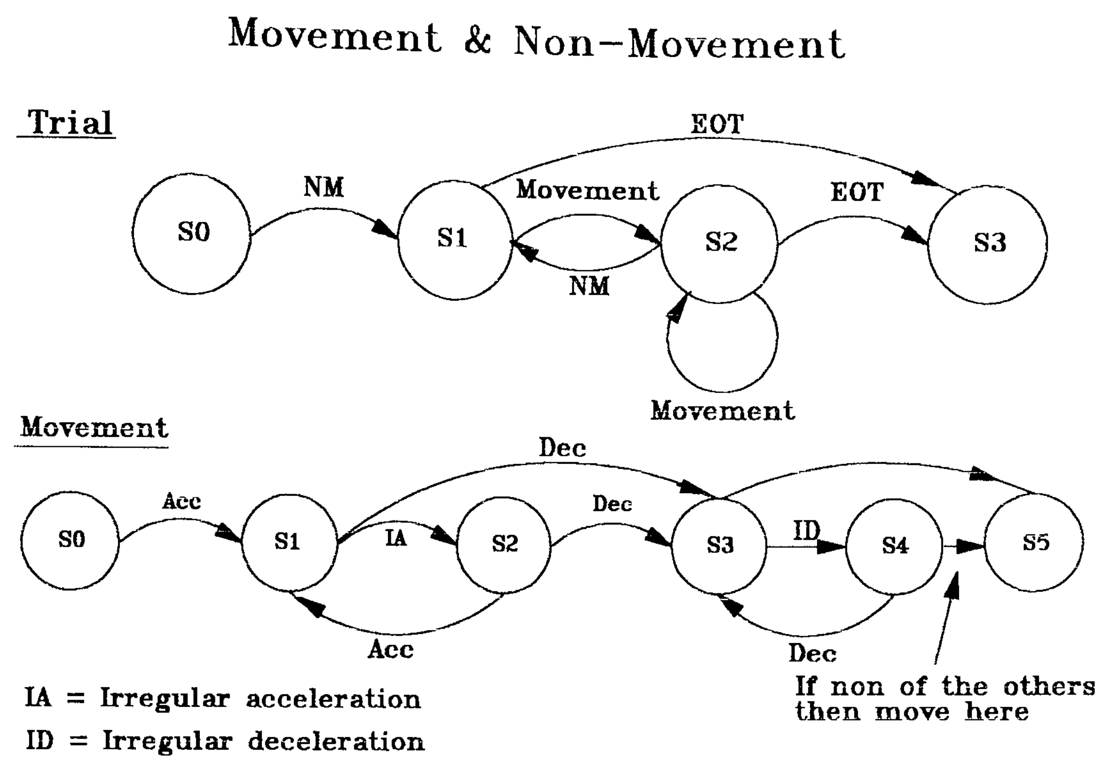


Figure 4.11b

The associated calculations are discussed later within this chapter.

107	Trial No.	
1	Target No.	
40	Size of Target	
2	Direction Code	
3	Distance Code	
3	No. of Movements	
nm	Type of Movement (non-movement)	
0.0000000000E+00		Start (0ms)
3.9200000000E+02		Finish (392ms)
3.9200000000E+02		Duration (392ms)
3.8393731636E-01		% of total trial time 38%
0.0000000000E+00		On or Off target (off)
movement		
3.9200000000E+02		Start
8.6300000000E+02		Finish
4.7100000000E+02		Duration
4.6131243879E-01		% of total trial time
2.7980350248E+02		Distance (279 units)
9.7685166213E-01		Curvature (0.98)
1.0000000000E+00		Effectiveness
0.0000000000E+00		Error of Aim (0 Deg.)
0.0000000000E+00		Error of extent (0)
5.9406263796E-01		Average Vel. (0.59 units)

nm

8.6300000000E+02	Start
1.0210000000E+03	Finish
1.5800000000E+02	Duration
1.5475024486E-01	% of total trial time
-1.0000000000E+00	On or Off target (on)
1.9525624190E-01	Location within target

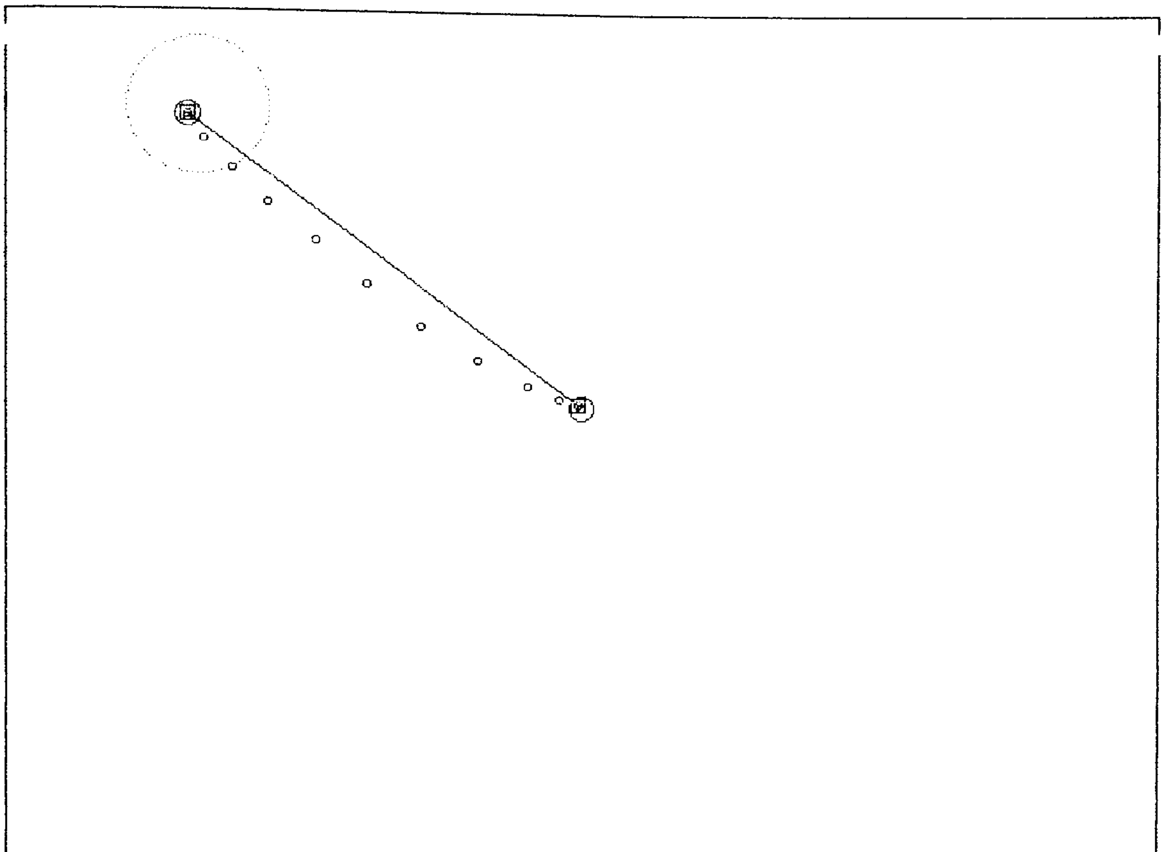


Figure 4.11c

The trace in Figure 4.11c shows the same movement trajectory as that in Figure 4.11a but it has some additional elements added to it. The straight line joins the beginning and end of the parsed movements (only one in this case) with the surrounding squares denoting the beginning and endpoints of the partitioned submovement.

The same movement data will now be parsed using the submovement criteria advocated by earlier researchers (see section 4.3). Figure 4.11d shows the transition diagram describing these syntactic rules.

When this movement syntax is applied to the movement data it yields identical results to those found using the previous rule. Thus, using the syntactic rules of previous studies, we would conclude that the subject had made a single curved movement without the need for movement corrections.

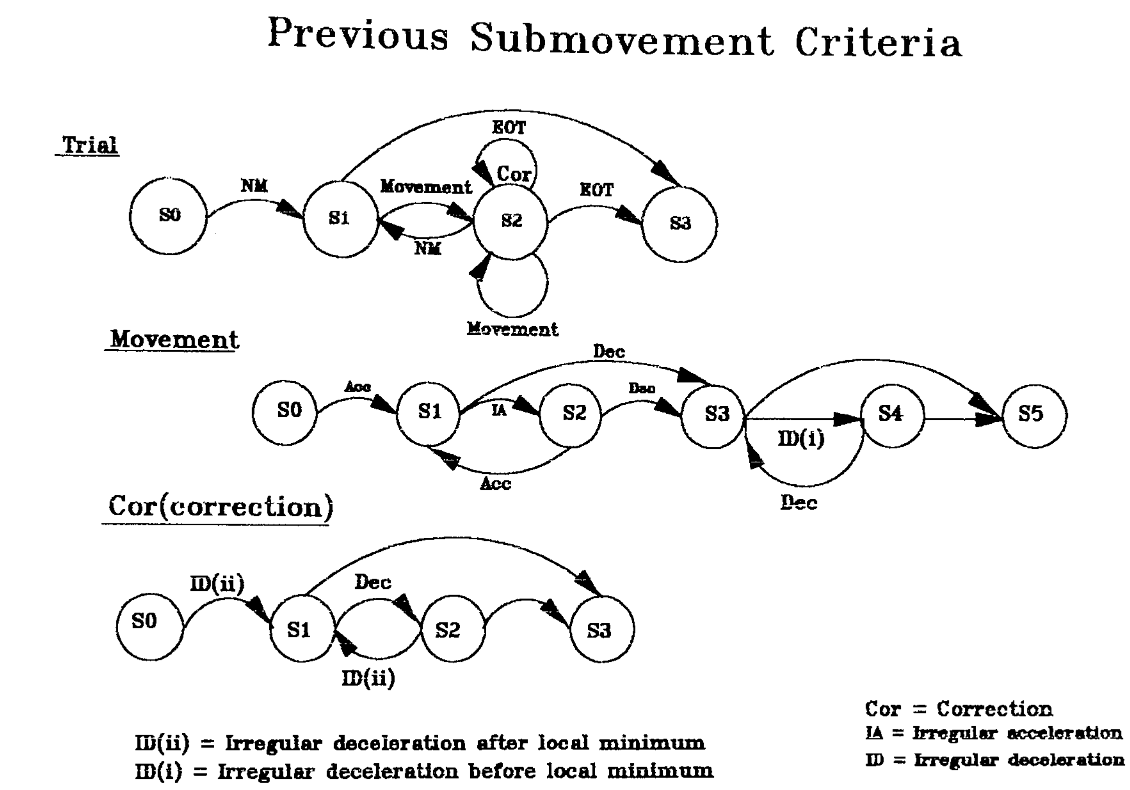


Figure 4.11d

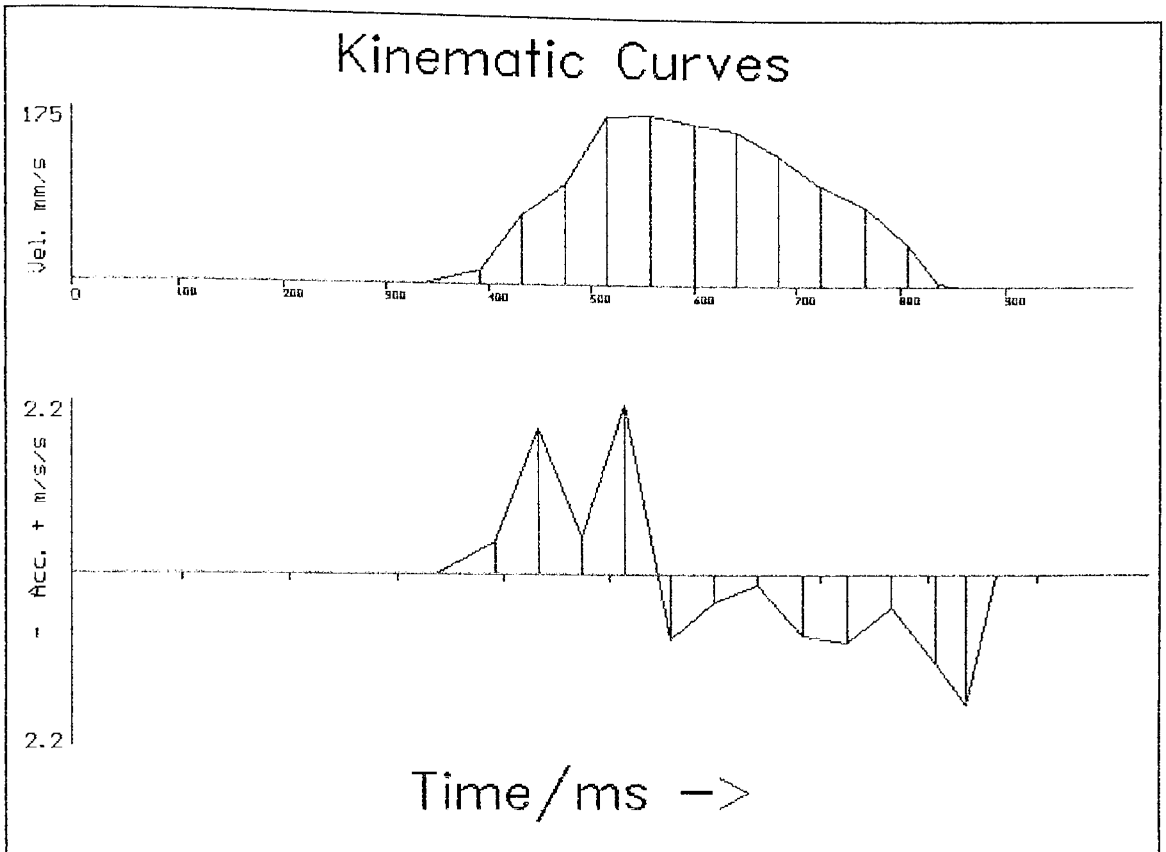


Figure 4.11e

However, inspection of the kinematic curves (Figure 4.11e) associated with the movement reveal that this 'single' movement has irregularities in its accelerative curve. These were not detected by the previous syntactic rule due to the irregularities not occurring in the final decelerative portion of the movement. Applying the syntactic rule, based on the possibility of irregularities throughout a movement (illustrated in Figure 4.10), using the parsing rule indicated earlier (see Figure 4.9) produced the parsed movement trace, as shown in Figure 411f, and the parsed output as shown in Appendix A8.

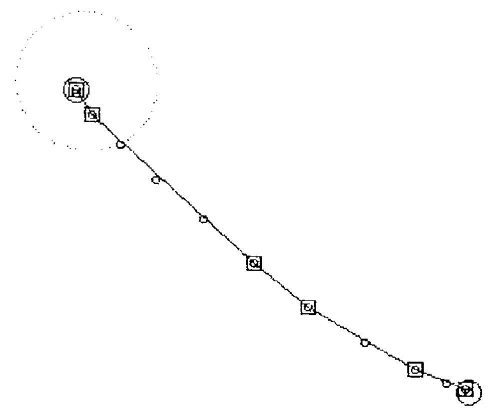


Figure 4.11f

This analysis of the movement data, based on the latter syntactic description, suggests that the movement consisted of a number of phases of movement. An initial acceleration phase whose aim was 'off target' was followed by an irregular acceleration phase which brought the movement trajectory 'on target'. This in turn was followed by a deceleration phase which had two irregular episodes which may considered as 'braking' within the movement. Thus, these three irregular acceleration phases appear to be acting as movement corrections with the initial irregular phase serving to correct for direction, and the latter irregularities in deceleration serving to ensure that the movement stopped over the target area.

The initial analysis of the movement, based on the initial parsing rules, suggested that an efficient single curved movement had accomplished the input task. However, subsequent more detailed analysis revealed that this 'single' movement contained corrective phases which compensated for the errors in direction and extent that arose throughout the movement's execution. This supports an analysis based on the latter, more complex, syntactic rule, rather than the previous two rule sets. Further, considering the two earlier syntactic rules, if a line is drawn between the beginning and the end of the movement phase (see Figure 4.11c) it can be seen that many of the movement points are 'missed' by the line, but using the latter syntactic rule, the movement trajectory can be approximated by a series of connected straight lines (See Figure 4.11f).

This suggests that a movement trajectory that involves irregular acceleration and deceleration phases may not be well approximated by a single straight line (as derived from the earlier two syntactic rules), but might be better approximated by a number of connected lines derived from the later syntactic rule. Whether or not a movement phase can be approximated by a straight line has implications for the sorts of measures that we can apply to this movement phase (discussed shortly).

Before we discuss the analytical measures that we wish to apply to the parsed movement, I should like the reader to consider the movement trace shown in Figure 4.12a.

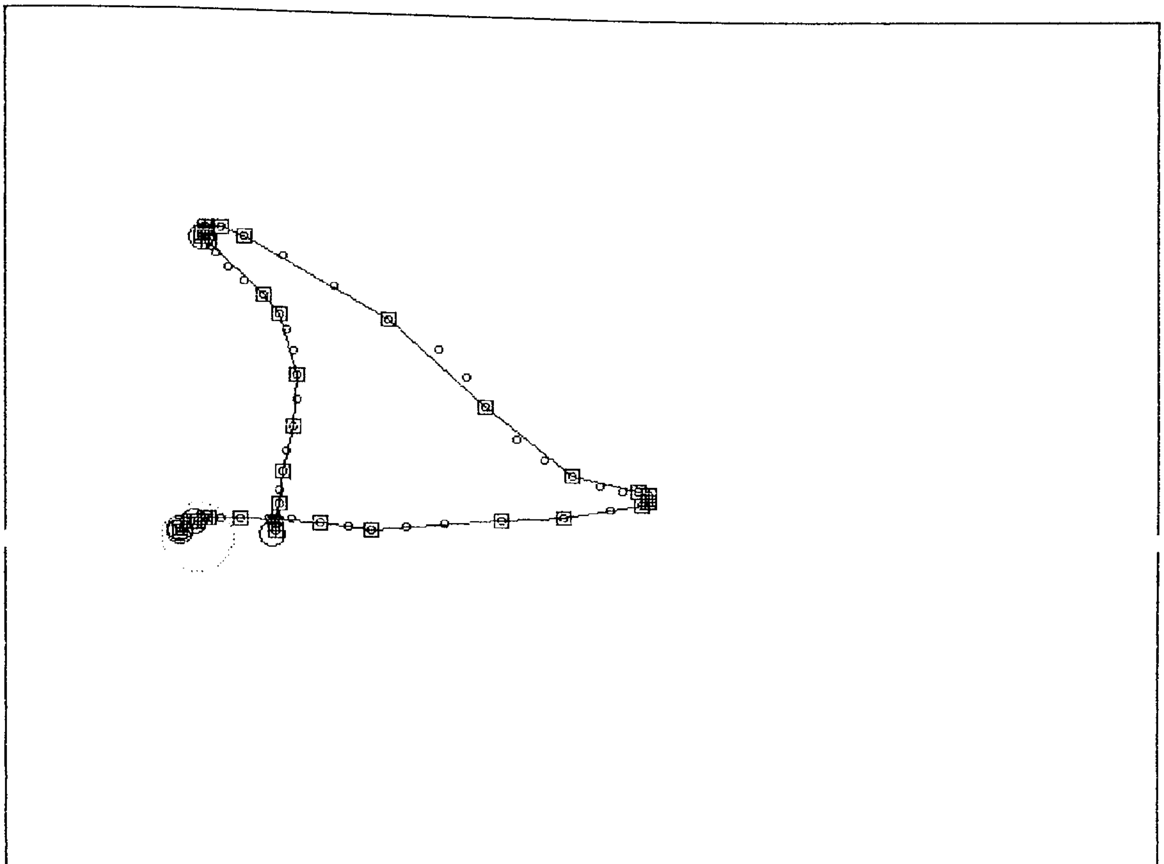


Figure 4.12a

This shows the movement trace for a relatively inexperienced mouse user. As can be seen, this is a far more complex trace than that the previously discussed movement. Whilst not typical, it was certainly not found to be an atypical occurrence across a number of device users. The kinematic curves are shown in Figure 4.12b.

This movement example is shown in order to convince the reader of the often complex device movements, and to demonstrate the need for the sufficiently powerful analytical tools that had to be developed throughout this research programme.

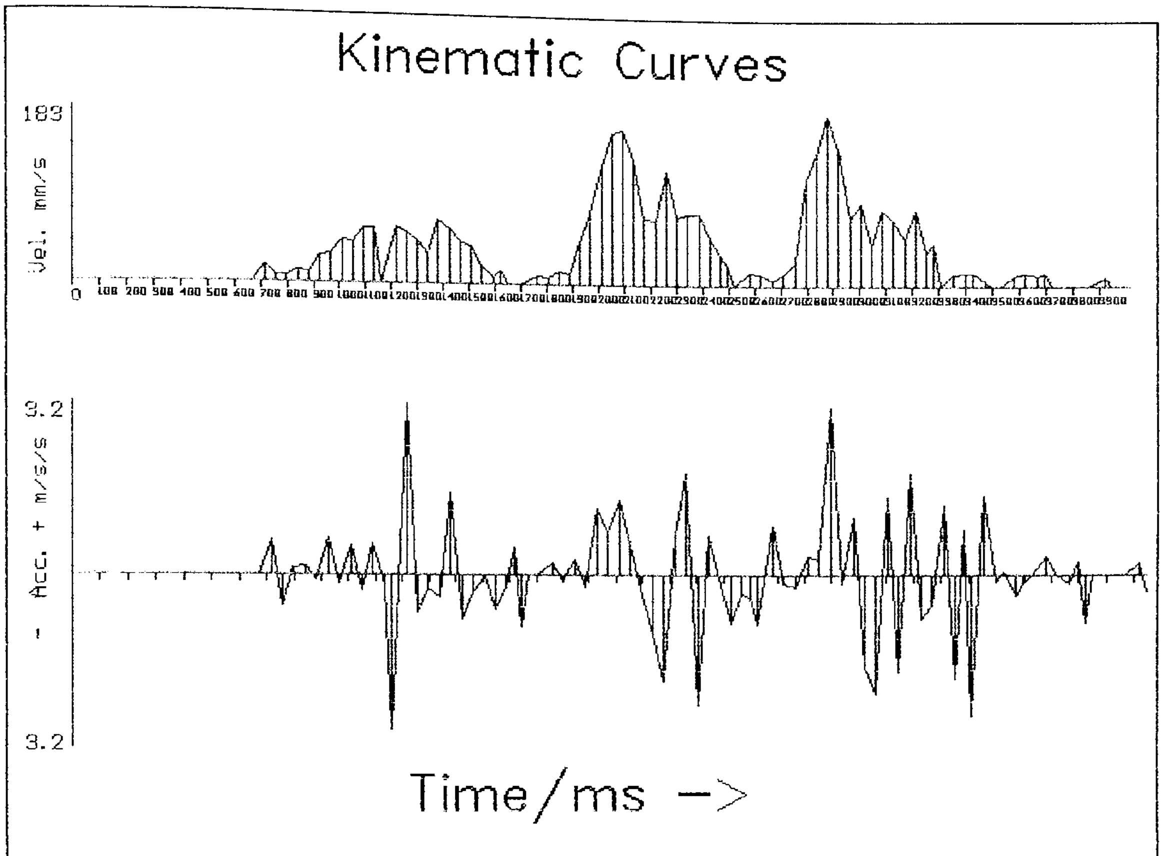


Figure 4.12b

4.6 Discrete movement analysis

All discrete submovements (including non-movements) are described with reference to: their type, start time, end time, duration, and percentage of the total trial time. Non-movements are also described as either occurring within the target area or outside of the target area. Submovements which involve movement have several additional measures attached to them: distance travelled, curvature, effectiveness, error of aim, error of extent, and average velocity. These will now be discussed in more detail.

4.6.1 Distance travelled

How far does a submovement travel? There are two distances of particular interest: the distance between the start point of the movement and the endpoint (measured by a single straight line), and the actual distance travelled throughout the submovement's execution. The best approximation that we have to the actual distance travelled is the sum of straight lines connecting the sampled submovement's trajectory (see Figure 4.13). This is the measure taken for distance travelled within this thesis. This measure corresponds to that of Movement Path Distance (Radwin et al, 1990; Lin et al, 1992) see section 2.4.1.1.3 .

4.6.1 Curvature

To be able to determine how well the movement is approximated by a straight line, the length of the single straight line which joins the start point to the endpoint is divided by the actual distance travelled. The consequence of taking this ratio is a number between 1.0 (no curvature) and 0.0 (infinitely curved!). This is a similar measure to RMS cursor deviation (Radwin et al, 1990; Lin et al, 1992) see section 2.4.1.1.3 .

4.6.2 Effectiveness & efficiency

The term 'effectiveness' was chosen as this measure gives an indication of the submovement's contribution towards achieving the input task. It is the ratio of the distance reduced to the target (as a consequence of the submovement) and the distance to be reduced at the movement onset. An effectiveness of 1.0 would indicate that the submovement had reached the target area. Whereas effectiveness of 0.25 would indicate that only a 25% reduction in target distance had occurred. A negative ratio would indicate a counterproductive movement. That is one in which the distance to the target would be greater, as a consequence of the submovement, than prior to the submovement's initiation.

Implicit within this measure of effectiveness is the assumption that only the distance to the target is of concern. An alternative measure might also take into account the duration of the movement. For example, a ratio of distance reduced by a submovement as described above, divided by the duration of the submovement, might provide an alternative description of submovement effectiveness. The former measure of effectiveness might be considered as '% effective distance reduced' and the latter as 'efficiency of movement' where an efficient movement is one which reduces the maximum amount of distance in the shortest possible time. Another way of conceiving of submovement efficiency is that it represents the 'effective velocity'.

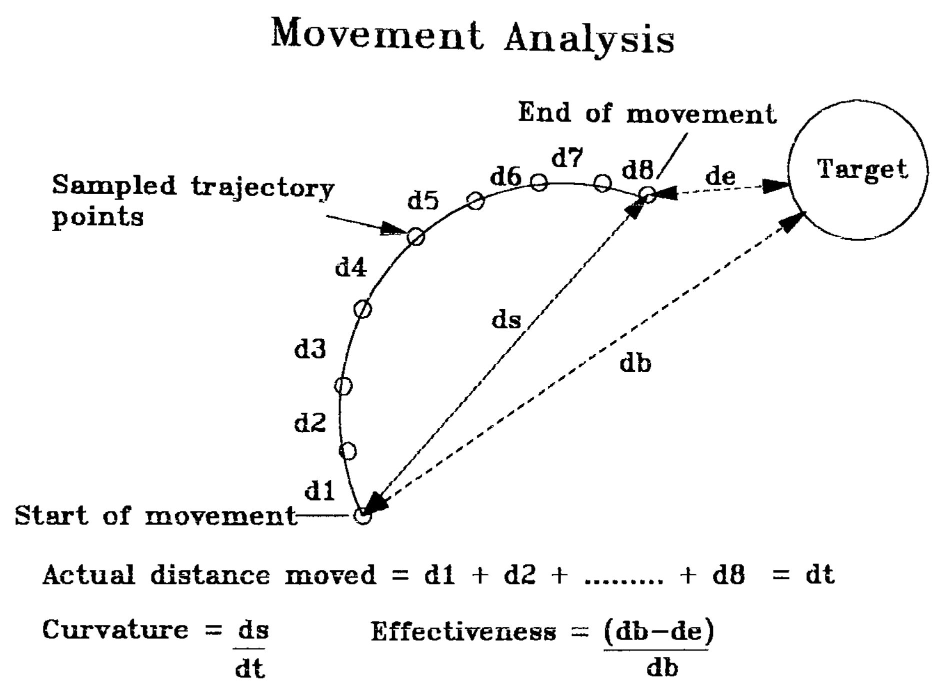


Figure 4.13

4.6.3 Errors of aim & extent

Errors of aim and extent (see section 2.4.2.2.1.2) have been used to describe a movement. An error of extent is said to have occurred if the amplitude of the submovement, as measured by a straight line joining the submovement's start and endpoints, would not 'land' on the target area if the direction of the submovement were to pass through the target centre. In calculating the magnitude of the error of extent it is either the additional straight line distance required to 'land' within that target area (at the closest point) or the amount of, straight line, distance that the submovement's amplitude would have to be reduced by (in the case of overshoots) to land within the target area. This error distance is expressed as a ratio to the required straight line distance. In the case of overshoots this ratio will be negative (see Figure 4.14).

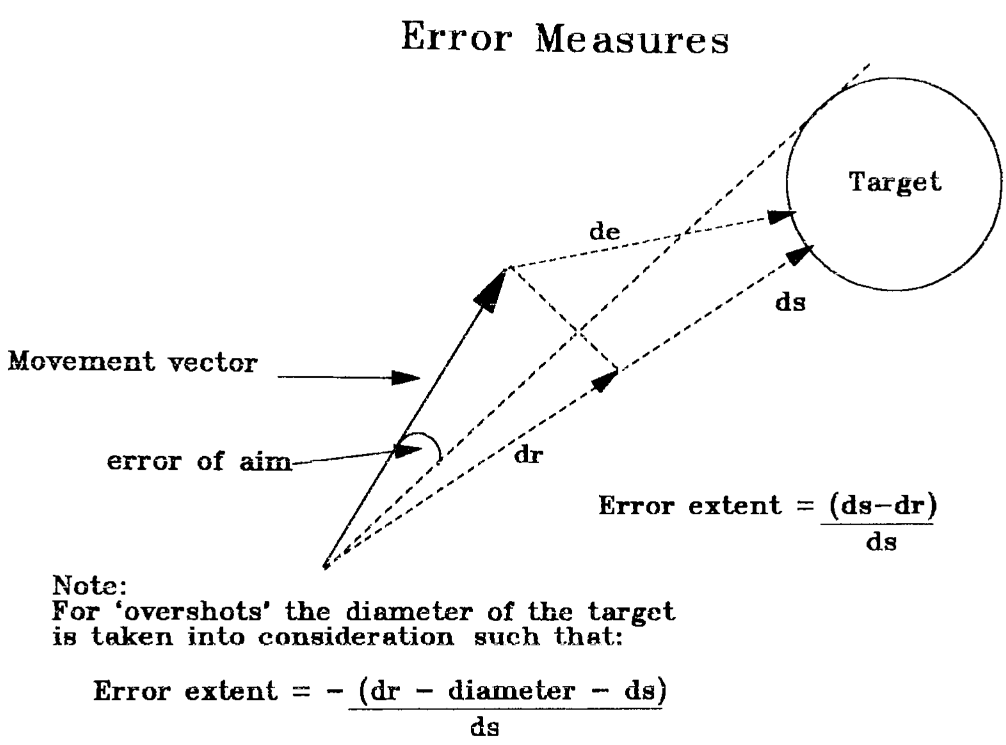


Figure 4.14

Errors of aim are considered to occur if the direction of the submovement, as measured by a straight line joining the start and endpoints, would not pass through the target area regardless of movement amplitude. The magnitude of the error of aim is measured in degrees and is taken from the closest target edge (see Figure 4.14). Error of aim is only a useful measure for submovements which can be approximated by a straight line.

4.6.4 Average velocity

This measure is simply the actual distance travelled divided by the submovement duration.

4.6.5 Target location

When a movement terminates with the target area and the end of trial has been indicated 'target location' gives an indication of the centrality of the endpoint position within the target area. It is the ratio of the distance from the target centre the submovement endpoint, to the radius of the circle. It takes a value between 0 and 1 with movements terminating at the target's centre having a value of 0, and those at the target's edge taking a value of 1.

4.7 Conclusions

Annett et al (1958) was one of the few studies which acknowledged the arbitrary nature of movement description and showed how different methods of movement measurement contributed to different conclusions being drawn. Most investigators have presented their descriptions of movement as self-evident and have neglected to justify their choice of movement description or measurement.

This chapter has shown even when only considering the partitioning of movement into submovements a number of different criteria may be employed. Moreover, the choice of movement description has been shown to influence the conclusions reached regarding a specific movement. In particular previous movement criteria have been examined and have been found to be too constrained with respect to their definitions of movement corrections. Furthermore, they have not presented an analysis of such submovement, just their occurrence or otherwise.

By drawing an analogy between movement partitioning and language parsing we have been able to express the problems of movement description in the terms of language grammar and have been able to employ the powerful techniques associated with computer parsing. This has enriched the ways in which we think of movement and has facilitated an automated movement analysis. Further more, it encourages an exploration of the consequences of using different movement grammars to describe movement. In this instance, different movement grammars have been largely employed to derive suitably different levels of movement description, depending upon the purpose of the analysis.

CHAPTER FIVE

RECORDING MOUSE MOVEMENT

This chapter describes the development of a measurement system for determining the position of the mouse device at a given instant in time. The problems surmounted in implementing this measurement system were difficult ones. In particular, the problems arising through 'micro-mismanagement' are discussed. The levels of precision and accuracy, required for such measurements within the research programme are discussed. Further, details of an empirical evaluation of the measurement system are provided. The evaluation exercise allowed the limits of the accuracy and precision of the measurement system to be known. Finally, these experimental findings are interpreted with reference to earlier input device studies.

5.0 Introduction

The movement descriptions in Chapter Three were applied to the kinematic movement patterns which in their turn were derived from the positional and temporal data associated with the device. This required the development of a measurement system capable of determining the spatial position of the input device at a particular instant of time. The implementation of such a measurement system can be reduced to:

- The construction of a suitable clock
- Locating the position of the input device
- Linking the device position to the clock

There are important advantages derived from having the required measurements made by the computer system with which the user interacts with the device under investigation. They are briefly as follows:

- The presence of already existing accurate and precise timing signals within the computer.

- Facilitating the linkage of device position and status, to the clock.

- Providing mechanisms for the storage of measured data.

- Facilitating data analysis of the stored data.

The computer system selected for the experimental program was an OPUS II (IBM PC clone). The choice of an IBM PC clone as the experimental machine had many advantages. As Segalowitz & Graves (1990) have indicated, the IBM PC family of computers (and presumably their clones) are widely used in behavioural research. The claim by Crosbie (Crosbie, 1989) that the IBM PC has become the *de facto* standard computer device within psychology departments is probably overstretching the case. However, the widespread use of the IBM PC has led to the existence of a body of knowledge appropriate to these investigations.

Other researchers have investigated the use of the IBM PC in reaction time experiments, and in particular discuss and suggest approaches to millisecond timing on the IBM PC. More generally, the choice of an IBM machine gave the researcher access to an extensive software library, and a large range of computer hardware which assisted in the development and implementation of this measurement system. The computer language selected for most of the computer programming within this project was Turbo Pascal (version 5.5).

Despite the mouse device providing spatial information, and the presence of very precise and accurate clocks within the computer system, the implementation of the required measurement system was very problematic and time consuming.

The structure of this Chapter will correspond to that of the division of the measurement system above. Initially, it deals with the requirements for, and implementation of, a suitable clock. This is followed by: the measurement of the input device's spatial location, and a description of the overall measurement system (the linking of spatial and temporal information). Finally, the implications from this Chapter for other input device studies are considered.

5.1 The timing requirements

The requirements of a clock are determined by the purpose for which it is to be used (Whelan & Hodgson, 1978) and may be stated in terms of the clock's accuracy and precision.

The precision of a clock is the degree to which the time can be described, for example hundredths or tenths of a second, and accuracy is a measure of the discrepancy between the value measured by one clock when compared to that measured by an agreed standard clock. Whelan & Hodgson (1978) define an accurate measurement as one having a relatively small systematic error and a precise one as possessing a small random error.

The construction of kinematic patterns of device movement require a clock of suitable precision and accuracy. Given the requisite timing resolution, a relatively high degree of clock accuracy is also required. A 'good' clock will assist in the identification of submovements by automated parsers. Similarly, the quality of the clock must be sustained throughout a complete movement trial, rather than the 'one shot' timers typically found in reaction time experiments

The resolution, or precision, of the clock had to be such so that submovements could be discerned. Research into kinematic movement patterns has discovered submovements of as little as 90ms duration (Meyer et al, 1988). However, given that within this study the input device's spatial position could, potentially, have been sampled at intervals as short as 5ms, a clock precision of 1 ms was selected.

The determination of the required clock accuracy is more problematic than the required precision. If the measurements from a clock are only to be compared against similar measurements from the same clock, then one is primarily concerned with its consistency, and that the resolution is appropriate to the task at hand. However, if measurements from one clock are to be in some way compared with those of another then one becomes concerned with the accuracy of the measurements made. Given that this project required statements to be made regarding the durations, velocities, and accelerations found in movement patterns, some measure of the relative accuracy of such measurements would be appropriate to facilitate meaningful comparisons with other studies.

5.1.1 The computer derived timing system

Many researchers have identified difficulties arising from the need for relatively precise computer based timing (Creeger et al 1990; Crosbie, 1989; Segalowitz & Graves 1990; Swan, 1989; Bovens & Brysbaert, 1990). One of the main problems facing an investigator, using an IBM PC for experimental purposes, is the lack of an accurate clock to which the programmer has access (Swan, 1989). The system clock is precise, in that it provides timing information to 1ms resolution, but is relatively inaccurate in that this value is only updated every 20ms or so. That is, its precision is greater than its accuracy.

Bovens & Brysbart (1990), and Crosbie (1989,1990) discuss and present solutions to the timing problems associated with establishing millisecond timing on the IBM PC. The reader is directed to these publications and to Syck (1990) for a detailed description of the timing difficulties. The implementation of a millisecond timer in this case was developed from that proposed by Heathcote (1988), which was based on the earlier work of Buhrer et al (1987). The following is a brief description of the timing routine used.

A 16 bit timer 'chip' is loaded with a 'threshold' digit, and then an associated counter is incremented at regular intervals until its value equals that of a predetermined value (set by the programmer) in the timer chip. When this occurs a hardware interrupt is generated which causes the processor to activate a preprogrammed piece of computer code (user written interrupt routine) which increases a software integer within the computer program. The value of this integer variable represents the 'ticking' of the clock, and corresponds to an approximate interval of 1ms.

5.1.2 Verification of the clock's accuracy

The possibility of computer based experimentation within the Reaction Time (RT) paradigm has been considered as a mixed and dangerous blessing.

> *'The advent of microcomputers has greatly facilitated studies based on this general paradigm [RT], but it has also introduced a number of potential errors in timing the events that are under experimental control.'*
>
> Creeger et al (1990, p. 34)

Segalowitz & Graves (1990), indicate that most behavioural studies utilizing computer based timing fail to provide documentation on the quality of the timing used. Given the difficulties associated with computer based timing this can be regarded as a serious omission from the studies concerned.

Most of the proposed solutions to the 1ms interval clock have relied on generating a hardware interrupt at regular intervals. The interval between hardware interrupts is largely determined by the numerical value loaded into the timer chip. Using the same value (4A9 hexadecimal) Heathcote (1990), and Buhrer et al (1987) produced a clock with a period of 1000.15 μs. That is, 1ms + 0.015%. No details of the method used to determine this figure are presented. Moreover, no standard error about the mean is given. It seems unlikely that the time interval, if randomly sampled on a number of occasions, would always yield an identically accurate value for this level of precision. An empirical evaluation of the clock used in this study is now presented.

5.1.3 Empirical validation of the clock

As indicated earlier, an experimental clock based on that suggested by Heathcote (1988), was implemented on an IBM PC XT clone. The precision of the clock was given as 1ms but no information was provided as to its accuracy. The aim of the following experiment was to establish the clock's precision and accuracy levels.

Given that a millisecond timer had been established on the computer, there were two possible ways of determining its accuracy and precision.

i) Generate an external event of a known duration and compare this to the duration as measured by the computer's clock.

ii) Use the computer's clock to generate an external event which can then be measured by another clock of a known accuracy and precision.

Both approaches are equally valid, however the latter method was adopted due to the availability of a suitable measurement device.

The device in question was a digital storage oscilloscope. This device, like a conventional oscilloscope, is able to provide a graphical representation of the waveform being measured over time. However, unlike a conventional oscilloscope, the storage scope is able to 'freeze' the measured waveform for a particular period of time.

5.1.3.1 Method

The computer program implementing the experimental clock was adapted so that instead of causing an integer variable to be incremented at 1ms intervals it caused the computer's external speaker to switch between output voltage levels of zero or five volts (approx) for each 'tick' of the clock. Given the clock interval of 1ms, and the connection of the speaker wires to the oscilloscope, this produced a square wave output of 1ms duration.

The external speaker was removed (muting the sound), and replaced with a 100 ohm load resistor. The output voltage across the speaker wires was then observed using The Tektronix Digital Storage oscilloscope model 2221A 100MHz (Serial No.B010617). At random intervals over a four hour period the output waveform was recorded, using the storage function of the oscilloscope. This yielded a sample of 101 waveforms. The precision and accuracy were determined by calculating the confidence intervals, based on the standard error of the mean duration, for the pulse width using the Student's t sampling distribution (Blalock, 1972; pages 211-213).

5.1.3.2 Results

An approximate squarewave output was observed using the digital storage scope in the oscilloscope mode. Table 5.0

shows a summary of durations measured from the stored waveforms.

Summary Table of the Stored Waveforms

Mean Pulse Width	1000.63 μs
Standard Error	0.31 μs
Sample size	101
Range	15.00 μs
Minimum	992.50 μs
Maximum	1007.50 μs

Table 5.0

The 99% confidence intervals, using the calculated sample mean and standard error, yielded a value of 1000.63 +/- 0.82 μs ($t=2.63$, $P<0.01$, $df = 100$) for the pulse width of the observed waveform.

5.1.3.3 Discussion

We are now able to make reasonably confident statements regarding the accuracy of the clock used within this experimental programme. We can be 99% confident that an interval of our timer will be within the range:

$$\boxed{1000.63 \pm 0.82\ \mu s.}$$

This figure does not lead to concern over the precision and accuracy of the timer for use with the previously described tasks.

The percentage gain of the clock (0.063%) is higher than that value (0.015%) obtained in the few studies that have presented such information (Buhrer et al, 1987; and Heathcote, 1988). Although, in both studies information regarding the method of gain determination was not presented. Over a ten second period this gain would lead to an accumulated error of 6.3 ms (0.063%).

It should be noted that the computer program instructions to turn the speaker on and off may have had different delays before coming into effect. If the instruction to turn the speaker off took longer to come into effect than that to turn it on, then a systematic error would have occurred causing the pulse width to be widened. This could have been investigated by measuring the time delay between consecutive on/off signals. However, given that the levels of precision and accuracy of the clock were found to be suitable to the experimental task any further possible experimentation was not undertaken.

Having constructed an accurate and precise clock the next stage was to determine the precision and accuracy of the spatial location of the mouse.

5.2 The position of the mouse

Given that the starting location of the input device is known, then in principle, following a movement its subsequent position can also be known. This is providing particular factors are known and certain conditions are met. As indicated in Chapter Two many input devices, including the mouse, communicate the device's relative movement by transmitting a stream of data bytes representing the relative displacement. This information is utilized by the mouse driver software in positioning the screen cursor and/or acting on the status of the device buttons. Prior to considering those factors underlying the location of the input device, a general discussion on the relationship between the input device's movement and that of the screen pointer is presented.

5.2.1 Device and pointer movement

Input device movement leads to a corresponding movement of the screen pointer. In Chapter Two this relationship was described as:

$$\text{Gain} = \frac{\text{Screen pointer movement}}{\text{Device movement}} \qquad \text{Equation 5.0}$$

The following analysis of gain is presented as an aid to understanding the concept and assisting in its measurement and calculation. Two additional parameters are suggested; those of Device Resolution (DR) and of Pointer Resolution (PR).

The Device Resolution is the number of 'device counts' that are generated by a unit distance movement of the input device. That is:

$$DR = \frac{\text{Number of device counts}}{\text{Movement of the device}} \qquad \text{Equation 5.1}$$

This parameter is determined by the hardware of the input device, and may be regarded as a predetermined fixed value. For example, the optical mouse generates a device count of 23 for every centimetre of movement.

The Pointer Resolution is the resultant screen pointer distance moved that corresponds to a unit device count and is given by:

$$PR = \frac{\text{Pointer distance moved}}{\text{Number of device counts}} \qquad \text{Equation 5.2}$$

The relationship between the number of counts received from the device and the resultant screen pointer movement is not a fixed parameter and may be manipulated by the system designer or, in some cases, by the user. It is also partly determined by the size of the screen pixels.

The relationship between these two parameters and the Gain of the input system is given by:

$$\text{Gain} = \frac{\text{Pointer movement}}{\text{Device movement}} = \text{PR} \times \text{DR} \quad \text{Equation 5.3}$$

Thus, if the Mouse Resolution and Pointer Resolution are known, then their product yields the input system Gain. Given that the devices under consideration move in two dimensions there will exist two sets of the above parameters; one for the X dimension and another set for the Y dimension.

5.2.2 Determining the location of the input device

From the above analysis, if the Device Resolution parameter is known, then as a result of its movement, the new position of the device can be determined by use of the 'count' data transmitted from the device to the computer. Clearly, if the device were a mouse, then it is assumed that the device would not be lifted during the period for which statements would be made regarding its position. For, if it were lifted and not replaced at the same location, the position as recorded by the computer would be incorrect.

In order to detect such device lifting , most of the mouse devices used within this experimental programme were adapted such that the lifting of them caused a signal to be sent to the computer informing 'it' of the lifting action. This was achieved by rewiring one of the mouse buttons to an external gravity switch; such that a lifting action by the user would simulate the action of a particular mouse button press.

5.2.3 The device resolution of two mice

The Device resolution parameter was empirically determined for two input devices. These were a rollerball type mouse (Primax, SOP 029) and an optical type mouse (Mouse Systems optical mouse). From personal experiences with rollerball type mice it was suspected that under certain circumstances the computer system failed to move the screen pointer in accord with the movements made by the device. This 'slippage' probably arose due to the frictional nature of the mechanism detecting the movement of the device. Therefore, it was hypothesized that the degree of slippage would be greater for device movements producing the greatest accelerations of the rollerball. Similarly, it was suggested that different device users might produce different degrees of slippage.

5.2.3.1 Design of the experiment

The design of the experiment was a Device(2) X Subject(2) X Movement(2) factorial design with Device and Movement being within subject factors. Where: the two levels of device were the optical and rollerball mouse; the two levels of movement were fast and slow; and the two subject levels were two different users.

5.2.3.2 Method

The device order was counterbalanced for each subject. Both subjects completed 42 trials with each device. For half of the trials, in each 42 trial block, the subjects were instructed to complete a slow movement, for the remaining half they were asked to execute a fast movement. The speed order was randomised within each trial block.

Each trial consisted of the device being moved in the X plane in a left to right direction. On the completion of each trial the mouse count was noted, and the device displacement measured to the nearest millimetre. The ratio of the recorded mouse count to the mouse displacement in mm was calculated for each trial.

In the case of the optical device a further 48 trials were carried out, requiring bottom to top movements in the Y plane, by one of the subjects . This permitted the device resolution in the Y plane to be calculated for the optical mouse.

In order to minimise the degree of slippage for the rollerball mouse, its slippage was observed for various types of surface. That surface which yielded the least slippage was the optical mouse pad, and this was used as an operating surface for both devices.

The data analysis was carried out using the SPSS[1].

5.2.3.3 Results

Tables 5.1 and 5.2 summarise the results of the above experiment.

Device Resolution for the Optical Mouse

	Movement speed		
	Slow	Fast	Means
Subject 1	2.36	2.37	2.365
Subject 2	2.34	2.35	2.345
Means	2.35	2.36	2.355

Table 5.1

Device Resolution for the Rollerball Mouse

	Movement speed		
	Slow	Fast	Means
Subject 1	2.03	1.59	1.81
Subject 2	2.06	1.66	1.86
Means	2.045	1.625	1.835

Table 5.2

The Device and Speed factors were found to be significant ($P<0.05$). The Subject factor was not found to be significant. A significant ($P<0.05$) interaction between the Device and Speed factors was also found.

Table 5.3 presents the values of the Device Resolution parameters for the X and Y planes of movement.

1. Statistical Package for the Social Sciences

X & Y Device resolution for the Optical Mouse		
	Plane of Movement	
	X	Y
Means	2.36	2.38
Std. Dev.	0.03	0.04
Std. Error	0.0033	0.006
N	84	48

Table 5.3

5.2.3.4 Discussion

This experiment demonstrates that the device resolution parameter for the rollerball type mouse (used in this experiment) was not a fixed value. Its value was dependent upon the way it was used. In particular, the device resolution was found to reduce with faster movements. This is explained with reference to the rollerball mechanism (see Chapter Two) and the degree of slippage experienced with the acceleration of the device.

This finding indicates that the device resolution will be partly determined by the types of movement made by the user. Therefore device resolution may vary within users, and possibly between them, depending upon the characteristics of the movements made. As demonstrated earlier, the overall input system gain is partially determined by the device resolution parameter, and in the case of the rollerball type mouse this value of gain will be dependent upon the way the device is used. Given, that the variability in the mouse device parameter arose through the frictional nature rollerball mechanism, it is then proposed that the notion of a fixed system gain for all such devices is misleading.

Although some slight variability in the mouse resolution parameter was observed for the optical mouse, This could easily be accounted for by the precision of the measurements made in determining the distance moved by the device.

Slippage was only demonstrated in the rollerball device for movements made in the X plane following a left to right direction. However, given the nature of the cause of the slippage, it was deemed unnecessary to investigate slippage in the other planes and directions of movement.

Given that no significant differences were found between or within subjects for the optical mouse, all the Device resolution values for this device were pooled and the and the standard error about the mean calculated for this sample (Blalock, 1972). Similarly, those values determined in the Y plane of movement were treated in a similar fashion. Table 5.4 summarises this process. These values were then used as the basis for the estimation of confidence intervals (Blalock, 1972 pp 211-213) using the Student's t sampling distribution, which yielded at the 99% confidence interval:

DRs For The Optical Mouse
Device resolution X = 2.36 +/- 0.008
Device resolution Y = 2.38 +/- 0.015

Table 5.4

such limits could have been calculated for the rollerball mouse, but given the earlier finding regarding its unreliable Device Resolution they would not be very useful. The differences in the above standard errors for device resolutions between X & Y was due to the sample sizes used to determine their mean values.

5.3 Putting it all together

At the beginning of this Chapter three requirements were set out for a suitable measurement system.

1) A suitable clock

2) A method of determining the spatial location of the mouse

3) The linking together of 1 and 2

Having dealt with items 1 and 2 above, it only remains for item number 3 to be addressed. One may think this a straightforward and simple process, but in practice this was not the case. The difficulties centred around a difficulty in measurement which become poignant to atomic physicists earlier this century, namely;

> *'That event and observer are not separate'*
>
> Bronowski (1960, pp 83-84)

The very act of making a measurement (observation) on a system alters the state of that system. In most cases of measurement the effect of the very act of measurement appears to have little or no influence on the state of the system. However, when considering the timing of external events by a computer system the above should be borne in mind; a view advocated by Creeger et al:

> *'Using personal computers to obtain precise timing of events requires a sort of micro-management in which timing of events is not affected by ongoing computational processes.'*
>
> Creeger et al 1990, pp 34.

In order to understand the basis for the need for micro-management the reader will be introduced to the concept of computer system interrupts. A detailed description of the computer interrupts on the IBM PC can be gained from Swan (1989) and Syck (1990).

5.3.1 Interrupts

A microcomputer, when switched on, is always carrying out a number of 'housekeeping' tasks of which the user is mostly unaware. Although, in most microprocessors (those of the Von Neuman type) the machine is not capable of implementing two machine instructions at the same time, it is nearly always involved in some element of multitasking. That is, it has several tasks 'on the go' and divides its processing time between them. To an outside observer running a computer program it may appear as if the computer is only engaged in executing the commands of their program, but it is in fact interleaving this task with a number of others which have to be carried out in order for the correct operation of the machine to be maintained.

Multitasking is achieved through the use of interrupts. When an interrupt occurs the processor finishes the execution of current instruction and then attends to the interrupt signal which may cause a different set of computer instructions to be executed. These sets of instructions are referred to as 'Interrupt Service Routines' (ISRs).

When the instructions associated with the interrupt signal have been carried out the computer processor returns to processing those instructions it was engaged on prior to the interrupt.

Interrupt Service Routines may themselves be interrupted by other interrupt signals. Interrupts may have a priority over other interrupts preventing one ISR from being interrupted by a less important interrupt request. Certain interrupts may be 'masked' so as to prevent their immediate processing by the computer.

One class of interrupts of particular relevance to this research programme are hardware interrupts. A hardware interrupt is an interrupt signal that is associated with a particular piece of equipment, for example the RS232 communications port, or the keyboard. When a byte arrives at the RS232 port, or a key is pressed, an interrupt signal is generated which causes the processor to execute the specific ISR which has been written to process the particular hardware interrupt.

5.3.2 Serial Mice

Serial mice transmit their information to the computer via the RS232 port of the computer. This information is packaged in a number of separate sequential quanta described as bytes. As the communication channel through which they pass is serial, they arrive at the port with small intervals between them. When such a byte of information arrives at the Serial port a hardware interrupt is generated so as to inform the computer of its arrival.

The software item referred to as a mouse driver is in fact a specialized Interrupt Service Routine which deals with 'managing' the mouse device. These are generally supplied by the mouse manufacturer and are installed onto the system prior to the device's use.

5.3.3 Timers & interrupts

Most of the timers discussed earlier have operated by causing an external timing chip (the Intel 8253) to generate a hardware interrupt at 1ms intervals. They also provide an ISR which when executed causes a software integer variable to be incremented. This variable is defined within the experimenter's computer program and will appear to increment without the experimenter's program instructing it to do so (it is incremented by the ISR). In a similar way when a user moves the mouse the screen pointer is moved but without the user having to specifically implement the instructions in the form of a computer program. It is moved by the ISR supplied by the mouse driver package.

5.3.4 The Problems of Obtaining Precise Timing

Given the above discussions it becomes apparent that the computer interrupt system has to be manipulated so as to safeguard the precision and accuracy of the experimental clock.

In order to preserve the quality of the clock, described previously, it was necessary to prevent the timing ISR itself being interrupted by another ISR. This was achieved by ensuring that all the other 'maskable' interrupts, with the exception of the RS232 port, were masked so as to prevent them interrupting the timer ISR and that the End Of Interrupt (EOI) instruction was not given until the end of ISR. The timing ISR also had the highest priority amongst ISRs ensuring that in instances of conflict the timer ISR would always be processed first. However, an interrupt could not be interrupted, even by another one with a higher priority, unless the EOI instruction had been previously carried out during the current ISR.

Certain activities, in particular accessing the hard or floppy disks, are not themselves interrruptable so as to ensure their correct operation. Therefore, using these pieces of equipment during periods for which precise timing is required is likely to compromise the quality of the timing (Creeger et al, 1990). Hence, during periods of critical timing, the software was written so as to ensure that no such activities took place.

5.3.5 When Did the Device Movement Occur?

Given the need for complex micro-management, and the shortcomings of commercial device drivers, an appropriate device driver was developed for this experimental programme.

The developed software, as well as providing those facilities normally associated with a device driver, also recorded the arrival time, as measured by the experimental clock, of the device's positional location and button status. This time, positional and status data of the device, and the screen pointer position were all stored in the computer's memory until this information could safely be written to the hard disk.

The arrival time was determined by 'noting' the arrival time of the first byte (typically 3-5 bytes are transmitted) containing the new positional data. When all the data bytes had arrived at the serial port the new positional location and status were processed in an appropriate manner. The noting of the arrival time of the first byte was necessitated by the fact that the time taken for all the data (for each new update) to be sent is dependent upon the quantity of information to be transmitted, and the rate of data transmission. This could vary between 5ms and 45ms depending upon the device being used. This time, for the first data byte to be transmitted from the device to the computer, introduced a constant delay (for a particular device) into the measurement system.

The devices used in this study transmitted data to the computer when a change in the device's position or button status had occurred. That is, when changes in the internal counters which represented the device's status occurred, this information was then processed and transmitted by the device. On arrival at the computer RS232 port each byte generated a hardware interrupt. This could potentially lead to a conflict between the timer interrupt and the device interrupt requests. In such circumstances, as stated above, the timer ISR would be given priority, thus preventing the immediate processing of the RS232 port information. However, as the time taken to process the timing ISR was of the order of a few microseconds, only a small error would be introduced into the measurement of the byte arrival time.

5.3.5.1 Micro-mismanagement of the mouse

Segalowitz & Graves (1990), described a reaction time experiment using a Microsoft mouse device and associated driver software (version 6.14). They found that the time to transmit the data serially was relatively constant for the non-moving mouse (31ms) but that when the mouse was moving the time interval between successive device positional updates was slower and more varied (30 - 60 ms). They concluded that a moving mouse could not be used, reliably, for response timing. Similarly, Crosbie (1990) using the same type of device and driver software determined the transmission time to be 34ms.

Seglowitz & Graves attributed the variance in data transmission to the hardware of the device, rather than an anomaly within the timing software. Further they determined that the time to transmit the data from the mouse to the computer would take 32ms (should have been

33ms as is now shown). They calculated this from the product of the number of data bytes (5 for the Microsoft format), the number of bits per data byte to be transmitted (8) and divided this by the baud rate (1200 bits per second). Giving:

Transmission time = 8 x 5 / 1200 = 33 ms

However, the baud rate also takes into account the start and stop bits (Blasewitz & Stern, 1982; Lancaster, 1982; Zaks, 1982) which are inserted into the 'bit stream' prior to data transmission for every data byte. Thus, their calculations should have taken the number of bits per byte to be 10 rather than just 8, giving:

Transmission time = 10 x 5 / 1200 = 41.67ms

This time is longer than some of the transmission times found by Seglowitz & Graves (1990). This apparent discrepancy was the impetus for a small experiment.

5.3.5.2 Transition time experiment.

Using the timing algorithm described earlier (see section 5.1.1) the time for successive mouse data transmissions was noted for a moving mouse. The mouse was continually moved throughout a ten second interval and the 'arrival times' (as measured by the experimental timer) of the data transmitted from the mouse noted. The results are shown below (Table 5.5a).

Transmission Time for Mouse Data	
mean transmission time	28.22ms
Standard deviation	1.14ms
Standard error	0.08ms
N	221
Range	10

Table 5.5a

The large variability in transmission intervals and faster than expected transmission times were of some concern, given the need for precise and accurate timing.

However, these puzzling findings were resolved when it became apparent that if the EOI instruction was placed at the end of the ISR for the mouse driver (an intuitive location), it prevented the driver software from being interrupted by the timing ISR which was occasionally 'missed' as it could not interrupt the mouse driver ISR despite this being of a higher priority. This was the case for the results obtained above. However, when the EOI instruction was the first instruction to be executed in the mouse driver ISR the following results (Table 5.5b) were obtained.

Transmission Time for Mouse Data

mean transmission time	41.58ms
Standard deviation	0.49ms
Standard error	0.02ms
N	689
Range	1

Table 5.5b

All of these time intervals were either 42ms or 41ms, as would be expected from the 1ms resolution of the clock.

5.3.5.3 Discussion of experiment

This experiment has shown that a mismanagement of the ISRs can lead to substantial errors (mean error, 33%) in the computer based timing of an external event. In this case it occurred due to the clock ISR not being able to interrupt the mouse driver ISR (despite the timing ISR having a higher priority than the mouse driver ISR) due to the EOI instruction being placed at the end of the ISR rather than at the beginning of the mouse driver ISR. Had the EOI been placed at the beginning of the mouse driver ISR the timer ISR would have been able to interrupt the mouse driver ISR and update the software clock.

Due to the timer ISR not being able to interrupt the mouse driver ISR some beats of the hardware clock were 'missed', which in turn resulted in the software clock not being updated causing it to 'run slower'. Moreover, these timing errors are not systematic, and therefore cannot be corrected in a post-hoc manner. The times obtained for the data transmission time using the correctly managed mouse driver and timer ISRs accorded well with those predicted by the calculations based on a baud rate of 1200 bits/s and 5 bytes of 10 bits each (1 start bit, 8 data bits, and 1 stop bit).

The findings from this experiment suggest that Microsoft mouse driver software fails execute an EOI sufficiently early within its associated ISR for the purposes of obtaining accurate timing. Further, it is suggested that the difficulties encountered by Seglowitz & Graves in moving the mouse can be attributed to micro-mismanagement rather than the mouse device itself.

5.3.6 Screen delay

The final source of micro-mismanagement occurs through making the assumption that when the programmers instructions to write something to the screen are executed the screen is updated instantaneously. Depending upon the current state of the computer system and the complexity of the item being written to the screen, it may take over 20ms for the item to be displayed (Heathcote, 1988, and Creeger et al 1990).

This arises as a consequence of the way that IBM PCs implement the display system. Information determining what is to be displayed is first written into the computer system's video memory which is then sequentially scanned by the display hardware. If the screen information is written to the video memory halfway through a hardware scan then only that part of the screen information in that video memory that is still to be scanned will be displayed. The remaining screen information stored in the already scanned video memory will have to wait until a new scan of the video memory is executed.

Solutions to this problem (Heathcote, 1988; and Creeger et al 1990) are achieved by preventing the video memory being displayed to the screen and then writing the information to be displayed into the video memory. When the video memory is to be scanned and displayed again, and all the required display information has been written to the video memory, the screen updating process (screen refresh) is then enabled causing the video memory to be displayed to the screen.

The reenabling of the screen refresh, within this research, is linked to the timer. This provides accurate timing following the presentation of the screen task event.

5.3.7 Movement sampling rate

Many of the earlier kinematic studies (see section 3.5) have used sampling rates in the order of 5ms. Given the transmission rate of serial mice, a sampling rate of 42.5 ms is obtained. The Nyquist Sampling Theorem indicates that a sampling rate of twice, the highest frequency present, in the signal waveform itself, should be used. Winter (1990) indicates that there are relatively high costs associated with high sampling rates for movement phenomena, and that human movement researchers have tended to 'play it safe' and over sample the data. Winter indicates that, for human movement, a sampling rate of 24 Hz (40ms) is often acceptable. Jagacinski et al (1980), using a 5ms sampling rate, found a minimal submovement duration of 90ms which suggests that a sampling rate of 45ms should be adequate to detect submovement phases. As

Medawar has indicated, it is not the magnifying power of the microscope, that gives it its 'strength', but its ability to resolve an item into component parts (Medawar, 1984).

5.4 Conclusions

Given that task completion time is one of the most often used performance indices for the assessment of input devices (see section 2.4.1.1), it is surprising to find no description of the timing systems employed in such studies. This chapter has indicated that such timing measures are prone to errors of accuracy and precision. One hopes that the quality of the clocks used within these studies was appropriate to the measurement task. However, as many device studies have stated their timing to a precision of hundredths of a second, their silence suggests this hope may not have been fulfilled. Researchers employing millisecond timing should be encouraged to make explicit statements regarding the implementation of their timing systems.

It is also to be hoped that in the case of comparative studies between different devices the systematic errors arising through the measurement of performance times equally affected the timing for all devices, and that performance differences between the devices were not an artifact of the timing system employed.

Similarly, in order to permit comparisons across device studies, for similar tasks, the timing should be of a similar quality between the studies. However, without the specific details of the timing used within such studies these aspects of the investigations cannot be critically evaluated.

Given the widespread use of rollerball type mice, the finding that the device resolution parameter for the rollerball type mouse was variable, and dependent upon the type of movements made, is important. It has implications for the way in which the concept of input system gain is to be interpreted. The input system gain has to also be considered as dependent upon the type of movements made with the device. Given the frictional nature of the device, the amount of slippage will also be dependent upon the ball and surface frictional characteristics. Therefore gain is only then meaningful when it is defined for certain types of movement, made by a particular person, with a specific device model, on a particular surface. In no reported studies involving rollerball type mice has the input system gain been specified in this way.

Clearly, given the two dimensional nature of the mouse device, gain is only applicable to movements made whilst the mouse is in contact with the associated movement surface. No studies involving the mouse device have attempted to rigourously monitor the lifting of the device during use.

The problems associated with accurate millisecond timing on IBM PCs have received much more attention than those associated with micro-management. Yet this study has demonstrated that micro-mismanagement can lead to substantial errors in event timing, despite the use of highly accurate and precise clocks.

CHAPTER SIX

EXPERIMENTAL TASK & DESIGN

This chapter describes the experimental task extensively employed in the empirical work on which this thesis is based. This chapter also describes the rationale for certain decisions regarding the task implementation and the selection of particular experimental designs. The experimental task underwent many revisions as a consequence of an extensive pilot study which is also described. A primary conclusion from this exercise is that the complexity of computer derived experimental tasks of this nature, are such, that one can never be convinced of their being artifact free; and when an artifact free task has been obtained, slight amendments can often have unforeseen and serious consequences.

6.0 Introduction

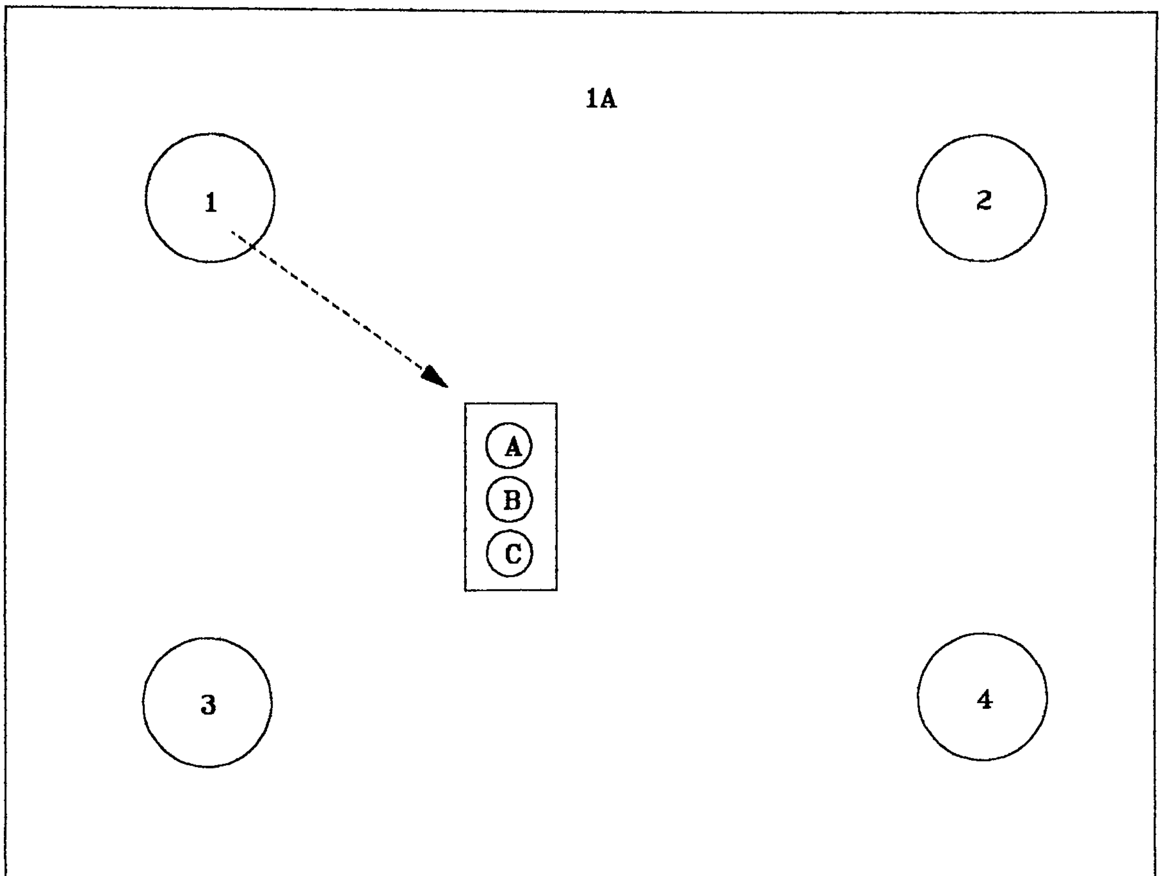
Researchers in the social sciences are often confronted with the dilemma of obtaining a 'tight' experimental design without compromising the ecological validity of their study. This research programme was not exempt from this conflict, and concessions were made to the experimental demands at the cost of ecological validity. In particular attempts to design a task, which was similar to that experienced by users in 'the field', was frustrated by the constraints placed on the task by the measurement techniques employed. Clearly, the effects of placing people in an experimental situation cannot be overlooked, but unless a naturalistic observational design is employed, which in this case was not practicable, this cannot be overcome.

Similarly experimenters want to be able to generalize their findings to a larger population than that of the experimental sample, and to this end they employ inferential statistics. However, the use of these statistics often requires that the sample has been appropriately obtained, which often conflicts with the other needs of the researcher.

It is conflicts, such as these, which will guide the debates within this chapter. However, to begin with, I should like to describe the experimental task, and then elaborate upon the compromises made.

6.1 The experimental task

At the onset of the experiment, the mouse was centred on the mouse pad by means of bespoke template, which facilitated the centring of the device on the pad. The user carried out a number of target acquisitions employing a computer mouse. Four circular targets, numbered 1-4 were displayed at the four corners of the screen. At the onset of the trial, a number and a letter (for example 3A) would be displayed at the top centre of the screen. This informed the users that they were to move the device, which in turn moved the screen pointer, so as to position the screen cursor within the circular target area, as indicated by the first number. Having located the pointer tip within the target area, they indicated their selection by pushing an appropriate mouse button. Having successfully acquired the target, the task information (3A) disappeared from the screen, there followed a small delay (less than 0.5s required for disk accessing), and a secondary target array appeared, as shown in Figure 6.0. The users were then required to select the one of the three small circles which contained the letter, that had followed the number (A), at the beginning of the trial.



Task Display

Figure 6.0

Having correctly acquired both the first and second targets, the trial would be repeated for a different combination of primary target number, and secondary target letter. The subject was not required to reposition the device in between trials, as one trial followed directly from the successful completion of the previous trial. Each primary target (1-4) had an associated secondary target array such that: for target 1, secondary array (A B C), for target 2 (D E F), for target 3 (G H I), and for target 4 (J K L). These letter triplets were always arranged in a vertical array, not unlike traffic lights, with the same letter sequence from the top to the bottom. Acquiring the secondary target was not central to the experiment but it primarily served to correctly position for the next trial. Appendix A3 contains the computer program listing for the main task.

The primary targets were circular and were always located in the same place, but the secondary target arrays appeared in different screen locations throughout the experiment. Subjects were required to complete 108 error free trials, which typically took them between ten and thirty minutes.

Three sizes of primary target were encountered (6mm, 15mm, and 30mm screen diameter) which were presented in blocks of 36 trials. Throughout each block, of a particular target size, subjects were required to make either horizontal, vertical, or diagonal movements, of amplitudes (7.5mm, 38mm, and 106mm on the screen).

If the user made a mistake, either in acquiring the primary or secondary target, the whole trial sequence (primary & secondary target) was repeated. If subjects lifted the mouse, at all, during the experiment, the experimenter was required to recentre the mouse before repeating the trial. If the subjects moved the device in the small time interval, between the correct selection of the primary task and the presentation of the secondary task, they were made aware of this by the computer and asked to acquire an additional very small target; this had the effect of repositioning them at their location prior to the premature movement.

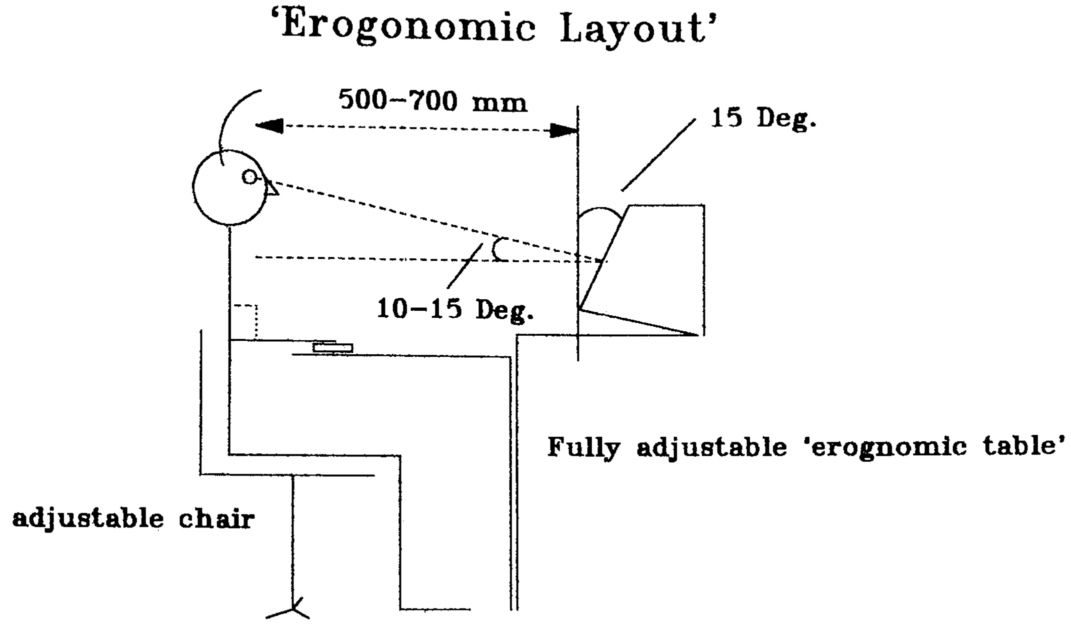


Figure 6.1

Before the experimental trials began, the chair height, the subject's distance to the screen, the mouse pad height, and screen height were adjusted so as to conform to ergonomic guidelines for terminal working conditions (TUC, 1985, see Figure 6.1).

The subject was then given a near visual acuity eye test, (the program listing is given in Appendix A2) which was specifically designed and implemented for this experiment. Several secondary target arrays were presented (with the letters in a non sequential order) and the subject was required to correctly identify the target letters. Additionally the subject was introduced to the screen cursor (an arrow) which was then presented either just inside, or outside, of a series of circular targets (See Figure 6.2). Their task was to correctly identify the tip of the cursor as either being just inside, or outside, of the target boundary.

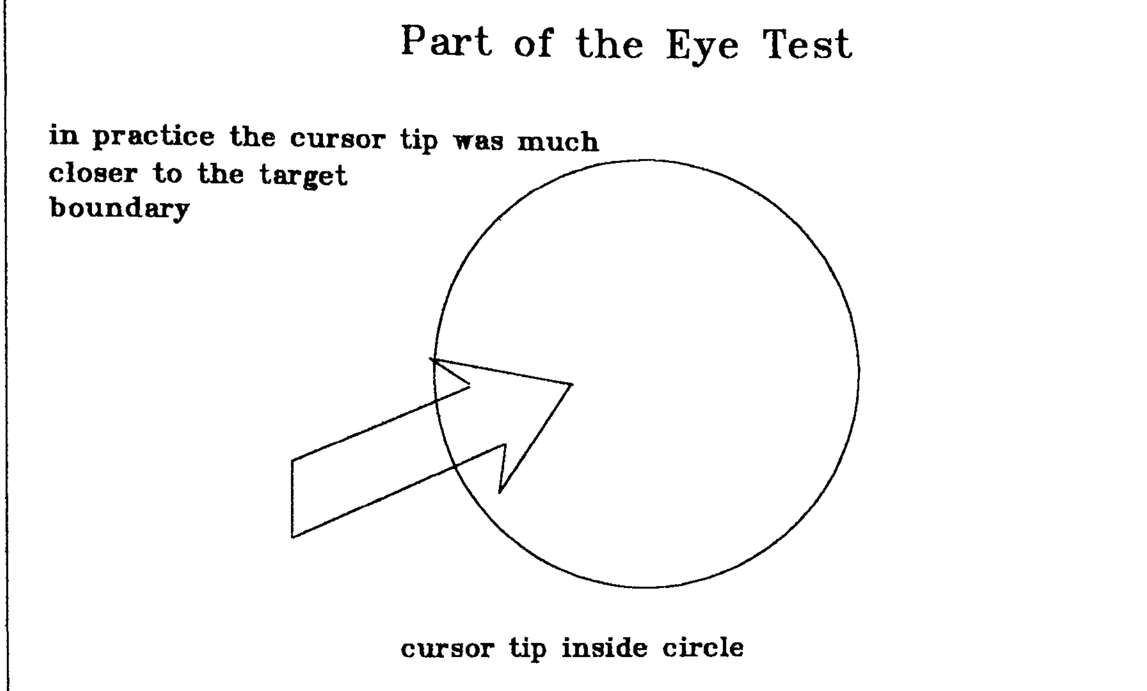


Figure 6.2

Subjects were given a preliminary demonstration of the task, after which they carried out 24 practice trials, prior to their commencing the main experimental session. The entire experimental session (including: demonstration, eyetest, practice, and main experimental session) took from 25 minutes up to one hour, per subject.

6.2 Rationale of the task

An informal task analysis of the user's input task revealed that many of the previous experimental tasks bore little resemblance to those found in 'practice' (see section 2.4.2.2.6). In particular a user's task involves:

- Acquisition of fixed location targets
- The selection of one target, from many
- Using hierarchical menu systems
- Additional cognitive loading

- Movements originating from a number of different mouse pad locations

These considerations were borne in mind for the design and implementation of the above experimental task. The acquisition of fixed location targets corresponds to the primary target selection. In the case of both primary, and secondary target acquisition, there is the need for target choice. The primary, and subsequent secondary, target selections corresponded to a two level, hierarchical, menu system. The subjects are required to remember the secondary target letter. The sequence of experimental trials is such that the subjects are not required to relocate the mouse, from trial to trial, unless the mouse has been lifted.

The secondary targets also ensured that the subjects were correctly positioned (at the correct angle, distance etc.) for the next primary target acquisition. In fact, it was the primary target acquisitions for which the movement data were collected, and not the secondary targets.

Given that the mouse would not normally have to be relocated at the end of a primary target acquisition, the position of secondary target also compensated for individual differences in the endpoint of the cursor, following the selection of a primary target. That is, it was also used to 'take up the slack', resulting from users not having to acquire the exact centre of the preceding primary target.

The secondary target acquisitions could have also been recorded, but the lack of experimental control, arising from the above compensations to the primary acquisitions, and the additional costs, associated with movement recording and subsequent analysis, mitigated against this information being recorded.

6.2.1 Task variables

The task variables, and their associated values, were decided with regard to the discussion of these topics in

Chapter Two (see section 2.4.2). The variables selected were: movement amplitude, target width, target orientation, target location, and system gain.

The screen locations, at the four corners of the display, for the primary targets were selected so as to maximise the possible movement amplitudes required to acquire them, and to reflect the fact that most fixed location targets, in practice, are often to be found at either the top, or bottom, of the display.

The required movement amplitudes, and target widths, were selected so as to present a wide range of task difficulties, and too ensure that subjects would be unable to correctly predict the required target, based on knowledge of their start location alone, prior to the commencement of the trial. The movement amplitude was taken as the minimal straight line distance from the starting location to the border of the target area (see section 2.4.2.2.2).

The required movement amplitude directions: horizontal, vertical, and diagonal, were selected, as not all targets could be approached from all possible angles; due to the location of the primary targets. However, they do reflect the findings of previous work and are consistent with earlier descriptions of the required target approach angle.

The system gain was either 2 (low), or 5 (high). These values were selected as they reflected the lowest, and highest values, respectively, available to the task. The highest value was restricted by the smallest target width, and the lowest value was constrained by the need for the target to be reached without necessitating the lifting of the mouse; which was determined by the size of the mouse pad. A larger mouse pad could have been used, but they are not often found in practice, and are also difficult to obtain.

6.2.2 Design

The following experimental description applies to the experimental task that served as the basis for the subsequent experiments. Variants on this task, and any additional experimental variables (such as age, gain, device), will be described in the, following, relevant experimental chapters.

The task may be considered to consist of five experimental factors: target width (3 levels), movement amplitude (3 levels), target angle (3 levels), target location (4 levels), and system gain (2 levels). All of these factors, with the exception of system gain, were taken as within subject factors, with system gain being a between subjects factor. This yields, for a particular subject, (3x3x4x3) 108 experimental combinations. Each subject carried out every combination of the within subject conditions. That is, the experimental session consisted of 108 experimental trials.

The experimental trials were presented in three successive blocks of 36 trials. For each block the target size remained constant, but varied between the individual blocks. The presentation order of these size blocks was counterbalanced across subjects. Each size block contained nine blocks of 4 trials (9x4 = 36 trials). Each block of 4 trials required the user to acquire 4 different primary targets (as defined by their location). The order of the 4 targets, within each of the nine blocks, was randomized, with the proviso that no two blocks of 4 targets had the same target order.

The movement amplitudes, and approach angles, were randomized within the nine blocks of 4 trials, ensuring, that for each block, of a particular target size, every target was approached from every direction, for each possible movement amplitude.

These task orders were determined by a computer program (listed in Appendix A1) which then calculated the required screen positioning details for the secondary targets associated with these selection tasks. The output from this program also provided task details that were essential in analysing the movement data.

6.2.3 Task constraints

The need to ensure that location of the device was known, when only relative movement data were available for the device, and the requirement for the device not to be lifted, necessitated the use of an optical mouse. In everyday tasks, people are seen to lift the mouse, and very few optical mice are to be found in use. Additionally people are not required, in practice, to resist moving the device before the presentation of the next target. These constraints may call into question the ecological validity of the experiment. Experiments, found in later chapters, will explore the possible influences of these constraints on people's behaviour.

6.2.4 Data analysis

The data recorded from the main task was converted into positional and temporal information using the computer program listed in Appendix A4. This was then converted into the associated X and Y velocities using the program listed in Appendix A5 whose outputs were combined by vector addition (Program listing Appendix A6) so as to produce the overall device velocity. This was then redifferentiated so as to produce the associated acceleration data for the trial. The outputs from these analyses were then processed by the movement parser (program listed in Appendix A7) which provided the data files for the subsequent SPSS analyses.

Although inferential statistical analysis is presented for the following experiments, caution is urged in accepting their appropriateness and the validity of the conclusions to be drawn. The studies, conducted within this thesis, have relied upon 'opportunistic samples', which violates a central assumption, the use of random sampling (assumption 1), underlying the appropriateness of inferential statistics. Furthermore, most of the experiments presented here have relied upon opportunistic student samples which casts further doubt upon the appropriateness of inferential statistics. These arguments will not be rehearsed here, as they are well documented elsewhere (Cochrane & Duffy, 1974; Schultz, 1969).

Violations of the additional assumptions underlying analysis of variance based tests: such as equal population variances of the treatment populations (assumption 2), and the normality of the criterion measures within the treatment population (assumption 3), are considered to be less serious than that of violating the assumption of random sampling, for the validity of such analysis (Glass, Peckham & Sauders, 1972; Ferguson, 1966; Norusis, 1986).

> *'Investigations into these assumptions [above] have generally concluded that assumptions 2 and 3 [above] can be violated in most cases without seriously distorting the stated alpha level.'*
>
> McNeil, Kelly, & McNeil (1975, p44)

McNeil, Kelly, & McNeil also urge researchers to consider the proportion of the variance accounted for by the experimental factors in addition to the associated significance levels. They present a relatively simple method by which the proportion of the variance accounted for by a particular factor (R^2) can be determined.

i) Determine the total sum of squares (SS_t)

ii) Divide the sum of squares for the factor (SS_f) of interest by the total sum of squares. The resultant ratio is the proportion of the total variance accounted for by that factor

$$R^2 = \frac{SS_f}{SS_t} \qquad \text{Equation 6.0}$$

Taken from McNeil et al (1975, p.53)

6.3 Pilot study

Prior to the experimental task, described earlier, there had been an earlier version of the task, now referred to as the 'pilot study', which was run on almost fifty subjects (N = 48). During this pilot study a number of problems came to light which led to an accumulating dissatisfaction with this experimental task, culminating in its abandonment in favour of the task described earlier. The problems were as follows:

- incorrect timing (see section 4.3.5.1)
- anticipation of target location by subjects
- the task, for many subjects, was too easy
- subjects ignored displayed instructions
- errors in the programming of the task
- some subjects relied on guessing the secondary targets, rather than attempting to remember them
- measures of device experience

Of the above, it is the programming error in the task, which was the most worrying. Given the complexity of the task program, it was found that some of the program 'bugs' only became manifest under rare task conditions. Some of these bugs became apparent after several months of extensive testing of the program, often arising as a consequence from earlier modifications to the program, attempting to 'fix' a previous bug. Having now established what is believed to be a 'bug' free program there was, and still is, a reluctance to modify the experimental task, except for minor amendments to the task, in order to avoid introducing any further 'bugs' into the system.

6.4 Definition of terms

Before the main experimental findings are presented it may be worth recapping on some of the terms that will be used throughout the remainder of this thesis.

Task completion time is the average time taken to complete the experimental trials.

Trial completion time is the time to acquire a particular primary target, or the time to complete an individual trial.

Discrete movements are referred to as submovements, and may either be episodes of movement, or non-movement.

Simple submovements display no evidence of movement trajectory correction whilst complex submovements do indicate such amendments to the course of the movement.

Reaction time is the interval between the stimulus presentation and the initiation of the first submovement.

Movement time is the interval between the initial submovement and the termination of a submovement within the target area.

Time on target is the summation of the intervals between the termination of a submovement within the target area and either the button being pressed or a movement terminating outside the target area.

Time on target and movement time can both be further partitioned into separate movement, and non-movement, intervals.

A primary target is one of the four fixed target areas located at the corners of the display.

A secondary target is one of the three small circular targets presented after the correct acquisition of a primary target.

Primary and secondary errors occur when either a primary or secondary target selection error is made

6.5 Conclusions

Experimentation using relatively complex computer based tasks, is a dangerous and mixed blessing. Computers enable experimentation, where perhaps it might not otherwise be possible, provides the possibility for automated complex data analysis, and affords a high degree of experimental control, but it needs extensive testing and evaluation over its complete operational range, if error free programs are to be obtained. In the case of the pilot study, the timing error invalidated much of the data collected, but it could have been detected prior to experimentation on subjects. In many cases the conditions for the 'misfunction' were so rare that they only arose in the case of one, or two subjects. It is most unlikely that all of the other problems would have been detected on a relatively small sample pilot study. Furthermore, apparently minor amendments can often have far reaching, unforeseeable, and undesirable consequences, in a complex, computer based, task system.

CHAPTER SEVEN

EXPERIMENT ONE

An investigation using the experimental design described in the previous chapter was analysed using techniques and methods described in the preceding chapters (4 & 5). Evidence of discrete submovements, and episodes of non-movement, were found whose characteristics were task dependent. In particular the incidence of submovements, and non-movements was found to increase with increasing task difficulty. In all but the easiest of task conditions the initial submovements were on average found to cover approximately half of the distance to the target area. Differences in overall task completion time attributable to different gain conditions were found to be task dependent with completion times being slower for the small target conditions. Equations based on Fitts' Law were found to give good post-hoc accounts of movement time based on the target size and movement amplitude but were not so effective at predicting, a priori, movement times. Poorer performers (when compared to good performers) were slower at initiating movement, moving to the target, and indicating target selection.

7.0 Introduction

Having developed suitable software and hardware tools for the recording and analysis of mouse movement, this chapter describes the application of these tools to the data collected for an input task. The task employed was that described in Chapter Six.

7.0.1 Design

A factorial design of five factors (target width, movement amplitude, movement angle, target location, and gain) was employed. However, due to the analytical complexity, and the possible subsequent difficulties in interpretation arising from such an experimental design, a three-way analysis of variance with factors of; target width, movement amplitude, and gain, was undertaken with movement angle and target location being regarded as controlled variables. Target width, movement amplitude, and gain were selected as factors due to their theoretical interest, and the relatively larger influences of target size and movement amplitude on task completion time. However the influences of movement angle and target location were not altogether ignored and separate one-way analyses of variance are also presented for these factors.

Gain was a between subjects factor. Subjects were allocated randomly to either a system gain of 2 (low) or 5 (high) condition , controlling for previous device experience, sex, and age.

7.0.2 Subjects

The subjects were an opportunity sample drawn from within Aston University. The majority of subjects were undergraduate psychology students possessing varying levels of previous device experience. Device experience was determined by estimating the number of days per annum (based on self report data) for which some device usage occurred. Data are presented for 48 subjects (29 female and 19 male) of mean age 23.85 years (S.D. 6.06, max. 50 & min. 19) who had a mean device experience of 56.63 days per annum (S.D. 91.37, max. 360, min. 0). Data for two additional subjects was rejected due to their failure to follow the experimental instructions adequately.

7.0.3 Additional analyses

This section will present a detailed analysis of the 'effects' of target width, movement amplitude, and system gain on performance. Differences in the movement characteristics between; good and poor performers, old and young, and previous device experience are also considered.

People's performance was partitioned into relatively 'good' and 'poor', based upon the median task completion time (1800 ms). Similarly, previous experience and age (continuous scales) were grouped into high and low, experience or age groups, based upon the median values for these groups. Thus, for people aged over 22 (median age) years age = 1, otherwise age = 0, and for people using the mouse on 20 days or more of the year (median usage) use = 1, otherwise use = 0.

Statistical analyses was carried out using the Statistical Package for the Social Sciences, Personal Computer version (SPSS/PC, V. 3.1).

7.1 Results - an overview, completion time & errors

A mean trial completion time of 1818 ms (S.D. 332.70) and mean primary error rate of 1.21 misses per session (S.D 1.7) were found. Over half of the subjects made no primary target errors and less than 13% of subjects made more than 3 primary target selection errors. A secondary target error rate of 2.4 misses per session (S. D. 2.5) was found on the secondary targets with 79% of subjects making fewer than 4 secondary errors, but with only 22% of subjects making no errors in secondary selection.

39% of subjects did not lift the mouse at all during the experiment with less than 10% of subjects lifting the mouse more than twice. Only 20.8% of subjects were able to completely avoid moving the device before the presentation of the primary target, following a secondary target selection, with 77% of subjects moving on less than 4 occasions.

Errors in primary target selection were significantly associated with the smaller target ($P < 0.05$, Small - mean rank 2.33, Medium - mean rank 1.86, Large - mean rank 1.80, Chi^2 8.09, D.F 2) using the Friedman Analysis of Variance test. There was no significant difference between the larger and medium targets. Primary target errors were significantly correlated with secondary target errors (df = 47, $P < 0.05$, $r = 0.52$) and the number of movement errors prior to primary target presentation (df = 47 $P < 0.05$, $r = 0.40$) but not with task completion time.

7.1.1 Sex, age, experience, and gain

Sex and age were not found to be associated with task completion time. However, previous device experience (high, low) was found to be associated with task completion time ($P < 0.01$, Chi^2 6.8, D.F. 1) using the Chi^2 test, with 75% of highly experienced users also being 'good' users. Furthermore, experienced users were significantly quicker when compared to the less experienced users ($P < 0.001$, mean - high experience 1656ms, mean - low experience 2010ms, t 4.3, D.F. 46) using the Student's t-test (independent).

The only significant difference between male and female users was found in the lifting of the device. Women lifted the device more often than men ($P <0.05$, mean lifts- men 0.89, mean - women 1.69, t 2.04, D.F. 45.98) using the Student's t-test (independent).

Older users were found to make fewer primary errors than younger users ($P<0.05$, old - mean 0.64 misses, young - mean 1.69, t 2.21, D.F. 46) using the Student's t-test (independent). There were no other significant age differences.

The low gain condition produced significantly more device lifts ($P < 0.005$, Low - mean lifts 2.04, High - mean 0.71, t 3.46, D.F. 46) but significantly fewer secondary target selection errors ($P < 0.005$, Low - mean error 1.33, High - mean 3.46, t - 3.21, D.F. 33.59) using the Student's t-test (independent). Overall task completion time was found to be independent of system gain.

7.1.2 Task factors & completion time

Initially, a one-way Analysis of Variance (ANOVA) for repeated measures was carried out for the effects of: target size, movement amplitude, target angle, target location, and learning effects, on task completion time. Summary tables are presented in Appendix B.1. Prior to this analysis the trial completion times were averaged for each level of each factor for every subject.

Task completion time was most strongly influenced by movement amplitude and target size. Longer completion times were identified with smaller target diameters and longer movement amplitudes. Although the mean task completion times reduced with each subsequent trial block these differences were not found to be significant. Similarly, although a main effect of approach angle was found, subsequent post-hoc tests failed to identify any significant differences between the group means.

Different target locations yielded significantly different task completion times (T1 = 1650ms, T2 = 1759ms, T3 = 2019ms, T4 = 1843ms) with target T1 (top left) resulting in the shortest completion times, and target T3 (bottom left) yielding the longest completion times. Possible reasons for this finding are discussed later within this chapter.

7.2 Time by gain, target size, & movement amplitude

The following analyses are a more detailed investigation into the effects of target size and movement amplitude at the different gain conditions. The 'effects' on performance of these variables are explored with respect to task completion time, reaction time, movement time, and time on target. Further analyses, based on the movements made, are then presented. The ANOVA summary tables for the following 3-way analyses are shown in Appendix B.2.

7.2.1 Task Completion Time

There were main effects of target size and movement amplitude upon completion time but there was no main effect of gain. Increasing the movement amplitude and reducing the target size were associated with an increased task completion time. The shortest mean task completion times were 1167ms (close-up to the largest targets) and the longest times were 2647ms (farthest from the smallest targets) with the difference being 1.48 seconds.

Figure 7.0a shows target size by movement distance, and Figure 7.0b shows the gain by target size interaction.

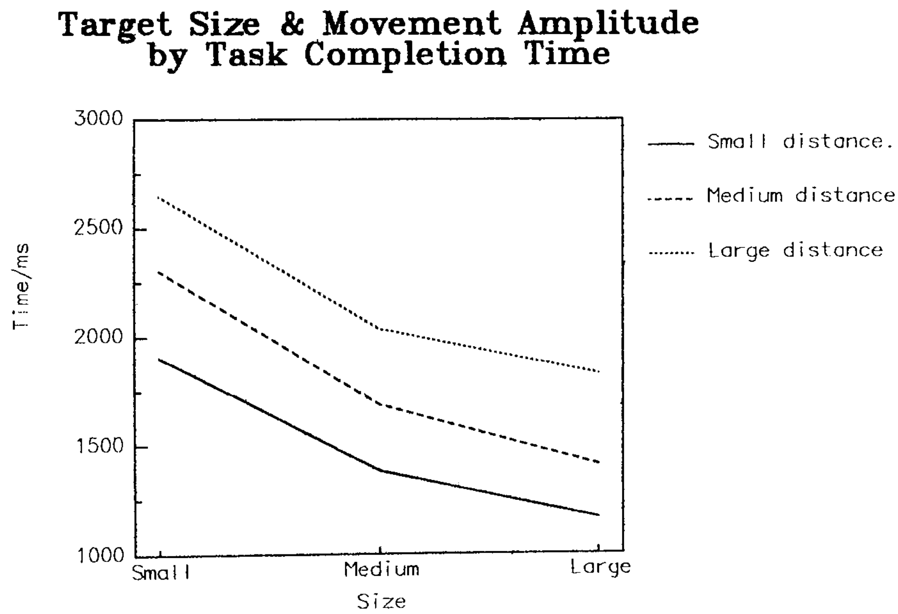


Figure 7.0a

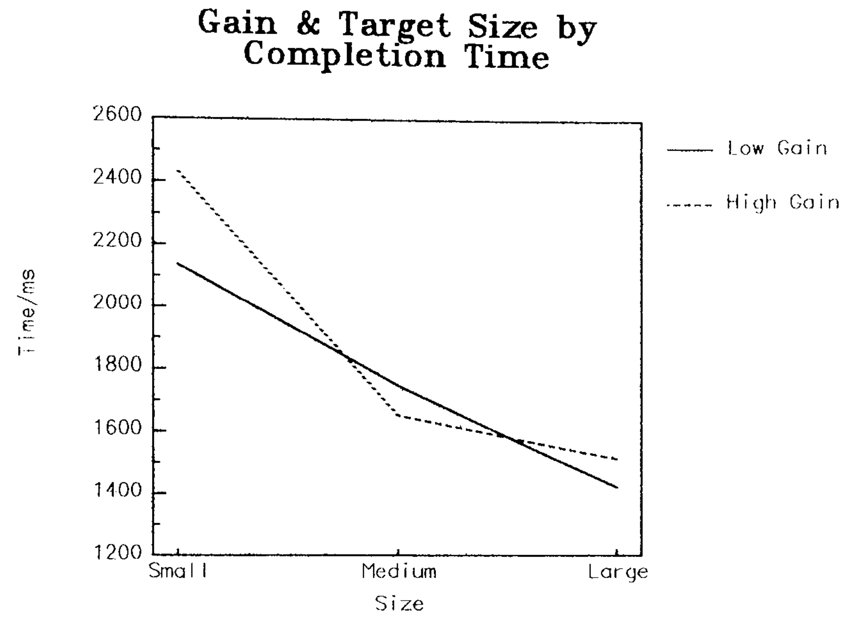


Figure 7.0b

Although the interaction term between target size and movement amplitude was significant it accounted for less than 0.2% of the total variance compared to 33% by target size and 22% by movement amplitude. The gain by target size interaction term was significant with the high gain condition being associated with the slower task completion times (2433 ms compared to 2136ms) for the small targets ($P < 0.01$, Tukey T 213.2). There was no significant interaction between movement amplitude and gain.

7.2.1.1 Reaction time

There were significant main effects of target size (1.3% of the total variance) and movement amplitude (1.4% of total variance) on reaction time, but there was no significant main effect of gain.

Figure 7.1 shows target size and movement amplitude by reaction time.

Tukey Post-Hoc tests revealed that none of the mean reaction times for different target sizes significantly differed ($P > 0.5$, T 54.6). Similarly, Tukey Post-Hoc tests revealed a significant difference (P < 0.05, T 38.8) between the small (mean 704 ms) and medium movement (661 ms) amplitudes but not between these and the large movement (696 ms) amplitude. The significant target size and movement interaction accounted for 0.9% of the total variance.

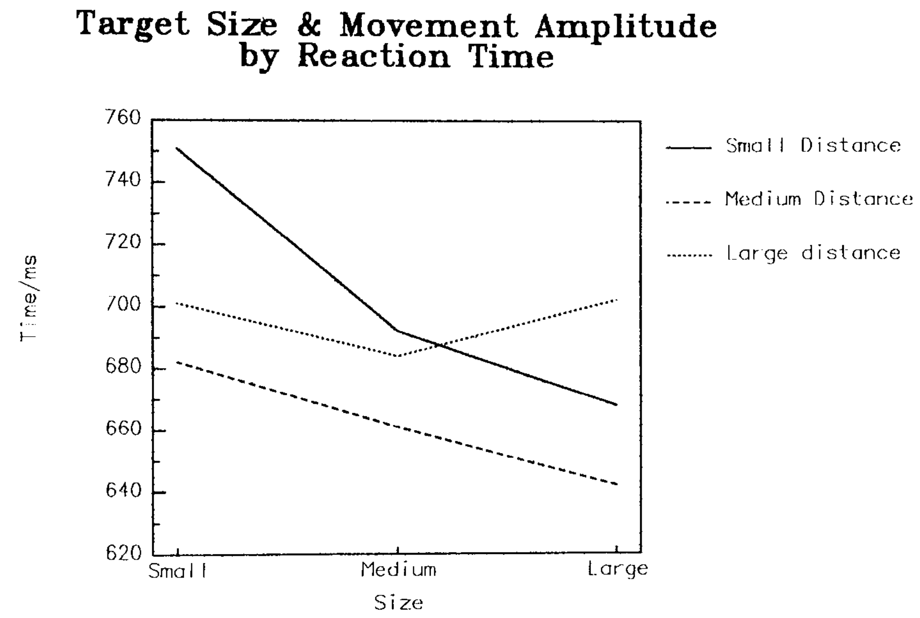


Figure 7.1

7.2.1.2 Movement time

There were main effects of target size (22.8% of total variance) and movement amplitude (47.8% of total variance), but not gain, upon movement time. Quicker movement times were associated with larger target sizes and with shorter movement amplitudes. The target size and movement amplitude interaction accounted for 1% of the total variance, and suggests that movement times for small targets are disproportionately quicker for small movement amplitudes.

Figures 7.2a and 7.2b show movement time by movement amplitude and target size, and movement time by system gain respectively.

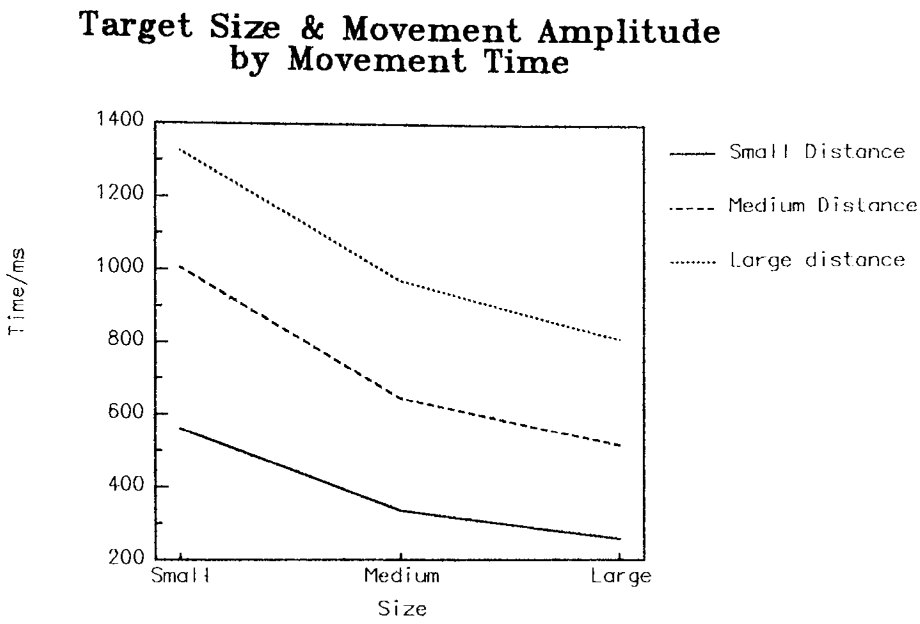


Figure 7.2a

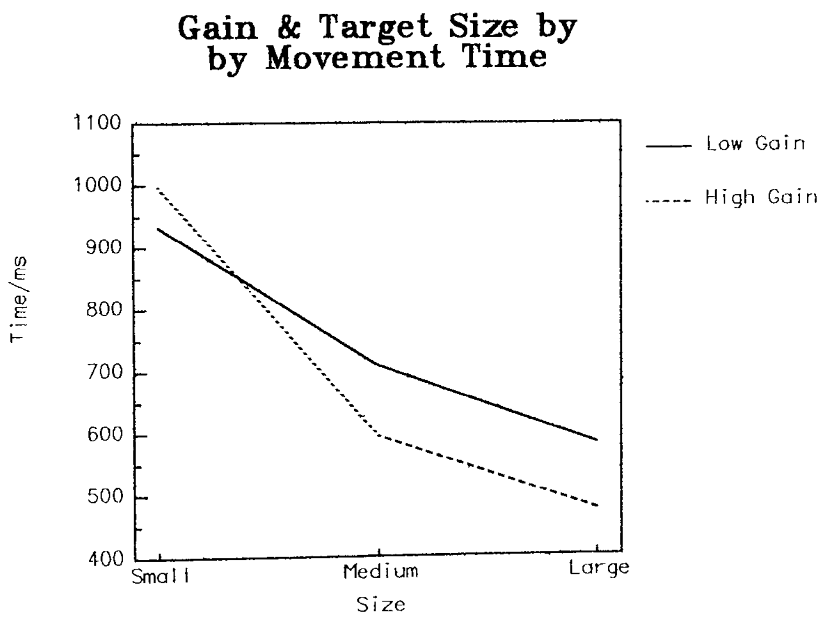


Figure 7.2b

Tukey Post-hoc tests showed the mean differences between all target sizes were significant ($P < 0.01$, T 104.4) and the mean differences between all movement amplitudes were significant ($P < 0.01$, T 90.7).

There was a significant interaction between target size and gain (1.2% of the total variance) where the high gain condition resulted in significantly quicker movement times for the large and medium target size conditions ($P < 0.05$, Tukey T 101.5) than for the small target condition. This may reflect the higher occurrences of 'holes in one' (see section 7.3.1) associated with this task condition.

7.2.1.3 Time on target

There were main effects of target size (36.8% of total variance), movement amplitude (0.6% of the total variance), and gain (8.4% of total variance) upon time on target. Time on target reduced for increasing target size and increased for the high gain condition (low gain 349ms, and high gain 484ms).

Tukey Post-hoc tests revealed that none of the time on target means significantly differed between different movement amplitudes ($P > 0.05$, T 46.0) but all of the means significantly differed between target sizes ($P < 0.05$, T 83.0). Figures 7.3a and 7.3b show the group means.

The significant movement amplitude and target size interaction (0.3% of the total variance) suggests that for large targets, the time on target for the large movement amplitudes is greater than for medium or small movement amplitudes. This was supported by post-hoc tests which showed the time on target for larger movement amplitude was significantly greater than was the case for the medium and small movement amplitudes ($P < 0.05$, Tukey T 51.2).

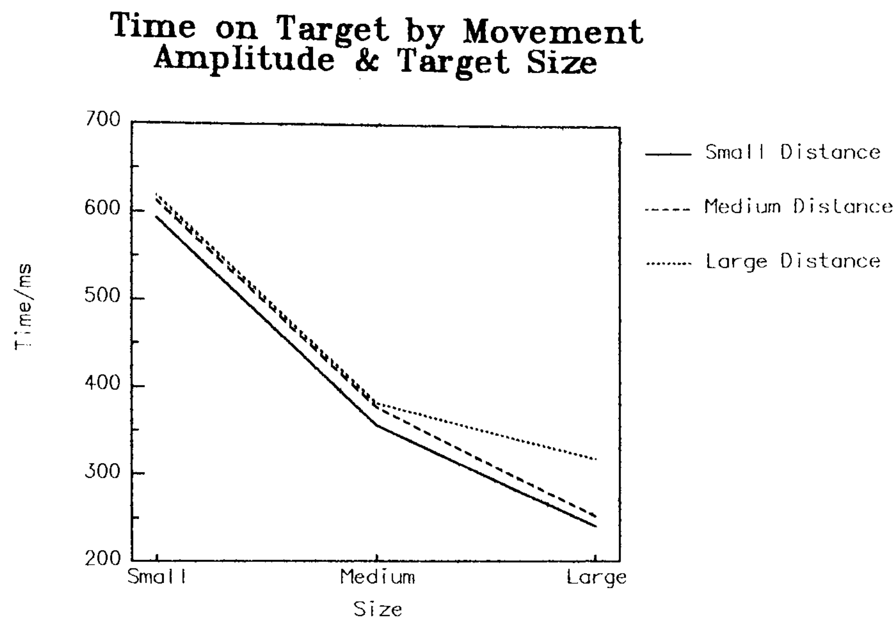


Figure 7.3a

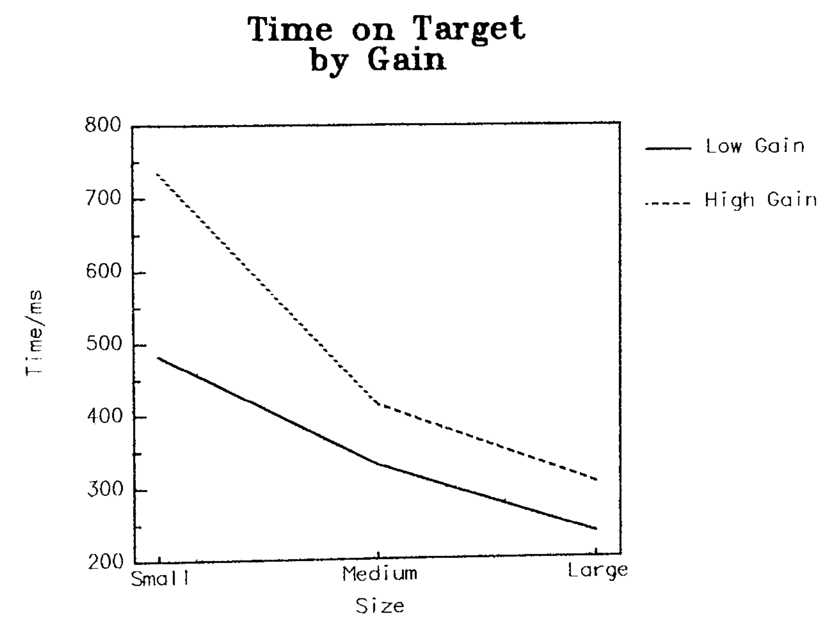


Figure 7.3b

There was a significant interaction between target size and gain. Post-hoc tests showed that the time on target for small target sizes was significantly greater under the high gain condition ($p < 0.01$, Tukey T 121.8).

Significant main effects of gain (10.7% of total variance), target size (42.3% of total variance), and movement amplitude (0.3% of total variance) on time on target (non-movement) were found. The high gain condition was associated with longer periods of non movement on the target area (mean time high 377 ms, and mean time low 242 ms).

Tukey post-hoc tests revealed that the difference between means for small targets and other targets differed significantly ($P < 0.01$, T 92.5), but there was no significant differences between movement amplitude means ($P > 0.05$, T 34.9).

Figures 7.4a and 7.4b show the group means.

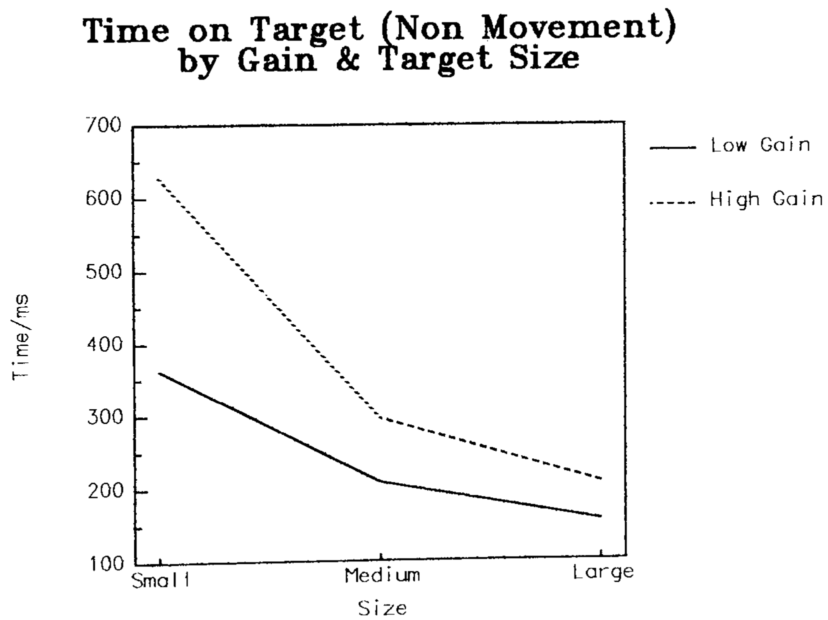


Figure 7.4a

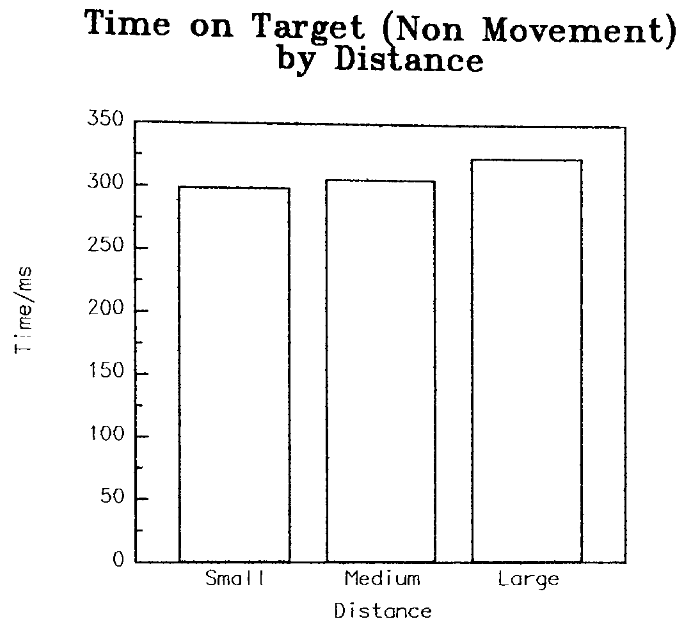


Figure 7.4b

The significant interaction between target size and gain (5.2% of total variance) indicates that for small targets the time spent in non-movement is greater for the high gain condition. This was supported by post-hoc tests ($P < 0.01$, Tukey T 107.9).

Significant main effects of target size (5.2% of total variance) and movement amplitude (1.8% of total variance) on time on target in movement were found.

Tukey post-hoc tests revealed that the difference between means for medium and large sized targets differed significantly ($P < 0.05$, T 26.0), but no significant differences between movement amplitude means ($P > 0.05$, T 34.9) were found.

Figures 7.5a, 7.5b, and 7.5c show the group means.

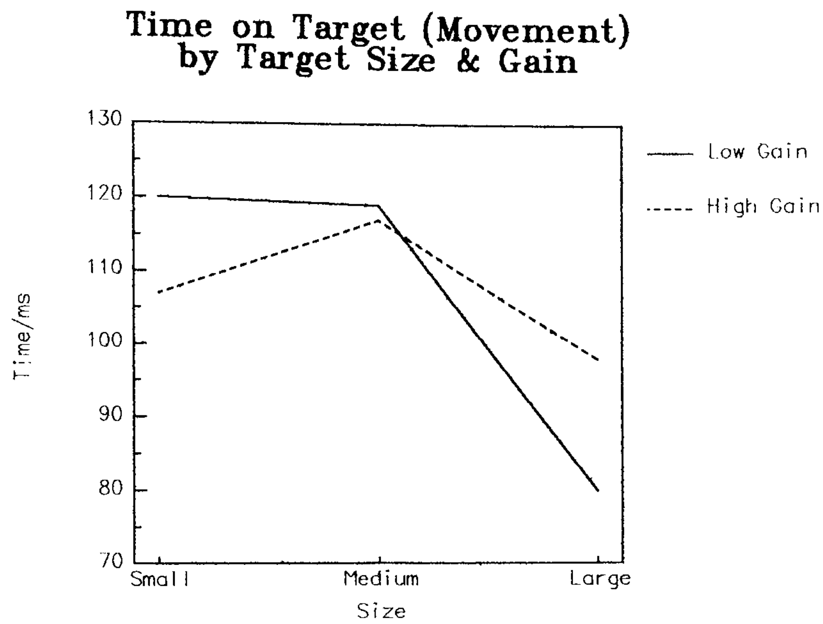


Figure 7.5a

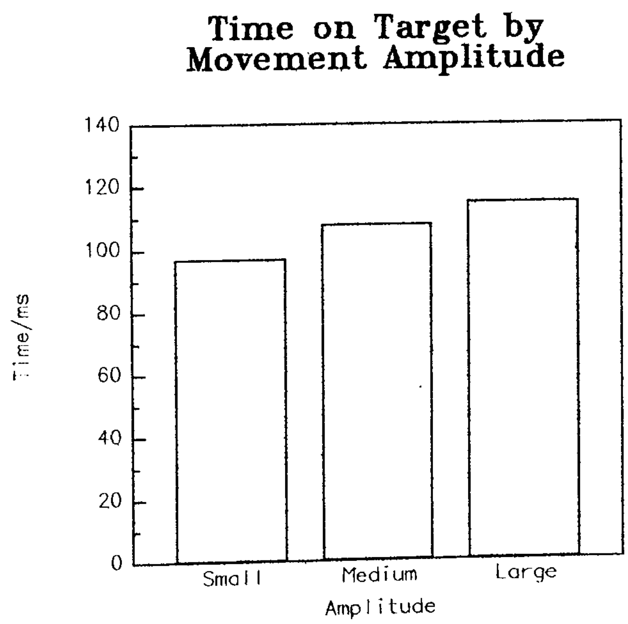


Figure 7.5b

The significant interaction between target size and movement amplitude (1.3% of total variance) suggests that for large targets large movement amplitudes are associated with greater periods of movement on the target. This was supported by post-hoc tests which found, for large target sizes, significantly greater periods of movement for the large amplitude condition when compared to the medium and small amplitude conditions ($P < 0.05$, T 23.2).

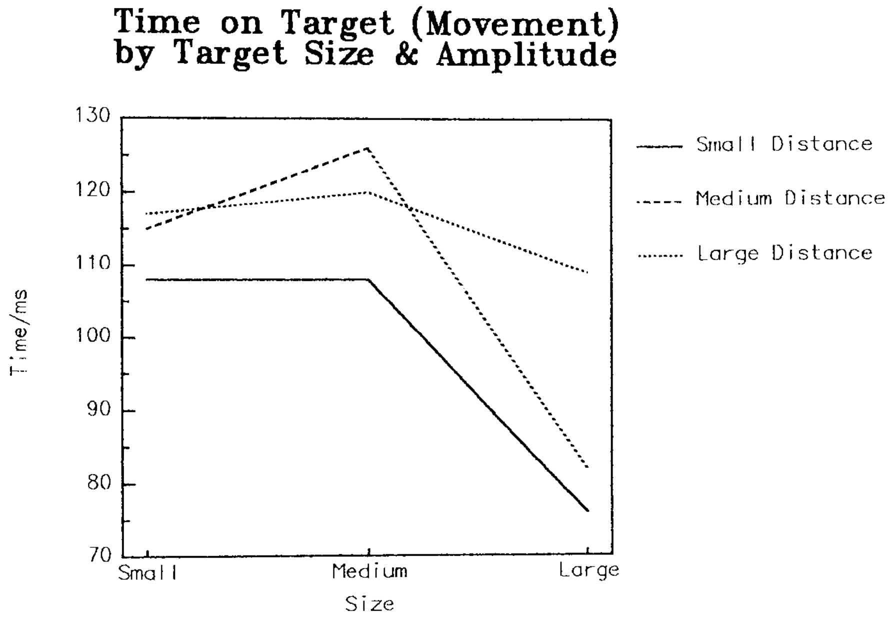


Figure 7.5c

7.2.2 Interim discussion.

Target size and movement amplitude were found to be strong influences on task performance as measured by task completion time. Target size was found to account for 33% of the total variance in task completion time and movement amplitude 22%. Given the larger range of movement amplitudes when compared to the range of target sizes, it is suggested that target size is relatively more important than movement amplitude in determining performance.

Although the different target sizes and movement amplitudes have been shown as equal intervals when plotted it should be borne in mind that the intervals between these groups (small, medium, and large) are in fact unequal (see section 6.1). Target sizes (diameters) approximately double for increasing size categories and the movement amplitudes increase five fold (from small to medium) and then three fold (from medium to large). If there were a linear relationship between target sizes, movement amplitudes and task completion time the graphs presented above would in fact show curvilinear relationships. This suggests that task completion time may have a non-linear relationship to target size and movement amplitude. This issue will be explored shortly.

Target size and movement amplitude accounted for relatively little of the variance in reaction time. If reaction time is taken to be an indication of task complexity (see section 3.3) then target size and movement amplitude are not good measures of such task complexity. However, the relatively longer reaction times for small and medium sized targets when the required movement amplitude is small does invite speculation regarding the underlying cognitive processes of movement execution. Why should reaction times be greater when required movement is relatively small, especially when the overall completion time for small targets requiring small movement amplitudes is disproportionately less? This question will be returned to later in this chapter.

Target size accounted for 22.8% of the total variance, and movement amplitude 47.8%, (a total of 70.6%) for movement time. Most theories of discrete motor acts would predict such task effects, if not the actual movement times.

The time on target was mostly accounted for by target size (36.8% of total variance) and was spent largely in non-movement. However, the increased time on target for the large movement amplitudes and large target size condition is probably accounted for by the additional time spent in movement on the target for this task condition. Movement time on target may have included the small 'jerk' that often occurred through the depression of the mouse buttons.

Why should people take longer to indicate their selection for smaller targets? Again we are invited to speculate upon the underlying cognitive processes of discrete movement. If we examine the times on target for the different target sizes (large 181ms, medium 252ms, and small 495ms) they do not exclude a decision based on visual feedback (see section 3.2.6). This question will be returned to shortly.

The effects of target size and movement amplitude on task completion time arise largely from the effects of target size and movement amplitude on the time spent in moving to the target and from the effect of target size on the time spent on target.

Tables 7.0a,7.0b, and 7.0c show the correlations between the earlier measures of task difficulty (Sections 2.4.1.10, 2.4.6, 3.2.1, 3.2.4, & 3.2.5) which provided formulations to predict movement time, and the actual movement times recorded within this study. Where A is the movement amplitude, W is the target width, r is the correlation coefficient, and r^2= variance. All correlations were found to be significant ($P<0.001$, N 432).

Predicting Movement Time

For all trials:

Equation	r	r^2
A	0.64	0.41
$A^{0.5}$	0.66	0.44
A/W	0.77	0.59
$(A/W)^{0.5}$	0.83	0.69
$\ln(2A/W)$	0.84	0.71
$\ln(A/W + 0.5)$	0.84	0.71
$\ln(A/W + 1.0)$	0.84	0.71

Table 7.0a

Predicting Movement Time

For low gain trials:

Equation	r	r^2
A	0.68	0.46
$A^{0.5}$	0.69	0.48
A/W	0.73	0.53
$(A/W)^{0.5}$	0.79	0.62
ln(2A/W)	0.81	0.66
ln(A/W + 0.5)	0.81	0.66
ln(A/W + 1.0)	0.81	0.66

Table 7.0b

Predicting Movement Time

For high gain trials:

Equation	r	r^2
A	0.61	0.37
$A^{0.5}$	0.62	0.38
A/W	0.82	0.67
$(A/W)^{0.5}$	0.87	0.76
ln(2A/W)	0.88	0.77
ln(A/W + 0.5)	0.88	0.77
ln(A/W + 1.0)	0.88	0.77

Table 7.0c

However, considering the predictive equation provided by Card, Moran, & Newell (based on Fitts' Law, see section 2.4.1.10) given earlier:

$$MT = 0.8 + 0.1(A/W + 0.5) \quad \text{Equation 7.0}$$

The coefficients provided by regression analysis for the data collected within this study yielded different values to those in the above equation. Namely:

$$MT = 0.11 + 0.33(A/W + 0.5) \quad \text{Equation 7.1}$$

Using the predictive values for the equation (Equation 7.0), rather than fitting the data to equation 7.1, resulted in a reduced r^2 of 0.36 compared to 0.71 obtained by using equation 7.0. The value of r^2 being derived from:

$$r^2 = 1 - \frac{\text{Error variance}}{\text{Total Variance}} \quad \text{Equation 7.2}$$

Adapted from Ferguson (1966, p126).

Where Error variance refers to the variance of the discrepancies arising between the predicted and actual movement times, and Total variance refers to the variance of the actual movement times. The predictive equation (7.0) overestimated the movement time by an average of 268 ms.

Thus we find a marked reduction (approx. 50%) in the predictive power of the equation when the coefficients are taken from previously determined empirical values rather than calculating them after the data has been collected. A further reduction in the predictive power of the Card et al formulation occurs when their suggestion to add 200ms to the constant term in their equation, where the button press is to be included within the movement time, is adopted. Using:

$$\text{Time (MT+Button press)} = 1 + 0.1(A/W + 0.5) \quad \text{Equation 7.4}$$

resulted in an r^2 of 0.24. This reduction was to be expected given that time on target was found to be task dependent, rather than being constant across the different task conditions. The equation overestimated the overall mean time by 51ms.

Including the reaction time in addition to the movement time (mt + rt) resulted in an r^2 of 0.29. using the predictive equation (7.0). The equation underestimated the mean time by 419ms.

The different gain conditions had no main effect on task completion time, although small target sizes were disadvantaged by the high gain condition. However, movement time was quicker for the medium and large target sizes in the high gain condition. This advantage was offset by the increased time spent in non-movement on the target for the high gain condition. The disadvantage for small targets in the high gain condition would appear to be accounted for by the increased time spent in non-movement within the target area.

Why should the time on target spent in non-movement be different between the different gain conditions? One possibility which also applies to the earlier question of time on target is parallel processing. If the decision to press the button could be anticipated for some movements prior to their termination, then the time on target for such movements would be less than for those movements not anticipated to terminate correctly. The likelihood of a correct termination would depend upon the ability of the subject to anticipate the movement's correct termination which would be influenced, in part, by the subject's previous movement experience, and the size of the target area. Subjects may be unfamiliar with such a high gain for the mouse device and a consequence of an increase in gain is a reduction in the size of virtual target area. One of the consequences of the higher gain condition was a relatively smaller virtual target area when compared to the low gain condition. For smaller screen target areas it is also possible that the cursor itself, or the location of its tip, might be more difficult to distinguish against the background of the target, and target number.

7.3 Submovements by target size, movement amplitude, and gain

The following subsections are concerned with the incidence and types of movement which occurred, and how they related to the differing task conditions.

7.3.1 Number of submovements

There were significant main effects of target size (38.5% of the total variance), movement amplitude (24.2% of the total variance) and gain (3% of the total variance). Reducing the target size, increasing the movement amplitude, and the high gain condition, are associated with an increase in the number of movements made to reach the target.

Figures 7.6a, 7.6b and 7.6c show the significant group means.

Post-hoc tests showed that the mean number of movements significantly differed between all target sizes ($P < 0.01$, Tukey T 0.46), and all movement amplitudes ($P < 0.01$, Tukey T 0.39).

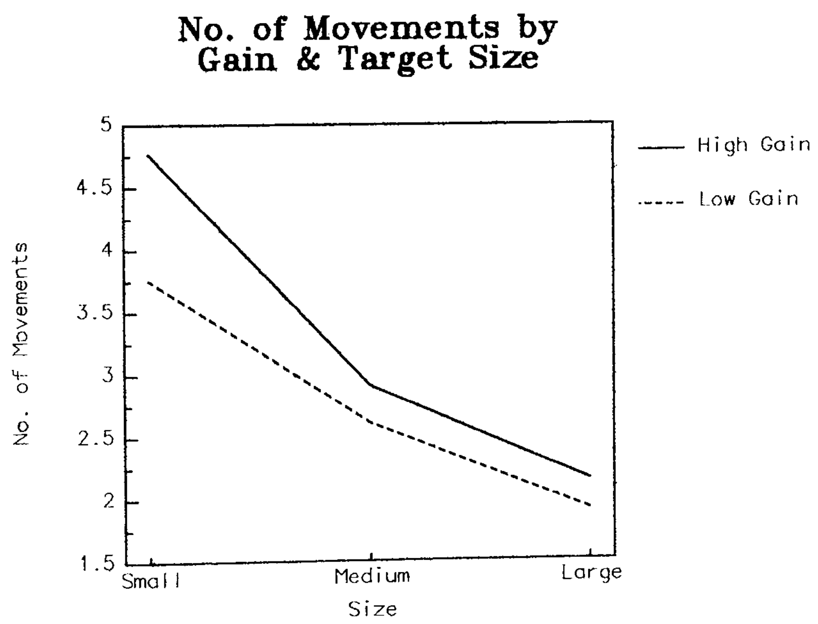


Figure 7.6a

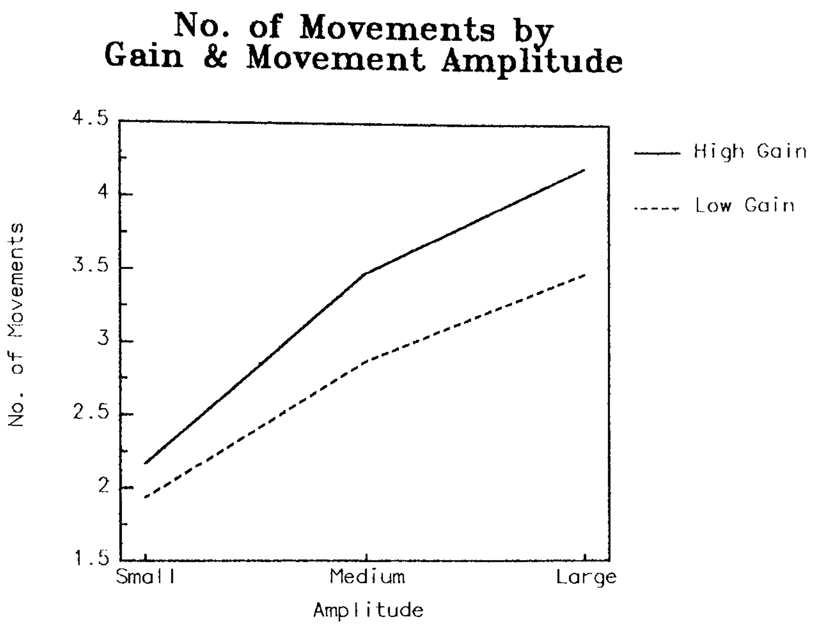


Figure 7.6b

The significant interaction between target size and movement amplitude accounts for 1.6% of the total variance and suggests that a relatively low number of movements are required for the small amplitude trials. The gain by target size interaction (1.4% of the total variance) suggests that for small targets the high gain condition results in a disproportionately higher number of movements ($P<0.01$, Tukey T 0.54).

The significant three-way interaction between target size, movement amplitude, and gain accounts for only 0.2% of the total variance, and will not be considered any further.

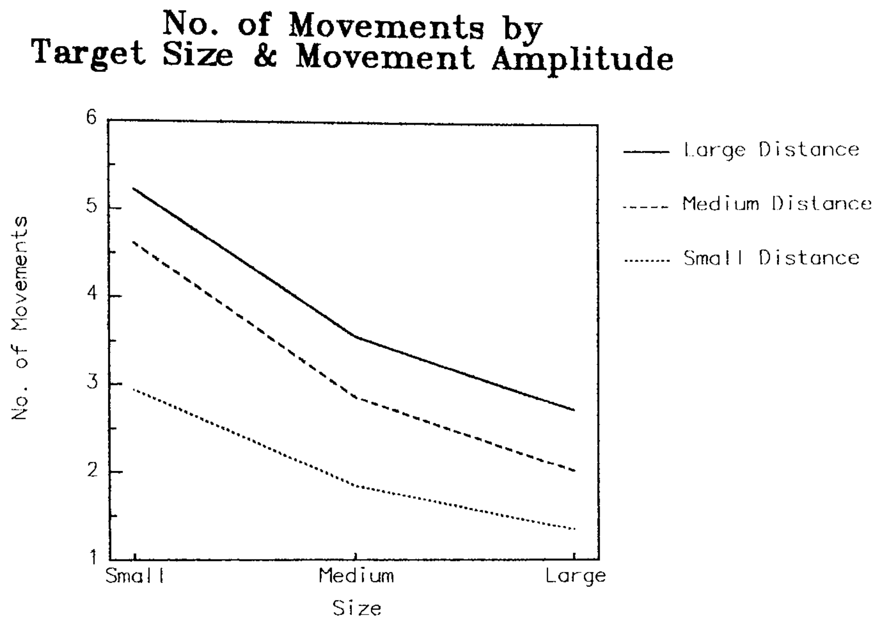


Figure 7.6c

7.3.2 The nature of the movements

Movements were either identified as complex or simple. Complex movements were those identified as having a movement correction during its execution and simple movements as those that showed no evidence of such corrections. The proportion of complex movements within the total number of movements was calculated for each contingency of target size, movement amplitude, and gain.

Significant main effects of target size (18% of the total variance), movement amplitude (11.9% of the total variance), and gain (19.8% of the total variance) were found on the relative frequency of complex movements. Increasing target size, increasing movement amplitude, and low gain condition were associated with relatively more frequent occurrences of complex movements. Post-hoc tests found significant differences between all target sizes (small 20.9%, medium 28.8%, and large 36.2%, P<0.01, Tukey T 6.3), and all levels of movement amplitude (small 22.1%, medium 29.2%,, and large 34.5%, P<0.05, Tukey T

4.4). The low gain condition had a mean frequency of 35.2% compared to 22.0% for the high gain condition.

7.3.4 Number of non-movements

There were main effects of gain (8.8% of the total variance), target size (52.1% of the total variance), and movement amplitude (3.8% of the total variance). The number of non-movements increased for smaller target sizes, the higher gain condition, and larger movement amplitudes. The differences between all target size means (P<0.01, Tukey T 0.54) and movement amplitude means (P<0.01, Tukey T 0.24) were significant.

Figures 7.7a, 7.7b, and 7.7c show the means for the significantly different groups.

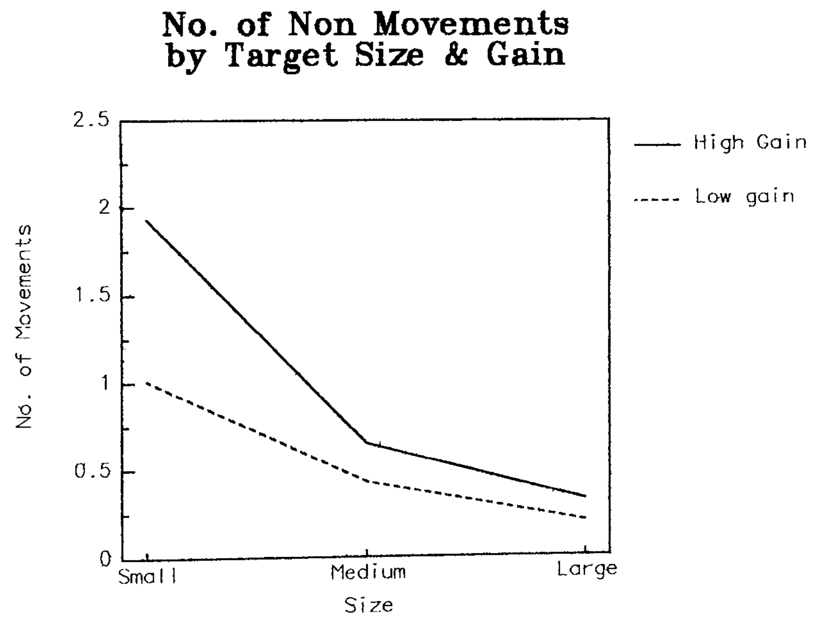


Figure 7.7a

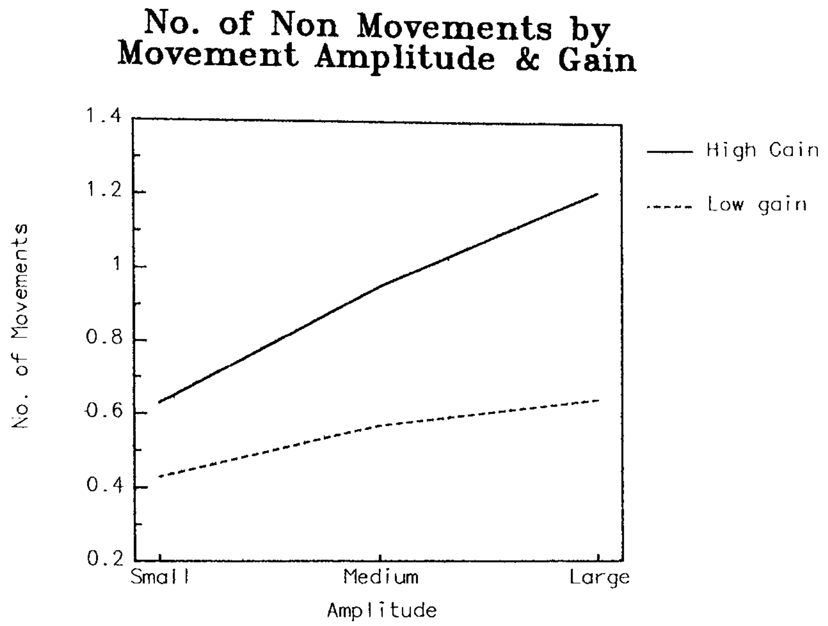


Figure 7.7b

The significant interaction between target size and movement amplitude (0.8% of the total variance) suggests that the differences between the movement amplitude means become larger for the small target size condition.

The significant interaction between gain and target size (6% of the total variance) suggests that the number of non-movements for the small target size increases disproportionately for the high gain condition (P <0.01, Tukey T 0.28)

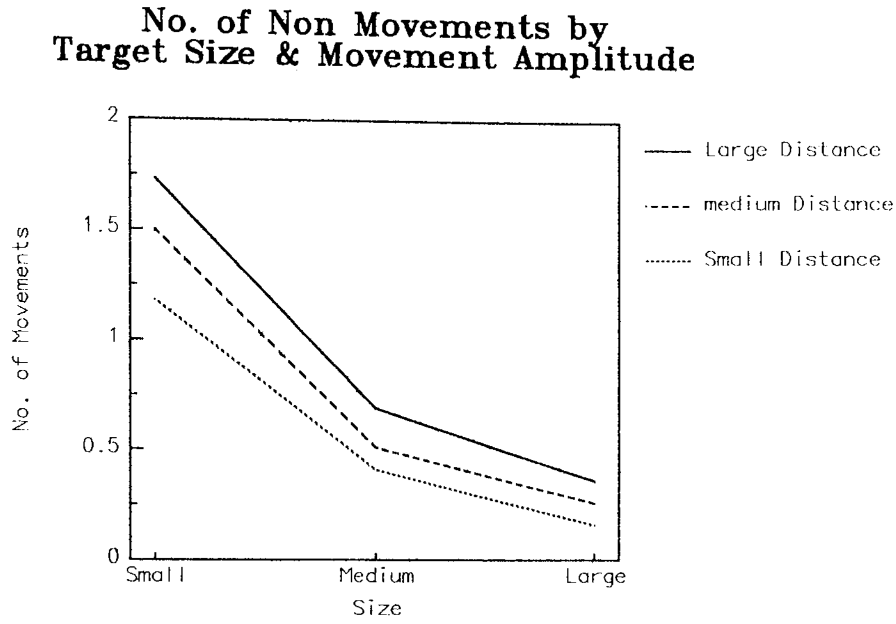


Figure 7.7c

The significant interaction between movement amplitude and gain (0.5% of the total variance) suggests that as movement amplitude increases the number of non-movements in the high gain condition disproportionately increases.

7.3.3 Interim discussion

Increasing the movement amplitude or reducing the target size leads to more movements being made, and an increase in the incidence of non-movements. Target size predominantly accounted for the number of non-movements. This may have been due to the reduced target area interfering with the ability to preplan the subsequent movement whilst executing the previous movement. Alternatively the uncertainty of a correct movement termination for the smaller targets could have led to an episode of non-movement if the terminating movement missed the target area. 'Holes in one' were most likely for the small movement amplitudes when large and medium target sizes are presented. However, three or more movements are often required for the large amplitude movements, and the small target sizes. One or more occurrences of non-movement are likely for the small target condition.

Increasing the movement amplitude, the low gain condition, and increasing the target size lead to an increase in the relative frequency of complex movements. The overall mean incidence of complex movements is lower (28.6%) than would be expected if every trial (on average 3.02 moves) were to have at least one complex movement.

Given that movement amplitude accounted for a higher percentage of the variance in movement time than target size, and that target size accounts for a larger percentage of the variance in the number of movements and non-movements, then the velocity of the movements is most likely to also be influenced by the movement amplitude.

Although there is a main effect of gain on the number of movements (high gain condition 3.28 moves, low gain condition 2.76 moves) and the number of non-movements (high gain condition 0.97, low gain condition 0.55) the actual differences are relatively small for the medium and large target sizes. The need for fewer movements in the low gain condition may arise from these movements benefiting from the relatively higher number of movement corrections found in the low gain condition.

The advantage of the high gain condition for movement time is probably accounted for by the movements being executed more quickly in this condition (shorter durations) despite the small increase in the number of non-movements and movements. However, when the small targets are considered the difference between the mean number of movements and non-movements for the different gain conditions becomes significantly greater (movement: high condition 4.77, low condition 3.75, non-movement: high condition 1.93, low condition 1.01), and the 'on average' additional movement and non-movement is sufficient to offset the time advantage of the high gain condition.

The following sections consider the 1st and 2nd movements in more detail.

7.3.5 The 1st movement

7.3.5.1 Duration

There were significant main effects of gain (20% of the total variance), target size (2.6%), and movement amplitude (48.4%). Increasing the target size, increasing the movement amplitude, and the low gain condition, are associated with longer 1st movement durations.

Post-hoc tests revealed that the large target size mean was significantly different from those of the small and medium sized targets ($P < 0.05$, Tukey T 23.2), and the movement amplitude means all significantly differed ($P < 0.01$, Tukey T 35.0).

Figures 7.8a and 7.8b show the means for the significantly different groups.

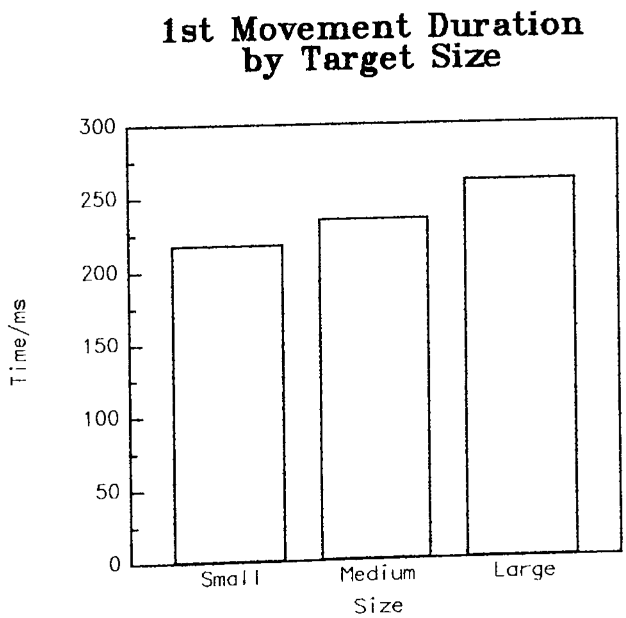


Figure 7.8a

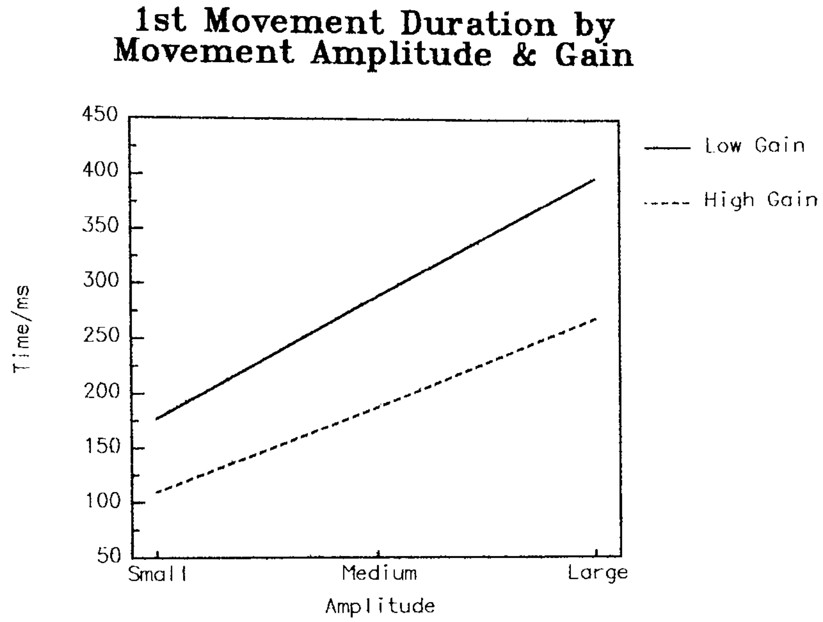


Figure 7.8b

The significant interaction between gain and amplitude (1.3% of total variance) suggests that the difference between the low and high gain condition increases for increased target size. The gain means significantly differed for all movement amplitudes (P < 0.01, Tukey T 40.9).

7.3.5.2 Velocity

Significant main effects of gain (16.4% of the total variance), target size (2.7% of the total variance), and movement amplitude (49.3%) on 1st movement velocity were found. Increasing target size, increasing the movement amplitude, and the low gain condition resulted in faster 1st movements.

Figures 7.9a and 7.9b show the means for the significantly different groups.

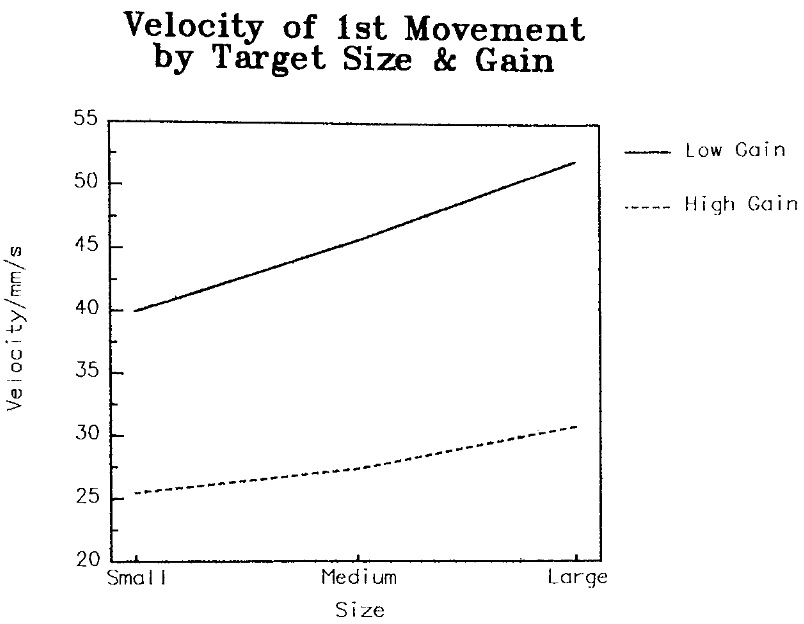


Figure 7.9a

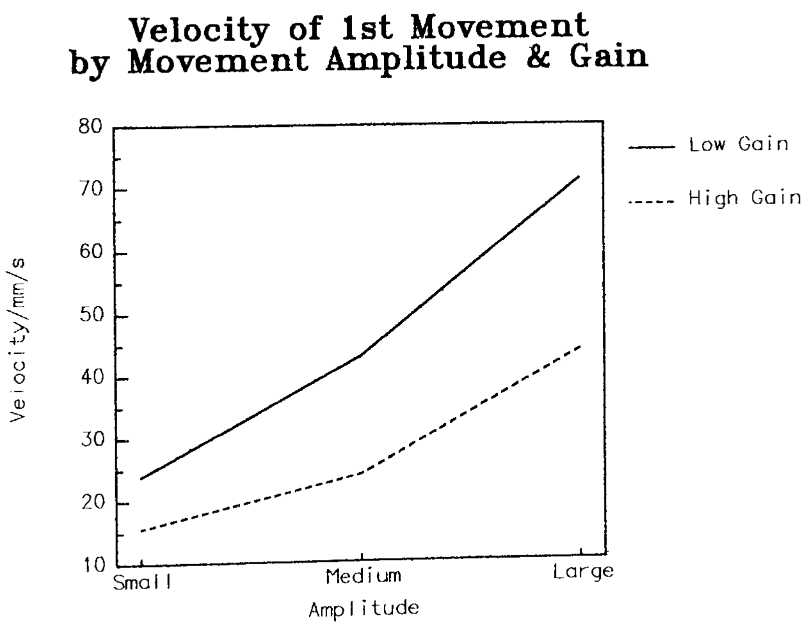


Figure 7.9b

Post-hoc tests revealed that the large target size mean was significantly different from those of the medium and small target sizes ($P < 0.05$, Tukey T 3.85), and that all movement amplitude mean velocities were significantly different ($P<0.01$, Tukey T 7.08).

The significant interaction between gain and target size (0.7% of the total variance) suggests that the difference in velocity between the high and low gain conditions increased for the large target size. The differences between gain mean velocities for all levels of target size were significant ($P<0.01$, Tukey T 5.65).

The significant interaction between gain and movement amplitude (2.7% of the total variance) suggests that the difference in velocity between the high and low gain conditions increases with increasing movement amplitude. The differences between gain means for all levels of movement amplitude were significant ($P<0.01$, Tukey T 8.25).

7.3.5.3 Effectiveness

The main effect of target size on 1st movement effectiveness (9.8% of the total variance) was such that increasing the target size increased the movement's effectiveness. The mean effectiveness for the small targets(49.4%) significantly differed from those of the medium (62.5%) and large (71.4) target sizes ($P < 0.05$, Tukey T 13.0).

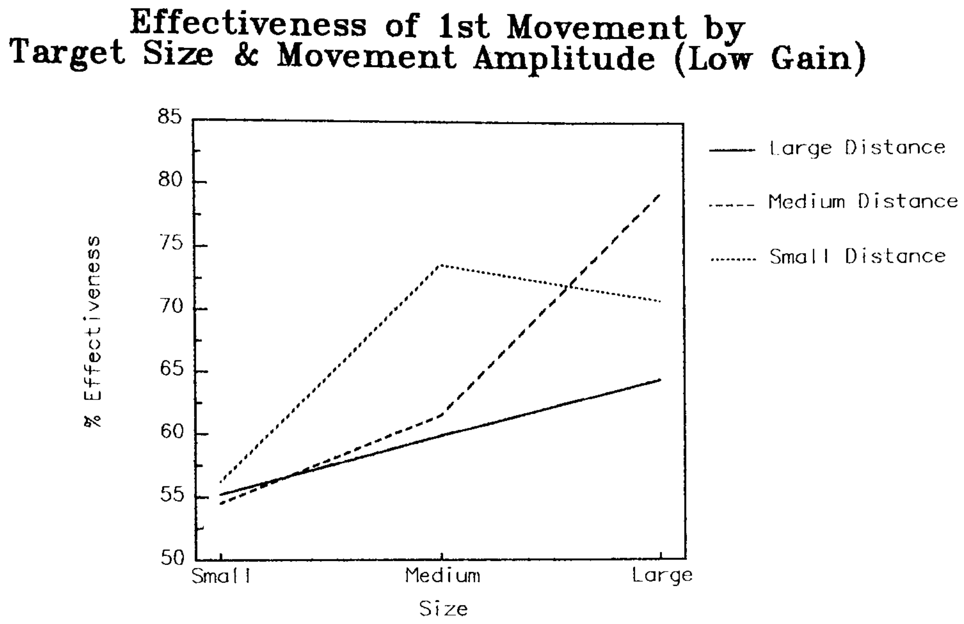


Figure 7.10a

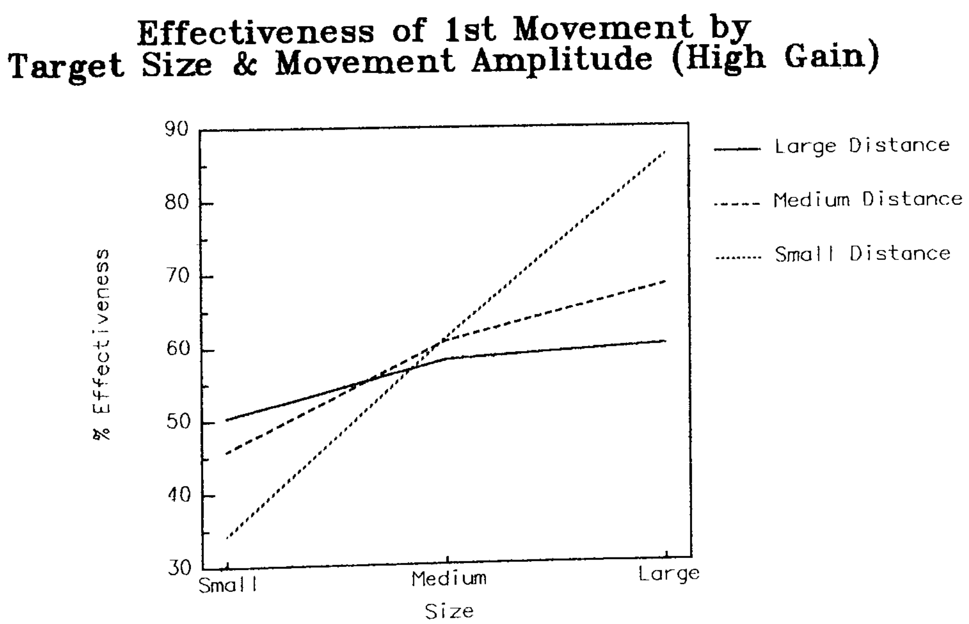


Figure 7.10b

The significant three-way interaction between gain, amplitude, and size (2% of the total variance) are illustrated in Figures 7.10a and 7.10b.

Considering the low gain condition none of the movement amplitude means significantly differ for the small, medium, and large target sizes ($P>0.05$, Tukey T 17.1). However, considering the high gain condition, for the large target size, the small movement amplitudes are significantly more effective than the medium or long movement amplitudes ($P<0.05$, Tukey T 17.1). This may reflect the higher incidence of holes in one associated with this task condition.

7.3.5.4 Errors of extent

The errors of extent were predominantly due to undershoots of the target rather than overshoots.

Significant main effects of target size (11.1% of the total variance) and movement amplitude (6.5% of the total variance) were found. Increasing the target size and reducing the movement amplitude resulted in smaller errors of extent.

Post-hoc tests found a significant difference between the error mean of the small target and those of the medium and large target conditions ($P<0.05$, Tukey T 9.8), and the small movement amplitude error mean was significantly different from those of the medium and large movement amplitude means ($P <0.05$, Tukey T 9.8).

The significant interaction between target size and movement amplitude (2.2% of the total variance) suggests that for medium sized targets the small movement amplitudes' errors of extent are significantly less than those for medium or large movement amplitudes, and that for large sized targets the smaller movement amplitudes have significantly smaller errors of extent than the large movement amplitudes ($P<0.05$, Tukey T 11.2).

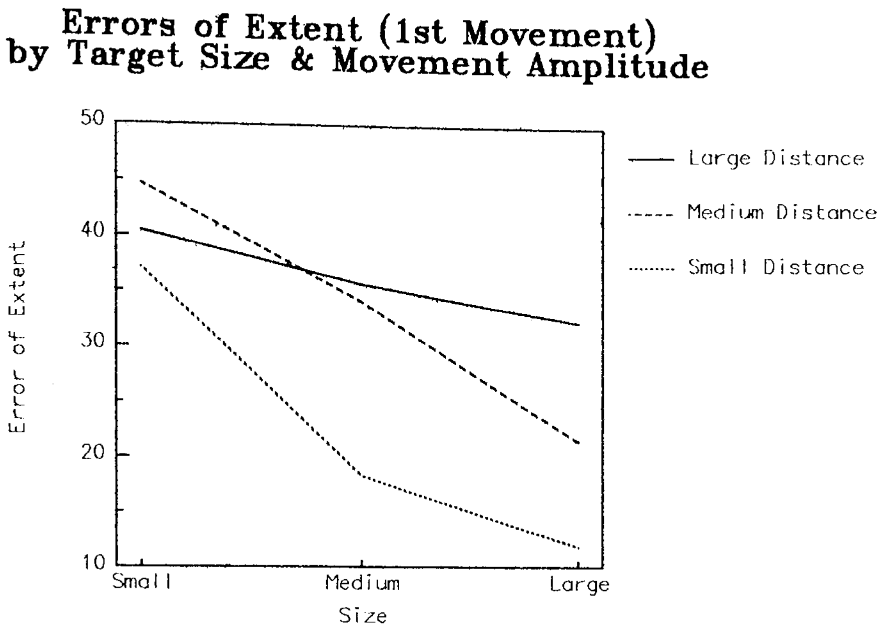


Figure 7.11

7.3.5.5 Errors of aim

Significant main effects of target size (5.9% of the total variance) and movement amplitude (9.4% of the total variance were found). Increasing the target size and reducing the movement amplitude resulted in reduced errors of aim. Figures 7.12a and 7.12b show the significantly different means.

Post-hoc tests showed the mean error of aim for small targets to be significantly different to that of the large targets ($P<0.05$, Tukey T 3.52), and the large movement amplitude mean error of aim to be significantly different for those of the medium and short movement amplitude ($P<0.05$, Tukey T 3.42).

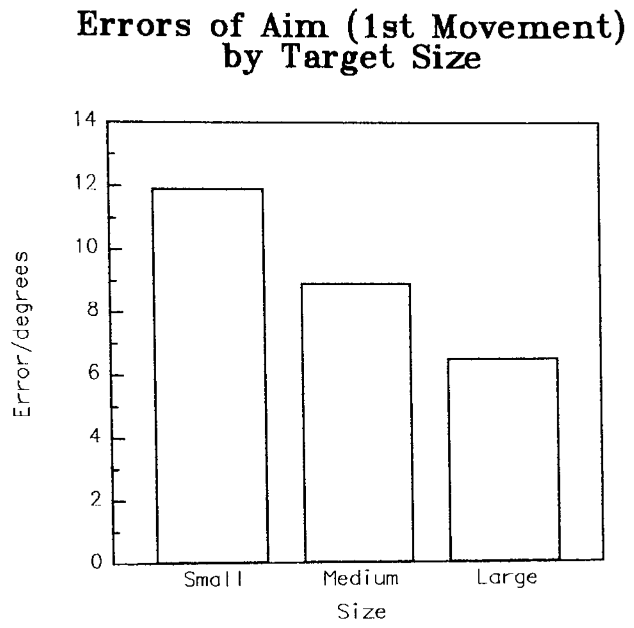


Figure 7.12a

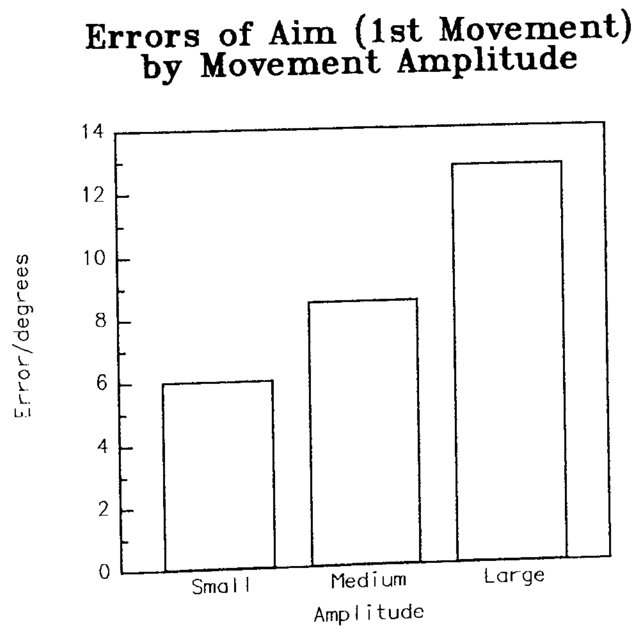


Figure 7.12b

Percentage of Error in Effectiveness Accounted for by Error in Extent			
	Movement Amplitude		
	small	Medium	Large
Target size			
Small	72%	82%	89%
Medium	91%	93%	94%
Large	89%	92%	92%

Table 7.1

7.3.5.6 Interim discussion

Larger movement amplitudes, and the low gain condition, produced faster 1st movements. Large target sizes were also associated with faster 1st movements but target size accounted for much less of the variance in movement velocity than either movement amplitude or gain.

The duration of the 1st movement increased with increasing target size, increasing movement amplitude, and the low gain condition. These conditions were also conducive to movements with corrections being made. Tables 7.2a & 7.2b show how well the 1st movement's duration (averaged grouped data) correlated with various measures of task difficulty; where A is movement amplitude, and W is target width. All correlations, with the exception of target width, were significant ($P<0.01$, $N = 216$).

In both the high and the low gain conditions the first movement's duration was mostly accounted for by movement amplitude with the best formulation (as determined by the largest r^2 value - McNeil, Kelly, & McNeil, 1975, p302) being the square root of the movement amplitude. The variance accounted for by the square-root of movement amplitude increased in the high gain condition (from 61% to 72%).

Predicting 1st Movement Duration

For low gain trials:

Equation	r	r^2
W	0.18	0.03
A	0.76	0.58
$A^{0.5}$	0.78	0.61
A/W	0.47	0.22
$(A/W)^{0.5}$	0.53	0.28
ln(2A/W)	0.56	0.31
ln(A/W + 0.5)	0.56	0.31
ln(A/W + 1.0)	0.56	0.31

Table 7.2a

Predicting 1st Movement Duration

For high gain trials:

Equation	r	r^2
W	0.19	0.03
A	0.83	0.69
$A^{0.5}$	0.85	0.72
A/W	0.49	0.24
$(A/W)^{0.5}$	0.57	0.32
ln(2A/W)	0.61	0.37
ln(A/W + 0.5)	0.61	0.37
ln(A/W + 1.0)	0.61	0.37

Table 7.2b

If the distribution of endpoints is to be taken as an indication of the intended aim of the movement then the 1st movements were aimed at a point located halfway, to three quarters, of the distance from the start point to the target's edge. Movements consistently undershot the target. This may have arisen due to the task conditions and or the experimental instructions. The targets were located at the corners of the screen so that overshoots could have resulted in the loss of immediate screen pointer feedback (the device position was still recorded but the cursor was held at the edge until the device position had become displayable on the screen again); or for the low gain condition, could have taken the device off the mouse pad which would have resulted in the mouse being lifted and the need for the trial to be repeated.

The 1st movement's effectiveness was greater for increased target size, but no significant main effect of movement amplitude was found. However, significant interactions between movement amplitude, target size and gain were found. Inspection of the associated graphs did not reveal an easily interpretable pattern of effects. Larger target sizes were associated with higher velocities and longer 1st movement durations which would have resulted in more distance being covered by the movement and an increase in the movement's effectiveness.

Analyses of the relative contributions of errors of aim and extent to the movement's effectiveness (table 7.1) indicated that the 1st movement's effectiveness, or the lack of it, was mostly attributable to errors of extent rather than errors of aim. However, it can be seen that the effectiveness of smaller movement amplitudes was influenced relatively less by errors of extent than was the case for the larger movement amplitudes.

7.3.6 2nd Movement

7.3.6.1 Duration

There were significant main effects of gain (11% of the total variance), and movement amplitude (30% of the total variance). Increasing the movement amplitude, and the low gain condition, are associated with longer 2nd movement durations.

Post-hoc tests revealed that the movement amplitude means all significantly differed (P<0.01, Tukey T 37.1).

Figures 7.13a and 7.13b show the means for the significantly different groups.

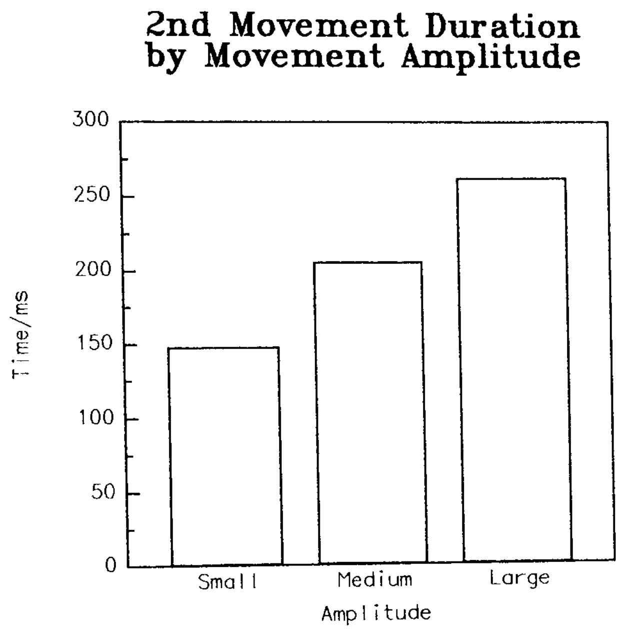


Figure 7.13a

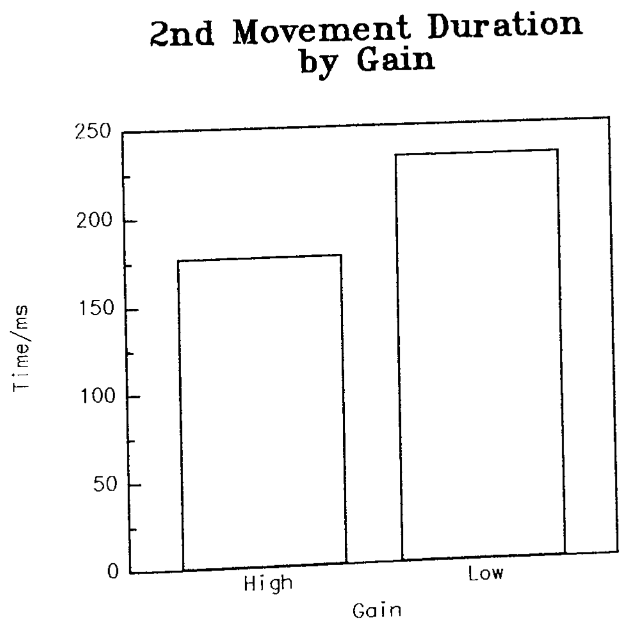


Figure 7.13b

Tables 7.3a & 7.3b show how well the 2nd movements' duration (averaged data) correlated with various measures of task difficulty. Where A is movement amplitude, and W is target width. All correlations, with the exception of target width, were significant ($P<0.01$, $N = 216$).

Predicting 2nd Movement Duration

For low gain trials:

Equation	r	r^2
W	0.06	>0.01
A	0.56	0.32
$A^{0.5}$	0.57	0.33
A/W	0.35	0.12
$(A/W)^{0.5}$	0.40	0.16
ln(2A/W)	0.44	0.19
ln(A/W + 0.5)	0.44	0.19
ln(A/W + 1.0)	0.44	0.19

Table 7.3a

Predicting 2nd Movement Duration

For high gain trials:

Equation	r	r^2
W	0.07	>0.01
A	0.59	0.35
$A^{0.5}$	0.61	0.37
A/W	0.35	0.12
$(A/W)^{0.5}$	0.42	0.18
ln(2A/W)	0.49	0.24
ln(A/W + 0.5)	0.48	0.23
ln(A/W + 1.0)	0.47	0.22

Table 7.3b

As was the case for the 1st movement's duration the 2nd movement's duration is best accounted for by the formulation based on the square-root of the movement amplitude with the higher correlations occurring in the high gain condition. However, the variance accounted for by this expression in the second movement's case is not so high.

7.2.6.2 Velocity

Significant main effects of gain (19.2% of the total variance), target size (3.9% of the total variance), and movement amplitude (39.7%) on 2nd movement velocity were found. Increasing target size, increasing the movement amplitude, and the low gain condition resulted in faster 2nd movements.

Figures 7.14a, 7.14b, and 7.14c show the means for the significantly different groups.

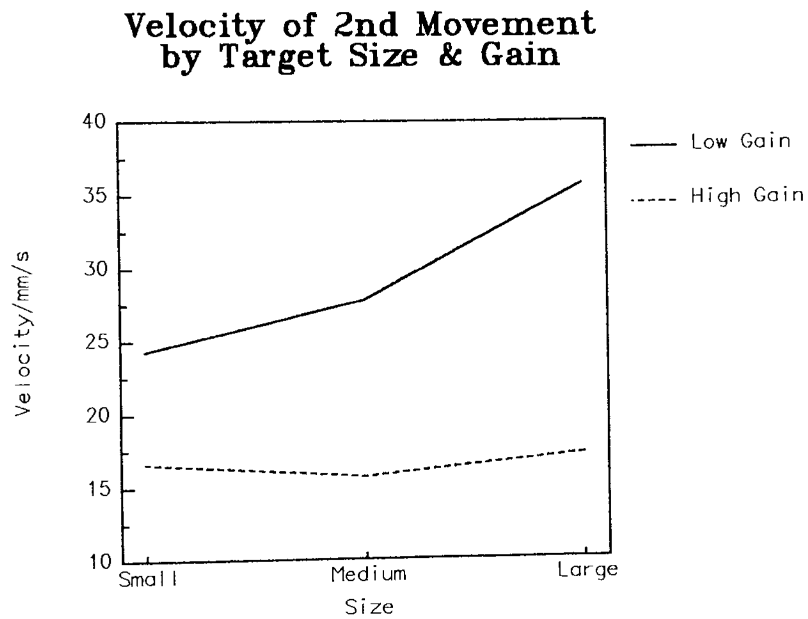


Figure 7.14a

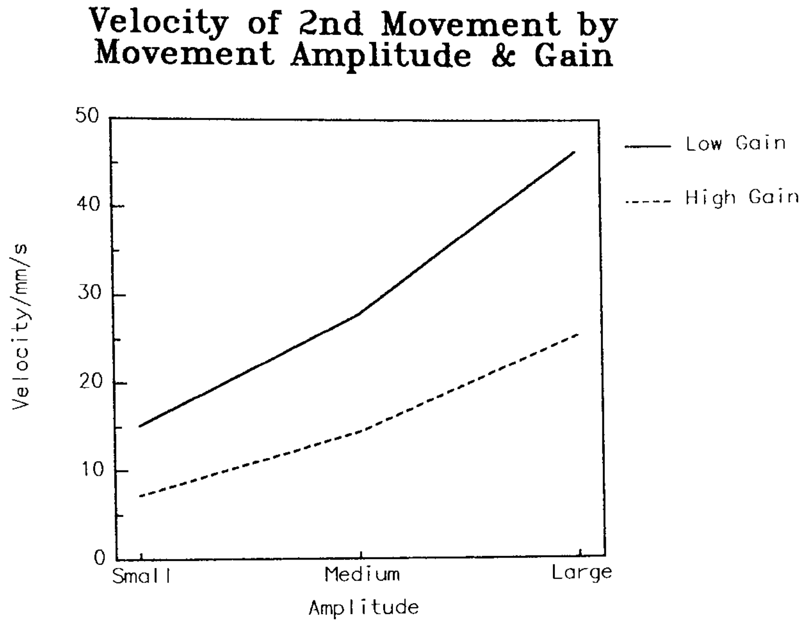


Figure 7.14b

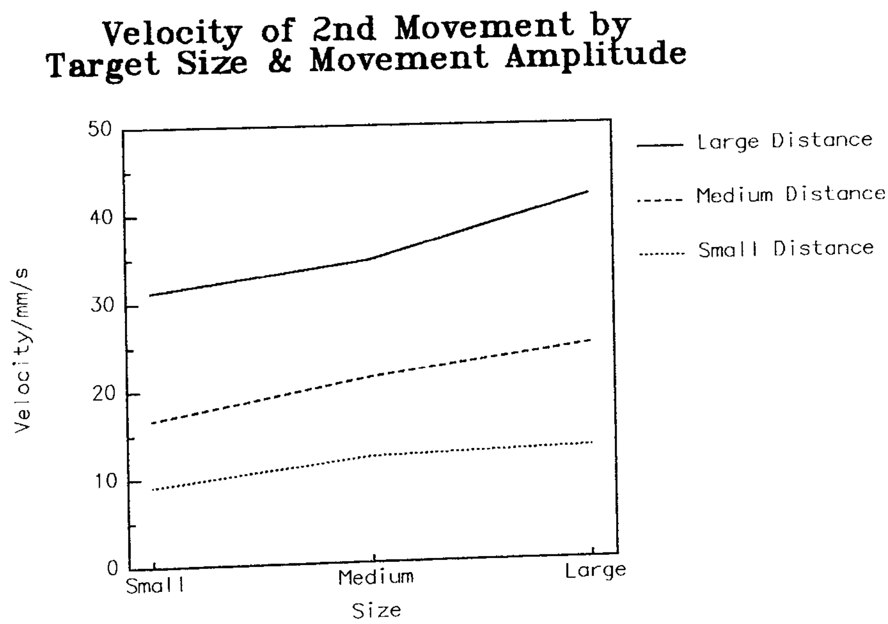


Figure 7.14c

Post-hoc tests revealed that the large target size mean was significantly different from those of the medium and small target sizes ($P < 0.05$, Tukey T 3.52), and that all movement amplitude means were significantly different ($P<0.01$, Tukey T 4.4).

The significant interaction between gain and target size (1.3% of the total variance) suggests that the difference in velocity between the high and low gain conditions increased for the large and medium target sizes. The differences between gain means for all levels of target size were significant ($P<0.01$, Tukey T 4.31).

The significant interaction between gain and movement amplitude (2.6% of the total variance) suggests that the difference in velocity between the high and low gain conditions increases with increasing movement amplitude. The differences between gain means for all levels of movement amplitude were significant ($P<0.01$, Tukey T 6.5).

The significant interaction between target size and movement amplitude (1.3% of the total variance) suggests that the velocity is disproportionately faster for the large target condition with large movement amplitudes.

7.3.7 Comparing 1st & 2nd movements

The first and second movements (when they occurred) were compared for movement duration and movement effectiveness. Second movements (mean duration 205.00 ms) were found to have significantly shorter durations than first movements (mean duration 237.63ms) when all trial values were pooled ($P < 0.001$, t 4.67, DF 47). However, movement effectiveness was not found to significantly differ between 1st and 2nd movements.

7.3.8 Overall discussion for the influence task factors on device movement.

7.3.8.1 Target angle

The lack of any significant differences in task completion time due to differences in the target angle is consistent with those findings of some previous studies (Lin et al, 1992, and Card et al, 1978). This may arise from no such effects occurring, or may be due to an inadequate classification of target angle. The angle classification selected (vertical, horizontal & diagonal) was selected as it had yielded significant differences in the past (Radwin et al, 1990, Trankle & Deutschmann, 1991, and Barker et al, 1990). However, the slower acquisition times for targets located at the bottom of the screen, and people's reluctance to overshoot the target area, might indicate that movements made towards the body are slower than those made away from the body. Unfortunately a reanalysis of the experimental data using such a classification is not easily achieved; although it does suggest that such an analysis would be worthwhile at some later date.

7.3.8.2 Target location

The finding that targets located at the bottom of the screen took longer to acquire than those at the top of the screen has important implications for the layout of fixed menu systems employing a similar task to that employed within this study. However the presentation of the user's task details at the top of the screen may limit the applicability of this finding to other mouse input tasks. If the additional time taken to select targets at the bottom of the screen area was accounted for by the location of the task details at the top of the screen this would restrict the generalizability of this finding. Further analysis of these experimental data would reveal where the time differences for the target locations occurred (reaction time, movement time, or selection time). This information could help to determine the likelihood of these findings generalizing to other input tasks using a mouse. This analysis could be carried out at a future time.

7.3.8.3 Target size, movement amplitude, and gain

The conclusion of Trankle & Deutschmann (1991) and Lin et al (1992) that movement amplitude and target size have the strongest influence on task completion time was supported here. Reducing the movement amplitude, or increasing the target size, results in shorter task completion times.

Reducing the movement amplitude and increasing the target size magnitudes results in a non-linear reduction in movement time, such that a 'Law of Diminishing Returns' would appear to apply for such changes. Thus in considering task completion time the advantage gained by increasing the target size from 6mm to 15mm diameter is less than the corresponding advantage achieved by increasing the target size from 15mm to 30mm diameter.

The effect of differing gain conditions on task completion time, or the lack of such effects, was consistent with previous findings (Lin et al, 1991, and Trankle & Deutschmann, 1991). However, within this study, and in that of Lin et al, the differences in target acquisition times were found to be task dependent. In particular users were disadvantaged for small size targets in the high gain conditions. Trankle & Deutschmann (1991) also reported that users were disadvantaged for large movement amplitudes and a low gain condition. This was not found to be the case here despite a bigger movement amplitude (106mm compared to 100mm) condition being used within this study. However, Trankle & Deutschmann did utilize a lower value of gain (1 compared to 2) in their study which would have demanded a larger device movement amplitude for their low gain/large screen amplitude condition.

Table 7.4 shows a summary of the findings presented earlier. It shows the significant main effects, or the lack of them, for the factors of movement amplitude, target size, and gain.

Relative explanatory power of different task factors in the variance of the different dependent measures

Task factor	Amplitude	Target size	Gain
Dependent measure			
Total time	22%	33%	ns
Reaction time	1%	1%	ns
Movement time	48%	23%	ns
Time on target	1%	37%	8%
Time on target (non movement)	>1%	42%	11%
Time on target (movement)	2%	5%	ns
Number of movements	24%	39%	3%
Complex or simple?	12%	18%	20%
Number of non movements	4%	52%	9%
Duration of 1st move	48%	3%	20%
Velocity of 1st move	49%	3%	16%
Effectiveness of 1st movement	0%	10%	ns
Errors of extent	7%	11%	ns
Errors of aim	9%	6%	ns
Duration of 2nd move	30%	ns	11%
Velocity of 2nd move	40%	4%	19%

ns - not significant.

Table 7.4

To suggest that there are little or no differences in performance due to differing gain conditions is clearly erroneous. However, the differences are such that their combined influences are largely counterbalanced in their effects on task completion time for all but the small sized targets.

It has been suggested that the lack of an effect of system gain on task completion time arises from a trade-off between the movement time to the vicinity of the target and the time spent in final location of the target (see Trankle & Deutschmann, 1991 for a brief review). However, this study has shown that for medium and large sized targets the trade-off has arisen from the movement time and the time to press the button once the pointer was located on the target, and that for small sized targets in a high gain condition the trade-off was unequal and resulted in longer overall task completion times.

The 1st and 2nd movements (for the same screen target size and movement amplitude) made under the high gain condition were slower, of a smaller movement amplitude, and had a shorter movement duration than those made for the low gain condition. Although not all of the movements were analysed it is assumed that these differences would also be found for any additional movements. It is suggested that the shorter duration of the terminating movements for the high gain condition resulted in the movements being less predictable than was the case for the relatively longer duration of the movements in the low gain condition.

Inspection of Table 7.4, and the figures presented earlier, suggest that the differences attributable to the different gain conditions are often consistent with those that would be expected for the corresponding changes in the required device movement amplitude, or the virtual mouse pad target dimensions as a consequence of the changes in gain. Thus we find that movement durations and velocities which increase with larger movement amplitudes are also associated with the lower gain condition which increases the movement amplitude required of the mouse device. The number of non-movements, and the time spent on target, both increase for reduced target sizes, which is also found to be the case for the high gain condition. When the target size and movement amplitude have approximately equal influences on a parameter, and the direction of their effects the parameter are such that they are opposed (e.g. completion time - faster times are associated with large targets and small movement amplitudes) then different gain conditions will have little, or no overall effect on the parameter. Conversely, when the parameter is one for which the directions of effect (for size and amplitude) are the same, then gain will also have an effect (e.g. the incidence of complex movements).

Given the associated conceptual difficulties arising from the notion of an effect of gain (see section 2.4.2.3.2) the suggestion that the differences in performance found within this study, for the different gain conditions, may be considered solely in terms of the corresponding changes in the device parameters (virtual target sizes and device movement amplitudes) is intended as an aid to our understanding of these findings rather than making a potentially refutable statement.

Device 'accelerators' (see section 2.4.1.6) which are intended to provide the device with a higher device gain for relatively fast movements and a lower device gain for relatively slow movements, attempt to minimize the movement time to the target's vicinity (high gain), and then to reduce the problems associated with target location (low gain). However, the findings here suggest that the disadvantage of the high gain condition is not target location (except perhaps for very small targets), but in being able to predict the correct termination of a movement and consequently anticipate the button press. Given these findings it is not surprising that Trankle & Deutschmann (1991) found the use of an accelerator to be counter-productive rather than improving the task completion time.

7.3.8.4 Fitts' Law and its ilk

Formulations such as those based on Fitts' Law were found to provide reasonably good post-hoc descriptions of the movement time, and or the movement and button selection times. However they were not so effective for providing a priori predictions of such times when assumptions have to be made regarding certain unknown coefficients. The values of these coefficients are largely determined by task specific factors, such as the number of target choices, the time required to conduct an appropriate visual search, and the time required to determine the appropriate action. Suggesting values for these coefficients would require extensive task analysis which still might not yield a highly predictive equation.

Fitts' Law predicts that there should be no effect of system gain on overall movement time (Trankle &

Deutschmann, 1991). However, the results found here suggest that such a lack of influence of different gain conditions will be task dependent. In particular this would not be true for a high gain condition with relatively small targets.

7.3.8.5 The Iterative Corrections Model

This model predicts that movements will consist of discrete submovements of a constant duration (approximately 200ms) and relative accuracy (see section 3.2.2). Discrete submovements were found, but when comparing first and second movements their durations were found to be significantly different.

The duration and relative accuracy of the first movement are predicted to be independent of the task demands, but this was not found to be the case. This model does not give an account for the occurrence of non-movement periods. Other studies have found similar findings which contradict the assumptions of the model (see section 3.2.2).

However, the findings here are consistent with those found by Crossman & Goodeve (1963), and although they assumed the submovement durations to be equal they did not test this hypothesis. They found that the submovements covered a fixed proportion of 50% of the distance to the target centre. A similar value was found here, although its magnitude was found to be task dependent. Similarly, they reported episodes of non-movement within the target acquisition period. Their description of movement as consisting of separate impulses of reducing amplitude, of roughly equal durations would describe the findings of this study.

7.3.8.6 The Stochastic Optimal Submovement Model

This model predicts that each trial consists of one or two submovements (and for particularly difficult tasks tertiary movements) whose durations are proportional to the square-root of the ratio of movement amplitude to target size, and whose endpoints will be normally distributed about the target centre (see section 3.2.5). It also predicts that the incidence of secondary, and tertiary movements will increase with increased movement amplitude and reduced target size, and that there are no periods of non-movement. It assumes that people execute their movements so as to minimize the overall task completion time with the planning of the possible subsequent movement taking place during the execution of previous movement.

The findings from this study present a number of problems for this model of movement: subject's first movement terminations were not distributed about the target centres, periods of non-movement occurred frequently for the more difficult task conditions, the sub-movement durations were found to be related to the square-root of movement amplitude rather than the square-root of the ratio between movement amplitude and target size.

The errors of extent found within this study suggest that subjects executed the first movement so that a second movement could be initiated at a point, on average, located at approximately 50 - 75% of the initial distance towards the target. An exception to this strategy occurs for those targets requiring a very small movement amplitude, where a 'hole in one' is far more likely to be attempted. Associated with the very small amplitude movements is a greater reaction time. Perhaps this extra processing time indicates an extra task complexity associated with 'hitting' the target with a single movement.

If people do attempt to optimize overall movement time, the findings here suggest that rather than one optimal movement strategy being employed, people may adopt different optimal strategies depending upon the task variable conditions. Further, if these movement strategies are artifacts of this particular experimental task then it suggests that people may adopt different movement strategies not only for particular task variables but also for different types of movement task.

The number of movements made for a particular trial did increase with increased task difficulty, as predicted by this model. Post-hoc descriptions of movement time found it to be related to the square-root of the ratio of movement amplitude to target size, as was predicted by this model. However, similar predictions for the durations of the first and second movements were not so well supported, with the square-root of the movement amplitude, providing a better account of these movement's durations.

There was evidence of movements being preplanned 'on the fly'. The incidence of non-movements indicates that some movements, subsequent to the initial movement, were executed without an intervening non-movement period.

7.3.8.7 The type of movements

In section 3.2.6 the distinctions between ballistic and monitored movements were explored. Ballistic movements were considered to be movements of a relatively short duration which were made independently of visually based corrections. The experimental implementation of this study was such that the movement's utilization of visual feedback could not be investigated directly. However, the duration of discrete movements and the occurrence of movement corrections could be detected. Over half of the mouse movements made during this study exhibited no signs of being corrected. The incidence of complex movements increased for an increasing movement amplitude, reducing target size, and for the low gain condition. The velocities and durations of first and second movements also increased for the same manipulations of task conditions. Similarly, the first and second movement durations were best accounted for by a mathematical formula relating the movements' duration to the square-root of the movement amplitude; a relationship identified with ballistic movements (Gan & Hoffman, 1988 and Jagacinski et al, 1980). It is suggested, within this study, that as a movement's duration increases so does the likelihood of a movement correction based on visual information. Further, as movements following the initial movement have been shown to have smaller durations than those of the first movement, it is likely that most of the movements found within this study can be considered as ballistic. These findings are consistent with those of Murrell & Entwisle (1960). This does not support a model of movement which would suggest that each trial consisted of an initial ballistic movement followed by a phase of monitored movements to locate the target. However for the difficult task conditions the initial movement was more likely to be corrected during its execution.

7.4 Good versus poor performers.

The differences in performance, for a number of movement characteristics, between relatively good and poor users is shown in Table 7.5. The number of cases (N) was 48 in each instance.

Good Versus Poor Performers

Movement Characteristic	Mean Good	Mean Poor	t	P
Task completion time	1.56s	2.08s	8.9	<0.001*
Reaction time (RT)	0.62s	0.76s	4.6	<0.001*
Movement time (MT)	0.62s	0.81s	4.9	<0.001*
Time on Target (TOT)	0.32s	0.51s	6.0	<0.001*
MT (non-movement)	0.70s	1.13s	3.5	<0.001*
MT (movement)	0.55s	0.69s	4.0	<0.001*
TOT (non-movement)	0.24s	0.39s	5.3	<0.001*
TOT (movement)	0.08s	0.13s	5.3	<0.001*
No. Sub-movements	2.67	3.36	4.0	<0.001*
No. Non-movements	0.63	0.88	2.8	<0.010*
% complex movements	37.7%	41.7%	0.8	>0.050
Distance to target centre	44.0%	40.6%	2.5	<0.050*
1st Sub-movement				
Duration	0.24s	0.24s	0.0	>0.050
Velocity	0.11	0.09	2.5	<0.050*
Effectiveness	65.4%	56.8%	2.2	<0.050*
Errors of Extent	21.9%	33.1%	3.1	<0.005*
Errors of Aim	7.8^{o}	10.4^{o}	1.6	>0.050
2nd Sub-movement				
Duration	0.21s	0.20s	0.8	>0.050
Velocity	0.06	0.06	0.7	>0.050
Effectiveness	59.5%	64.3%	0.5	>0.050
Errors of Extent	03.1%	21.3%	2.7	<0.050*
Errors of Aim	5.6^{o}	8.3^{o}	1.2	>0.050

* significant at the p<0.05

Table 7.5

Given the definition of good users it is not surprising to discover that they are quicker than poor users. Poor performers take longer to initiate the first movement, a greater time to travel to the target, and spend more time on the target prior to making their selection. These findings are partially consistent with those of Wolf(1992); she found differences in movement and selection times between experienced and inexperienced users, but reported no differences in reaction time.

The additional time taken by poorer performers in initiating and executing the movement activity does not result in more effective movements, or reduce the number of movements, or non-movements, that are made to reach the target. Similarly the additional time taken on target does not lead to a marked reduction in the number of target selection errors (see section 7.1). Although, poor performers do spend more time in movement within the target area which may account for pointer's location nearer the target centre, the differences between good and poor performers for final pointer location is only marginal, and still would not account for the difference between good and poor performers in the time spent in non-movement.

Poorer performers execute slower first movements, of about the same duration and aim as those of good performers, which consequently cover less of the distance towards the target and result in a less effective movement. This leads to more movements having to made to reach the target. Furthermore poorer performers are more likely to punctuate their movements with non-movement phases.

The higher incidence of non-movements made by poorer performers suggests that the ability to preplan a subsequent movement whilst executing another movement is an ability improved through practice. The slower reaction times of poorer performers lend weight to this argument.

7.5 Errors in target selection

A mean average of 1.25 wrong target selections were made per session. The incidence of wrong target selections was found to be independent of the task conditions and performance.

Tables 7.7a and 7.7b show the mean reaction times and the first movement duration respectively, for target selection correct, and incorrect responses for the different target size and movement amplitude conditions.

			Reaction time				
Size	Amplitude	N	mean time		t	DF	Significance
			correct	error			P <
small	small	28	763.3	600.6	3.65	27	0.001*
small	medium	32	695.3	593.6	3.04	31	0.003*
small	large	39	692.8	576.2	5.70	38	0.001*
medium	small	25	684.8	513.0	5.15	24	0.001*
medium	medium	34	660.7	530.7	4.49	33	0.001*
medium	large	37	669.1	575.1	4.31	36	0.001*
large	small	20	679.3	523.6	3.11	19	0.003*
large	medium	27	641.8	504.4	4.62	26	0.001*
large	large	28	694.9	590.3	2.95	27	0.004*

Table 7.6a

First Movement Duration

Size	Amplitude	N	mean time correct	mean time error	t	DF	Significance P <
small	small	28	112.0	91.5	1.32	27	0.100
small	medium	32	194.4	144.7	3.30	31	0.001*
small	large	39	322.2	121.6	10.27	38	0.001*
medium	small	25	139.5	127.8	0.65	24	0.276
medium	medium	34	235.4	104.0	7.77	33	0.001*
medium	large	37	329.5	125.0	10.18	36	0.001*
large	small	20	174.0	72.6	7.11	19	0.001*
large	medium	27	253.4	117.7	7.20	26	0.001*
large	large	28	339.5	113.5	11.04	27	0.001*

* = significant, P<0.05

Table 7.6b

In all of the above cases the error reaction times and error movement durations were less than those associated with correct target selection trials. It can be seen that all of the error movement reaction times were significantly less than those for correctly selected targets. Similarly, in all but two cases the error movement duration was significantly less than those durations associated with correctly selected targets.

The smallest significant difference for reaction time was 94.1ms (13.7% of mean RT) and the largest significant difference was 171.8ms (25% of mean RT). The smallest significant difference for movement duration was 49.7ms and the largest difference was 226.0ms.

Movements believed to reflect incorrect target selection were found to differ in their movement characteristics to target correct responses. Reaction times for incorrect responses were found to be less than those associated with correct responses. A similar finding by Megaw (1972) was considered to reflect an incorrect response selection arising from a premature onset of the movement.

The finding that erroneous movement durations were significantly shorter than correct response movement suggests that the onset of the movement termination may be used as a measure of ECRT (Error Correction Reaction Times). These times would be much less than those of the erroneous movement's duration and indicate ECRT's of much less than 100ms would be found.

7.6 Conclusions

The conclusions have been separated into four sections: mouse movement, theoretical implications, empirical implications, and the possible future implications for input design based on the movement characteristics.

7.6.1 Mouse movement

Target acquisition with a mouse device was often achieved by a sequence of discrete submovements directed towards the target area which were occasionally punctuated with episodes of non-movement. The velocity of the second submovement reduced when compared to the first submovement. The durations and effectiveness of submovements, and the incidence of discrete movements and non-movements, were found to vary with task conditions and previous device experience. However, for any particular task condition the effectiveness of the submovements were not found to be significantly different. Although the durations of the second submovement were significantly less than those of the initial submovements they were of a similar magnitude. The initial submovements undershot rather than overshot the target area.

It is suggested that certain task conditions (a small movement amplitude and relatively large targets) are likely to result in an initial movement that will terminate within the target area.

7.6.2 Theoretical implications

Many theories of discrete motor movement assume that people's movements are made so as to terminate within the target area, and the likelihood of a correct termination is dependent upon the limitation of the human motor system and the task circumstances. The findings of this experiment suggests that people were making first movements which often terminated well before the target area; although there was an exception for medium and large targets requiring a small movement amplitude. Such behaviour may have arisen from the particular task conditions, but these finding still suggest that any model of discrete motor behaviour, for the mouse task at least, should acknowledge general movement strategies other than those based on the assumption that each movement is aimed at the target centre. The possibility that such movement behaviour was a consequence of the task demands lends further support to the need for more sophisticated models of discrete movement which take into account the influence of task demands (such as task instructions not to lift the mouse and the possible loss of pointer feedback) other than those directly associated with the task (such as target size and movement amplitude).

A priori predictions of task completion time, based on formulations such as Fitts' Law, are problematic due to the difficulty in specifying, in advance, the required coefficients associated with such equations. The values of such terms have been found to be task dependent and can lead to substantial errors in predicted completion times for some tasks. Furthermore, the suggestion that additional, fixed value, times can be added to completion times to reflect the time for additional decision processes, such as that involved in pushing the mouse button, has also been challenged. However, such formulations do provide indications of the general relationship found between target size and movement amplitude. That is, they are often able to define the family of mathematical curves relating certain task conditions to task completion time, but are unable to provide the appropriate curve to the task, as indicated by post-hoc analysis.

Differences between the gain condition and previous device experience were found to be associated with differences in overall task completion time. Most models of discrete movement fail to take into account differences in movement behaviour arising from such influences.

It should be noted that the findings of this chapter have been taken from averaged grouped subject data and do not necessarily reflect individuals' movement characteristics. However, this does not invalidate the conclusions drawn from averaged group data, as such analysis allows us to consider and make statements regarding the general tendencies in data. Most theories of motor movement have been validated against such averaged data. However, it is important to distinguish between statements made regarding the averaged group data and those used to describe individual behaviour.

7.6.3 Empirical implications

It is suggested that greater task completion times were associated with targets located at the bottom, as opposed to the top, of the display area. However, this may have arisen from the implementation of the task and requires further work to establish its generalizability.

Task completion times are found to reduce for increased target size and reduced movement amplitude. Generally, the reduction in task completion time was greater for increases in target size when compared to the reduction for proportionally equal reductions in movement amplitude. Additionally, small sized targets should be avoided due to the greater number of errors associated with them.

It is suggested that small targets should not be used with a high input gain.

Inexperienced device users are likely to have slower task completion times than experienced device users. This can be offset, to some degree, by employing much larger targets and much smaller movement amplitudes.

7.6.4 Future implications

The finding that first movements often fall short of the target area, especially for inexperienced users, could be used as a basis for future research attempting to optimize the input process based on the characteristics of the first movement. Similarly, the finding that much of the time on target was spent in non-movement, especially for the small targets, could also provide potential enhancements to the input process.

CHAPTER EIGHT

AGE RELATED DIFFERENCES IN TASK PERFORMANCE

EXPERIMENT TWO

This chapter highlights the need for research into age related differences for people's use of computer input devices. Most older users were able to successfully use the mouse device, even for the more difficult task conditions. Older device users were found to be slower, requiring about one third as much time again, in acquiring targets. Older users required more time to initiate, move to, and then select the targets when compared to younger users. The additional time taken by the older users did not lead to improved performance accuracy. The performance of older and younger novices improved through practice with both groups improving at about the same rate. Despite most older individuals successfully completing the experimental session, two individuals experienced difficulty with the task and were unable to complete the session. Therefore the mouse device may not be suitable for all older computer users.

8.0 Introduction

It has been noted (Denton, Feaver, & Spencer, 1986; Smith, 1990; Staufer, 1992) that the labour force, like the population is ageing. Given the changing nature of work (see section 1.0), the older[1] computer user within the workforce will change from a rarity to a routine occurrence (Charness & Bosman, 1990; Czaja, 1988); and is likely to use an additional input device to the keyboard (Smith, 1980), which for the immediate future will probably be a mouse.

The use of computer-based technologies emphasises perceptual, cognitive, and motor skills which have been shown to deteriorate in the older age groups (Bromley, 1988; Czaja, Hammond, Blascovich, & Swede, 1989; Welford, 1985). At present it is unclear whether the demands placed by new technology on the older worker can be met (Bromley, 1988). Despite the increase, and the anticipated increase in number of older workers, little attention has been paid to age factors in the design of new information technology systems (Charness & Bosman, 1990, Kahneman and Tversky, 1973).

8.1 New Information Technology And The Older Worker

Most studies have investigated age related differences in performance involving computer application programs, such as; word processors (Elias, Elias, Robbins, & Gage, 1987; Czaja, Hammond, Blascovich, & Swede, 1986 & 1989; Hartley, Hartley, & Johnson, 1984; Egan & Gomez, 1985; Gomez, Egan, Wheeler, Sharma, & Gruchacz, 1983; spreadsheets (Garfein, Schaie, & Willis, 1988; Gist, Rosen, & Schwoerer, 1988), and databases (Greene, Gomaz & Devlin, 1986). These studies have primarily been concerned with skill acquisition for 'application programs' and the effects of different training methods for the young and older user.

---

1 Typicially over forty years of age

All of the studies have found that the older worker was able to acquire the skills demanded for these applications. With respect to the rate of skill acquisition and task performance the findings are more equivocal. Some studies (Egan & Gomez, 1985; Czaja et al, 1989; Gist et al, 1988) found that the older user acquired the skills at a slower rate, and required more time to complete the tasks. However, Hartley et al (1984) did not report any age related difference in the rate of skill acquisition, but found that older users took longer to complete a given task. Similarly, Garfein et al (1988) found no differences between young and older users in skill acquisition rate and final performance levels.

Davies, Glendon, Stammers, Matthews, & Taylor (1992) investigating age related differences in the acquisition of office information technology skills (word processor, spreadsheet, and database skills) found that whilst, initially, younger learners performed tasks more quickly than older users, these age based differences in performance reduced with practice, where the older learners showed a more rapid improvement with subsequent task experience. Novice older users made fewer errors than novice younger users for the word processing task, but there were no significant differences, or age by error rate interactions, for the database and spreadsheet tasks.

When experienced older and younger word processor users were investigated, whilst there was no difference in accuracy of performance between the age groups, younger experienced operators were consistently quicker than older experienced operators.

In general, it has suggested that training periods for the novice older user should be twice the duration of those for younger users (Charness & Bosman, 1990).

There is a paucity of empirical research into age and input devices. Only one study has investigated age factors and the type of input device. The mouse device when compared to keyboard input has been found to reduce age differences in performance (Charness, Graham, Bosman, & Zandri, 1988). However, small target areas (single characters) presented difficulty for some of the older users. It was suggested that this might be accounted for by hand tremors associated with the older user (Charness & Bosman, 1990). Attempts to obtain further information regarding this experiment, by personal communication with the principal author, provided information on sample size (10 older & 10 younger subjects), and average 'time to target' (older = 4.5s, younger = 2.9s) but failed to provide any further useful experimental details (Charness, 1993, personal communication).

Despite the lack of research into age and input devices generally, and the mouse device in particular, there has been research into psychomotor skills and age.

8.2 Age and Psychomotor Skill

Reviews of ageing and psychomotor skill (Bromley, 1988; Welford, 1958; Spirduso & Macrae, 1990) have concluded that older people generally take longer to complete such tasks. However, such time differences may not be reflected equally across all task components (for a review see Welford, 1985).

Ageing is associated with the slower processing of visual information (Czaja, 1988). Walsh & Prasse (1980) found that the elderly took longer (15ms) to recognise letters presented on a visual display terminal. Simple reaction time slows by 20% from 20 to 60 years of age (Birren, Woods, & Williams, 1980). However, extensive practice by the older individual can lead to a reduction in age related differences for simple and choice reaction times

(Brinley & Botwinick, 1959; Botwinick & Thompson, 1967; Falduto & Baron, 1986).

Singleton (1959) was able to separate the reaction time and movement time components of a psychomotor task. Using an arrangement not unlike a manual gear stick subjects were required to move the lever from a central home position along one of four movement guides to a terminal point (enforced by a mechanical stop) and then back to the centre location. The movement guides were arranged in a cross pattern so that the movements progressed along straight lines following the cardinal compass directions (north, east, south, and west) relative to the subject. The subject was seated a small distance away from a display that indicated the direction that the movement was to take. The movement itself required little, if any, guidance to the terminal location. Reaction times were found to slightly increase with increasing age (20 years 450ms, 60+ years 520ms). The time to move from the centre to the terminal location showed a small increase for the fifty plus groups, but the largest time differences were found between the fifty plus age groups and the younger age group, for the time spent in initiating and executing the return movement to the centre location. Although the experimenter was unable to separate the time to initiate the return movement from the time to execute the movement, it was believed that most of this additional time was spent in initiating, rather than executing, the return movement.

Considering the above study, and the earlier studies by Leonard (1952) and Szafran (1951), it was suggested that for motor tasks which are predominated by a 'ballistic' component age effects are minimized (Welford, 1968). Older people were found to be most disadvantaged for motor tasks requiring frequent changes in direction, and periods of constant motion (Welford, 1958,1977).

This age effect has been attributed to older people requiring more time to initiate, guide, and monitor movements, rather than to execute them. That is, there slower performance is attributed to their central processing capabilities rather than their muscular atrophy. This is supported by studies which have found age differences in response latencies to increase with the increasing complexity of the motor responses (Jordan & Rabbitt, 1977; Plude, Hoyer, & Lazar, 1982; Tolin & Simon, 1968; Welford, 1965).

8.3.0 Experiment 2

Twenty younger (mean age 22.3 (SD 3.1)) and twenty older (mean age 51.6 (SD 4.8)) individuals, all recruited from local job clubs and training agencies, completed the input task (described in chapter 6) using the PC Optical mouse with a linear movement gain of five. Two additional older subjects were unable to complete the experimental task and their data are not included within this analysis.

The groups were matched for gender, handedness, educational level, and previous device experience. In most cases individuals had none, or little, mouse experience. All participants passed the near visual acuity eyesight test (described chapter 6), and the apparatus was tailored to their individual ergonomic requirements (described in Chapter 6).

8.3.1 Results

Older novice users took longer than younger novice users for all task conditions, and there were no significant interactions for any of the performance measures between age and task factors. Table 8.0 shows a summary of the experimental results (all times shown in seconds).

Results Summary					
Performance measure	Old	Young	DF	t	P<
Task Completion Time	3.348	2.471	24.5	-3.0	0.010
Reaction Time	1.019	0.804	28.1	-3.3	0.010
Movement Time	1.289	0.948	28.6	-2.7	0.050
Time on Target	1.039	0.719	23.7	-2.5	0.050
Movement Time					
Movement	0.931	0.727	29.8	-2.9	0.010
Non-Movement	0.359	0.222	27.7	-2.4	0.050
Time on Target					
Movement	0.182	0.136	24.8	-2.5	0.050
Non-Movement	0.857	0.583	24.0	-2.4	0.050
No. of Movements	6.18	4.29	24.7	-3.3	0.010
No. of Non-Movements	2.26	1.38	25.4	-2.9	0.010

Table 8.0

Figure 8.0 shows the component times by age.

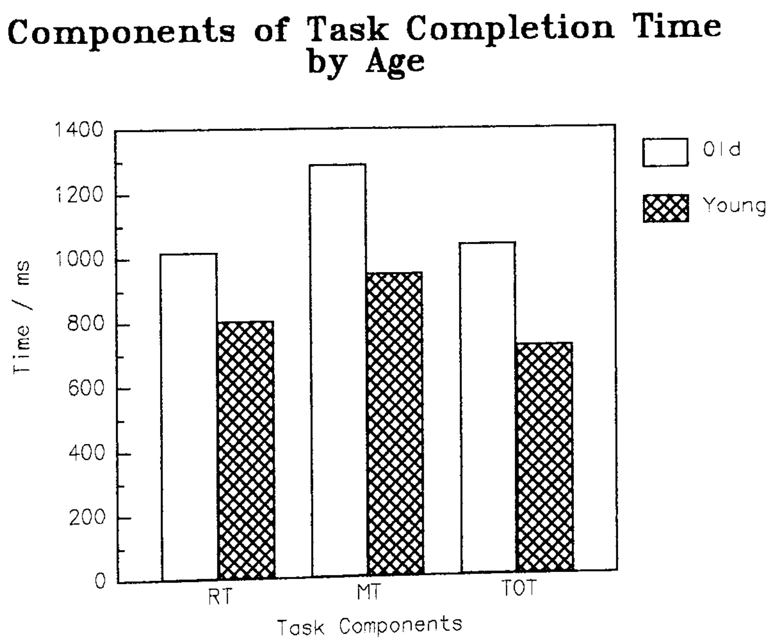


Figure 8.0

Older novice users took significantly longer to; initiate movement, move to the target, and indicate their selection, when compared to younger users. Older users required more submovements to reach the target, and had more 'rest' episodes, than the younger novice users. The variances associated with the older age group, for all of the above performance measures, were significantly greater when compared to those of the younger age group.

Despite spending more time on all components of the task the older users did not make significantly fewer errors, either for primary or secondary targets, and did not locate the pointer significantly nearer the target centre.

The experimental session trials (108) were partitioned into three consecutive blocks of 36 trials each. Table 8.1 shows the Anova summary table for task completion time by age and block, and Figure 8.1 shows the mean task completion times for the trial blocks.

Anova Summary Table
for Trial Completion Time
by Age & Block

Variation Source	SS	DF	MS	F	P<
WITHIN CELLS	99455640.30	38	2617253.7		
AGE	23000969.99	1	23000970.0	8.79	0.005
WITHIN CELLS	55785000.82	76	734013.2		
BLOCK	9298252.42	2	4649126.2	6.33	0.003
AGE BY BLOCK	2459704.25	2	1229852.1	1.68	0.194
Total SS	189999567.78				

Table 8.1

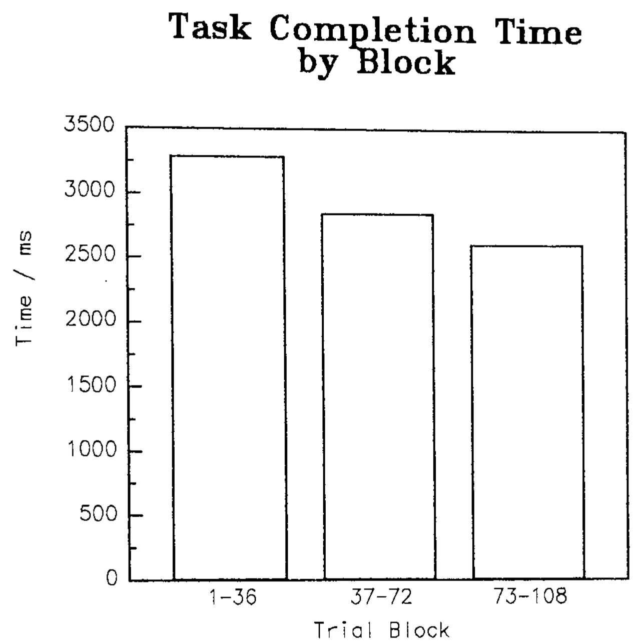


Figure 8.1

Main effects of Age (12% of the total variance) and Block (4.9% of the total variance were) found. Old and young users improved through practice, and older users took longer to acquire targets than younger users. Post-hoc tests found the first trial block to be significantly slower than the last trial block ($P<0.01$, Tukey T 579.8).

Movement angle was not found influence task completion time. However, target location was found to influence task completion time. Table 8.2 shows the Anova summary table for task completion time by target location and age. Figure 8.2 shows task completion time by target locations.

Anova Summary Table
for Trial Completion Time
by Location & Age

Variation Source	SS	DF	MS	F	P<
WITHIN CELLS	132607522.20	38	3489671.6		
AGE	30667907.75	1	30667908.0	8.79	0.005
WITHIN CELLS	9904441.64	114	86881.1		
TARGET	5706759.45	3	1902253.1	21.89	0.001
AGE BY TARGET	264036.84	3	88012.3	1.01	0.390
Total SS	179150667.88				

Table 8.2

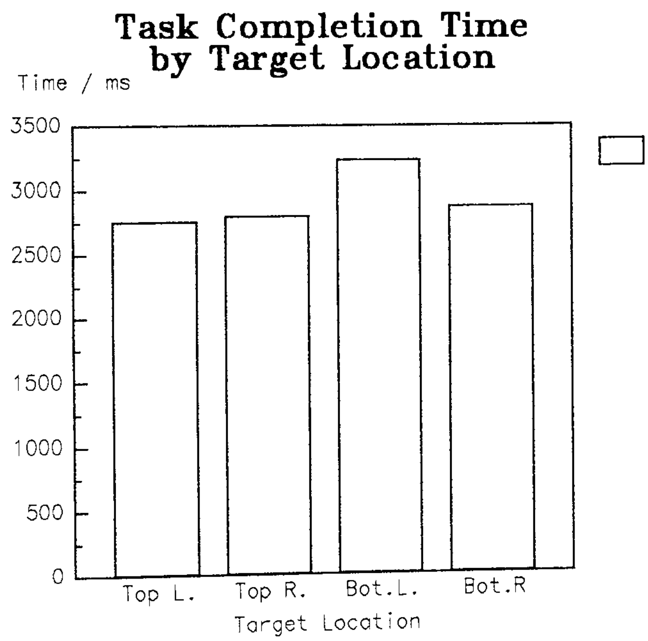


Figure 8.2

A main effect of Target Location (3.2% of the total variance) was found. Post-hoc test found Target three (bottom left) to be associated with significantly slower task completion times (P<0.01, Tukey T 213.9) when compared to the other three target locations. Differences in Task Completion Time between the other target locations were not found to be significant.

8.3.2 Discussion

Mouse based computer input took longer (36% longer) for older, when compared to younger, device users which was to be expected from previous investigations. However, the finding that age did not significantly interact with any other task conditions conflicts with earlier findings (Charness et al, 1988). Charness et al reported that older users were especially disadvantaged for small sized targets. The targets within this study were of a comparable size to those used by Charness et al (they used single characters), and the gain used within this experiment (5) was as high a value of gain as could be used without compromising the difficulty of the task. The relatively large completion times in the Charness et al study suggests that their task was more challenging than the task used within this study. Without more task and subject details a resolution of these disparate findings cannot be achieved.

The increased variability in the performances of the older age group, when compared to the younger age group, are consistent with the findings from other studies of age related differences in performance (Bromley, 1988).

The times associated with the different task components were longer for the older users. The difference between young and old for; Reaction time, Movement time, and Time on Target, is just over 200ms.

The reaction time difference may represent the slower processing of visual information and subsequent movement initiation. However, the magnitude of this difference is much greater than that which would be expected for the processing time of visual information alone (see section 8.2). It was noticeable that several of the older subjects wore bifocal glasses and this resulted head in positioning movements which enabled wearers to read the task details at the top of the display. It is believed that such head movement slowed down the onset of device movement. However, as most of the older subjects had their vision corrected, this information was not recorded and subsequent analysis along such lines is not possible. In the case of future studies the prescription for any subject's visual correction should be noted.

Although no detailed submovement analysis is presented within this chapter, older users provided on average more movements, and had more episodes of non-movement, than younger users. The older users spent longer in non-movement when moving to the target area than did younger users. However, the mean times for each non-movement period for the two age groups are very similar (Old 159ms, Young 160ms), but the mean duration of the submovements for the two age groups are less so (Old 151ms, Young 170ms). This suggests that younger users are making movements of longer duration than older users, which leads to younger people requiring fewer movements to acquire a target. It also suggests that younger people are more able to make their submovements on 'the fly' (see section 3.2.5) than older users. The additional movements required by older users in reaching the target area suggests that there are age related differences in the effectiveness of the movements being made.

Older people spend more time on target in non-movement (274ms more) and movement (46ms more) than younger users. As there are no significant time on target by age interactions, button anticipation (see section 7.2.2) probably does not account for the differences between young and old users. Older people may have been acting in a more cautious way. However, if this were so then they should have made significantly fewer errors of selection, but they did not. Presumably they required more time to determine when to press the button? If hand tremor had been the cause of older users requiring more time on target, as suggested by Charness & Bosman (1988), then more of the additional time spent on target by the older individuals should have been spent in movement, but this was not the case. Similarly, if hand tremor in the older user was responsible for the additional time spent on the target area, then an interaction between age and target size for time on target should have been found, but it was not. Furthermore, hand tremor for movement in three dimensions may be reduced when the movement takes place in two dimensions, such is the case for mouse based input.

Some studies have reported that the difference in task completion time between older and younger subjects reduces with task practice (see section 8.1), but this was not found to be the case here, as there was no block by age interaction. However, considering the lack of any significant block effect in the earlier experiment (see section 7.1.2.5) and in the following experiment (9.1.1) it is likely that there is an initial rapid period of learning which slows down after a few hundred trials (for this experimental task). It is possible that in the case of older learners this high rate of improvement may be sustained whilst it reduces for the younger learners.

It should be noted that over 200 people have successfully completed the experimental task described within this chapter. The only two subjects who were unable to complete the experimental session came from the older pool of subjects described with this subject. If this sample were representative of the older population it would suggest that 9% of the elderly population might have difficulty with operating a mouse. Even if these difficulties were found to reduce through practice it suggests that special attention would have to be paid to the task training of older users.

8.4 Conclusions

Most of the older individuals were able to successfully acquire targets using the mouse input device. Older users were not found to be selectively disadvantaged for any of the task conditions. Although older users were found to task longer, when compared to their younger counterparts, they made errors at approximately the same rate. The often reported speed/accuracy trade-off was not observed. Older users were found to be slower across all components of the target acquisition task, and they required more submovements, and episodes of non-movement, in order to acquire displayed targets.

There was no significant differences in the rate of performance improvement between young and older individuals. A longitudinal study could be undertaken to determine if the learning rates remained the same after extended practice.

There may exist a subpopulation of the older working community who could have difficulty, at least initially, in using the mouse device.

CHAPTER NINE

PERFORMANCE WITH PRACTICE

EXPERIMENT THREE

This chapter investigates people's acquisition of device skill. The target acquisition behaviour of relatively experienced and inexperienced mouse users was observed for either a cognitively demanding, or undemanding task. Experienced and inexperienced users improved through practice and inexperienced users' performance times had converged with those of the experienced users after the third experimental session. Performance for the difficult task converged with performance on the easy task after the first block. Further there was no significant interaction between task difficulty and previous device experience. Analysis of movement traces suggested that experienced users make a smaller number of 'poorer' first movements than inexperienced users, and that through practice the number of poorer movements can be reduced. These movement traces also supported the suggestion that movements were made so as to avoid overshooting the target, and that near 'misses' arose through errors of extent. However, inspection of these movement traces suggested that users were not consistently undershooting the target by a fixed proportion, and that 'holes in one' and very poor movements often occurred. These findings suggest that caution should be urged when dealing with averaged movement data.

9.0 Introduction

'Practice makes perfect'

Anon.

Numerous studies of motor behaviour have shown that people's performance for a motor task can change as a result of practice on that task. Generally, skilled performance increases as a direct function of the amount of practice (Johnson, 1984). When a relatively permanent improvement in performance is observed, typically as a consequence of practice, learning is assumed to have taken place (Magill, 1991).

A useful method to assess learning is to keep a performance record throughout the period of skill acquisition (Magill, 1991).

9.0.1 The Power Law of learning

> *'It has been known since at least the study of Snoddy (1926) that performance time, when considered as a task criterion in perceptual motor skills, tends to decrease with practice as a function of a power law.'*
>
> Newell (1991, p. 216).

One of the best known examples of this phenomenon was reported by Crossman (1959) who investigated the manual manafacturing activities of cigar rollers by means of a cross-sectional study. He found that the cigar rolling times, over a range of seven years could be described (10,000,000 cigar making trials) by a power law (also mentioned in section 2.4.1.8). This finding has been replicated for a number of motor performance tasks (Newell, 1991), and for mouse device studies (Card et al, 1978; and Radwin et al, 1990).

Despite widespread empirical support for a Power Law of learning, Newell (1991) suggests that there are important limitations to this description of skill acquisition.

1. It has only been demonstrated for tasks where time is the performance variable.

2. Qualitative changes in behaviour may occur leading to discontinuities in performance time.

3. Task variables, other than time, may also change with practice.

Some studies of skill acquisition have reported finding 'plateaux' in performance curves (Book, 1925; Bryan & Harter, 1887; Franks & Wilberg, 1982). Book (1925) distinguished between 'breathing places' which showed no improvement in performance for 6-8 experimental sessions, and plateaux which showed no improvement in performance over a larger number of experimental sessions (17-33). However, relatively few skill acquisition studies have reported finding such plateaux in performance (Adams, 1987), and it has been suggested that their occurrence may be attributable to artifacts of the performance measure used (Magill, 1991, Adams, 1987). Further, Keller (1958) has suggested that although performance measures may show no evidence of improvement, task learning may still be taking place.

Crossman (1959) not only found that task completion times reduced according to a power law for a large number of trials over a relatively long period, but that the advantage for good over poor performers, was in their relatively higher frequency of fast task completion times rather than faster task completion times per se.

9.0.2 Qualitative changes in performance

Generally, skill acquisition is not only associated with improvements in overall task completion time but qualitative changes in task performance are also seen (Rosenbaum, 1991; and Johnson, 1984). Fitts (1964) proposed a three stage model.

1. Cognitive

2. Associative

3. Autonomous

During the cognitive stage students are learning the demands of the task. At this stage behaviour is erratic and contains many gross errors. Having learnt the basic mechanics of the required skill the student then moves on to the associative stage where they refine their skill. During the associative phase errors are less frequent and less gross. After much task practice students enter the autonomous stage of skill acquisition. The mechanics of the skill have become automatic and the student's attention is directed to the more critical phases of the skilled action so as to ensure a highly consistent performance.

Adams (1971) proposes a two stage model which has correspondences with that above; the verbal motor stage (Fitts' cognitive & assocative stages), followed by the motor stage (Fitts' autonomous).

Gentile (1972) proposes a two stage model of motor skill learning. At the first stage the student learns to distinguish the relevant stimuli from the irrelevant, and to determine the most effective movement pattern for the task; during the second stage the student focuses on achieving the required goal regardless of the situation, and secondly to attain a consistent performance.

9.0.3 Theories of motor learning

> *'Currently, there is no prevailing theoretical view of motor skill acquisition; indeed, there has not been one since Hull's theory fell from favour during the 1950s.'*
>
> Newell (1991 P.215).

A fundamental assumption underlying traditional and most of the recent theories of motor skill acquisition is that learning arises from the acquisition of better representations of action (Newell, 1991). For Adams, motor learning was achieved through the consolidation of the memory and perceptual traces (see section 3.1.1) and in the case of Schmidt et al (1979) it occurred through consolidation of the recall and recognition schemas (see section 3.1.2). These theories of motor skill acquisition are termed 'prescriptive accounts' by Newell (1991) and are considered to be seriously challenged by the emergence of the ecological perspective on motor skill acquisition.

Bernstein (1967) considered the process of practice as the search for the optimal solutions to the problem at hand which involved mastering the degrees of freedom problem (see section 3.1). Learning from this perspective is considered to be the coordination of the perceptual environment with the action environment in a way consistent with the task constraints.

9.1 Experiment Three

The aim of this study was to investigate people's acquisition of device and task skill. Data were collected from an opportunity sample of 24 subjects who were drawn predominantly from the clerical support staff of Aston Business School. The sample comprised equal numbers of men and women with a mean age of 31.25 years (S.D. 8.7) and mean device usage of 73.5 days per year. Subjects were classified into high and low mouse device users (using a median split value of 23 days usage) taken from self report data with age and sex being controlled for.

Two experimental tasks, based upon the task described in Chapter Six, were employed. Task A, the easy task, corresponded exactly to the task described earlier, but Task B, the hard task, was a variant of the task described previously. In the hard task condition, subjects were only presented with the letter of the secondary target, and not the first number which indicated the appropriate primary target to acquire. For example, 3E would be presented for the easy task, but only E would be presented in the hard task condition. Thus subjects in the hard task condition would have to determine which primary target to acquire in order to select the correct secondary target; whereas in the easy task condition this information is already presented. Subjects were allocated to either the hard, or easy, task condition controlling for experience, sex, and age. In order to ensure that the subjects assigned to the different task conditions were of comparable performance, a small pretest (based on the easy task) was carried out for each individual and subsequent t-tests revealed no significant differences in task completion times for the two groups.

Device users participated in four experimental sessions (blocks) with each session being separated by a three day period. Although they did not practice the experimental task within this period they could have used a mouse as a part of their daily activity. Prior to the initial session each subject passed the eyesight test described in Chapter Six, received task instructions, and had 26 practice trials (representative of their task condition). For each subsequent session a small practice session (10 trials) was carried out prior to the main experimental task.

The experiment was of a factorial design with three factors (experience, task, and experimental session or block). Experience (2 levels) and task (2 levels) were between subject factors and block (4 levels) was a within subject factor. The total number of correct target acquisitions, per session, was approximately 226, and for the complete experiment was just over 1000 (this includes primary and secondary target acquisitions).

9.1.1 Task completion time

Table 9.0 shows the ANOVA summary table for task completion time, and Figures 9.0a & 9.0b the significantly different group means.

Task Completion Time by
Experience, Task, & Block

Variation Source	SS	DF	MS	F	P<
WITHIN CELLS	8108000.05	20	405400.00		
EXP	2654044.92	1	2654044.90	6.55	0.019
TASK	298602.68	1	298602.68	0.74	0.401
EXP BY TASK	8538.10	1	8538.10	0.02	0.886
WITHIN CELLS	2860457.28	60	47674.29		
BLOCK	3586793.19	3	1195597.70	25.08	0.001
EXP BY BLOCK	495942.94	3	165314.31	3.47	0.022
TASK BY BLOCK	938922.19	3	312974.06	6.56	0.001
EXP BY TASK BY BLOCK	189271.07	3	63090.30	1.32	0.275
SS Total	19140572.36				

Table 9.0

Task Completion Time by Performance & Block

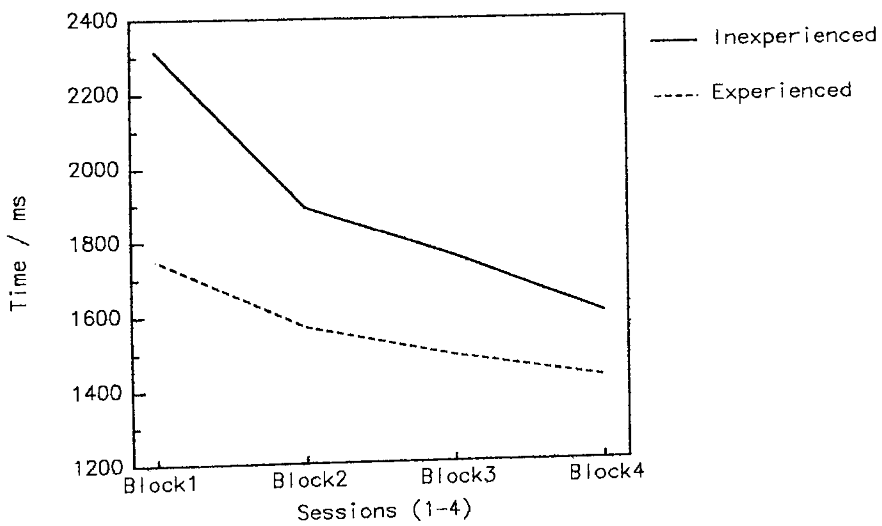


Figure 9.0a

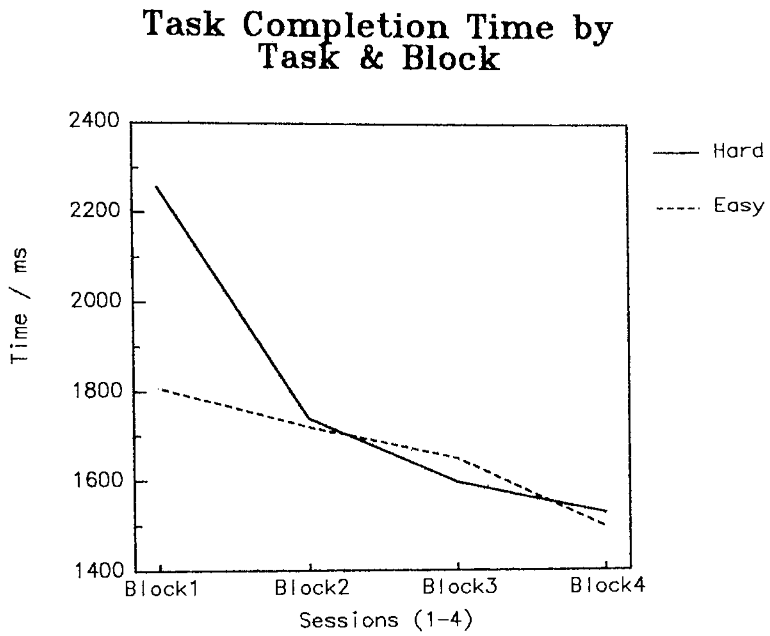


Figure 9.0b

Significant main effects of experience (13.9% of the total variance) and Block (18.7% of the total variance) were found. Experienced users were quicker than less experienced users, and all users improved with practice. Post-hoc tests revealed Block one to be significantly slower than all subsequent blocks, and Block four was significantly less than block two ($P<0.05$, Tukey T 160). However there were no significant differences between Block two and Block three, and Block three and Block four.

Significant interactions between experience and block (2.6% of the total variance) and task and block (4.9% of the total variance) were found. Inexperienced users showed a more rapid improvement in performance when compared to experienced users; especially from block one to block 2. This was supported by post-hoc tests which found significant differences in task completion time between experienced and inexperienced users for all but the last block ($P<0.05$, Tukey T 185.4). The harder task took significantly longer for the first block only ($P<0.05$, Tukey T 185.4).

Table 9.1 shows the correlation coefficients (r) between subjects' pretest performance and their subsequent performance.

Pretest and Subsequent Performance

Block	r task easy	r task hard
1	0.97**	0.66*
2	0.88**	0.61
3	0.96**	0.65
4	0.89**	0.79*

N = 12
P< * - .01 ** - .001

Table 9.1

A subject's pretest performance provided a good indication of subsequent performance for the easy task, but was not such a good predictor of subsequent performance under the hard test condition.

9.1.2 Errors

Table 9.2 shows the ANOVA summary table for the number of primary target selection errors. Figure 9.1 shows the significantly different group means.

Primary Errors by Experience & Task

Variation Source	SS	DF	MS	F	P<
WITHIN CELLS	369.25	20	18.46		
EASY	100.04	1	100.04	5.42	0.031
EXP	28.17	1	28.17	1.53	0.231
EASY BY EXP	40.04	1	40.04	2.17	0.156
WITHIN CELLS	563.42	60	9.39		
BLOCK	162.75	3	54.25	5.78	0.002
EASY BY BLOCK	187.54	3	62.51	6.66	0.001
EXP BY BLOCK	35.08	3	11.69	1.25	0.301
EASY BY EXP BY BLOCK	51.71	3	17.24	1.84	0.150
Total SS	1538.00				

Table 9.2

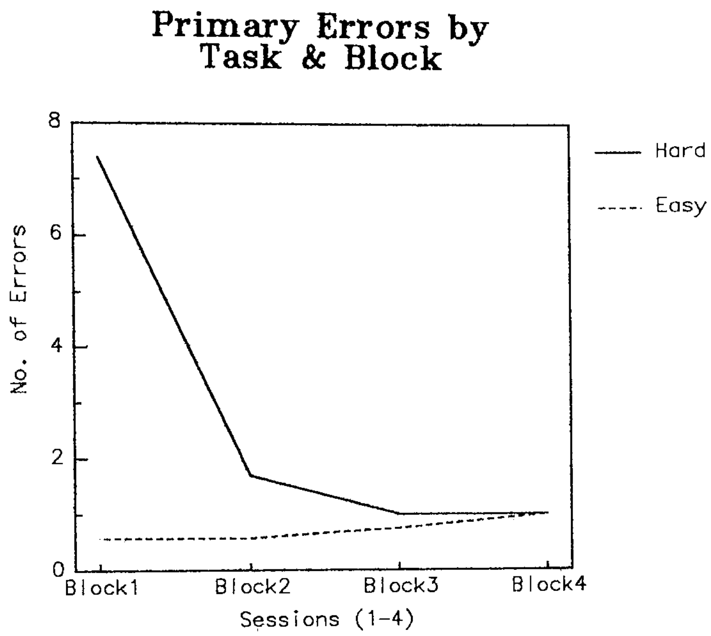


Figure 9.1

Main effects of task (6.5% of the total variance) and Block (10.6% of the total variance) on the number of primary errors were found. The significant task by block interaction (12.2% of the total variance) suggests that the number of errors for the two tasks converge especially after the first experimental block. Post-hoc tests revealed that the difference in errors between the two task were only significantly different for the first block ($P<0.05$, Tukey T 2.6).

There were no significant findings for the incidence of secondary target selection errors.

9.1.3 Reaction time, movement time, & time on target

Table 9.3 shows the ANOVA summary for reaction time, and figures 9.2a & 9.2b show the significantly different group means.

Reaction Time by
Task & Experience

Variation Source	SS	DF	MS	F	P<
WITHIN CELLS	1659010.88	20	82950.54		
EXP	92527.13	1	92527.13	1.12	0.303
EASY	226117.75	1	226117.75	2.73	0.114
EXP BY EASY	20426.28	1	20426.28	0.25	0.625
WITHIN CELLS	771726.95	60	12862.12		
BLOCK	1059033.58	3	353011.19	27.45	0.001
EXP BY BLOCK	127325.78	3	42441.93	3.30	0.026
EASY BY BLOCK	374191.17	3	124730.39	9.70	0.001
EXP BY EASY BY BY BLOCK	22502.48	3	7500.83	0.58	0.628
Total SS	4352862.48				

Table 9.3

**Reaction Time by
Task & Block**

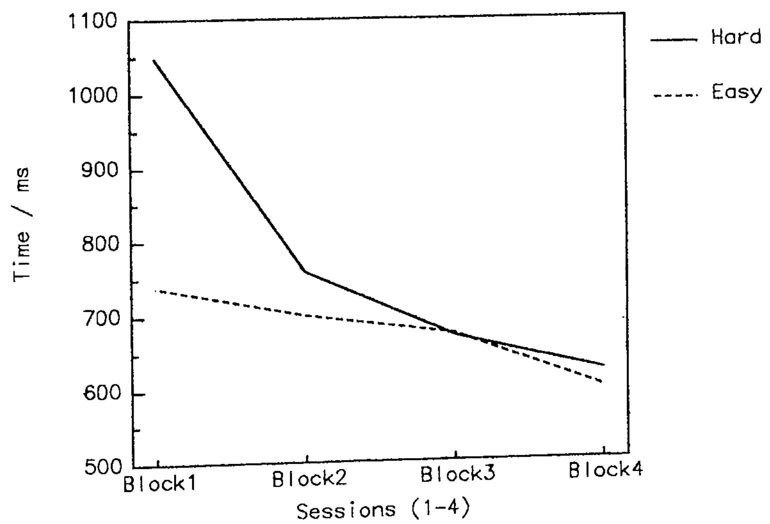


Figure 9.2a

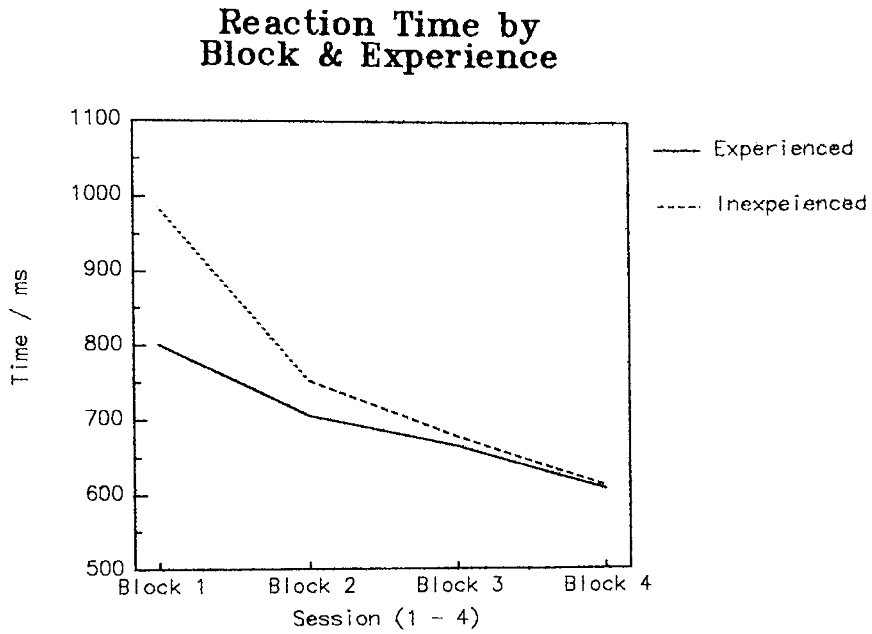


Figure 9.2b

A significant main effect of Block (24% of the total variance) was found. The first block RT was significantly longer than the subsequent blocks, and the second block RT was significantly longer than the fourth block (P<0.05, Tukey T 82.9). There were no significant differences between block two & block three, and block three & block four. The significant block by task interaction (8.6% of the total variance) suggests that differences in RT for the different task conditions reduced for subsequent blocks. This was supported by post-hoc tests which found the difference in RT between the two task conditions significant for only the first block (P<0.05, Tukey T 96.3).

The significant block by experience interaction (2.9% of the variance) indicates that the difference in RT between experienced and inexperienced users reduces for subsequent blocks. This was supported by post-hoc tests which found the only significant difference in RT between experienced and inexperienced users to occur for the first block (P<0.05, Tukey T 96.3).

Table 9.4 shows the ANOVA summary table for movement time, and Figures 9.3a & 9.3b show the significantly different group means.

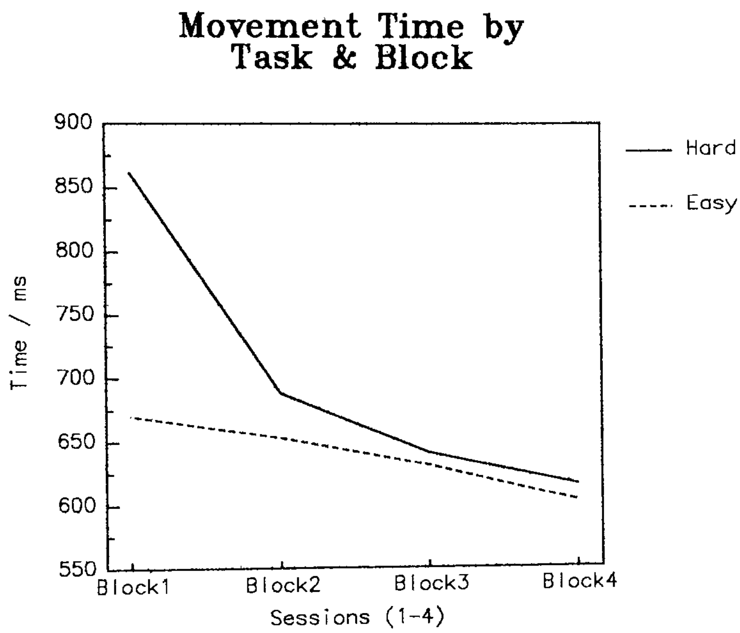


Figure 9.3a

Movement Time by
Task & Experience

Variation source	SS	DF	MS	F	P<
WITHIN CELLS	993067.86	20	49653.39		
EXP	624605.20	1	624605.20	12.58	0.002
EASY	80449.74	1	80449.74	1.62	0.218
EXP BY EASY	508.48	1	508.48	0.01	0.920
WITHIN CELLS	455062.16	60	7584.37		
BLOCK	383327.75	3	127775.92	16.85	0.001
EXP BY BLOCK	74558.26	3	24852.75	3.28	0.027
EASY BY BLOCK	107370.25	3	35790.08	4.72	0.005
EXP BY EASY BY BLOCK	29582.76	3	9860.92	1.30	0.283
Total SS	2748532.46				

Table 9.4

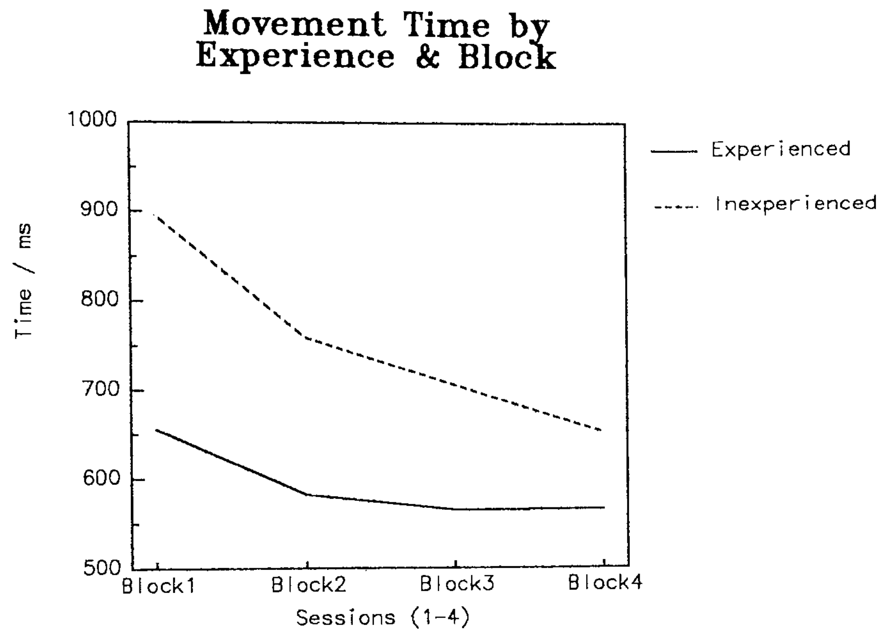


Figure 9.3b

Significant main effects of experience (22.7% of the total variance), and block (14.0% of the total variance) on movement time were found. Experienced users were quicker than inexperienced users and all users improved with practice. However post-hoc test revealed a significant difference in movement time between the first and final block only (P<0.05, Tukey T 162.8).

A significant task by block interaction (3.7% of the total variance) was found which suggests that for subsequent blocks the differences in movement time between the task conditions reduces. This was supported by post-hoc tests which found a significant difference in movement time between the different task conditions for the first block only (P<0.05, Tukey T 77.55).

A significant experience by block interaction (2.6% of the total variance) was found which suggests that movement time for experienced and inexperienced users converged for subsequent blocks.

Table 9.5 shows the ANOVA summary table for time on target, and Figures 9.4a & 9.4b show the significantly different group means.

Time on Target by
Experience & Task

Variation Source	SS	DF	MS	F	P<
WITHIN CELLS	1308473.17	20	65423.66		
EXP	285816.94	1	285816.94	4.37	0.005
EASY	45244.07	1	45244.07	0.69	0.415
EXP BY EASY	45274.47	1	45274.47	0.69	0.415
WITHIN CELLS	332479.14	60	5541.32		
BLOCK	62495.23	3	20831.74	3.76	0.015
EXP BY BLOCK	9630.94	3	3210.31	0.58	0.631
EASY BY BLOCK	16741.59	3	5580.53	1.01	0.396
EXP BY EASY BY BLOCK	54251.31	3	18083.77	3.26	0.027
Total SS	2160406.86				

Table 9.5

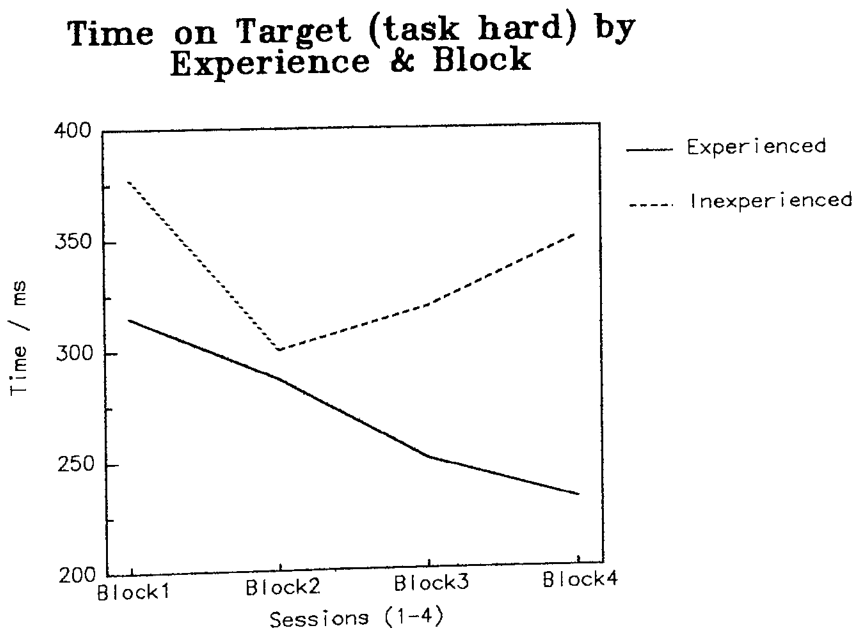


Figure 9.4a

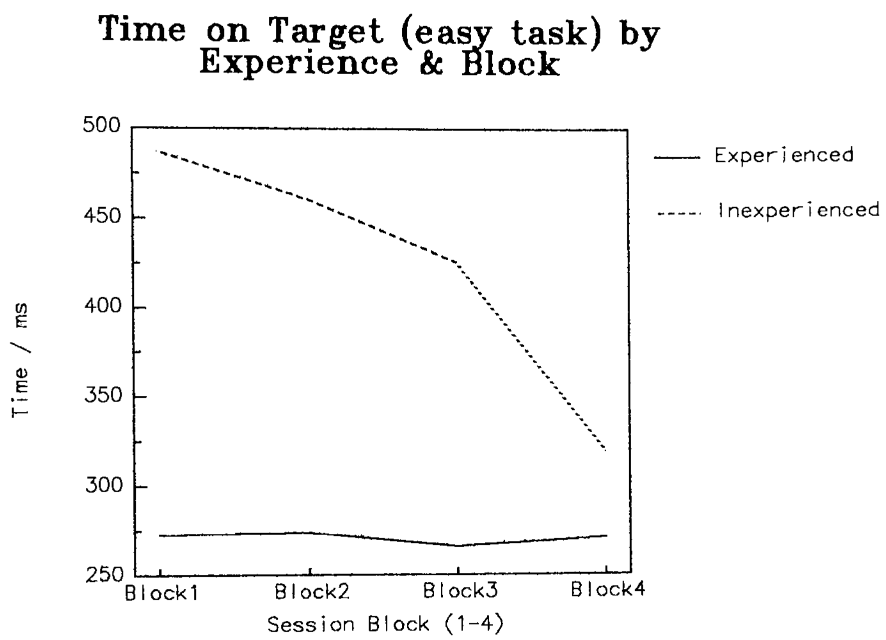


Figure 9.4b

Significant main effects of experience (13% of the total variance), and block (2.9% of the total variance) were found. Inexperienced users spent more time on the target than experienced users, and all users spent less time on the target for subsequent blocks. Post-hoc tests failed to find significant differences in time on target for different experimental blocks (P>0.05, Tukey T 186.9).

The significant three-way interaction between block, task, and experience (2.5% of the total variance) suggests that for the easy task the differences between experienced and inexperienced users converge for the final block. This is supported by post-hoc tests which found significant differences in time on target between experienced and inexperienced users for all but the final block (P<0.05, Tukey T 67.5). For the difficult task condition inexperienced users were found to significantly differ in time on target for the last two blocks (P<0.05, Tukey T 67.5) which suggests that for this task condition time on target diverged for the latter blocks.

9.1.4 Submovement analysis

Table 9.5 shows the ANOVA summary table for the number of submovements made, and figure 9.5 shows the significantly different group means.

Significant main effects of experience (17% of the total variance) and block (14% of the total variance) were found. Experienced users made fewer submovements than inexperienced users and all users showed a reduction in submovements made with practice. Post-hoc tests revealed that there was a significant difference in the number of submovements made between the first and last blocks only ($P<0.05$, Tukey T 0.59).

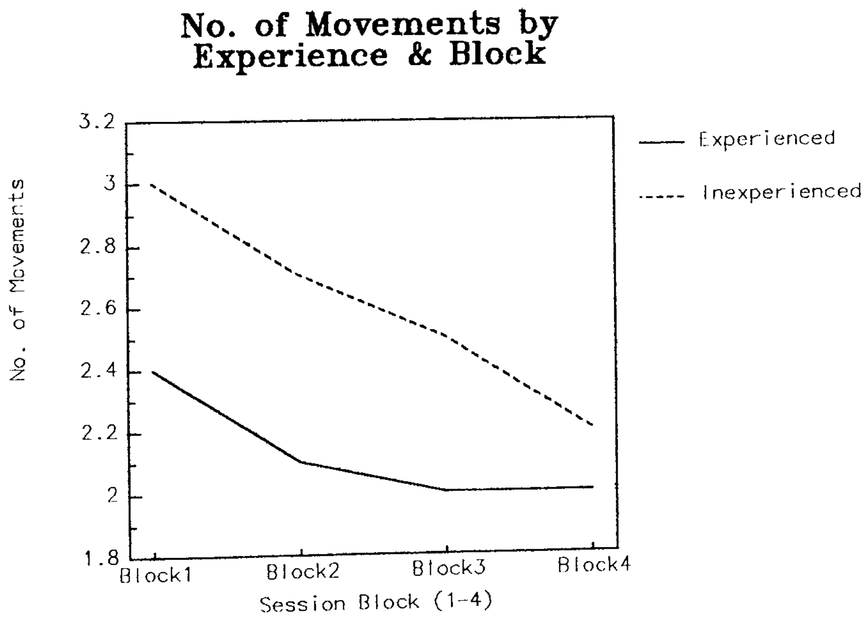


Figure 9.5

Number of Movements by Task & Experience					
Variation Source	SS	DF	MS	F	P<
WITHIN CELLS	13.45	20	0.67		
EXP	5.42	1	5.42	8.05	0.010
TASK	0.00	1	>0.00	>0.00	0.973
EXP BY TASK	0.07	1	0.07	0.10	0.755
WITHIN CELLS	6.85	60	0.11		
BLOCK	4.31	3	1.44	12.59	0.001
EXP BY BLOCK	0.63	3	0.21	1.83	0.151
TASK BY BLOCK	0.22	3	0.07	0.65	0.588
EXP BY TASK BY BLOCK	0.35	3	0.12	1.02	0.389
Total SS	31.30				

Table 9.6

Table 9.6 shows the ANOVA summary table for the number of non-movements made, and Figures 9.6a & 9.6b show the significantly different group means.

Number of Non-Movements by Experience & Task					
Source of Variation	SS	DF	MS	F	P<
WITHIN CELLS	1.58	20	0.08		
EXP	0.93	1	0.93	11.75	0.003
TASK	0.24	1	0.24	3.04	0.097
EXP BY TASK	0.12	1	0.12	1.52	0.231
WITHIN CELLS	0.61	60	0.01		
BLOCK	0.50	3	0.17	16.30	0.001
EXP BY BLOCK	0.12	3	0.04	4.02	0.011
TASK BY BLOCK	0.11	3	0.04	3.46	0.022
EXP BY TASK BY BLOCK	0.05	3	0.02	1.76	0.164
Total SS	4.26				

Table 9.7

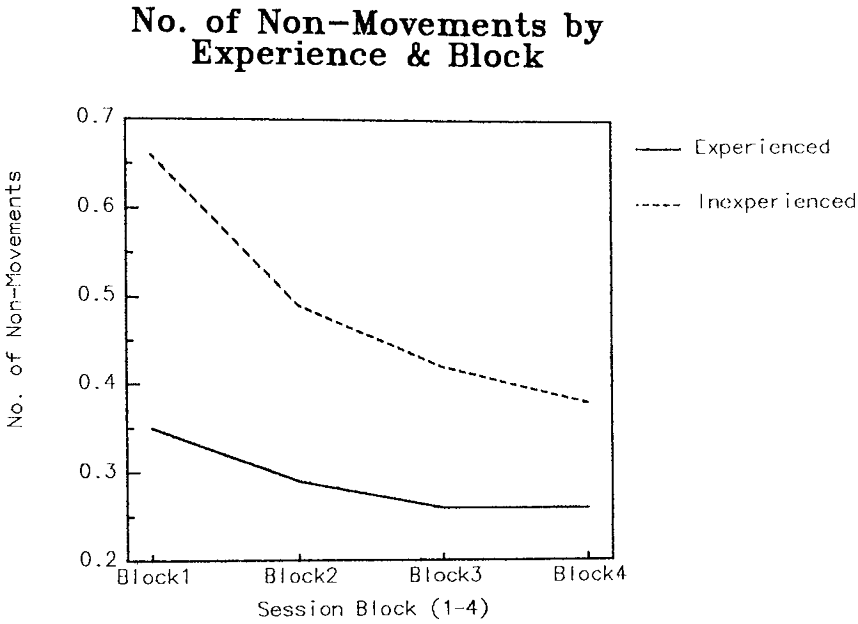


Figure 9.6a

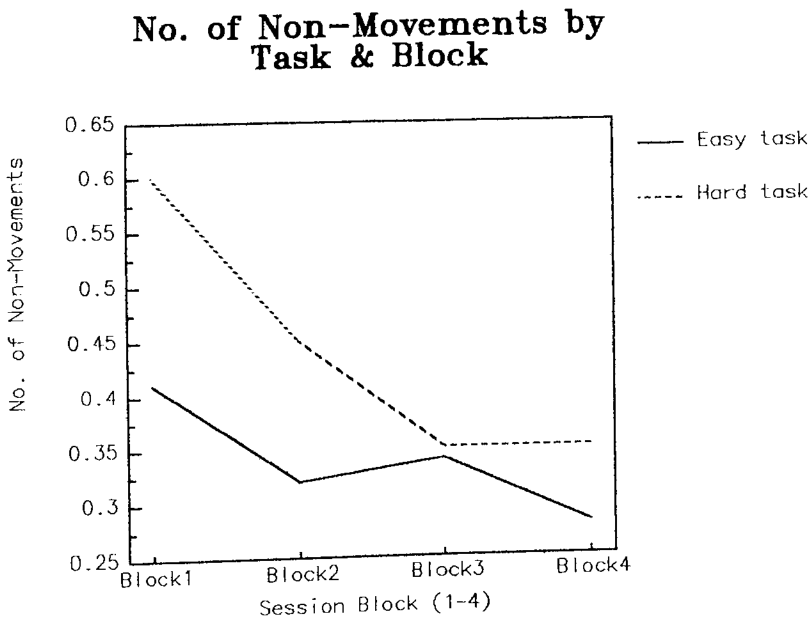


Figure 9.6b

Significant main effects of experience (21.8% of the total variance) and block (11.7% of the total variance) were found. Experienced users made less non-movements than experienced users and all users made less non-movements for subsequent trials. Post-hoc tests revealed that there were significant differences in the number of non-movements made between the first block and the last two blocks.

The significant interactions between experience and block (2.8% of the total variance) and between task and block (2.6% of the total variance) suggest that the number of non-movements for the two conditions converge with practice. Post-hoc tests between the easy and hard task conditions for the various levels of block found significant differences in the number of non-movements for the first and second blocks ($P<0.05$, Tukey T 0.09). Numbers of non-movements for experienced and inexperienced users were significantly different for all levels of block.

9.1.5 Discussion & conclusions

Subjects' performance, as measured by task completion time and the number of errors, improved with practice. Improvements in completion time would appear to be consistent with a power law of learning. However, the application and usefulness of such post-hoc curve fitting exercises has already been questioned within this thesis (see section 7.2.2) and the number of points to be fitted by such a curve in this instance is relatively small (four).

The criteria for stabilization of performance, namely asymptotic improvement in completion times with no significant change in time from one block to the next (see section 2.4.1.8), would suggest that mouse performance stabilized after the second block of trials for experienced and inexperienced users and both task conditions. This would have occurred after approximately 250 target selections which is consistent with earlier findings (see section 2.4.1.8). This is supported by the convergence in task completion time of the two task conditions after the second block of trials. However, the task completion times for experienced and inexperienced users did not converge until the final block.

No interactions between task difficulty and previous device experience for task completion time, or errors, were found. This suggests that when people are learning to use a computer application requiring the use of a mouse there may be no advantage in providing separate mouse tuition to relatively inexperienced device users prior to learning the application package.

The pretest was found to be a good predictor of subsequent task completion times for the easy task. this suggests that pretesting of device usage may provide a good method by which to assess a person's future performance with the device. However, the pretest task should be the same as that for which the predictions are being made.

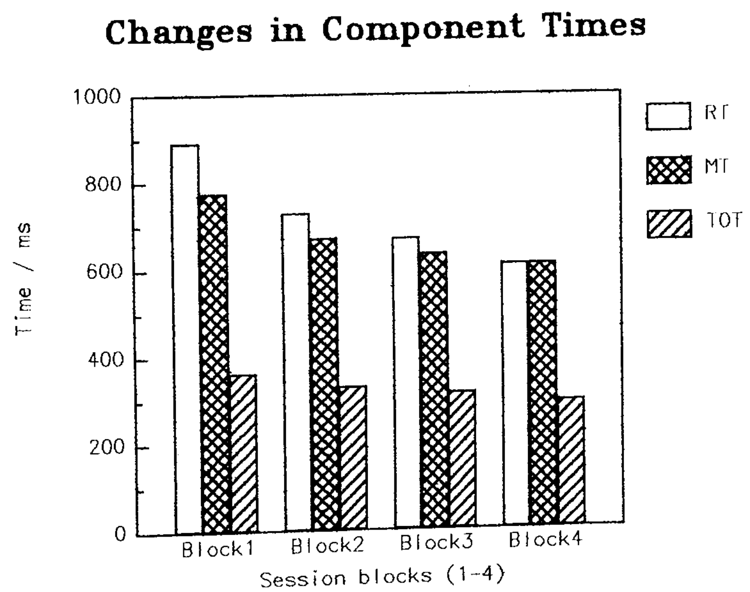


Figure 9.7

Reaction time, movement time, and time on target were found to reduce for subsequent blocks. Figure 9.7 shows how these time components changed for different blocks.

However, they need to be considered in the light of the task and experience interactions discussed shortly.

9.1.5.1 Task difficulty

The more difficult task led to significantly slower task completion times for the first trial block, but such differences were not found in the subsequent blocks. A similar pattern of findings occurred for the number of primary target errors, with significantly more errors occurring for the harder task, but only for the initial trial block, and not for the subsequent blocks. Given the nature of the more difficult task such results would have been expected, but what was surprising was the pace at which they diminished. Users appear to have learnt the mappings between primary and secondary targets in a single trial block (108 trials).

The difficult task, for the initial trial block, was associated with longer reaction time periods. This was presumably a consequence of the additional cognitive processing involved in determining the appropriate primary target. This may have involved subjects in alphabet rehearsal in order to determine the required primary target. Similarly, significantly longer periods of movement were associated with the difficult task for the first trial block only. This may indicate concurrent cognitive processing which leads to a detriment in movement time. Alternatively it may arise through an initial submovement towards an incorrect target requiring subsequent submovements towards the correct target which may result in longer movement times. However, this would lead to more submovements being made for the difficult task over the initial trial block, and this was not supported by the data (see table 8.5) where no significant task by block interaction for the number of submovements was found.

The experience of the user interacts in a counter-intuitive way with the time on target for the different task conditions. Experienced users showed no significant change in the time on target for the easy task ($P>0.05$, Tukey T 56.8). However, inexperienced users spent increasingly less time on target as a result of practice, with the time on target for the first block being significantly longer than that for the final block ($p<0.05$, Tukey T 56.8). These findings are consistent with those found earlier within this thesis (see section 7.3.1).

Interpreting the time on target for the difficult task condition is more challenging, especially for the inexperienced users. The results for the inexperienced users in the hard condition are not very easily explained, and convincing arguments to explain these findings cannot be developed from the literature on learning and skill acquisition.On the other hand the experienced users showed significant changes in time on target between the first block and the third block, and the first block and the last block. ($P<0.05$, Tukey T 56.8). The time on target in the hard task condition for experienced users converged with their time for the easy task condition. Why should initial longer times on target be associated with the difficult task condition for experienced users, and why does time on target reduce with subsequent practice? It was suggested in Chapter seven that good users were able, given conducive task conditions, to anticipate the button press. If this is so, then the additional cognitive demands of the difficult task condition may interfere with such anticipatory behaviour.

9.1.5.2 Experience

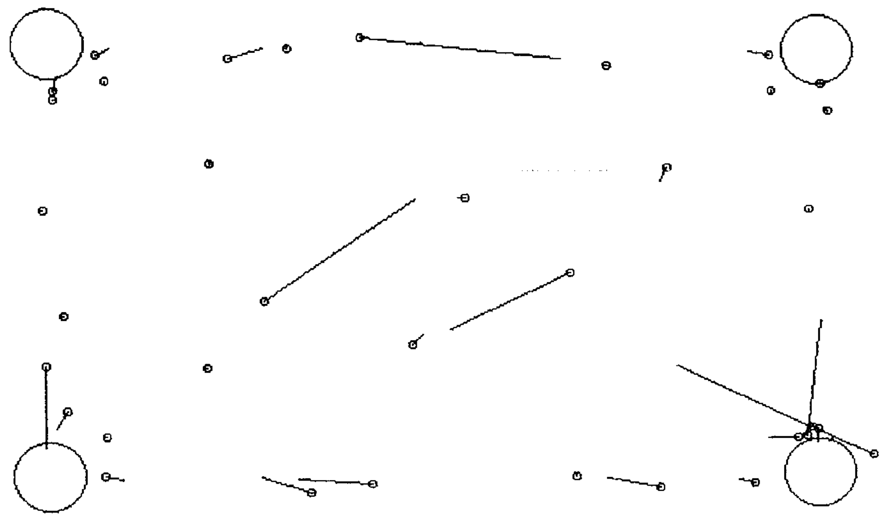
Experienced users were significantly quicker than inexperienced users, especially for the first block. However, the task completion times for the two groups had converged by the final block. There was no significant difference in error rates between experienced and inexperienced groups. Despite task completion times being significantly different between experienced and inexperienced users for all but the final block, reaction time was found to be significantly different between the two groups for first block only.

Inexperienced users took longer to move to the target than experienced users. Movement time, for the experienced users, improved more sharply from the first to second block than between the subsequent blocks. This was also true of the inexperienced users but their movement times converged to those of the experienced users. Experienced users spent significantly less time on target than inexperienced users (see 9.1.5.1 for a more detailed discussion).

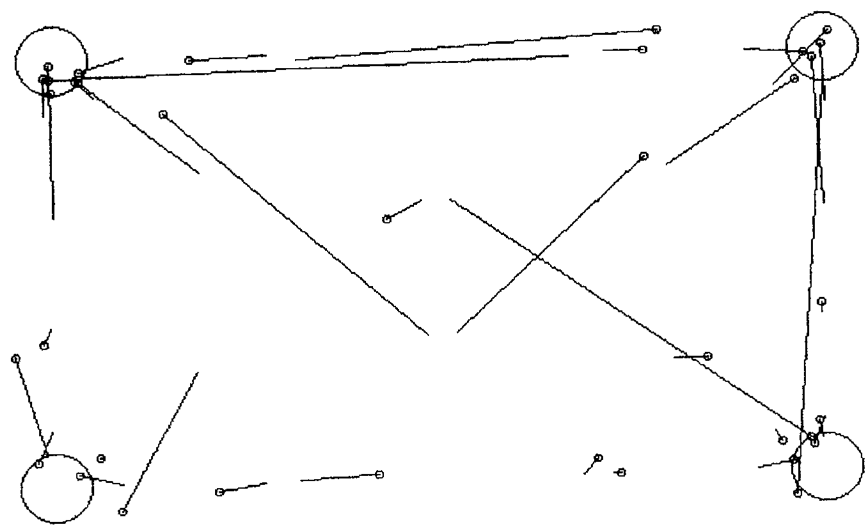
The differences between experienced and inexperienced users are consistent with those findings found earlier within this thesis (see section 7.3.1). Generally, the performances (task completion time and the number of submovements made) of experienced and inexperienced users improve through practice, with the inexperienced users' performance converging to that of the experienced users' performance. However, reaction time converges earlier than time on target which converges quicker than movement time. There is clearly still 'room for improvement' in movement time, and possibly time on target, for the inexperienced users when compared to the experienced users.

9.1.6 Some individual data

The above results, and those of chapter seven and eight, have dealt with grouped data. In order to obtain a better understanding of the submovements made during this experiment individual submovement traces are presented. Figures 9.8a & 9.8b show the movement traces of the first submovement for a novice mouse user made during this experiment. Figure 9.8a shows the first trial block first movements for the medium target sizes and Figure 9.8b shows the same target condition for the fourth block of trials.



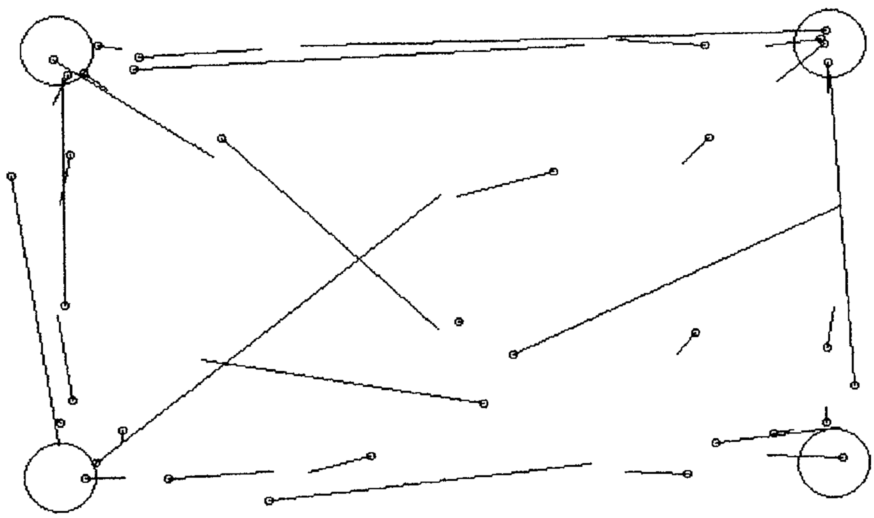
1st Trial Block (Inexperienced User)
Figure 9.8a



4th Trial Block (Inexperienced User)
Figure 9.8b

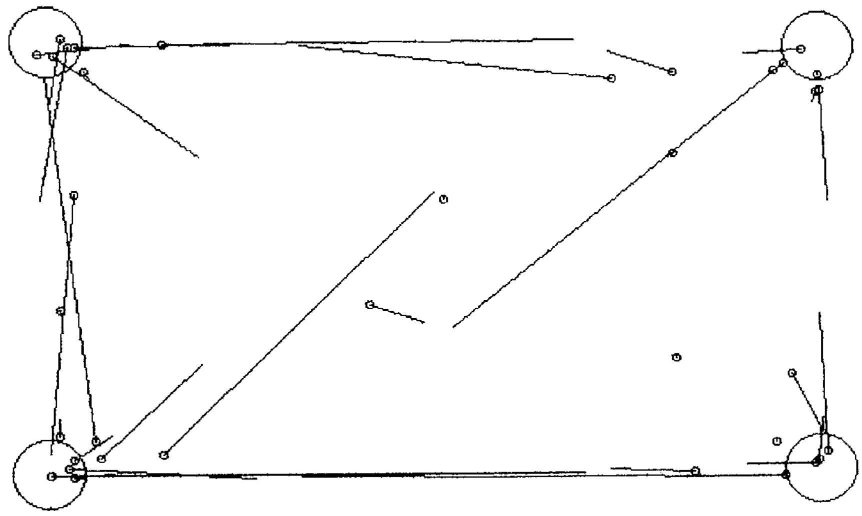
The first submovement is represented by the line, the circled end of the line represents the end of the first submovement and the non-circled end of the line indicates the start of the submovement.

Figures 9.8c & 9.8d show similar graphs for the same task conditions but for an experienced user. Figure 9.9a shows the submovement traces for the first block and 9.9b shows the movement traces for the final block.



1st Trial Block (Experienced User)

Figure 9.8c



4th Trial Block (Experienced User)

Figure 9.8d

Comparing the trials for first and last block of the novice user differences between the two diagrams can be seen. In figure 9.8a there are a number of inappropriately short amplitude movements, no first movements terminate within the target area, and an absence of relatively large amplitude movements. In figure 9.8b there are far less small amplitude movements, many of the movements terminate within the target area, there are a number of large movement amplitudes. It can be seen that the novice user initially made mostly short, largely erratic, and often unproductive first movements. In the forth block we find many examples of good first movements and far fewer poor movements.

Considering the experienced user (block one - figure 9.8c, and block four - figure 9.8d) it can be seen, that even for the first block, the proportion of 'good' first movements is higher than either the first or final block movements of the novice user. The final block of trials for the experienced user contains relatively few short movement amplitudes (which substantially undershoot), many 'holes in one', and many movements which almost hit the target.

Although distinctions in the level of individual movement skill have been drawn from these movement traces, establishing where subjects are in terms of the earlier stage models of skill acquisition (section 9.0.2) is problematic. It is influenced by the choice of dependent variable, for example the number of selection errors might yield a different picture of skill level than the proportion of poor movements. Furthermore, the stage models are vague with respect to how exactly one determines which skill stage the subject has reached.

Neither of these users were making consistently good or poor movements. The experienced user made far more good movements than the inexperienced user, but, none the less still made some relatively poor movements. This finding has similarities with that of Crossman (1959) with his cigar rolling study. Further, it suggests that there is considerable room for improvement in the experienced user's performance.

These findings raise questions about the conclusions, derived from averaged data, that can be made regarding movement characteristics. Earlier findings within this thesis suggested that subjects were consistently undershooting the target, except for small amplitude trials, by about 50%. However, many of the first submovements for the experienced user, and the novice user on her final block terminated within the target area. Of those which failed to terminate within the target area many might be considered to be 'a little too short'. However some of the submovements are sufficiently erratic for it to be impossible to determine their intended aim. These movement traces suggest that averaged data may yield misleading accounts of people's submovement behaviour and their acquisition of skill. Whilst averaged data might have suggested that people's first submovements were progressively improving with respect to diminishing errors of extent, the movement traces suggest that the proportion of good first submovements is increasing, but that poor first submovements are still being made. However, many movement studies are based on such averaged data. Where subjects have attained a more consistent performance then such averaged descriptions may not be so misleading. These findings indicate that an analysis based on an individual's submovement characteristics might be more appropriate than determining trends from averaged group data.

These movement traces lend support to the suggestion that subjects were reluctant to overshoot the target area. Even experienced subjects showed improvements in their task completion times (see section 8.1.5.2) and in at least one case improved upon their submovement characteristics. This particular experienced user reported using the mouse 360 days of the year. Given this level of experience, it is likely that they would be familiar with handling a mouse so any changes we observe are most likely to reflect their growing familiarity with the task.

CHAPTER TEN

COMPARING DEVICES

EXPERIMENTS FOUR & FIVE

This chapter describes two experiments, one comparing three mice, and the other comparing a mouse, mouse pen, and thumbwheel. All of the devices used within this chapter are considered to be functionally equivalent. However, differences between the three conventional computer mice, and between three different devices, suggest that functional equivalence does not ensure performance equivalence. Significant performance differences in reaction time, movement time, time on target, and the number of errors were found between the Optical mouse, used throughout the earlier experiments, and two rollerball type mice. Moreover, a significant difference in task completion time, for the medium and large target conditions, between the two rollerball devices was found. These findings suggests that generalizing from one input study, involving particular devices and tasks, may be problematic. Previous mouse device experience was found to significantly correlated with subsequent mouse device performance but not with subsequent thumbwheel or MousePen performance. It is suggested that the positive transfer of previous device experience to a different device could be used in implementing and input device taxonomies

10.0 Introduction

The device used in the previous experiments was of necessity an optical mouse (see section 5.4). However, the rollerball type device is probably the most commonly purchased mouse (Computer Shopper, 1991). Therefore a comparative study between the experimental Optical mouse and two rollerball type devices was undertaken which investigated the ecological validity of the previous findings. Furthermore, earlier within this thesis (see section 2.5), it was suggested that the diversity of computer mice implementations was problematic for the notion of a generic mouse input device. A comparison between the performance of three mice for the same input task would contribute to this debate.

Additionally, another experiment was conducted to investigate the differences in performance between two contemporary devices, suggested as alternatives to the mouse by their manufacturers (a thumbwheel & a mouse-pen), and a rollerball mouse which was used in the above experiments. This permitted the differences in performance between computer mice to be contrasted with the differences in performance between different devices, for similar input tasks. The thumbwheel and pen mouse were selected for their relatively low cost and widespread availability.

Given that rollerball type devices are being used within these experiments, with their associated slippage, it would not be appropriate to talk of device submovements (see section 5.4), so display pointer submovements will be considered instead.

In both comparative experiments a between subjects design for the different devices was selected. This was decided after considering the possible asymmetric transfer effects associated generally with within subjects designs (Poulton, 1973, 1982, Poulton & Freeman, 1966), and in particular for equipment testing (Poulton, 1969); and considering the increased time required of individual subjects that would result from such a design.

10.1 Experiment four

The aim of this experiment was to investigate people's use of three different mice. The devices selected were; the Optical PC Mouse (the device used for all the previous experiments), the Primax (SOP 029) rollerball type mouse, and the QS-159 rollerball type mouse. Each mouse had three buttons that were located towards the front of the device, on the side away from the user. All three mice transmitted their status data at the same rate via a serial link to the computer. Although the number of data bits encoding the mouse status was identical for all devices, their data formats were idiosyncratic necessitating amendments of the optical software device driver (see section 2.3) for the two rollerball devices. The devices varied in their shape and weight (Optical 98g, QS-159 112g, and Primax 110g).

The 'Device Resolution' parameters (see section 5.2.1) were determined for the two rollerball mice for relatively slow device movements (those associated with smaller amounts of slippage) made on the mouse pads that were subsequently used in this experiment. The input system gains of five (for both X and Y planes of movement) were then calculated using these device resolution parameters.

An opportunity sample of 61 first year undergraduate psychology students (18 male & 43 female of mean age 20.2 years, SD 4.1) performed the experimental task. All subjects had used a mouse within the past year, but most subjects reported using the mouse infrequently (less than once a month).

The task was based on that described in Chapter Six. However, subjects completed 72 instead of 108 trials and subjects were able to lift the mouse device without having to repeat the trial. The first 36 trials were taken as a practice session. Targets were presented in three blocks of 12 trials using the same size target, with each block having a different target size; small, medium, and large, with the movement amplitude being held constant across all trials. This movement amplitude corresponded to the large movement amplitudes used in the previous experiments. Subjects then repeated these 36 trials completing in 72 trials in total. The magnitude of movement amplitude and target sizes correspond to those described in Chapter Six.

A between subjects design was employed with the subjects having been randomly allocated to one of the three device groups; controlling for age, previous device experience, and sex of the subject. Data were only analysed for the last 36 trials. The data from the practice trials was not analysed.

10.1.1 Results

Table 10.0 and figure 10.0 illustrate the different task completion times for the three devices.

Task Completion Time by Device & Target Size

Variation Source	SS	DF	MS	F	P<
WITHIN CELLS	25960626.07	58	447597.0		
DEV	3734701.75	2	1867350.9	4.17	0.020
WITHIN CELLS	5034341.63	116	43399.5		
SIZE	18196133.11	2	9098066.6	209.64	0.001
DEV BY SIZE	1069609.11	4	267402.3	6.16	0.001
Total SS	53995411.67				

Table 10.0

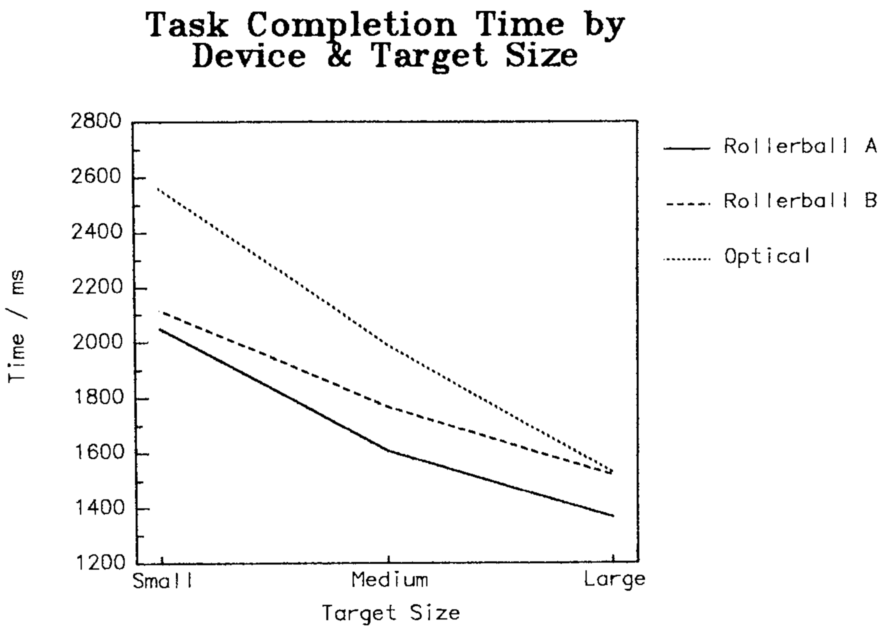


Figure 10.0

Significant main effects of device (6.9% of the total variance) and target size (33.7% of the total variance) on task completion time were found. Post-hoc tests revealed that the task completion time for Rollerball A was significantly less that of the Optical mouse (P<0.05, Tukey T 293.7). Task completion time reduced with larger target sizes for all devices. There was a significant device by target size interaction (2.0% of the total variance) which suggests that the differences in task completion time for the devices become greater with reducing target size. This was supported by post-hoc tests which found significant differences in task completion time; between the Optical mouse and the two rollerball mice for the small and medium target conditions, but between Rollerball A and the other devices for the large target condition (P<0.05, Tukey T 120.2). Furthermore, task completion times between Rollerball A and Rollerball B were significantly different for the medium task condition (P<0.05, Tukey T 120.2).

The Optical mouse was associated with more primary errors; Optical - 1.00, Rollerball A - 0.14, and Rollerball B - 0.19, ($P<0.05$, Chi^2 6.8, N=61), and more secondary errors; Optical - 2.42, Rollerball A - 0.67, and Rollerball B - 0.52, ($P<0.01$, chi^2 12.1, N=61) than either of the two rollerball devices. There were no significant differences in errors between the two rollerball devices.

Table 10.1 and figure 10.1 illustrate the differences in reaction time between the devices.

Reaction Time by Device & Target Size

Variation Source	SS	DF	MS	F	P<
WITHIN CELLS	3238444.99	58	55835.26		
DEV	452165.90	2	226082.95	4.05	0.023
WITHIN CELLS	1158730.83	116	9989.06		
SIZE	298557.98	2	149278.99	14.94	0.001
DEV BY SIZE	38073.33	4	9518.33	0.95	0.436
Total SS	5185973.03				

Table 10.1

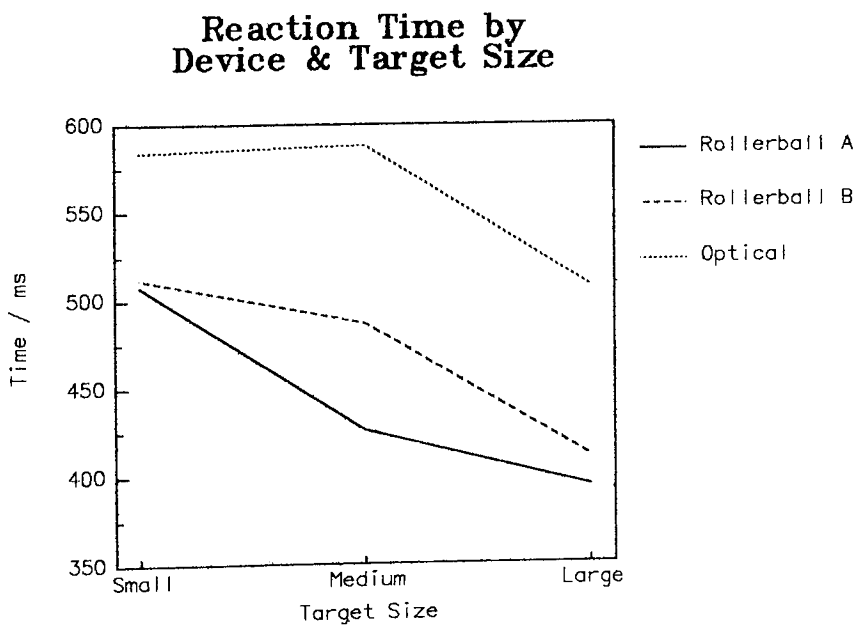


Figure 10.1

Significant main effects of device (8.7% of the total variance) and target size (5.8% of the total variance) on reaction time were found. Post-hoc tests found a significant difference in Reaction time, for all target sizes, between Rollerball A and the Optical Mouse (P<0.05, Tukey T 103.7). Reaction time tended to reduce for increased target size.

Table 10.2 and Figure 10.2 show illustrate the differences between devices for movement time.

Movement Time by Device & Target Size

Variation Source	SS	DF	MS	F	P<
WITHIN CELLS	8471073.76	58	146053.0		
DEV	566090.13	2	283045.1	1.94	0.153
WITHIN CELLS	2254904.50	116	19438.8		
SIZE	7840687.84	2	3920343.9	201.68	0.001
DEV BY SIZE	541231.15	4	135307.8	6.96	0.001
Total SS	19673987.38				

Table 10.2

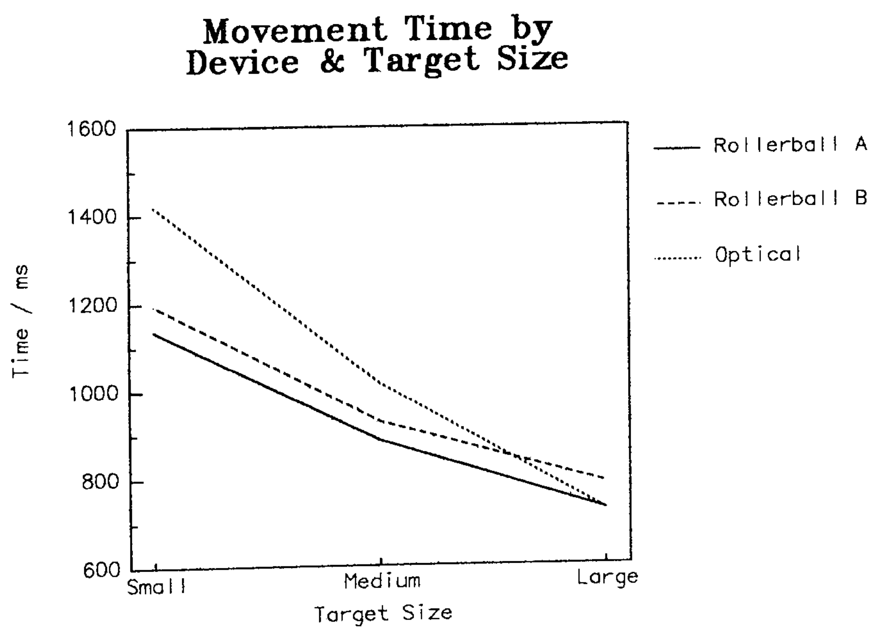


Figure 10.2

A significant effect of size (39.9% of the total variance) on movement time was found. There was a significant interaction effect between device & target size (2.8% of the total variance) on movement time which suggests that movement time for the devices converges for increased target size. This was supported by post-hoc test which found significant differences in movement time between the Optical mouse and the two rollerball devices for both the medium and small target conditions (P<0.05, Tukey T 80.5).

Table 10.3 and Figure 10.3 illustrate the device differences for time on target.

Time on Target by Device & Target Size

Variation Source	SS	DF	MS	F	P<
WITHIN CELLS	1980663.22	58	34149.37		
DEV	270764.66	2	135382.33	3.96	0.024
WITHIN CELLS	595785.52	116	5136.08		
SIZE	884681.46	2	442340.73	86.12	0.001
DEV BY SIZE	147897.21	4	36974.30	7.20	0.001
Total SS	3879792.07				

Table 10.3

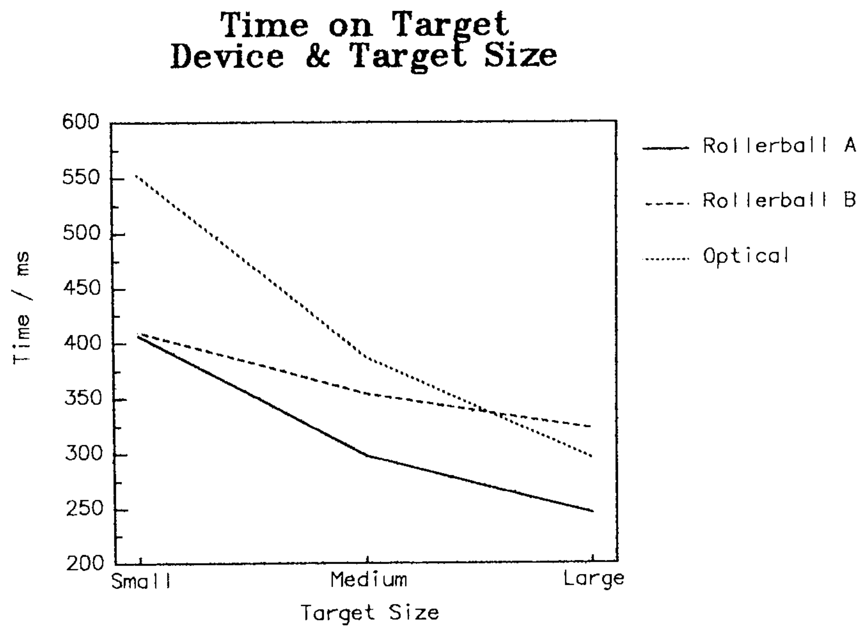


Figure 10.3

There were significant main effects of device (7.0% of the total variance) and target size (22.8% of the total variance) on time on target. Post-hoc tests found the time on target for Rollerball A to be significantly less than that of the Optical mouse (P<0.05, Tukey T 81.1).

The significant interaction between device & target size (3.8% of the total variance) suggests that there are greater differences in time on target between the devices for the small target condition. Post-hoc tests found time on target to significantly differ between; Rollerball A and Rollerball B for the medium and large target conditions, but not for the small target condition; between the Optical mouse and the two rollerball devices for the small target condition; and between the Optical mouse & Rollerball A for the medium and large target conditions.

Table 10.4 and Figure 10.4 illustrates the device differences for the number of submovements made during the movement time period.

Number of Submovements by
Device & Target Size

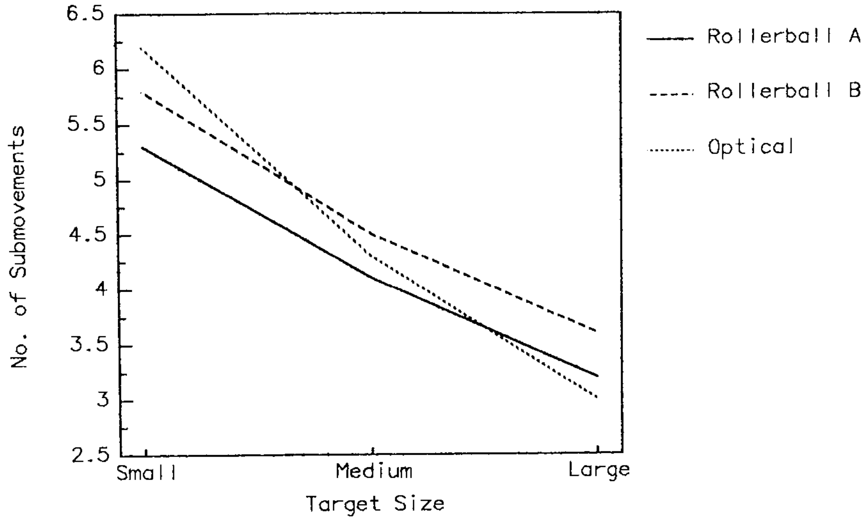


figure 10.4

No. of Submovements by Device & Target Size

Variation Source	SS	DF	MS	F	P<
WITHIN CELLS	158.74	58	2.74		
DEV	6.65	2	3.32	1.21	0.304
WITHIN CELLS	43.34	116	0.37		
SIZE	186.40	2	93.20	249.44	0.001
DEV BY SIZE	7.35	4	1.84	4.92	0.001
Total SS	393.48				

Table 10.4

There was a significant main effect of target size (47.4% of the total variance) on the number of submovement made. There was a significant device by target size interaction (1.9% of the total variance) which suggests that the Optical mouse requires disproportionately more submovements for the smaller task condition. Post-hoc tests found no significant differences in the number of submovements between the devices for the medium sized target condition (P<0.05, Tukey T 0.35) However, the differences in the number of submovements between all devices for the small target size condition, and between Rollerball B, and the other two devices, for the large target condition were significant (P<0.05, Tukey T 0.35).

Table 10.5 and Figure 10.5 illustrate the device differences for the number of non-movement episodes made.

No. of Non-Movements by Device & Target Size

Variation Source	SS	DF	MS	F	P<
WITHIN CELLS	26.61	58	0.46		
DEV	26.28	2	13.14	28.65	0.001
WITHIN CELLS	13.13	116	0.11		
SIZE	29.75	2	14.87	131.38	0.001
DEV BY SIZE	10.84	4	2.71	23.93	0.001
Total ss	106.61				

Table 10.5

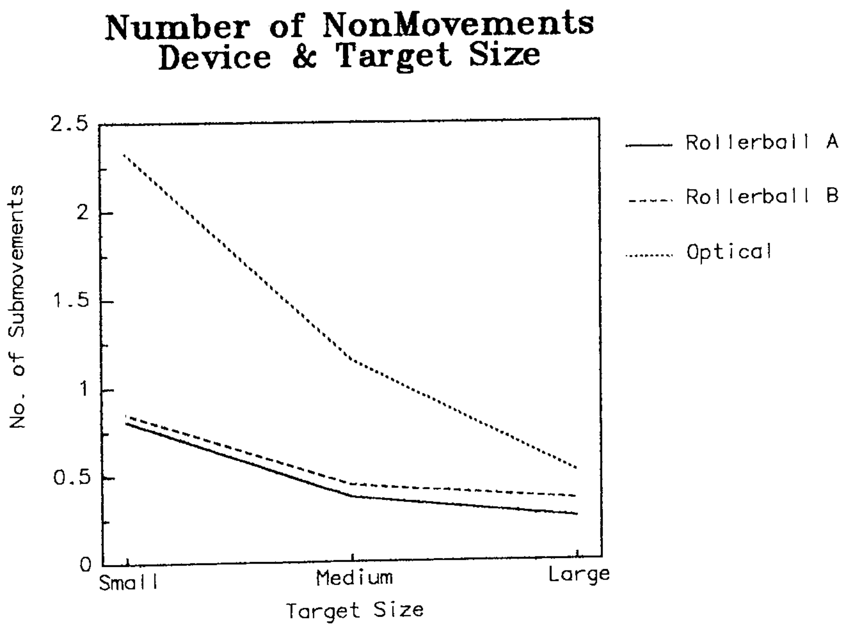


Figure 10.5

Significant main effects of device (24.7% of the total variance) and target size (27.9% of the total variance) were found. Post-hoc tests showed that the number of non-movements were significantly greater for the Optical mouse (P<0.05, Tukey T 0.3). The significant interaction between target size and device (10.2% of the total variance) indicates that the number of non-movements made by the Optical mouse disproportionately increase for reduced target sizes. Post-hoc tests found no significant difference in the number of non-movements between the two rollerball devices for any of the target conditions, but found the corresponding differences between the rollerball devices and the optical mouse were significant for all target levels with the exception of the difference between rollerball B and the optical mouse for the medium size target condition.

10.1.2 Discussion

Significant differences in both task completion time and the number of errors (primary & secondary) were found between devices. The pattern of significant differences in task completion time for the different devices was found to vary for different task conditions. This finding suggests that generalizing from one study using a particular task and mouse device, to other task situations involving a different task condition, and or mouse device is problematic. Taken as a whole, considering Figure 10.0, the two rollerball devices appear to be more similar in respect to performance time when compared to the optical device. However, if the means for the medium and large targets are considered in isolation from the small target condition, this pattern becomes ambiguous, and when only trials for the large target condition are considered the performance of the optical mouse and Rollerball B are seen to be more similar when compared to the performance of Rollerball A.

The finding that different physical implementations of a mouse device perform differently has been demonstrated in previous studies (Abernethy & Hodes, 1987; Hodes & Akaki, 1986) where differences in performance have been attributed to different device morphologies. However, there were differences between devices other than morphology and weight which may have also contributed to the differences in device performance. Some of the experienced users commented upon the 'feel' of the optical mouse, or the lack of it. This has been observed by other optical device users.

> *'What is noticeable, however, is that when anyone accustomed to mechanical mice has to use the machines [optical mice], a certain amount of swearing ensues. It's an acquired taste.'*
>
> Computer Shopper (p. 130, 1991).

The importance of the feel of a device has been stressed by other device researchers (Baecker & Buxton, 1987).

'Feel' is mostly associated with the various types of resistance encountered in moving a control (McCormick & Sanders, 1976). It is suggested that the proprioceptive feedback provided by the rollerball type devices might have improved their performance relative to the optical mouse.

Although most of the experimental subjects had reported having little previous device experience it is almost certain that such experience would have occurred with a rollerball type mouse. A negative transfer of previous device experience to the optical mouse may have accounted for the relatively poor performance of the optical mouse. However, this would not be consistent with the comparative performance time of Optical mouse for the large target conditions and the earlier findings where experienced rollerball device users did comparatively better than releatively less experienced device users.

The significant difference in reaction time between the optical mouse and Rollerball A is an interesting one, but one for which there was no a priori prediction. Given the slippage associated with the rollerball mice, it might have been expected that longer reaction times, rather than shorter ones, would have been associated with this device.

Why should people take longer to initiate movement with the Optical mouse? Some users commented upon the fact that the Optical device felt 'sticky' at the onset of motion. It could be that such subjective differences for the optical mouse in the onset of movement, and the relative lack of feedback in motion, contribute to the longer movement initiation period.

Movement time for the optical mouse was significantly greater than for the rollerball devices when small and medium sized targets were encountered. Inspection of the number of submovements made by each device (see Figure 10.4) indicates that the magnitude of the differences were relatively small and the rollerball devices and the optical device were not distinctly separated. However, the number of non-movement episodes made with the optical mouse are distinctly greater than the corresponding episodes of non-movement for the rollerball type devices (see Figure 10.5). Subjects using the rollerball type devices were able to sequence their submovements with less intervening non-movements than optical mouse users. This may have arisen from either the dynamics of the rollerball device towards maintaining motion, from the additional feedback provided by the rollerball devices, or from a combination of the two.

The time on target for the optical mouse was distinctly & significantly greater than that for the rollerball mice for the small target condition. However, as the target size increased distinctions in performance based on the type of mouse became less distinct. It is possible that the dynamic characteristics of the two rollerball devices facilitated the anticipation of target selection but as the targets became larger this advantage, relative to the optical device, became less. The difficulty in anticipation of the pointer's correct termination for the optical device may account for the higher number of incorrect selections associated with this device.

The similarity in performance between the two rollerball devices appears to have occurred due their similar reaction and movement times, and the dissimilarity in their performances through the differences between their time on target for the large and medium sized targets. The time on target for both Rollerball A and the Optical mouse reduced significantly for the large target condition when compared to the medium target condition, but this was not the case for the Rollerball B device. Why did this reduction in time on target between the medium and large target sizes not occur for the Rollerball B device? The construction of Rollerball B's buttons may have limited the degree to which the time on target could be reduced through anticipating a correct movement termination within the target area. The operation of Rollerball B's buttons could not have altered the trajectory of the mouse in such away as to increase the terminating uncertainty, as otherwise significantly more errors would have occurred with this device. The operation of the Rollerball B's button could not have simply added a constant delay to the time on target, as this would have introduced an equal delay for all target sizes.

10.2 Experiment 5

The aim of this experiment was to investigate people's use of three different input devices. The devices selected were; the Primax (SOP 029) rollerball type mouse, the Thumbelina thumbwheel (Appoint) type device, and a mouse pen (Appoint). All three devices transmitted their status data at the same rate via a serial link to the computer. Although the number of data bits encoding the device status was identical for all devices, their data formats were idiosyncratic necessitating amendments of the original software device driver (see section 2.3). The dimensions of the thumbwheel were as follows; height 0.9", width 1.7", and length 1.7", and it was operated in the 'hand held' fashion suggested by the manufacturer. It was billed as:

> *'The World's Most Versatile Mouse'*

by the device manufacturers.

The dimensions of the MousePen were as follows; height 0.7", width 0.7", and length 5.9". It was operated on the same mouse pad as the Primax mouse device. The MousePen device was billed as:

'The mouse you use like a pen.'

by the device manufacturers.

The 'Device Resolution' parameters (see section 5.2.1) of the three devices was determined using relatively slow device movements. The input system gains of five, or two, (for both X and Y planes of movement) were then calculated using these device resolution parameters.

An opportunity sample of 60 undergraduate students (30 male & 30 female of mean age 21.7 years, SD 2.05) performed the experimental task. The data for one subject was subsequently lost.

The task was based on that described in Chapter Six, and subjects under went all parts of the task and completed 108 experimental trials. However, subjects were able to lift the mouse device, or the MousePen, without having to repeat the trial.

A between subjects design was employed with the subjects having been randomly allocated to one of the three device groups; controlling for age, previous device experience, and sex of the subject. An extensive analysis of the data collected could have been carried out, as in Chapter Seven, but only analyses based on the influence of target size and device on task completion time and errors are presented within this thesis.

10.2.1 Results

Table 10.6 and Figure 10.6 illustrate the differences in task completion time between the three devices.

Target Size by Device

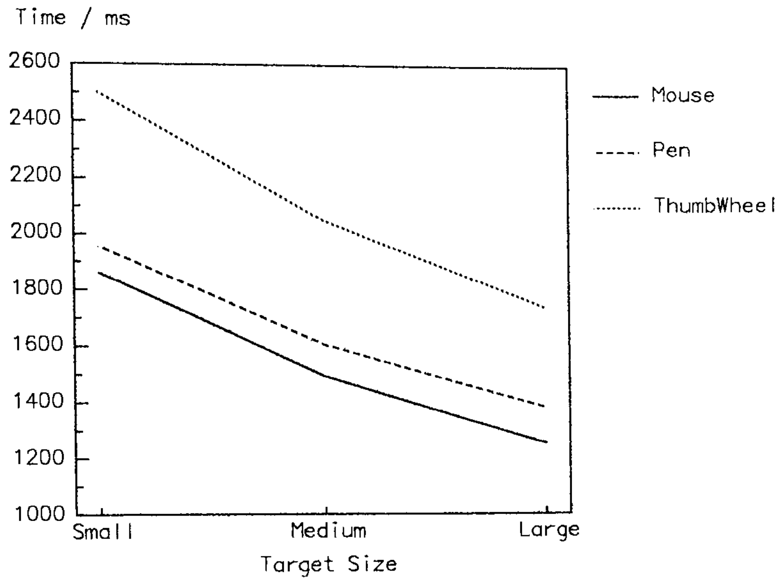


Figure 2

Figure 10.6

Task Completion Time by Device & Target Size					
Variation Source	SS	DF	MS	F	P<
WITHIN CELLS	8372381.43	56	149506.8		
DEV	10489389.32	2	5244694.7	35.1	0.001
WITHIN CELLS	1808809.55	112	16150.1		
SIZE	12424750.85	2	6212375.4	384.7	0.001
DEV BY SIZE	204326.39	4	51081.6	3.2	0.017
Total SS	33299657.54				

Table 10.6

Significant main effects of target size (37.3% of the total variance) and device (31.5% of the total variance) were found. Post-hoc tests revealed that task completion time for the thumbwheel was significantly greater than those associated with the other two devices; there was not a significant difference in task completion time between the MousePen and the Primax mouse (P>0.05, Tukey T 171.2).

The significant interaction of task completion between device and target size accounted for only 0.6% of the total variance and post-hoc tests failed to find any significant differences in task completion time between the different device and target conditions. There were no significant differences in the primary error rates between the devices.

Previous mouse device experience was found to significantly correlate with task completion time for the Primax mouse ($P<0.05$, $r = 0.49$, n 30) but not for task completion time of the other two devices.

10.2.2 Discussion

Although the tasks carried out in these two experiments are similar, the differences in instruction, practice, and the total number of trials completed make comparisons between the two studies difficult. For example the mean trial completion time for the Primax mouse in the first experiment was 1802 ms, and for the second study was 1537 ms. This difference was probably attributable to the trials for the second study including a number of 'easier' trials due to the additional levels of movement amplitude (medium and small movement amplitudes). However, relative comparisons within one study can be compared to relative device differences in the other. Although different levels of input gain were used within the second experiment no significant main effects of input gain, or device by gain interactions, on task completion time were found.

Therefore the task completion times for the two levels of gain for each device group were not separately analysed.

The task completion time of the MousePen was not significantly different from the Primax mouse, for all target conditions. Given that subjects had no previous experience of the MousePen it is probable that with equal levels of device experience that the task completion times for the Mousepen would have been less than those for the Primax mouse. Furthermore, as previous mouse experience was found to be significantly correlated with the Primax mouse task completion times, but not with those of the

MousePen, it is likely that the task completion times of inexperienced device users would be faster with the MousePen when compared to those for the same users with the Primax mouse device.

However, the task completion time of the Optical mouse in the first experiment was significantly different from that of the Rollerball A device. Furthermore there were significant differences in task completion time between the two rollerball devices for particular target conditions. The largest differences in task completion time between the two rollerball devices in the first experiment occurred for the medium target condition and had a value of 160 ms. Similarly, the largest difference in task completion time between the MousePen and the Primax mouse occurred for the large target condition and had a value of 131 ms. Thus the MousePen in the second experiment could be considered, in respect of performance, to be more like a mouse device than the Optical mouse, in experiment one when compared to the two rollerball devices, and for particular target conditions in experiment one, the Mousepen is more like the Primax mouse than the similarity between the two rollerball devices.

The use of the term 'mouse' by Appoint in describing their Thumbelina and MousePen products appears to be misleading. The thumbwheel device requires different body movements and operates in a fundamentally different way to a conventional mouse and the term 'mouse' when applied to this device is probably a misnomer. This is further supported by the large performance differences between this device and the other two devices in the second experiment. However, functionally it is equivalent to the conventional mouse, it can even be 'plugged in' in place of a conventional mouse without the need to alter any of the computer input software. However, functional equivalence is not the same as performance equivalence (Baecker & Buxton, 1987). Similarly, the MousePen has a functional equivalence to the conventional mouse, and its movement and operation are very similar to the conventional mouse. The term 'mouse' when applied to this device is more justified. Its performance, in experiment two when compared to the conventional mouse device, might be argued to be less dissimilar than the performance differences between the three conventional mice in the first experiment.

These findings confirm previous studies which have reported that device taxonomies, based on generic input device, are limited in their usefulness in solving the problems associated with the selecting the most appropriate method of input (see section 2.1.2). However such device taxonomies may be useful in predicting the transfer of device skill from previous device experience to a similar type of device, albeit a different implementation. In the second experiment previous mouse device experience was significantly correlated with subsequent mouse device performance, but was not found to be significantly correlated with subsequent MousePen or Thumbwheel performance. Further evidence of such a skill transfer can be found in the earlier chapters where previous mouse device experience was associated with subsequent Optical mouse experience (see sections 7.4.1 & 8.1.1). Given the rarity of Optical mice, this suggests that previous rollerball type mouse experience has positively transferred to the new type of mouse device.

10.3 Conclusions

The growth in the use of information technology has resulted in an expansion in the number of alternative devices which have an equivalence of input function. This equivalence of function has lead to a diversification of instances constituting the generic mouse class of input devices. However, such a functional equivalence of an input device does not necessarily lead to an equivalence in input performance. Moreover, even when the devices are operated in very similar ways, such as the optical and rollerball mice, differences in the technical mechanisms may result in significant performance differences. Furthermore, depending on the particular task conditions, input devices sharing the same functionality, same technical implementation, and having a similar morphology (such as rollerballs A & B in the first experiment) may significantly differ in their task performance.

Differences in performance between different devices are not necessarily confined to one component of the input task, but they are distributed throughout the input activity. That is, differences between submovement initiation, submovement execution, and target selection were found between different devices and within a single device for different task conditions.

These findings not only problematise the generalizabilty of the findings within this thesis, but they also question the external validity of most empirical studies investigating computer input. However, previous experience with a particular class of input device does appear to transfer to novel input tasks and to different implementations of the generic device. This is an important finding as it suggests a possible new direction for the classification of input devices.

CHAPTER ELEVEN

CONCLUDING REMARKS

11.0 Introduction

The original aim of this thesis was to determine how people moved the mouse device when acquiring and selecting displayed targets, and to utilize this knowledge in the design of input systems. However, in attempting to determine how mouse movements were made many, technological, methodological, and conceptual difficulties were encountered, the solution to which became central issues within this thesis.

The subsequent analyses of device submovements revealed an even greater complexity in input behaviour than might have been expected from earlier accounts of discrete motor movement. This complexity, in an example of discrete motor behaviour, provides a challenge to most theories of such motor acts (see section 3.2), and problematises the generalizability of the findings from previous input studies to different task contexts. This criticism can be applied, to some extent, to the studies within this thesis. The main issues raised within this thesis will now be reviewed.

11.1 Technical, methodological, & conceptual issues

This thesis has identified the problems in obtaining an accurate and precise timing of input events which appear to have been overlooked in many previous input studies (see section 5.4). The timing solutions proposed and developed within this thesis should assist future researchers in achieving accurate timing of computer input activities. Despite task completion time being the most commonly used performance index, it was found that it had not been identified in a consistent way between input studies (See sections 2.4.2.2.1.1 & 5.4), and subsequent reviewers had failed to take such inconsistencies into account. This thesis proposes a framework for the definition, and partitioning, of task completion time (see sections 4.1 & 4.5). Similarly, previous researchers investigating movement at the level of submovements have failed to provide an exposition of their submovement criteria and the possible consequences of selecting different alternative criteria (see section 4.7). These issues have been explored and discussed within this thesis. Furthermore, the analogy drawn between movement and language has provided a useful conceptualization of movement which facilitates the discussion, and actual activity, of submovement partition.

The development of a movement analyser, based on the principles of language parsing, has provided what is believed to be a powerful and versatile technique of analysing human movement (see section 4.5.1). It has yielded analysed movement data of a level and sophistication not found in the published literature. It has made possible the analysis of quantities of movement data at a level of detail which would not have been possible otherwise.

Input system gain, despite having been the subject of research attention, has often been misconceptualized (see section 2.4.2.3) and incorrectly defined in many earlier input device studies (see section 5.4). This thesis has clarified the issue of gain and has shown it to be inseparable from other task parameters.

11.2 Theoretical issues

This thesis has shown that many factors influence mouse performance in a complex way. Prescriptive theories of discrete motor movement are not able to deal with this complexity in a satisfactory way (see section 7.6.2). Additionally, most motor theories only consider movement behaviour from the perspective of a small number of performance variables (typically including movement time), and this reduces the behavioural complexity with which such theories have to deal. However, even if time were the only behavioural index, the complexity of the relationship between task completion time and contextual task factors is such that it challenges most theories of discrete motor acts. This complexity was found to increase when completion time was partitioned into subcomponents (see section 7.3).

Fitts' Law, and its derivatives were not found to be good predictors of movement time (see section 7.3.8.4). For a Fitts' Law formulation to account for the findings within the thesis it would have to accommodate: the effects of previous device experience and age on reaction time; the effects of age, device experience, target size, and movement amplitude on movement time; the effects of target size, device experience, and age on the time on target (see Table 7.4, and sections; 7.22, 8.4, & 9.1.5). Furthermore each of these terms would have to be modified with reference to task experience, task complexity, the location of the target on the display, and the particular device implementation. These findings were based on averaged group data, and the complexity could only increase if individual movement times were to be predicted.

Theories, such as the Iterative Corrections Model and the Stochastic Optimal Submovement Model, have not been fully supported by the findings of this thesis (see sections 7.3.8.5 & 7.3.8.6). Although the incidence of submovements increased with increasing task complexity, as measured by target size and movement amplitude, they were found to vary with a number of other factors such as previous device and task experience, and age. Moreover, such factors were also found to influence the submovement characteristics. Furthermore, an examination of some individual movement data (see section 9.1.6) suggests that submovement activity is relatively inconsistent, even for experienced device users. The finding that subjects were reluctant to overshoot the target has implications for theories which do not advocate an adaptable movement strategy which is determined, in part, by the task constraints.

Theories which are supported by averaged group data, but make assumptions regarding individual movement strategies can be misleading, and should be supported by individual movement data (see section 9.1.6). Similarly, despite task completion time having levelled off in the long term study, there were indications from individual movement traces that relatively poor submovements were frequently being made and that there was still room for considerable performance improvement. These movement traces also suggested that differences in overall performance between experienced and inexperienced users may have been largely accounted for by the proportions of 'good' movements made by the different experience groups. Although more subtle improvements in the movements made were also evident for both experience groups.

Most theories of discrete motor acts do not consider the possibility of incorrect target selection. Errors in target acquisition were probably of three types; selection of an incorrect target, device misoperation, and near misses, and evidence of the first and last instances were found within this study. Theories of discrete motor acts do not provide an account of such errors. This thesis suggests that near misses may occur through an incorrect anticipation of the movement's correct termination, and that the longer selection latencies associated with smaller targets are, at least in part, a reflection of the anticipatory processes. Movements which were indicative of subjects moving towards the wrong target were associated with shorter reaction times, which probably arose through the movement being prematurely initiated before adequate processing of the task demands had occurred (see section 7.5).

The diversity and 'resolution' of the performance measures taken, within this thesis, often revealed effects on performance which could have been overlooked, or 'masked', had a single performance measure such as task completion time been used. In the case of gain many of the differences between the two conditions would not have been discovered had just task completion time been used as the performance variable.

Given the diversity of functionally equivalent devices and the sensitivity of device performance to task related, and non-task related, factors the use of device taxonomies in determining the most suitable input device becomes more difficult process (see section 10.3). However, the findings from this study suggest that the type of mechanism (potentiometer wheels, optical, rollerball) used for translating device movement should be included in such taxonomies. Furthermore such taxonomies should also consider the positive, or negative, transfer of previous device experience to novel devices.

11.3 Implications for future design

The main implication of this research programme is that device performance is very sensitive to a number of factors, not all of which may be identifiable in advance of system implementation, and whose effects are not necessarily predictable. Differences have been found between two very similar mice, for the same input task, but only for some of the task conditions, and not others. Similarly, apparently different devices, can appear to perform equally well for all of the task conditions. This supports Buxton's belief that designs will have to be tested through actual implementations and prototyping (see section 2.1.3).

Two potentially useful findings, with respect to input system design, from the observation of the movement characteristics are:

- subjects predominantly undershoot the target area
- the amount of time spent on the target in non-movement

These movement characteristics, undershooting, and the time spent on target could be exploited in future input system design.

Targets located at the bottom of the display area were associated with longer task completion times. Although further research is needed to determine the external validity of this finding it has potentially important implications for task design. Similarly relatively poor and inexperienced device users are particularly disadvantaged for small targets and large movement amplitudes. Persistent input difficulty might be alleviated by avoiding these task conditions.

The mouse device was found to be suitable for most older users. However, there may exist a subpopulation of older people who would experience severe problems in using a mouse device. Older users took longer than younger users but were not found to be particularly disadvantaged for any of the task conditions.

11.4 Further work

There is still much analysis to be done on the data already collected. In particular case studies based on an individuals' movements should be undertaken.

Reanalysis of subjects' movement data should include the type of first movement made, the location of periods of non-movement, and a more detailed analysis for the different task locations. Movement data for the Mousepen and thumbwheel should be analysed at the submovement level of analysis. Subjects' movement data should be reanalysed using a syntactic rule which would facilitate an examination of movement corrections in more detail.

The two comparative studies should be repeated but the tasks should be the same to facilitate comparisons between the devices. Furthermore a comparative study between optical mice having different; shapes, weights, and button arrangements should be undertaken. Similarly a rollerball and optical mouse identical except for their mechanism should be compared.

Subjects' task experience, even for the 'longterm' study, cannot be considered as extensive. It could be the case that some of the variability in performance may reduce with extensive task experience. This could be achieved through extensive practice by subjects on the experimental task, or by observing people's mouse behaviour for an application package with which they are highly experienced.

Alternatively, the mouse driver software developed within this thesis could be installed onto a prospective subject's computer system, allowing their pointer movements to be recorded for subsequent analysis. It would not be possible to determine the task demands associated with this movement data, but this would also allow comparisons to be made between the findings from the experimental task and people's mouse movements for a 'real world' task.

Subjects who experience difficulty with the mouse should be sought and their device movements investigated in an attempt to determine the nature of the difficulties and possibly suggest solutions to their input problems.

REFERENCES

Abend, W., Bizzi, E., & Morasso, P., 1982, "Human arm trajectory formation", Brain, 105, 331-348.

Abernethy, C. N., and Hodes, D. G., 1987, "Ergonomically determined pointing device (mouse) design". Behaviour and Information technology, 6(3), 311-314.

Abrams, R. A., Kornblum, S., Meyer, D. E., & Wright, C. E., 1983, "Fitts' Law: optimization of initial ballistic impulses for aimed movements". Paper presented at the meeting of the Psychonomic Soc. San Diago: CA.

Adams, J. A., 1971, "A closed loop theory of motor learning". Journal of Motor Behaviour, 3, 111-149.

Adams, J.A., 1987, "Historical Review and Appraisal of Research on the Learning, Retention, and transfer of Human Motor Skills". Psychological Bulletin, 101 (4),41-74.

Alvey, , 1982, "A Programme for Advanced Information Technology": The Report of the Alvey Commitee. HMSO: London.

Andres, R. O., and Hartung, K. J., 1989, "Prediction of head movement time using Fitts' Law". Human Factors, 31(6), 703-713.

Angel, R. W., 1974, "Electromyography during voluntary movement. The two burst pattern.Electroencephalography and Clinical Neuropsysiology", 36, 493-498.

Angel, R. W., and Higgins, J. R., 1969, "Correction of false moves in pursuit tracking". Journal of Experimental Psychology, 82(1), 185-187.

Annett, J., Golby, C. W., & Kay, H., 1958, "The measurement of elements in an assembly task - The information output of the human motor system". The Quarterly Journal of Experimental Psychogy, 10(1), 1-11.

Arbib, M. A., 1980, "Interacting schemas for motor control". In: Stelmach G. and Requin J. (eds), Tutorials in Motor Behaviour, 71-81. Amsterdam: North-Holland.

Argyle, M., 1989, "The Social Psychology of Work". 2nd Ed, Pengiun, England.

Arnaut, L. Y., and Greenstein, J. S., 1986, "Optimising the touch tablet: The effects of control-display gain and method of cursor control". Human Factors, 28(6), 717-726.

Arnaut, L. Y., and Greenstein, J. S., 1990, "Is display/control gain a useful metric for optimising an interface". Human Factors, 32(6), 651-663.

Atkeson, C. G., and Hollerbach, J., 1985, Kinematic features of unrestrained vertical arm movements. Journal of Neuroscience, 5, 2318-2330.

Atkinson, M., Kilby, D., & Roca, I., 1982, "Foundations of General Linguistics". London: George Allen & Unwin.

Baecker, R. M., and Buxton, W.A.S, 1987, "Readings in Human Computer Interaction". Morgan Kaufmann: CA.

Barker, D., Carey, M. S., & Taylor, R. G., 1990, "Factors Underlying Mouse Pointing Performance". In E. J. Lovesy (Ed) Contempory Ergonomics, Procs. of the Ergonomics Society's 1990 Annual Conference, Taylor & Francis, 359-364.

Bartle, R., 1985, "Artificial Intelligence and Computer Games". London: Centry Comms.

Beggs, W.D.A, and Howarth, C. I., 1972, "The movement of the hand towards a target". Quarterly Journal of Experimental Psychology, 24, 448-453.

Beggs, W.D.A, and Howarth, C. I., 1972, "The accuracy of aiming at a target". Acta Psychologica, 36, 171-177.

Bennett, J.L., 1972, "The User Interface in Interactive Systems". Annual Review of Information Science and Technology, 7, 159-196.

Bernstein, N.A., 1967, "The Coordination and Regulation of Movement". London, Pergamon Press.

Birren, J.E., Woods, A.M., & Williams, M.V., 1980, "Behavioural slowing with age: Causes, organiztion, and consequences". In: Poon L. (ed), Ageing in the 1980s: Psychological Issues, American Psychological Association, Washington, 293-308.

Bizzi, E., and Abend, W., 1983, "Posture control and trajectory formation in single and multijoint arm movements". In: Desmedt J. (ed), Motor Control Mechanisms in Health and Disease, 31-45. New York: Raven.

Bizzi, E., Polit, A., & Morasso, P., 1976, "Mechanisims underlying achievement of final head position". Journal of Neurophysiology, 39, 435-444.

Blalock, H. M., 1972, "Social Statistics", 2nd Ed. McGraw-Hill: Tokyo.

Blasewitz, R. M., and Stern, F., 1982, "Microcomputer Systems". Hayden: NJ.

Book, W.F., 1925, "The Psychology of Skill". Clegg: New York.

Botwinick, J., and Thompson, L., 1976, "Practice of speeded response in relation to age, sex, and set". Journal of Gerontology, 22, 72-76.

Bouisset, S., and Lestienne, F., 1974, "The organization of a simple voluntry movement as analysed from its kinematic properties", Brain Research, 71, 451-457.

Bovens, N., and Brysbaert, M., 1990, "IBM PC/XT/AT and PS/2 Turbo Pascal timing with extended resolution". Behaviour Research Methods, Instrumentation, & Computers, 22(3), 332-334.

Bracker, J.S., and Pearson, J. N., 1986, "Worker obsolescence: The H.R. dilema of the 80s". Personnel Administrator, 31, 109-116.

Braunstein, M., and Anderson, N. S., 1961, "A comparison of the speed and accuracy of reading aloud and keypunching digits". IRE Trans. on Human Factors in Electronics, March, 56-57.

Bromley, D.B., 1988, "Human Ageing". Penguin: England.

Brooks, V. B., 1979, "Motor programs revisited". In: Talbott R. and Humphrey D. (eds), Posture and Movement: Perspective for Integrating Sensory and Motor Researchon the Mammalian Nervous System, 13-49. Raven Press.

Brown, P. J., 1979, "Writing Interactive Compilers and Interpreters". John Wiley & Sons.

Brownowski, J., 1960, "The Common Sense of Science". Penguin: England.

Bryan, W.L., and Harter, N., 1897, "Studies in The Physiology and Psychology of The Telegraphic Language", Psychological Review, 4, 27-53.

Buck, L., 1980, "Motor performance in relation to control-display gain and target width". Ergonomics, 23(6), 579-589.

Buhrer, M., Sparrer, B., & Weitkunat, R., 1987, "Interval timing routines for the IBM PC/XT/AT microcomputer family". Behaviour Research Methods, Instrumnetation, & Computers, 19, 327-334.

Bullock, D., and Grossberg, S., 1988, "Neural dynamics of planned arm movements: emergent invariants and speed-accuracy properties during trajectory formation". Psychological Review, 95(1), 49-90.

Bullock, M. I., and Harley, I. A., 1972, "The measurement of three-dimensional body movements by the use ofphotogrammetry". Ergonomics, 15(3), 309-322.

Burden, R. L., Faires, J. D., & Reynolds, A. C., 1981, "Numerical Analysis". 2nd ed. Boston: Prindle, Weber, & Schmidt.

Buxton, W, 1986, "There's more to interaction than meets the eye: some issues in manual input". In Norman, D. A. & Draper, S. W. (Eds), User Centred System Design: New Persoectives on Human Computer Interaction, Hillsdale, N.J: Lawrence Erlbaum Associates.

Card, S. K., English, W. K., & Burr, B. J., 1978, "Evaluation of mouse, rate-controlled isometric joystick, step keys, and test keys for text selection on a CRT". Ergonomics, 21(8), 601-613.

Card, S. K., Moran, T. P., & Newell, A., 1980, "The Keystroke-Level Model for User Performance Time with Interactive Systems". Comm. of the ACM, 23(7), 396-410.

Carey, M. S., 1985, "Selection criteria for input devices". Aston University, Applied Psychology Department, Interim Report.

Carlton, L. G., 1980, "Movement control characteristics of aiming responses". Ergonomics, 23(11), 1019-1032.

Carlton, L. G., 1981, "Processing visual feedback information for movement control". Journal of Exp. Psychology: Human perception and performance, 7(5), 1019-1030.

Carlton, M. J., 1983, "Amending movements: the relationship between degree of mechanical disturbance and outcome accuracy". Journal of Motor Behaviour, 15, 39-62.

Carroll, J. M., and Campbell, R. L., 1989, "Artifacts as psychological theories: the case of human-computer interaction". Behaviour and Information Technology, 8(4),247-256

Carroll, J., Young, R., & Long, J., 1991, "HCI Theory on Trial". In Robertson S. P., Olson G. M., and Olson R. J. (Eds), CHI '91, Reaching Through Technology, Procs. of Human Factors in Computing Systems, New Orleans: 1991, Addison-Wesley, 399-401.

Carroll, Lewis, 1865, "Alice's Adventures in Wonderland".

Charness, N., 1993, "Personal Communication", Feburay 23rd. Ontario, Canada.

Charness, N., and Bosman, E.A., 1990, "Human Factors and Design for the Older Adult". In: Birren J.E. & Schaie K.W. (eds), Psychology of Aging, 3rd ed., 446-463, Academic Press: San Diego.

Charness, N., Bosman, E., & Zandri, E., 1988, "Computer technology and age". Poster presented at the Atlanta Conference on Cognitive Ageing, April.

Checkland, P., 1976, "Systems Thinking, Systems Practice". John Wiley & Sons.

Chomsky, N., 1957, "Syntactic Structures". Moulton: The Hague.

Cochrane, R., and Duffy, J., 1974, "Psychology and scientific method". Bulleten of the British Pyschological Society, 27, 117-121.

Colley, A. M., 1989, "Cognitive motor skills". In: Holding D. (ed), Human Skills, 229-248, John Wiley and Sons.

Computer Shopper, , 1991, "Whats the point?". Computer Shopper, August, 130-145

Cooke, J. D., and Diggles, V. A., 1984, "Rapid error correction during human arm movements: evidence for central monitoring". Journal of Motor Behaviour, 16, 348-363.

Creeger, C. P., Miller, K. F., & Paredes, D. R., 1990, "Micromanaging time: measuring and controlling timing errors in computer-controlled experiments". Behaviour Research Methods, Instruments, & Computers, 22(1), 34-79.

Crosbie, J., 1989, "A simple Turbo Pascal 4.0 program for millisecond timing on the IBM PC/XT/AT". Behaviour Research Methods, Instruments, & Computers, 21(3), 408-413.

Crosbie, J., 1990, "The Microsoft mouse as a multipurpose response device for the IBMPC/XT/AT". Behaviour Research Methods, Instruments, & Computers, 22(3), 305-316.

Crossman, E. R., 1959, "A theory of the acquisition of speed skill". Ergonomics, 2, 153-166.

Crossman, E.R., and Goodeve, P.J., 1963, "Feedback control of hand movement and Fitts' Law". Paper preseneted at the meeting of the Experimental Psychology Society, July 1963,Oxford, Engalnd.

Crossman, E. R., and Goodeve, P. J., 1983, "Feedback control of handmovement and Fitt's Law". Quarterly Journal of Experimental Psychology, 35A, 251-278.

Czaja, S.J., 1988, "Microcomputers and the elderly". In: Hellander (ed), Handbook of Human Computer Interaction, 581-598, Elsvier Science: North Holland.

Czaja, S.J., Hammond, K., Blascovich, J.J., & Swede, H., 1986, "Learning to use a word-processing system as a function of training strategy". Behaviour and Information Technology, 5(3), 203-216.

Czaja, S.J., Hammond, K., Blascovich, J.J., & Swede, H., 1989, "Age related differences in learning to use a text-editing system". Behaviour and Information Technology, 8(4), 309-319.

Davies, D.R., Glendon, I., Stammers, R.B., Matthews, G., & Taylor, R.G. 1992, "Age, health status and the acquisition of office technology skills". Final Report to the E.S.R.C., Aston University, Birmingham.

Deininger, R. L., Billington, M. J., & Michaels, , 1968, "Operator Speed and Accuracy with Literal and Combination Codes for Processor Input"., IEEE Trans. on Man-Machine Systems, 9(1),11-14.

Deininger, R. L., Billington, M. J., & Riesz, R. R., 1966, "The Display Mode and the Combination of Sequence Length and Alphabet Size as Factors in Keying Speed and Accuracy". IEEE Trans. on Human Factors in Electronics, 7(3),110-115.

DeJong, J. R., 1957, "The effects of increasing skill on cycle time and its consequencesfor time standards". Ergonomics, 1, 51-60.

Denton, F.T., Feaver, C.H., & Spencer, B.G., 1986, "Prospective aging of the population and its implications for the labour force and government expenditures". Canadian Journal of Aging, 5, 75-95.

Diaper, D., 1987, Satisfying Goals: "An Introduction to People and Computers III". In Diaper D. and Winder R.(Eds), People and Computers III, Procs of the 3rd Conference of the British Computer Society Human-Computer Interaction Specialist Group, Exeter, 1987, Cambridge University Press, 1-8.

Dooijes, E. H., 1983, "Analysis of handwriting movements", Acta Psychologica, 54, 99-114.

Easton, T. A., 1978, "Coordinative structures - the basis for the motor program". In: Landers D. and Christina R. (eds), Psychology of Motor Behaviour and Sport. Champaign, IL,: Human Kinetics.

Edmonds, E. A., and Green, T.R.G, 1984, "The ergonomics of the user interface". Behaviour & Information Technology, 3(2), 97-99.

Edwards, E., 1965, "Research note on the study of hand motions", Perceptual and Motor Skills, 20, 1098.

Egan, D.E., and Gomaz, L.M., 1982, "Characteristics of people who can learn to use computer text editors: Hints for text editor design and training". In Proceedings of the ASIS Annual Meeting, 19, 75-79.

Egan, D.E., and Gomez, L.M., 1985, "Assaying, isolating and accomodating individual differences in learning a complex skill". Individual Differences in Cognition, Vol. 2, Accademic Press: New York.

Elias, P.K., Elias, M.F., & Gage, P., 1987, "Acquisition of word processing skills by younger, middle aged, and older adults". Psycholgy and Ageing, 2, 340-348.

English, W. K., Engelbart, D. C., & Berman, M. L., 1967, "Displpay-selection techniques for text manipulation". IEEE Trans.on Human Factors, 8(1), 5-15.

Epps, B. W., 1986, "Comparison of six cursor control devices based on Fitts' Law models". In D. J. Obourne (Ed) Contempory Ergonomics, Procs. of the Erogonomics Society's Annual Conference, 327-331.

ESPRIT, , 1983, "Project ESPRIT", 1983, Report of the EEC, Brussels.

Ewing, J., Mehrabanzad, S., Sheck, S., Ostroff, D., & Shneiderman, B. 1986, "An experimental comparison of a mouse and arrow-jump keys for an interactive encyclopedia". Int. J. Man-Machine Studies, 24, 29-45.

Falduto, L., and Baron, A., 1986, "Age related effects of practice and task complexity on card sorting". Journal of Gerontolgy, 41(5), 659-661.

Feldman, A. G., 1986, "Once more on the equilibrium-point hypothesis (model) for motor control". Journal of motor behaviour, 18, 17-54.

Ferguson, G. A, 1966, "Statistical Analysis in Psychology and Education", 2nd Ed. McGraw-Hill: London.

Fitch, H. L., Tuller, B., & Turvey, M. T., 1982, "The Bernstein perspective: III tunning of coordinative structures with special reference to perception". In Kelso J. (ed), Human Motor Behaviour, 271-282, Lawrence Erlbaum: NJ.

Fitts, P. M., 1954, "The information processing of the human motor system in controlling the amplitude of movement". Journal of Experimental Psychology, 46, 381-391.

Fitts, P.M., 1964, "Perceptual-Motor Skill Learning". In: Melton A.W. (ed), Categories of Human Learning, 243-285, New York, Academic Press.

Fitts, P. M., and Peterson, J. R., 1964, "Information capacity of discrete motor responses". Journal of Experimental Psychology, 67, 103-112.

Fleischer, A. G., and Becker, G., 1986, "Free hand-movements during the performance of a complex task". Erogonomics, 29(1), 49-63.

Fleischer, A. G., and Lange, W., 1983, "Analysis of hand movements during the performance of positioning tasks". Ergonomics, 26(6), 555-564.

Foley, J. D., 1980, "Methodolgy of Interaction". In Guedj R. A., tenHagen P.J.W., Tucker H. A., and Duce D. A. (Eds), Mehtodolgy of Interaction. IFIP working conference, France: 1979, Amsterdam: North-Holland.

Foley, J. D., Wallace, V. L., & Chan, P., 1984, "The Human Factors of Computer Graphics Interaction Techniques". IEEE Computer Graphics and Applications, 4(11), 13-48.

Franks, I.M., and Wilberg, R.B., 1982, "The Generation of Movement Patterns During The Acquisition of a Pursuit Tracking Task". Human movement science, 1, 251-272.

Gaines, B. R., 1984, "From Ergonomics To The Fith Generation: 30 Years of Human-Computer Interaction Studies". In: Shackel B. (Ed) Human-computer Interaction, Interact '84, 1-5. Amsterdam: North-Holland.

Galen, G. P., and Teulings, H. L., 1983, "The independent monitoring of form and scale factors in handwriting". Acta Psychologica, 54, 9-22

Gan, K. C., 1985, "Ballistic arm movement; its definition, characteristics and mathematical modelling", PhD thesis, University of Melbourne.

Gan, K. C., and Hoffmann, E. R., 1988, "Geometrical conditions for ballistic and visually controlled movements". Ergonomics, 31(5), 829-839.

Garfein, A.J., Schaie, K.W., & Willis, S.L., 1988, "Microcomputer proficiency in later-middle aged and older adults: Teaching old dogs new tricks". Social Behaviour, 13, 131-148.

Genter, D. R., 1987, "Timing of skilled motor performance: tests of the proportional duration model". Psychological Review, 94(2), 255-276.

Gentile, A.M., 1972, "A Working Model of Skill Acquisition With Application to Teaching".Quest, monograph XVII, 2-23.

Gibbs, C. B., 1962, "Controller design: Interactions of controlling limbs, time lags, and gains in positional and velocity systems". Ergonomics, 5, 385-402.

Gibbs, C. B., 1965, "Probability learning in step-input tracking". British Journal of Psychology, 56, 233-242.

Gist, M., Rosen, B., & Schwoerer, C., 1988, "The influence of training method and trainee age on the acquisition of computer skills". Personnel Psychology, 41, 255-265.

Glass, G. V., Peckham, P. D., & Sanders, J. R., 1972, "Consequences of failure to meet assumptions underlying the analysis of variance and covariance". Reviews of Educational Research, 42, 237-288.

Glauthier, T. J., 1967, "Computer Time Sharing: its origins and development". Computers and automation, 16, 23-27.

Glencross, D. J., 1977, "Control of skilled movements". Psychologicl Bulletin, 84, 14-29.

Glencross, D. J., and Barrett, N., 1989, "Discrete movements". In: Holding D. (ed), Human Skills, 107-145, John Wiley and Sons: Chichester.

Goodwin, N. C., 1975, "Cursor positioning on an electronic display using lightpen, lightgun, or keyboard for three basic tasks". Human Factors, 17(3), 289-295.

Greeger, C. P., Miller, K. F., & Paredes, D. R., 1990, "Micromanaging time: Measuring and controlling timing errors in computer controlled experiments". Behaviour, Research Methods, & Instrumentation, 22(1), 34-79.

Greene, P. H., 1972, "Problems of organization of motor systems". In: Rosen R. and Snell F. (eds), Progress in Theoretical Biology, Vol. 2., New York: Academic Press.

Greenstein, J. S., and Arnaut, L. Y., 1988, "Input Devices". In Helander M. (Ed), Handbook of Human-computer Interaction. Elsevier: North-Holland.

Greer, K., 1984, "Physiology of motor control". In: Smyth M. and Wing A. (eds), The Psychology of Human Movement. Accademic Press, Inc: London.

Gurley, B. M., and Woodward, C. E., 1959, "Light-pen links computer to operator". Electronics, 32, 85.

H.S.E., 1983, "Visual Display Units". H.M.S.O.

Haller, R., Mutschler, H., & Voss, M., 1984, "Comparison of input devices for correction of typing errors in office systems". In: Shackel B. (ed) Human-Computer Interaction, Interact '8 4, (2), 218-223. Amsterdam: North-Holland.

Hartley, A.A., Hartley, J.T., & Johnson, S.A., 1984, "The older adult as a computer user". In: Robinson P.K. and Birren J.E. (eds), Ageing and Technological Advances, 347-348, New York: Plenum.

Hay, L., 1979, "Spatio-temporal analysis of movements in children: motor programs versus feedback in the development of reaching". Journal of Motor Behaviour, 11, 189-200.

Hay, L., Bard, C., Fleury, M., & Teasdale, N., 1991, "Kinematics of aiming in direction and amplitude: a developmentalstudy". Acta Psychologica, 77, 203-215.

Heathcote, A, 1988, "Screen control and timing routines for the IBM microcomputerfamily using a high level language". Behaviour Research Methods, Instruments, & Computers, 20(3), 289-297.

Helander, M., 1988, "Handbook of Human Computer Interaction". North Holland: Elsevier Science.

Higgins, J. R., and Angel, R. W., 1970, "Correction of tracking errors without sensory feedback". Journal of Experimental Psychology, 84(3), 412-416.

Hill, G. W., Gunn, W. A., Martin, S. L., & Scwartz, D. R., 1991, "Perceived Difficulty and User Control in Mouse Usage". Procs. of the Human Factors Soc. 35th Annual Meeting, 295-299.

Hodes, D., and Akagi, K., 1986, "Study, design, and development of a mouse". In Procs. of the Human Factors Society 30th Annual Meeting, Santa Monica, CA: Human Factors Soc.

Hofstadter, D. R., 1979, "Godel, Escher, Bach: An Eternal Golden Braid". Penguin: New York.

Holding, D. H., 1989, "Human Skills". 2nd ed. John Wiley & Sons: Chichester.

Howarth, C. I., and Beggs, W.D.A, 1971, "The relationship between speed and accuracy of movement aimed at a target". Acta Psychologica, 35, 207-218.

Hulstijn, W., and Galen, G. P., 1983, "Programming in handwriting: reaction time and movement time as a function of sequence length". Acta Psychologica, 54, 23-49.

Jacob, R.J.K, Leggett, J.J., Myers, B.A., & Pausch, R., 1993, "Interaction styles and input/output devices". Behaviour & Informatition Technology, 12(2), 69-79.

Jagacinski, R. J., Repperger, D. W., Moran, M. S., Ward, S. L., & Glass, B. 1980, "Fitts' Law and the microstructure of rapid discrete movements". Journal of Experimental Psychology: Human Perception and Performance, 6(2), 309-320.

Johnson, E. A., 1967, "Touch displays: a programmed man-machine interface". Ergonomics, 10, 271-277.

Johnson, P., 1984, "The Acquisition of Skill". In: Smyth M. & Wing A. (eds), The Pyschology of Human Movement, 215-239, Academic Press, Inc: New York.

Jordan, T.C., and Rabbitt, P.M.A, 1977, "Response times to stimuli of increasing complexity as a function of ageing". British Journal of Psychology, 68, 189-201.

Kahneman, D., and Tversky, A., 1973, "On the psychology of prediction". Psychological Review, 80, 237-251

Karat, J., McDonald, J. E., & Anderson, M., 1986, "A comparison of menu selection techniques: touch panel, mouse and keyboard". Int. J. Man-Machine Studies, 25, 73-88.

Katter, R. V., 1969, "On-line interfaces for document information systems: considerations for the biomedical communications network". System Development Corperation. Santa Monica: California. Technical memo 4494/000/00.

Keele, S. W., and Posner, M. I., 1968, "Processing of visual feedback in rapid movements". Journal of Experimental Psychology, 77(1), 155-158.

Keele, S. W., 1968, "Movement control in skilled motor performance". Psychological Bulletin, 70(6), 387-403.

Keele, S. W., 1982, "Learning and control of coordinated motor patterns: the programming perspective". In: Kelso J. (ed), Human Motor Behaviour, 161-188, Lawrence Erlbaum: NJ.

Keirsey, D., and Bates, M., 1984, "Please understand me". Gnosology Books, Prometheus Nemesis.

Keller, F.S., 1958, "The Phantom Plateau". Journal of experimental analysis of behaviour, 1, 1-13.

Kelley, D. L., 1971, "Kinesiology". NJ: Prentice-Hall Inc.

Kelso, J.A.S, 1982, "Perspectives and issues in motor behavior". In: Kelso J. (ed), Human Motor Behavior, 3-62, Lawrence Erlbaum: NJ.

Kelso, J.A.S, and Holt, K. G., 1980, "Exploring a vibratory systems analysis of human movement production". Journal of Neurophysiology, 43, 1183-1196.

Kelso, J.A.S, and Stelmach, G. E., 1976, "Central and peripheral mechanisims in motor control". In: Stelmach G. (ed), Motor Control: Issues and Trends. Academic Press: NY.

Kelso, J.A.S, and Tuller, B., 1981, "Toward a theory of apractic syndromes". Brain and Language, 12, 224-245.

Klapp, S. T., 1975, "Feedback versus motor programming in the control of aimed movements". Journal of Experimental Psychology. Human Perception and Performance, 1, 147-153.

Klapp, S. T., 1980, "The memory drum theory after twenty years: comments on Henry's note". Journal of Motor Behaviour, 12(2), 169-171.

Klapp, S. T., and Greim, D. M., 1979, "Programed control of movements revisited: the role of target visability and symetry". Journal of Experimental Psychology: Human Perception and Performance, 5, 509-521.

Klapp, S. T., and Rodriguez, G., 1982, "Programming time as a function of response duration: a replicationof 'dit-dah' without possible guessing artifacts". Journal of Motor Behavior, 14(1), 46-56.

Kosslyn, S. M., 1980, "Image and Mind". Cambridge, Mass: Harvard University Press.

Kosso, P., 1989, "Sciences and Objectivity". Journal of Philosphy, 86, 245-257.

Kvalseth, T. O., 1977, "A generalized model of temporal motor control subject to motor constraints". Ergonomics, 20(1), 41-50.

Laabs, G. J., and Simons, R. W., 1981, "Motor memory". In: Holding D. (ed), Human Skills, 119-151. Chichester: John Wiley and Sons.

Lancaster, D., 1982, "Micro Cookbook". Howard W. Sams & Co, Inc: Indiana.

Lanczos, C., 1967, "Applied Analysis". London: Pitman.

Langolf, G. D., Chaffin, D. B., & Foulke, J. A., 1976, "An investigation of Fitts' Law using a wide range of movement amplitudes", Journal of Motor Behaviour, 8(2), 113-128.

Lashley, K. S., 1917, "The accuracy of movement in the absence of excitation from the moving organ". The American Journal of Physiology, 43, 169-194.

Laycock, J., and Peckham, J. B., 1980, "Improved piloting performance whilst using direct voice input". R.A.E., M.O.D. Technical Report 80019, Feb. 1980.

Lazovick, P. B., Trost, J. C., Reickord, A. W., & Green, R. S., 1961, "A Versatile man-machine communications console". In the Procs. of the Joint Computer Conference, Washington, AFIPS, 20, 166-173. Spartan: Washington.

Leonard, J.A., 1952, "Some experiments on the temporal relation between information and action". Unpublished thesis. Cambridge University Library.

Leonard, J. A., and Newman, R. C., 1965, "On The Acquisisition and Maintenance of High Speed and High Accuracy in a Keyboard Task". IEEE Trans. on Human Factors, 6(3), 281-304.

Lestienne, F., 1979, "Effects of inertial load and velocity on the breaking process of voluntary limb movements". Experimental Brain Research, 35, 407-418.

Lewis, C. H., 1990, "A Research Agenda for the Nineties in Human-Computer Interaction". Human-Computer Interaction, 5, 125-143.

Lichtenberger, W, and Pirtle, M. W., 1967, "A facility for experimenting in man-machine interaction". In Procs.of the Fall Joint Computer Conference, AFIPS, 27(1), 589-598. Spartan: Washington.

Licklider, J.C.R, 1960, "Man-computer symbiosis". IRE Trans. Human Factors in Electronics, March, 1, 4-11.

Licklider, J.C.R, and Clark, W. E., 1962, "Man-machine cooperatrion". AFIPS, (21), 113-139.

Lin, M. L., Radwin, R. G., & Vanderheiden, G. C., 1992, "Gain effects on performance using a head-controlled computer input device". Ergonomics, 35(2), 159-175.

Long, J., and Whitefield, A., 1989, "Cognitive Ergonomics and Human Computer Interaction". Cambridge University Press: England.

Mack, R., and Montaniz, F., 1991, "A Comparison of Touch and Mouse Interaction Techniques for a Graphical Windowing Software Environment". Procs. of the Human Factors Soc. Annual Meeting, 286-289.

MacKenzie, S., 1989, "A note on the information-theoretic basis for Fitts' Law". Journal of Motor Behaviour, 21(3), 323-330.

MacKenzie, I. S., Sellen, A., & Buxton, W., 1991, "A Comparison of Input Devices in Elemental Pointing and Dragging Tasks". Procs. of Human Factors in Computing Systems, CHI '91, Robertson, S.P., Olson, G. M., Olson, J. S. (Eds), 161-166.

Magill, R. A., 1985, "Motor Learning, Concepts & Applications". 2nd Ed., W.C. Brown: Iowa.

Magill, R.A., 1991, "Motor Learning, Concepts & Applications". 4th Ed., W.C. Brown: Iowa.

Marsden, C. D., Merton, P. A., & Morton, H. B., 1972, "Servo action in human voluntary movement". Nature, 238, 140-143.

McCormik, E. J., and Sanders, M.S., 1976, "Human factors in engineering and design". New York:McGraw-Hill.

McNeil, K. A., Kelly, F. J., & McNeil, J. T., 1975, "Testing Research Hypotheses using multiple linear regression". Southern Illinois University Press.

Medawar, P. B., 1967, "The Art of The Soluble". Methuen & Co. Ltd: London.

Medawar, P. B., 1984, "The Limits of Science". Oxford University Press,

Megaw, E. D., 1972, "Directional errors and their correction in a discrete tracking task". Ergonomics, 15(6), 633-643.

Megaw, E. D., 1974, "Possible modification to a rapid on-going programmed manual response". Brain Research, 71, 425-441.

Meyer, D. E., Abrams, R. A., Kornblum, S., Wright, C. E., & Smith, J. E. 1988, "Opimality in human motor performance: Ideal control of rapid movements". Psychological Review, 95(3), 340-370.

Meyer, D. E., Smith, J.E.K, & Wright, C.E., 1982, "Models for the speed and accuracy of aimed movements". Psychological Review, 89(5), 449-482.

Mitchelson, D., 1975, "Recording of movement without photography". In: Grieve D., Miller D., Mitchelson D., Smith J., and Smith A. (eds), Techniques for The Analysis of Human Movement. London: Lepus Books.

Monk, A., 1984, "Fundimentals of Human Interaction". Academic Press: London.

Morasso, P., Ivaldi, F.A.M, & Ruggiero, C., 1983, "How a discontinuous mechanism can produce continuous patterns in trajectory formation and handwriting". Acta Psychologica, 54, 83-98.

Moto-Oka, , 1982, "Fith Generation Computer Systems". Amsterdam, North-Holland.

Muckler, F. A., and Seven, S. A., 1992, "Selecting performance measures; objective versus subjective measurement". Human Factors, 34(4), 441-455.

Murata, A., 1991, "An Experimental Evaluation of Mouse, Joystick, Joycard, Lightpen, Trackerball and Touchscreen for Pointing - Basic Study on Human Interface Design". In Human Aspects in Computing: Design and Use of Interactive Systems and Work with Terminals, Bullinger, H. j. (Ed), Elsevier Science.

Murrell, K.F.H, and Entwisle, D. G., 1960, "Age Differences in Movement Pattern", Nature, 185, 939-940.

Naisbitt, J., 1982, "Megatrends: Ten New Directions Transforming Out Lives". New York:Warner Books

Naur, P., 1960, "Report on the algorithmic language ALGOL 60'", Comm. ACM, 3(5), 299-314.

Newell, A, and Simon, H. A., 1961, "GPS, a program that simulates human thought". In Biling, H. (Ed) Lernende Aotomaten, Munich: Oldenbourg KG.

Newell, K.M., 1991, "Motor Skill Acquisition". Annual Review of Psycholgy, 42, 213-237.

Newman, W. M., and Sproull, R. T., 1979, "Principles of interactive computer graphics". 2nd ed. London: Mcrgraw-Hill.

Nickerson, R. S., 1969, "Man-Computer Interaction: A Challange for Human Factors Research". IEEE Trans. on Man-Machine Systems, 10, Part II (4),164-180.

Norusis, M.J., 1986, "SPSS/PC+ for the IBM PC/XT/AT". SPSS: Chicago, SPSS INC.

Paillard, J., 1980, "The multichanneling of visual clues and the organization of visually guided responses". In: Stelmach G. and Recuin J. (eds), Tutorials in Motor Behaviour, 259-279. Amsterdam: North-Holland.

Palfreyman, J., and Swade, D., 1991, "The Dream Machine - Exploring The Computer Age". BBC Books Ltd, London: England.

Pew, R. W., 1984, "A distributed view of motor control". In: Prinz W. and Sanders A. (eds), Cognition and Motor Processes. Amsterdam: North-Holland.

Plude, D.J., Hoyer, W.J., & Lazar, J., 1982, "Age response complexity, and target consistency in visual search". Experimental Ageing Research, 8, 99-102.

Poulton, E.C., 1969, "Bias in experimental comparisons between equipments due to the order of testing". Ergonomics, 12(4), 675-687.

Poulton, E.C., 1973, "Bias in ergonomic experiments". Applied Ergonomics, 4(1), 17-18.

Poulton, E. C., 1974, "Tracking Skill and Manual Control". London: Academic Press, Inc.

Poulton, E.C., 1982, "Influential companions: Effects of one strategy on another in the within subjects designs of cognitive psychology". Pyschological Bulletin, 91(3), 673-690.

Poulton, E.C., and Freeman, P.R., 1966, "Unwanted asymmetrical transfer effects with balanced experimental designs". Psychological Bulletin, 66(1), 1-8.

Preece, J., and Keller, L. S., 1990, "Why, What and How? Issues in The Development of an HCI Training Course". In Diaper D., Gilmore, D., Cockton G., and Shackel B. (Eds), Human-Computer Interaction, Interact '90, Procs. of the IFIP TC 13 3rd International Conference on Human-Computer Interaction, Cambridge, Elsvier Science: Holland,

Price, L. A., and Cordova, C. A., 1983, "Use of Mouse Buttons". In: Janda A. (Ed), Human Factors in Computing Systems, CHI '83, Boston. ACM.

Radwin, R. G., Gregg, C. V., & Mei-Li, L., 1990, "A method for evaluating head-controlled computer input devices using Fitt's Law". Human Factors, 32(4), 423-438.

Ramsden, E. N., 1985, "A-level Chemistry". Stanley Thornes Ltd: London.

Ritchie, G. J., and Turner, J. A., 1975, "Input devices for interactive graphics". Int. J. Man-Machine Studies, 7, 639-660.

Rosenbaum, D. A., 1991, "Human Motor Control", Academic Press, Inc: New York.

Saltzman, E., and Kelso, J.A.S, 1987, "Skilled actions: a task dynamic approach". Psychological Review, 94(1), 84-106.

Schank, R. C., and Colby, K. M., 1973, "Computer Models of Thought and Language". Freeman: San Francisco.

Scheid, F., 1968, "Numerical Analysis". NY: McGraw-Hill.

Schmidt, R. A., 1975, "A schema theory of discrete motor skill learning". Psychological Review, 82, 225-260.

Schmidt, R. A., 1976, "Control processes in motor skills". In: Keogh J. and Hutton R. (eds), Exercise and Sport Sciences Reviews, (4). Santa Barbara.

Schmidt, R. A., 1982, "Generalized motor programs and schemas for movement". In Kelso J. (ed), Human Motor Behaviour, 189-218, Lawrence Erlbaum: NJ.

Schmidt, R. A., and McGowan, C., 1980, "Terminal accuracy of unexpectedly loaded rapid movements: Evidence for a mass-spring mechanism in programming". Journal of Motor Behaviour, 12, 149-161.

Schmidt, R. A., and White, J. L., 1972, "Evidence for an error detection mechanism in motor skills: a test of Adams' closed-loop theory". Journal of Motor Behaviour, 4, 143-153.

Schmidt, R. A., Zelaznik, H. N., & Frank, J. S., 1978, "Sources of inaccuracy in rapid movement". In Stelmach G. (ed) Information Processing in Motor Control and Learning. NY: Academic Press.

Schmidt, R. A., Zelaznik, H., Hawkins, B., Frank, J. S., & Quinn, J. T. 1979, "Motor output variability: a theory for the accuracy of rapid motoracts". Psychological Review, 86(5), 415-451.

Schmidtke, H., and Stier, F., 1961, "An evaluation of the validity of predetermined elemental time systems". Journal of Industrial Engineering, 12(3), 182-204.

Schultz, D. P., 1969, "The human subject in psychological research". Psychological Bulletin, 27(3), 214-228.

Sears, A., and Shneiderman, B., 1991, "High precision touchscreens: Design strategies and comparisons with a mouse". Int. J. Man-Machine Studies, 34, 593-613.

Segalowitz, S. J., and Graves, R. E., 1990, "Suitability of the IBM XT, AT, and PS/2 keyboard, mouse, and game port as response devices in reaction time paradigms". Behaviour Research Methods, Instruments, & Computers, 22(3), 283-289.

Shackel, B., 1959, "Ergonomics in the design of a large digital computer console. Design", 120, 36-39.

Shackel, B., 1969, "Man-computer interaction: the contribution of the human sciences". Ergonomics, 12, 485-499.

Shackel, B., 1984, "Designing for People in The Age of Information". Interact '84, Elsevier Science, 6-15.

Shackel, B, 1991, "Whence and Where - A Short History of Human-Computer Interaction". In Bullinger, H. J. (Ed) Advances in Human Factors/Ergonomics (18a), Procs. of the Fourth International Conference on Human-Computer Interaction, Stuttgart, Germany, 1991, Elsevier Science.

Shannon, C. E., and Weaver, W., 1949, "The mathematical theory of communication". Urbana, IL: University of Illinois Press.

Sheridan, M. R., 1979, "A reappraisal of Fitts' Law". Journal of Motor Behaviour, 11, 179-188.

Sheridan, M. R., 1984, "Planning and contolling simple movements". In: Smyth M. and Wing A.(eds), The Psychology of Human Movement, 47-81, Academic Press, Inc.

Shneiderman, B., 1982, "System message design: Guidelines and experimental results". In Badre, A. and Shneiderman, B. (Eds), Ablex Publishing Co., Norwood, NJ, 55-78.

Shneiderman, B., 1987, "Designing The User Interface: Strategies for Effective Human-Computer Interaction". Addison-Wesley: California.

Siddall, G. J., Holding, D. H., & Draper, J., 1957, "Error of aim and extent in manual point to point movement". Occupational Psychology, 31(3), 185-195.

Singleton, W.T., 1954, "The change of movement timing with age". British Journal of Psychology, 14, 166-172.

Smith, D.B.D, 1990, "Human factors and aging: an overview of research needs and application opportunities". Human Factors, 32(5), 509-526.

Smith, H. T., 1980, "Human computer interaction". In H. T. Smith, and T. R. G. Green(Eds.), Human Interaction with Computers, London: Academic Press.

Smyth, M. M., and Wing, A. M., 1984, "Movement, action and skill". In Smyth M. and Wing A. (Eds), The Psychology of Human Movement, 1-12, Academic Press, Inc.

Snoddy, G.S., 1926, "Learning and stability". Journal of Applied Psychology, 10, 1-36.

Spiegel, M. R., 1974, "Vector Analysis". NY: McGraw-Hill.

Spirduso, W.W., and MacRae, P.G., 1990, "Ageing and information processing rate". In Birren J.E. & Schaie K.W. (eds), Handbook of the Psychology of Ageing, 201-219. Academic Press, San Diego.

Staufer, M., 1992, "Technological change and the older employee: Implications for introduction and training". Behaviour and Information Technology, 11(1), 46-52.

Stein, R. B., 1982, "What muscle variable(s) does the nervous system control in limb movements?". Behavoural and Brain Sciences, 5, 535-541.

Stelmach, G. E., 1982, "Motor control and motor learning: the closed loop perspective". In:Kelso J. (ed), Human Motor Behaviour, 93-116, Lawrence Erlbaum: NJ.

Stetson, R. H., and Bourman, H. D., 1935, "The coordination of simple skilled movements". Archives Neeerlanderde Physiologie, 20, 177-254.

Stewart, T., 1989, "Editoral". Behaviour & Information Technolgy, 8(5), 321-322.

Stotz, , 1963, "Man-machine console facilities for computer aided design". In: The Procs. of the Spring Joint Computer Conference, AFIPS, 23, 323-328. Spartan: Washington.

Straayer, D. H., 1991, "Location of Isometric Joystick for Optimal Performance in TypicalComputer Task Mix". In Ed. Bullinger, H., J., Human Aspects in Computing: Design and Use of Interactive Systems and Work with Terminals, Elsevier Science.

Summers, J. J., 1989, "Motor programs". In Holding D. (ed), Human Skills, 49-69, John Wiley and Sons: Chichester.

Sutherland, , 1963, "Sketchpad: A man-machine graphical communication system". In: The Procs. of the Spring Joint Computer Conference, AFIPS, 23, 329-346. Spartan: Washington.

Swan, T., 1989, "Mastering Turbo Assembler". Hayden Books: Indiana.

Syck, G., 1991, "Turbo Assembler Bible". SAMS: Indiana.

Szafran, J., 1951, "Changes with age and with exclusion of vision in performance at anaiming task". Quarterly Journal of Experimental Psychology, 3, 111-118.

T.U.C., , 1985, "T.U.C. Guidelines on VDUs". Trades Union Congress: London.

Taylor, F. V., and Birmingham, H. P., 1948, "Studies of tracking behaviour. II. The acceleration pattern of quick manual responses", Journal of Experimental Psychology, 38, 783-795.

Taylor, R. G., and Hinson, N., 1988, "Individual differences in the use of a pointing device". In E. D. Megaw (Ed), Contempory Ergonomics, Procs. of the Ergonomics Society's 1988 Annual Conference, Taylor & Francis, 521-525.

Taylor, R. W., 1967, "Man-Computer Input-Output Techniques". IEEE Trans. on Human Factors in Electronics, 8(1), March, 1-4.

Toffler, A., 1980, "The Third Wave". New York: Morrow.

Tolin, P., and Simon, J.R., 1968, "Effects of task complexity and stimulus duration on perceptual motor performance of two desparate age groups". Ergonomics, 11, 283-290.

Trankle, U., 1989, "Cursor-positioning mit der Maus-Uberlegungen und Ergebnisse zur benutzerfreundlichen Gestaltung", Zeitschrift fur arbeitswissenschaft, 43, 141-146.

Trankle, U., and Deutschmann, D, 1991, "Factors influecing speed and precision of cursor positioning using a mouse". Ergonomics, 34(2), 161-174.

Tuller, B., Fitch, H. L., & Turvey, M. T., 1982, "The Bernstein perspective: II the concept of muscle linkage or coordinative structure". In: Kelso J. (ed), Human Motor Behaviour, 253-270, Lawrence Erlbaum: NJ.

Turvey, M. T., 1977, "Preliminaries to a theory of action with a reference to vision". In: Shaw R. and Bransford J.

(eds), Perceiving, acting, and knowing, 211-265, Hillsdale: Lawrence Erlbaum.

Turvey, M. T., Fitch, H. L., & Tuller, B., 1982, "The Bernstein Perspective: 1 the problems of degrees of freedom and context-conditioned variability". In: Kelso J. (ed), Human Motor Behaviour, 239-252, Lawrence Erlbaum: NJ.

Valentine, E. R., 1982, "Conceptual Issues in Psychology". George Allen & Unwin: London.

Vince, M. A., 1948, "Corrective movements in a pursuit task", Quarterly Journal of Experimental Psychology, 1, 85-103.

Wacholder, K., and Altenburger, H., 1926, "Beitrage zur Physiologie der willkurlichen Bewegung". 10. Enzelbewegungen, pflugers archiv die Gesamte Physiologie des Menschen und der Tiere, 214, 642-661.

Wadman, W. J., Denier, G., Genze, R. H., & Mol, C. R., 1979, "Muscle activation patterns for fast goal directed arm movements". Journal of Human Movement Studies, 1980, 6, 19-37.

Wallace, S. A., and Newell, K. M., 1983, "Visual control of discrete aiming movements". Quarterly Journal of Experimental Pyschology, 35a, 311-321.

Wallace, S. A., and Wright, L., 1982, "Distance and movement time effects on the timing of agonist and antagonist muscles: a test of the impulse timing theory". Journal of Motor Behavior, 14(4), 341-352.

Walsh, D.A., and Prasse, M.J., 1980, "Iconic memory and attentional processes in the aged". In: Poon L. (ed), New Directions in Memory and Ageing, 251-260, Hillside: NJ.

Welford, A.T., 1958, "Ageing and Human Skill", London: Oxford University Press.

Welford, A.T., 1965, "Performance, biological mechanisms and age: A theoretical sketch". In: Welford A.T. & Birren J.E. (eds), Behaviour, Ageing and the Nervous System, 1-23, Springfield, IL: Thomas.

Welford, A. T., 1968, "Fundamentals of skill". London: Methuen.

Welford, A.T., 1977, "Motor performance". In Birren J.E. & Schaie W.K. (eds), Handbook ofthe Psychology of Ageing, 450-496, New York: Van Nostrand Reinhold.

Welford, A.T., 1985, "Changes in performance with age: an overview". In: Charness N. (ed), Ageing and Human Performance, 333-369, Wiley: England.

Whelan, P. M., and Hodgson, M. J., 1978, "Essential Principles of Physics". John Murray: London.

Whitefield, A., 1986, "Human factors aspects of pointing as an input technique in interactive computer systems". Applied Ergonomics, 17(2), 97-104.

Whitfield, D., 1964, "Validating the application of ergonomics to equipment to equipment design: a case study". Ergonomics, 7, 165-174.

Whitfield, D., Ball, R. G., & Bird, J. M., 1983, "Some comparisons of on-display and off-display touch input devicesfor interaction with computer generated displays". Ergonomics, 26(11), 1033-1053.

Whiting, W. C., and Zernicke, R. F., 1982, "Correlation of movement patterns via pattern recognition". Journal of Motor Behaviour, 14(2), 135-142.

Winstein, C. J., and Schmidt, R. A., 1989, "Sensorimotor feedback". In: Holding D. (ed), Human Skills, 17-46, John Wiley and Sons: Chichester.

Winston, P. H., 1992, "Artificial Intelligence", 3rd Ed. Addison Wesley: CA.

Winston, P. H., 1979, "Artificial Intelligence". Addison-Wesley: CA.

Witkin, H. A., 1950, "Individual differences in the ease of perception of embedded figures", Journal of Personality, 19, 1-15.

Wolf, C. G., 1992, "A comparative study of gestural, keyboard, and mouse interfaces". Behaviour & Information Technology, 11(1), 13-23.

Woods, W. A., 1970, "Transition network grammars for natural langauge analysis". Communications of the Association for Computing Machinery, 13, 591.

Woodworth, R. S., 1899, "The accuracy of voluntary movements". Psychological Review, 3(2),whole of number 13.

Wright, C. E., and Meyer, D. E., 1983, "Conditions for a linear speed accuracy trade-off in aimed movements". Quarterly Journal of Experimental Psychology, 35A, 279-296.

Young, R. P., and Zelaznik, H. N., 1992, "The visual control of aimed hand movements to stationary and movining targets". Acta Psychologicia, 79, 59-78.

Zaks, R., 1982, "Programming the Z80". Sybex, Inc: CA.

APPENDICES

Contents

Appendix A1 Task Order Program

```
Program target_order;
uses dos,crt;
const
    blocks = 3; noTargets = 4; dir = 3; dist = 3;
    tblocks = 9; ar = 0.775; nt  = 108;  {9 * 4 * 3}
    x = 1; y = 2; direction = 1; distance = 2;
    dx = 13; dy1 = 10; dy2 = 26; dy3 = 42; noStargets = 3;
type
    a = array[1..notargets] of integer;
points = array[1..2] of integer;
var answer : a; op   : text;opt  : text;crap : string;
    targetArray : array [1..tblocks] of a;
    sTargetArray: array [1..tblocks] of a;
    loop1,loop2,loop,count        : integer;
    condArray    : array[1 .. noTargets,1..dir,1..dist] of
boolean;
    distdir      : array[1..tblocks,1 .. noTargets,1..2] of
integer;
    s            : array[1..blocks] of integer;
    d            : array[1..dist] of integer;
    pos          : array[1 .. tblocks,1..notargets,1..2] of
integer;
    dtot         : integer;
    z            : real;
    spos         : array[1 .. tblocks,1..notargets,1..2] of
integer;
    offset       : array[1 .. noStargets] of integer;
    order        : array[1 .. Blocks] of integer;
    tchar         : array[1..noTargets,1..noStargets] of
char;
    pposstack           : array[1..nt] of points;
    dirstack            : array[1..nt] of integer;
    diststack           : array[1..nt] of integer;
    ptarget             : array[1..nt] of integer;
    start         : points;
    starget       : integer;
    sposx,sposy   : integer;
Procedure getTargetArray( var res : a);
var loop1, x : integer;
Begin
     for loop1 := 1 to notargets do res[loop1] := 0;
     x := random(random(110));
     for loop1 := 1 to noTargets do begin
       repeat
        x := random(noTargets)+1;
       until res[x] = 0;
       res[x] := loop1;
     end{for}
End;{proc}
Procedure buildTargetArray;
var loop1,loop2 : integer;
    test : boolean;
Begin
    getTargetArray(targetArray[1]);
    For loop1 := 2 to tblocks do begin
    repeat
     test := true;
     getTargetArray(TargetArray[loop1]);
```

```
      for loop2 := 1 to loop1-1 do begin
       if (TargetArray[loop1][1] = targetarray[loop2][1])
and
           (TargetArray[loop1][2] = targetarray[loop2][2])
and
           (TargetArray[loop1][3] = targetarray[loop2][3])
and
           (TargetArray[loop1][4] = targetarray[loop2][4])
       then begin
               test := false;
       end;{if}
      end;{for loop2}
      until test;
    end;{for loop1}
End;{proc}
Procedure freecond(var target,direction,distance:
integer);
 var x , y : integer;
 Begin
 x := random(random(99));
 repeat
  x := random(dir)+1;y := random(random(199));
  y := random(dist)+1;
 until condArray[target,x,y];
distance := y; direction := x;
condArray[target,x,y] := false;
end;
procedure clearcond;
var loop1,loop2,loop3 : integer;
begin
for loop1 := 1 to noTargets do begin
 for loop2 := 1 to dir do begin
  for loop3 := 1 to dist do begin
   condArray[loop1,loop2,loop3] := true;
  end;
 end;
end;
End;
Procedure conditions;
var loop1,loop2,direction,distance : integer;
begin
 clearcond;
 for loop1 := 1 to tblocks do begin
  for loop2 := 1 to noTargets do begin
   freecond(targetarray[loop1,loop2],direction,distance);
   distdir[loop1,loop2,1] := direction;
   distdir[loop1,loop2,2] := distance;
  end;
 end;
End;
BEGIN
randomize;
assign(op,'c:\text.out');assign(opt,'c:\data.exp');
rewrite(op);rewrite(opt);
{T1[1,1] := 100; T1[1,2] := 50;
T1[2,1] := 540;T1[2,2] := 50;
T1[3,1] := 100; T1[3,2] := 300;
T1[4,1] := 540; T1[4,2] := 300;}
tchar[1,1] := 'A'; tchar[1,2] := 'B'; tchar[1,3] := 'C';
```

```
tchar[2,1] := 'D'; tchar[2,2] := 'E'; tchar[2,3] := 'F';
tchar[3,1] := 'G'; tchar[3,2] := 'H'; tchar[3,3] := 'I';
tchar[4,1] := 'J'; tchar[4,2] := 'K'; tchar[4,3] := 'L';
offset[1] := dy1; offset[2] := dy2; offset[3] := dy3;
s[1] := 8;s[2] := 20; s[3] := 40;
d[1] := 20;d[2] := 100; d[3] := 280;
clrscr;
writeln('Order of blocks');
for loop1 := 1 to blocks do begin;
 writeln('Block number ',loop1);
 readln(order[loop1]);
 writeln;end;{write to task file}
count := 0;
for loop := 1 to blocks do begin
writeln(opt,s[order[loop]]);end;
For loop := 1 to blocks do begin
 BuildTargetArray;
 conditions;
 clrscr;
 writeln(op,'For Block number ',loop,' radius of main
targets ',
   s[order[loop]]);
   for loop1 := 1 to tblocks do begin
    for loop2 := 1 to notargets do begin
     count := count + 1;
     dtot := d[distdir[loop1,loop2,1]] + s[order[loop]];
     pos[loop1,loop2,1] := 0; pos[loop1,loop2,2] := 0;
     if distdir[loop1,loop2,2] = 1 then begin
      if targetArray[loop1,loop2] = 1 then
      pos[loop1,loop2,1] := dtot;
      if targetArray[loop1,loop2] = 2 then
      pos[loop1,loop2,1] := -dtot;
      if targetArray[loop1,loop2] = 3 then
      pos[loop1,loop2,1] := dtot;
      if targetArray[loop1,loop2] = 4 then
      pos[loop1,loop2,1] := -dtot;
     end;
     if distdir[loop1,loop2,2] = 2 then begin
      z := dtot;
      if targetArray[loop1,loop2] = 1 then begin
       pos[loop1,loop2,1] := round(0.707*z);
       pos[loop1,loop2,2] := round(0.707*z*ar);
      end;
      if targetArray[loop1,loop2] = 2 then begin
       pos[loop1,loop2,1] := round(-0.707*z);
       pos[loop1,loop2,2]:= round(0.707*z*ar);
      end;
      if targetArray[loop1,loop2] = 3 then begin
       pos[loop1,loop2,1] := round(0.707*z);
       pos[loop1,loop2,2] := round(-0.707*z*ar);
      end;
      if targetArray[loop1,loop2] = 4 then begin
       pos[loop1,loop2,1] := round(z*-0.707);
       pos[loop1,loop2,2] := round(-0.707*z*ar);
      end;
     end;
     if distdir[loop1,loop2,2] = 3 then begin
      if targetArray[loop1,loop2] = 1 then
      pos[loop1,loop2,2] := round(dtot*ar);
```

```
      if targetArray[loop1,loop2] = 2 then
      pos[loop1,loop2,2] := round(dtot*ar);
      if targetArray[loop1,loop2] = 3 then
      pos[loop1,loop2,2] := -round(dtot*ar);
      if targetArray[loop1,loop2] = 4 then
      pos[loop1,loop2,2] := -round(dtot*ar);
     end;
     dirstack[count] := distdir[loop1,loop2,2];
     diststack[count] := distdir[loop1,loop2,1];
     ptarget[count] := targetArray[loop1,loop2];
     pposstack[count,x] := pos[loop1,loop2,x];
     pposstack[count,y] := pos[loop1,loop2,y];
     gotoXY(10,10);writeln('Processing record ',count);
    end;{loop2}end;{loop1} end;{loop}
 writeln('Correction factors');
 {correction factors etc}
 start[x] := pposstack[1,x]; start[y] := pposstack[1,y];
 writeln(opt,start[x]);
 writeln(op,'Start position x/y ',start[x],' ',start[y]);
 writeln(opt,start[y]);
 count := 0; repeat writeln(op);
count := count + 1; writeln(op,'Trial  Number',count);
   sTarget := Random(109);
   sTarget := Random(noStargets)+1;
   sPosx := pposstack[count+1,x] - dx;
   sPosy :=
       pposstack[count+1,y] - offset[sTarget];
   writeln(op);   writeln(opt,Ptarget[count]);
   writeln(op,'Trial no ',count,' Main target no
',PTarget[count],
    ' direction ',
     dirstack[count],' Distance ',diststack[count]);
   writeln(opt,sTarget);writeln(opt,sposx);
   writeln(opt,sposy);
   writeln(op,'secondary target no ',sTarget,
   ' Position of target array (x) ',sposx,
    ' (y) ',sposy);
   writeln(opt,tchar[ptarget[count],starget]);
   writeln(opt,pposstack[count+1,x]);
   writeln(opt,pposstack[count+1,y]);
   writeln(op,'Target letter ',tchar[PTarget[count]
   ,sTarget],' centre of secondary target (x) ',
   pposstack[count+1,x],' centre (y)
',pposstack[count+1,y]);
 writeln(op); until count = nt; close(op); close(opt);
END.
```

Appendix A2 Eye-Test Program

```
Program Eye_test;
USES dos,crt,graph;
Type
      ByteArray      = Array [0 .. 0] of Byte;    {Used for
screen pointer}
      ByteArrayPtr   = ^ByteArray;
VAR
   a              : String;
   px,py,
   GraphDriver,
   GraphMode      : Integer;
   Image : ByteArrayPtr;
Procedure ShowPointer(x,y : Integer);
 {Purpose : To initially display the pointer}
 {Entry   : x,y location of pointer }
 {Exit    : PX,PY set to new location }
 Begin
      PutImage(x,y,Image^,XORPut);             {Display
image}
      PX := x; Py := y;                       {Update cursor}
 End;{Procedure ShowPointer}
 Procedure HidePointer;
 {Purpose : To hide the pointer}
 {Entry   :  }
 {Exit    :  }
 Begin
      PutImage(Px,Py,Image^,XORPut);        {Erase pointer}
 End;{Procedure HidePointer}
Procedure display;
Begin
       GraphDriver := Detect;                  {Sets up
screen}
       InitGraph(GraphDriver,GraphMode,'');
       SetColor(yellow);
       SetBkcolor(blue);
       MoveTo(108,50);                         {draw the
pointer}
       LineRel(2,-2);                           {    ......
}
       LineRel(-6,-6);                          {    . .
}
       LineRel(2,-2);                           {    .   .
}
       LineRel(-6,0);                           {         .
}
       LineRel(0,6);                            {           .
}
       LineRel(2,-2);
       LineRel(6,6);
       {Capture the cursor shape}

       GetMem(Image,ImageSize(100,40,110,50)); {Top
left,Bot right}
       GetImage(100,40,110,50,image^);          {store in
memory}

                                                {Clear the
       ClearViewPort;
screen}
```

```
               outtextXY(50,10,'Press return to start eye
test');
               readln(a);
        outtextXy(247,146,'B');
        circle(250,150,8);
        outtextXy(247,163,'E');
        circle(250,166,8);
        outtextXy(247,178,'C');
        circle(250,182,8);
        moveto(237,140);
        linerel(26,0);
        linerel(0,52);
        linerel(-26,0);
        linerel(0,-52);
        readln(a);
        ClearViewPort;                                {Clear the
screen}
        outtextXy(247,146,'D');
        circle(250,150,8);
        outtextXy(247,163,'J');
        circle(250,166,8);
        outtextXy(247,178,'F');
        circle(250,182,8);
        moveto(237,140);
        linerel(26,0);
        linerel(0,52);
        linerel(-26,0);
        linerel(0,-52);
        readln(a);
        ClearViewPort;                                {Clear the
screen}
         outtextXy(247,146,'I');
        circle(250,150,8);
        outtextXy(247,163,'A');
        circle(250,166,8);
        outtextXy(247,178,'L');
        circle(250,182,8);
        moveto(237,140);
        linerel(26,0);
        linerel(0,52);
        linerel(-26,0);
        linerel(0,-52);
        readln(a);
        ClearViewPort;                                {Clear the
screen}
         outtextXy(247,146,'H');
        circle(250,150,8);
        outtextXy(247,163,'G');
        circle(250,166,8);
        outtextXy(247,178,'K');
        circle(250,182,8);
        moveto(237,140);
        linerel(26,0);
        linerel(0,52);
        linerel(-26,0);
        linerel(0,-52);
        readln(a);                                    {Clear the
        ClearViewPort;
screen}
```

```
End; {Procedure SetupPointer}
Procedure Test(size : Integer; side,hit : char);
var x : integer;
Begin
 clearViewPort;
 outtextXY(200,250,'h for inside  -  m for outside');
 Circle (320,175,size);
 if side = 'l' then x :=  320-size else x := 320 + size;

 if ((hit = 'h') and (side = 'l')) or ((hit = 'm') AND
(side = 'r')) then x := x + 2
 else x := x-2;
 showPointer(x,175);
 readln(a);
 if a = hit then begin
   sound(1000);
   outtextXY(10,30,'Correct');
   end
   else
   begin
    sound(200);
    OuttextXY(10,30,'Wrong!');
   end;
 delay(200);
 nosound;
 outtextXY(50,50,'Hit return to continue');
 readln(a);
 hidepointer;
End;
procedure demo;
Begin
Repeat
 clearViewPort;
 outtextXY(10,10,'This is the pointer');
 showPointer(307,175);
 readln(a);
 hidepointer;
 clearViewPort;
 outtextXY(10,10,'This shows the very tip of the pointer
just inside the circle');
 Circle (320,175,15);
 showPointer(307,175);
 readln(a);
 clearViewPort;
 outtextXY(10,10,'This shows the very tip of the pointer
just outside the circle');
 Circle (320,175,15);
 showPointer(303,175);
 readln(a);
until a = 'e';
end;
Begin;
 Display;
 demo;
 test(40,'l','h');test(20,'r','m');test(20,'l','m');
 test(8,'r','m');test(8,'l','h');test(8,'l','h');
 test(8,'r','h');test(20,'l','m');test(40,'r','h');
 test(20,'l','h');test(8,'r','h');test(20,'l','m');
End.
```

Appendix A3 Main Task Program

```
Program PhD001;
 {Date 2oth September 1991 : Last amended 23rd January
1992}
 {Purpose : To record mouse details during device usage}
 Uses Dos,Crt,Graph;
 Const
      GainX = 2;
      GainY = 2;
      UGx = 1.11;
      UGy = 0.84;
      cmx = 320;
      cmy = 175;
      csx = 320;
      csy = 175;
      err     =  100;    {error on movement}
      test = true;
      PortCom     = $3F8;         {Communications port}
      Bios_Dseg   = $40;          {BIOS Data Segment}
      NewMask     = $EE;        {Mask to enable interupts}
      C8259_0     = $20;        {8259 Chip port register 1}
      C8259_1     = $21;        { "     "     "        "     2}
      EOI         = $20;        {Byte indicates End of Int}
      Addr_6845   = $63;       {Base address of video regs}
      Crt_ModSet  = $65;         {Video mode reg}
      HiFast      = $04;        {Hi byte of timer interval}
      LoFast      = $AA;      {Lo byte of timer interval}
      dx          = 13;      {offset for secondary targets}
      dy1         = 10;    {offsets for secondary y values}
      dy2         = 26;
      dy3         = 42;
      cposx       = 0;          {hotspot x}
      cposy       = 0;         {hotspot y}
      ntp         = 4;      {number of primary targets}
      nts         = 3;
               {number of targets on secondary target}
      x           = 1;      {x and y of point see type}
      y           = 2;
      nt          = 108;          {number of trials 36 x 3}
      nb          = 3;             {number of blocks}
      nbt         = 36;            {number of block trials}
      sr          = 8;             {secondary radius}
 Type
      DataRec = Record
       Period : LongInt;
       Xvalue : integer;
       Yvalue : integer;
       Mxvalue : integer;
       Myvalue : Integer;
       Bvalue : byte;
      End;{record}
      Trial = Record
       ft2    : Integer;
       fposx : Integer;
       fposy : Integer;
       fcposx: Integer;
       fcposy: Integer;
       flet   : string[10];
       ft1    : Integer;
```

```
    End;{Record}
    ByteArray      = Array [0 .. 0] of Byte;   {Used for
                                     screen pointer}
    ByteArrayPtr   = ^ByteArray;
    points = array[1..2] of integer;
Var
    serror : integer;
    smissed : boolean;
    moved,
    Lifted,
    LiftUp,
    error : Boolean;
    mcount1,
    mloop1,
    mloop2         : Integer;   {loops in main prog}
    blockrads      : array[1..nb] of integer; {t1 diam's}
    outfile        :  file of Datarec;
    Trials         : Array [1..nt] of trial;
    pt             : PaletteType;
    temp1          : Integer;
    PointerOn,                  {Display pointer true}
    ButtonPushed,               {Flag for button pushed}
    Rec,                                 {Save data}
    Button,screenPointer  : Boolean;    {Button pushed}
    offsets    : Array[1..nts] of integer; {secondary}
    Regs                      : Registers;  {access to
                  {Registers for ints}
    OldMask,               {Store old mask details}
    Temp,                  {Byte read from RS232}
    Enable,Disable,           {Screen refresh}
    Dummy,                    {Dummy variable}
    X1,X2,Y1,Y2,Butt Byte;{Bytes from RS232 given
                               meaning}
    a                         :string[10];
    az : string[1];
    rx,ry : real;
    px,py : Integer;
    dataarray           : array[0 .. 4000] of datarec;
    daptr                     : integer;
    counter1                  : integer;
    HitTrue,HitPrimary                  : Boolean;
    Xcounter,                {displacement counters}
    Ycounter                  : byte;
    Xpos,Ypos,                  {Cursor positions}
    Count,CCount,                          {Counters}
    GraphDriver,GraphMode,   {used for video routines}
    Display_Port,
    Newx,Newy,
    displacementx,
    displacementy,
    VSYNC_Port,
    EGA_FCR,
    Seg1C,Ofs1C,        {Timer int. Vect}
    MaxX,MaxY,               {screen maximims}
    MinX,MinY,
    HM,SCS,Year,MD,            {Date and time}
    BlockRadius,                 {Target Radius}
    OldX,OldY,
    toldx,toldy,
```

```
      mposx,mposY     : Integer;     {Pointer positions}
      startx,starty                         : integer;
      mx,my                          : integer;
      OldVect                        : pointer; {vector address}
      RecTime,
      DataTime,
      t                          : LongInt;{millseconds timer}
      Image                          : ByteArrayPtr;
      ts                   : Array[1 .. ntp] of bytearrayPtr;
      tp                         : Array[1 .. ntp] of points;
      Port_6845                      : Integer absolute
Bios_Dseg:Crt_Modset;
      Current_Mode                   : Byte absolute
Bios_Dseg:Crt_Modset;
      Adaptor_type                   : 1 .. 3;
      dataitem                       : datarec;
      endOftrial                     : datarec;
      subno : integer;
      torder : string;
      inf    : text;
      {PROCEDURES AND FUNCTIONS}
 {read in data for trials}
 Procedure ReadInData;
 var loop1 : integer; ip  : Text;
 Begin
  assign(ip,'c:\data.exp');
  reset(ip);
  assign (inf, 'c:\det.dat');
  rewrite(inf);
  writeln('Input sub no.');
  readln(subno);
  writeln('Input running order eg 1 2 3 and press
return');
  readln(torder);
  For loop1 := 1 to nb do begin
    readln(ip,blockrads[loop1]);
  end;{loop1}
  readln(ip,startx);
  readln(ip,starty);
  For loop1 := 1 to nt do begin
    with trials[loop1] do begin
      readln(ip,ft1);
      readln(ip,ft2);
      readln(ip,fposx);
      readln(ip,fposy);
      readln(ip,flet);
      readln(ip,fcposx);
      readln(ip,fcposy);
    end;
   end;{loop}
  close(ip)
 End;{proc}
  {test for target hit}
  function Hit (tx,ty,tr : Integer   ) : boolean ;
  var bigr,bigx,bigy : longInt;
  Begin
   bigx := tx-cposx-oldx; bigy := ty-cposy-oldy;bigr :=
tr;
```

```
   if bigx*bigx + bigy*bigy >= bigr*bigr then hit :=
false;
   if bigx*bigx + bigy*bigy < bigr*bigr then  hit := true;
  end;
  {set up target positions and stacks ect}
  Procedure setupData;
 {Purpose : To determine primary target locations etc}
 {Entry   :  }
 {Exit    :  }
 Begin
  tp[1,x] := 100; tp[1,y] := 50;
  tp[2,x] := 540; tp[2,y] := 50;
  tp[3,x] := 100; tp[3,y] := 300;
  tp[4,x] := 540; tp[4,y] := 300;
  offsets[1] := dy1; offsets[2] := dy2;  offsets[3] :=
dy3;
 End;{setupData}
 Procedure DrawPrimary;
 {Draws primary targets}
 Var
  loop1 : integer;
 Begin
    For loop1 := 1 to ntp do begin;
     outtextXY(tp[loop1,x]-3,tp[loop1,y]-4,chr(48+loop1));
     circle(tp[loop1,x],tp[loop1,y],blockRadius);
    end;{loop1}
 End;{proc}
 Procedure DrawSecondary (x,y,num : Integer);
 {Displays secondary targets}
 Begin
        PutImage(x,y,ts[num]^,normalPut);
{Display image}
 End;{proc}
 {Pointer routines - mouse arrow}
 Procedure SetupPointer;
 Procedure ClearSecond(x,y,num :integer);
 {Displays secondary targets}
 Begin
         PutImage(x,y,ts[num]^,xorPut);
{Display image}
 End;{proc}
 {Purpose : To define the shape of the pointer}
 {Entry   :  }
 {Exit    :  }
 Begin
       GraphDriver := Detect;                 {Sets up
screen}
       InitGraph(GraphDriver,GraphMode,'');
       SetColor(yellow);
       SetBkcolor(blue);
       setaspectratio(76,96);
       MoveTo(108,50);            {draw the pointer}
       LineRel(2,-2);                  {    ......     }
       LineRel(-6,-6);                 {    . .        }
       LineRel(2,-2);                  {    .   .      }
       LineRel(-6,0);                  {          .    }
       LineRel(0,6);                   {            .  }
       LineRel(2,-2);
       LineRel(6,6);
```

```
        {Capture the cursor shape}
        GetMem(Image,ImageSize(100,40,110,50)); {Top
left,Bot right}
        GetImage(100,40,110,50,image^);         {store in
memory}
        ClearViewPort;                          {Clear the
screen}
        outtextXy(247,146,'A');
        circle(250,150,8);
        outtextXy(247,163,'B');
        circle(250,166,8);
        outtextXy(247,178,'C');
        circle(250,182,8);
        moveto(237,140);
        linerel(26,0);
        linerel(0,52);
        linerel(-26,0);
        linerel(0,-52);
  GetMem(ts[1],ImageSize(237,140,263,192)); {Top left,Bot
right}
        GetImage(237,140,263,192,ts[1]^);         {store in
memory}
        ClearViewPort;                          {Clear the
screen}
        outtextXy(247,146,'D');
        circle(250,150,8);
        outtextXy(247,163,'E');
        circle(250,166,8);
        outtextXy(247,178,'F');
        circle(250,182,8);
        moveto(237,140);
        linerel(26,0);
        linerel(0,52);
        linerel(-26,0);
        linerel(0,-52);
  GetMem(ts[2],ImageSize(237,140,263,192)); {Top left,Bot
right}
        GetImage(237,140,263,192,ts[2]^);         {store in
memory}
        ClearViewPort;                          {Clear the
screen}
         outtextXy(247,146,'G');
        circle(250,150,8);
        outtextXy(247,163,'H');
        circle(250,166,8);
        outtextXy(247,178,'I');
        circle(250,182,8);
        moveto(237,140);
        linerel(26,0);
        linerel(0,52);
        linerel(-26,0);
        linerel(0,-52);
  GetMem(ts[3],ImageSize(237,140,263,192)); {Top left,Bot
right}
        GetImage(237,140,263,192,ts[3]^);         {store in
memory}
        ClearViewPort;                          {Clear the
screen}
         outtextXy(247,146,'J');
```

```
       circle(250,150,8);
       outtextXy(247,163,'K');
       circle(250,166,8);
       outtextXy(247,178,'L');
       circle(250,182,8);
       moveto(237,140);
       linerel(26,0);
       linerel(0,52);
       linerel(-26,0);
       linerel(0,-52);
 GetMem(ts[4],ImageSize(237,140,263,192)); {Top left,Bot
right}
       GetImage(237,140,263,192,ts[4]^);          {store in
memory}
       ClearViewPort;                            {Clear the
screen}
 End; {Procedure SetupPointer}
 Procedure DisplayPointer(x,y : byte);
 {Purpose : To display the pointer}
 {Entry   : x,y location of pointer }
 {Exit    : OldX,OldY set to new location }
 Begin
      {Calculate displacement - 2's compliment 8 bit}
      If x > 127 then displacementx := (((NOT X) + 1)and
255)*-1
      else displacementx := x;
      if y > 127 then displacementy := (((NOT Y) + 1)and
255)*-1
      else displacementy := y;
      mposx := mposx + displacementX;
      MposY := MposY - displacementY;
      rx := Oldx + displacementx*GainX*UGx;
      ry := Oldy - displacementy*GainY*UGy;             {add
displacement}
      newx := round(rx);
      newy := round(ry);
      if ((NewX >= minX) AND (newX <= maxX ) AND
         (NewY >= minY) AND (newy <= maxY))
         then Begin
      {Display pointer}
      if screenPointer then begin
       PutImage(PX,PY,Image^,XORPut);       {Erase old
pointer}
       PutImage(newx,newy,Image^,XORPut);   {Display new
image}
       Px := newx; Py := newy;
      end;
      end;{if }
      OldX := newx; OldY := newy;     {Update cursor}
 End;{Procedure DisplayPointer}

Procedure ShowPointer(x,y : Integer);
 {Purpose : To initially display the pointer}
 {Entry   : x,y location of pointer }
 {Exit    : PX,PY set to new location }
 Begin
      PutImage(x,y,Image^,XORPut);                {Display
image}                                        {Update cursor}
      PX := x; Py := y;
```

```
 End;{Procedure ShowPointer}
 Procedure HidePointer;
 {Purpose : To hide the pointer}
 {Entry   :  }
 {Exit    :  }
 Begin
      PutImage(Px,Py,Image^,XORPut);       {Erase pointer}
 End;{Procedure HidePointer}
Procedure RecData;
 {Purpose : saves data to array}
 {Entry   :  }
 {Exit    :  }
 Begin
  if daptr <= 4000 then begin
  with dataarray[daptr] do         {Save time,x,y,button
status}
  Begin
   Period := DataTime;
   Xvalue := OldX;
   Yvalue := OldY;
   MxValue := mposx;
   MyValue := mposy;
   Bvalue := Butt;
  End;{with}
  daptr := daptr + 1
  end;{if}
 End;{Proc}
Procedure saveData;
 {Purpose : saves data to disk}
 {Entry   :  }
 {Exit    :  }
 Begin
 counter1 := 1;
 with dataArray[0] do begin
   period := -1;
   Xvalue := BlockRadius;
   Yvalue := mcount1;
   if NOT(error)  then Bvalue := 0 ELSE Bvalue := 1;
   if lifted then Bvalue := Bvalue + 2;
   if daptr = 4000 then Bvalue := Bvalue + 4; {ie too
long}
   if moved then Bvalue := Bvalue + 8;
 End;{with}
 write; (outfile,dataArray[0]);{write trial header to
disk}
repeat
  write(outfile,dataarray[counter1]);
  counter1 := counter1 + 1;
 until counter1 = daptr;
  daptr := 1;
 End;{Proc}
 {RS232 routines}
 Procedure RS232Int;
 {Purpose : Provides user RS232 interrupt routine}
 {Entry   :  }
 {Exit    : cursor position and button status }
 Interrupt;                              {Byte arrived at
RS232}
 Begin
```

```
        port[C8259_0] := EOI;                {enable chip}
        InLine($FB);                          {SLI}
        RecTime := t;                    {time of reciept}
        temp := Port[PortCom]; {Read Byte strip parity}
        if  test then begin
         if (temp and $F8) = $80 {Bits 10000} then
          begin
          count := 1;         {First byte}
          DataTime := RecTime;        {Store data block time}
         end;{if}
        Case Count of
         1: Begin
             Butt := (Temp AND 7); {Read the button status}
             ccount := 2;
            End;{1}
         2: Begin
             Xcounter := Temp;          {Update X pos}
             ccount := 3;
            End;{2}
         3: Begin
             Ycounter := temp;  {Upper nibble}
             ccount := 4;
             {if pointerOn then
DisplayPointer(Xcounter,Ycounter);}
             {if rec then recdata;}
            End;{3}
         4: Begin
             {Xcounter := Temp+Xcounter;{Update X pos}
             ccount := 5;
            End;{4}
         5: Begin
             {Ycounter := Temp + Ycounter;   {upper nibble}
             ccount := 1;
             If ButtOn then
             begin
              if butt AND 3 <> 3 then ButtonPushed := True;
              if butt > 3 then liftup := true;
              {screenPointer := false;}
             end;{if}
             If PointerON then
DisplayPointer(Xcounter,Ycounter);
             if Rec  then
             RecData;
            End;{5}
        6: Begin
           write('6',temp);
           end;{6}
        End;{case}
        End;{if}
        if not test then begin
         if (temp and 192) = 192 {Bits 11000..} then
          begin
           count := 1;        {First byte}
           DataTime := RecTime;   {Store data block time}
          end;{if}
         Case Count of
          1: Begin
              Butt := (Temp AND 48); {read the button status}
              Ycounter := (Temp and 12)*16;
```

```
          Xcounter := (Temp and 3)*64;
          ccount := 2;
         End;{1}
      2: Begin
          Xcounter := Temp+Xcounter;{Update X pos}
          ccount := 3;
         End;{2}
      3: Begin
          Ycounter := temp+Ycounter;  {Upper nibble}
          ccount := 1;
          If PointerON then
DisplayPointer(Xcounter,Ycounter);
          If ButtOn then
          begin
           if butt AND 32 = 32 then ButtonPushed := True;
           if butt AND 16 = 16 then liftup := true;
          end;{if}
          if Rec  then
          RecData;
         End;{3}
     6: Begin
         writeln('Error in byte order at RS232');
         repeat until keypressed;
        end;{6}
     End;{case}
     End;{if test}
     count := ccount;            {prepare for next byte}
 End;{Procedure RS232Int}
 Procedure InitRS232;
 {Purpose : To setup the RS232 port for 1200 baud pc
mouse}
 {Entry   :  }
 {Exit    :  }
 Begin
     {Configure the port}
     With Regs Do
     Begin
      AH := 0; DX := 0;            {Set up com1:}
      if test then AL := $83 else AL := $E7; {Parameters:
9600,Even, etc}
      Intr($14,Regs);              {Do it}
     End;{With}
     {Change RS232 interrupt vector}
     GetIntVec($0C, OldVect);       {Save old vector}
     SetIntVec($0C, @RS232Int);     {New vector}
     {Set Chip 8259 IRQ line}
     OldMask := Port[ C8259_1 ];  {Save old mask}
     Port[ C8259_1 ] :=  NewMask AND OldMask;
    {Enable 8250 int on data-read}
     Temp := Port[PortCom+3];
     Port[PortCom+3] := (Temp AND $7F);{DLab set to 0}
     Port[PortCom+1] := 1;         {Int Enable register}
     {Clear 8250 status Register}
     Repeat
      Temp := Port[PortCom];          {Clear by reading}
      Temp := Port[PortCom + 5];            {Line Status}
      Temp := Port[PortCom + 6];            {Modem Status}
      Temp := Port[PortCom + 2];            {Intrupt ident}
      Temp := (Temp AND 1);  {bit 1 of ident should be 1}
```

```
      Until Temp = 1;
      {Set bit 3 of modem control reg}
      Temp := Port[PortCom + 4];
      Port[PortCom + 4] := (Temp OR $08);   {Done}
 End;{Procedure InitRS232}
 Procedure RestoreRS232;
 {Purpose : To Restore the RS232 port & interrupt routine}
 {Entry   :  }
 {Exit    :  }
 Begin
      Temp := Port[C8259_1];
      Port[C8259_1] := (Temp or $10);     {Disable 8259
IRQ}
      Temp := Port[PortCom + 3];
      Port[PortCom+3] := (Temp AND $7F);  {DLab = 0}
      Port[PortCom + 1] := 0;            {Int disabled}
      Temp := Port[PortCom + 4];
      Port[PortCom + 4] := (Temp AND $F7);{Clear bit 3
modem control reg}
      SetIntVec($0C,OldVect);            {Restore old
interrupt vector}
 End;{Procedure RestoreRS232}
 {Timming routines}
 Procedure TimerInt;
 {Purpose : Provides user timer interrupt routine}
 {Entry   :  }
 {Exit    : t becomes a mS counter }
 Interrupt;
 Begin
  Inline($FA);                         {Cli clear ints}
  t := t + 1;                          {increment counter}
  Inline($FB);                         {Sli}
 End;{Procedure}
 Procedure ChangeTimingFrequencey(HiByte,LoByte: Byte);
 {Purpose : Alters interrupt frequencey}
 {Entry   : number to be decremented }
 {Exit    :  }
 Begin
  Port[$43] := $36;                   {select mode 3}
  Port[$40] := LoByte;                {new divisor lo byte}
  Port[$40] := HiByte;                {new divisor hi Byte}
 End;{Procedure}
 Procedure Get_Time_And_Date(Var
year,MonDay,HrMin,SecCentis: Integer);
 {Purpose : Save current time and date}
 {Entry   :  }
 {Exit    : date and time stored }
 Begin
  with Regs do
  Begin
   AX := $2A00;                         {Int 21}
   MsDos(Regs);                         {get time and date}
   Year := CX;
   MonDay := DX;
   MsDos(Regs);
   HrMin := CX;
   SecCentis := DX;
  End;{with}
 End;{Procedure}
```

```
Procedure TimerOn;
{Purpose : Stores old vector and sets new}
{Entry   :  }
{Exit    :  }
Begin
 with Regs do
 begin
  AX := $351C;                      {Int 21}
  MsDos(Regs);                      {Get vector - could
have used GetVec}
  Seg1C := ES;
  Ofs1C := BX;
  DS := Cseg;
  DX := Ofs(TimerInt);
  AX := $251C;
  MsDos(Regs);
 End;{with}
Get_Time_And_Date(Year,MD,HM,SCS);
ChangeTimingFrequencey(HiFast,LoFast);
End;{Procedure}
Procedure ResetTime;
{Purpose : Restores old time and date}
{Entry   :  }
{Exit    :  }
Begin
 {Reset date}
 with Regs do
 Begin
  AX := $2B00;                     {function 2B}
  CX := Year;
  DX := MD;
  MsDos(Regs);                     {Call interrupt}
 End;{with}
 {Reset Time}
 with Regs do
 Begin
  AX := $2D00;                     {function 2D}
  CX := HM;
  DX := SCS;
  MsDos(Regs)                      {Call Interrupt}
 End;{with}
End;{Procedure}
Procedure TimerOff;
{Purpose : Restores old timing interval}
{Entry   :  }
{Exit    :  }
Begin
 with Regs do
 Begin
  AX := $251C;
  DS := Seg1C;
  DX := Ofs1C;
  MsDos(Regs);                     {sets DS:DX as
interrupt vector}
 End;{with}
 ChangeTimingFrequencey(0,0);
 ResetTime;
End;{Procedure}
{Display routines}
```

```
 Procedure VideoSyncOn;
 {Purpose : ensures that display item is displlayed at a
specific time}
 {Entry   :  }
 {Exit    :  }
 Begin
  Adaptor_Type := 3;
  Display_Port := Port_6845 + 4;
  VSYNC_Port := Port_6845 + 6;
  Enable := Current_Mode AND 247;
  if Adaptor_Type = 3 then with Regs do
  Begin
   AH := $F;
   intr($10,Regs);
   if AL in [7,$F] then EGA_FCR := $3BA
   Else EGA_FCR := $3DA;
   Display_Port := $3C0;
   Enable := $20;
   Disable := $F;
  End;{with}
 End;{Proc}
 Procedure DisplayTarget( t : integer);
 {Purpose : Displays the target on the screen}
 {Entry   : coordinates for target to be displayed }
 {Exit    :  }
Begin
 if Adaptor_Type = 3 then          {EGA}
 begin
  dummy := Port[EGA_FCR];          {Enable indexing}
  Port[Display_Port] := $12;       {select 12}
 end;{if}
 Port[display_port] := Disable;              {Disable
refresh}
  OutTextXY(320,10,trials[t].flet);   {Display stimulus}
  OutTextXY(308,10,chr(48+trials[t].ft1));
 Port[Display_port] := Enable;               {draw to
screen}
End;{Proc}
Procedure rerun;
var x,y : integer;
Begin
   if mcount1 = 1 then begin
    x := startx;
    y := starty;
    mx := 0;
    my := 0;
   end
   ELSE begin
    x := trials[mcount1-1].fcposx;
    y := trials[mcount1-1].fcposy;
    mx := tp[trials[mcount1].ft1,1];
    my := tp[trials[mcount1].ft1,2];
   End;
   ScreenPointer := false; HidePointer;
   clearViewPort;
   button := true;
   OuttextXY(10,10,'You missed the target first time!
Click the button to continue and then ');
```

```
   outtextxy(60,50,'Move the pointer to the centre of the
circle and press button');
   buttonPushed := false;
   repeat until buttonPushed;
   clearViewport;
   {now to locate for start of next trial}
   HitPrimary := false; error := false;
   Circle(mx+x,my+y,4);
   showpointer(oldx,oldy);screenPointer := true;
   hitTrue := false;
   While ((NOT hittrue) and (not liftup))   do begin
    ButtonPushed := false;
    repeat until buttonPushed;
    buttonPushed := false;
    if hit(mx+x,my+y,4)
    then begin
     hitTrue := true;toldx := oldx; toldy := oldy;
     sound(1000);
     end
    else  begin
     hittrue := false;
     sound(100);
    end;
    delay(200);
    nosound;
   end;{while}
   {test for lift}
   while liftup do begin
    ScreenPointer := false; HidePointer;
    liftup := false;
    clearViewPort;
    button := true;
    OuttextXY(10,10,'You lifted the mouse! Move mouse to
pad centre and ask start');
    OuttextXY(60,50,'then locate at the centre of the
circle');
    buttonPushed := false;
    readln(az);liftup := false;
    oldx := csx; oldy := csy; mposx := cmx;mposy := cmy;
    clearViewport;
    {now to locate for start of next trial}
    HitPrimary := false; error := false;
    Circle(mx+x,my+y,4);
    showpointer(oldx,oldy);screenPointer := true;
    hitTrue := false;
    While ((NOT hittrue) and (not liftup))   do begin
     ButtonPushed := false;
     repeat until buttonPushed ;
     buttonPushed := false;
     if hit(mx+x,my+y,4)
     then begin
      hitTrue := true;toldx := oldx; toldy := oldy;
      sound(1000);
     end
     else  begin
      hittrue := false;
      sound(100);
     end;
     delay(200);
```

```
      nosound;
     end;{while}
    end;{while}
   mcount1 := mcount1 -1;
  end;
Procedure ErrorTrial;
{error has occured so re run}
var et : trial; moved2 : boolean;
Begin
    if mcount1 = 1 then begin
     et.fcposx := startx;
     et.fcposy := starty;
    end
    ELSE begin
     et.fcposx := trials[mcount1-1].fcposx;
     et.fcposy := trials[mcount1-1].fcposy;
    End;
    et.ft1 := trials[mcount1].ft1;
    et.ft2 := trials[mcount1].ft2;
    et.flet := trials[mcount1].flet;
    et.fposx := et.fcposx - dx;
    et.fposy := et.fcposy - offsets[et.ft2];
    {secondary trial defined}
    Repeat
          ScreenPointer := false; HidePointer;
         {error due to lifting the mouse}
         if lifted then begin
           liftup := false;lifted := false;
           clearViewPort;
           button := true;
           OuttextXY(10,10,'You lifted the mouse! Move
mouse to pad centre and ask to start');
           buttonPushed := false;
           readln(az);
           readln(az);lifted := false;liftup := false;
           oldx := csx; oldy := csy;MposX := cmx;MposY :=
cmy;
           clearViewport;outtextXY(100,10,'target letter
');
           outtextXY(220,10,et.flet);
         end;{if}
         if moved then begin
           moved := false;
           clearViewPort;
           button := true;
           OuttextXY(10,10,'You moved before the target was
presented!');
           OutTextXY(10,30,'Push the button and then move
the mouse to target letter');
           buttonPushed := false;
           repeat until buttonPushed;
           {oldx := csx; oldy := csy;
           mposX := cmx; MposY := cmy;}
           clearViewport;outtextXY(100,10,'target letter
');
           outtextxy(220,10,et.flet);
         end;{if}

         toldx := Oldx; toldy := Oldy;
```

```
        HitPrimary := false; error := false;
       {primary target missed now present secondary}
     mx := tp[et.ft1,x];
     my := tp[et.ft1,y];
     if mcount1 = 1 then begin
      mx := 0;
      my := 0;
     end;
    if ((toldx-oldx)*(toldx-oldx)>err) or ((toldy-
oldy)*(toldy-oldy)>err) then moved := true;
    DrawSecondary(mx+et.fposx,my+et.fposy,
                                 et.ft1);
    showpointer(oldx,oldy);screenPointer := true;
      hitTrue := false;
      While ((NOT hittrue) and (not liftup))  do begin
      ButtonPushed := false;
      repeat until buttonPushed;
      buttonPushed := false;
      if hit(mx+et.fcposx,my+et.fcposy,8)
       then begin
         hitTrue := true;toldx := oldx;toldy := oldy;
         sound(1000);
       end
      else  begin
       hittrue := false;
       sound(100);
      end;
      delay(200);
      nosound;
    end;{while}
      screenpointer := false;hidePointer;
      clearviewport;
      showPointer(Oldx,Oldy);screenpointer:=true;
      {now to rerun primary}
      hitTrue := false;
      hitPrimary := false;
      {reference for primary targets}
      mx := tp[et.ft1,x];
      my := tp[et.ft1,y];
      screenpointer := false; hidepointer;
      if ((toldx-oldx)*(toldx-oldx)>err) or ((toldy-
oldy)*(toldy-oldy)>err) then moved := true;
      drawprimary;
      rec := true;timerOn;
      showPointer(oldx,oldy);
      {display task and start timer}
      datatime := 0;recdata;
      DisplayTarget(mcount1);
      t := 0;
      screenPointer := true;
      While ((NOT hittrue) and (not liftup))   do begin
      ButtonPushed := false;
      repeat until (buttonPushed);
      buttonPushed := false;
      if hit(mx,my,BlockRadius)
       then begin
         rec := false;
```

```
          hitTrue := true;hitPrimary := true;toldx := oldx;
toldy := oldy;
          sound(1000);timerOff;
        end
       else  begin
        rec := false;
        hittrue := false;error := true;
        sound(100);
        {need to setup secondary task so as to rerun}
       end;
     delay(100);
     nosound;
    end;{while}
    timerOff;
    if liftup then lifted := true;
    if ((toldx-oldx)*(toldx-oldx)>err) or ((toldy-
oldy)*(toldy-oldy)>err) then moved := true;
    SaveData;
    if moved then begin
     moved2 := true;
     end
    else
     moved2 := false;
   until (NOT error) AND (NOT LIFTUP) AND (NOT MOVED2);
  End;{Proc}
   {Begin Program}
   BEGIN
   pointerOn := false;
   screenPointer := false;
   clrscr;
   readInData;
    nosound;
    setuppointer;setupData;
    MaxX := 629;MaxY := 349;count := 1;
    MinX := 0; MinY := 0;
    ButtonPushed := false;liftup := false;lifted := false;
    Button := false;
    PointerOn := false;
    rec := false;
    clearViewPort;
    OuttextXY(60,10,'Move mouse to pad centre and press
return ');
    repeat until keypressed;
    oldx := csx; oldy := csy;mposx := cmx; mposY := cmy;
    assign(outfile,'c:\subData.exp');
    daptr := 1;
    rewrite(outfile);
    startx := tp[trials[1].ft1,x] + startx;
    starty := tp[trials[1].ft1,y] + starty;
    mcount1 := 1;
    nosound;
    clearViewPort;
    ShowPointer(csx,csy);oldx := csx;oldy := csy;mposx :=
cmx;mposy := cmy;
    InitRS232;
    TimerOn;
    VideoSyncOn;
    moved := false;
    pointerOn := true;screenPointer := true;button := true;
```

```
   OuttextXY(100,10,'Move pointer to centre of circle and
press button to start');
   buttonPushed := false; hitTrue := false;
   Circle(startx,starty,4);
   Repeat
    Repeat until buttonPushed;buttonPushed := false;
    toldx := oldx; toldy := oldy;
    if Hit(startx,starty,4) AND (NOT LiftUp) then hitTrue
:= true;
    if liftup then liftup := false;
   until hitTrue;
   ScreenPointer := false; HidePointer;
   clearviewPort;
   ShowPointer(oldx,oldy);screenPointer := true;
   serror := 0;
   {Main Program Loops}
   For mloop1 := 1 to nb do begin
     BlockRadius := blockRads[mloop1];
     for mloop2 := 1 to nbt do begin
     if mcount1 <= nt then begin
     repeat
      smissed := false;
      error := false;
      hitTrue := false;
      hitPrimary := false;
      {reference for primary targets}
      mx := tp[trials[mcount1].ft1,x];
      my := tp[trials[mcount1].ft1,y];
      screenpointer := false; hidepointer;
      if ((toldx-oldx)*(toldx-oldx)>err) or ((toldy-
oldy)*(toldy-oldy)>err) then moved := true;
      drawprimary;
      rec := true;timerOn;
      showPointer(oldx,oldy);
      {display task and start timer}
      datatime := 0;
      recdata;
      DisplayTarget(mcount1);
      t := 0;
      screenPointer := true;
      While ((NOT hittrue) and (not liftup))  do begin
      ButtonPushed := false;
      repeat until buttonPushed;
      buttonPushed := false;
      if hit(mx,my,BlockRadius) AND NOT LiftUp
       then begin
         rec := false;
         hitTrue := true;hitPrimary := true;
         sound(1000);timerOff;
       end
      else  begin
       rec := false;
       hittrue := false;
       error := true;
       if liftup then lifted := true;
       sound(100);
       {need to setup secondary task so as to rerun}
      end;
    delay(100);
```

```
     nosound;
     end{while};
     liftup := false;
    timerOff;
    saveData;
         if (error) or (Moved) then errorTrial;
         lifted := false;moved := false;
         screenPointer := false;hidePointer;
         clearViewPort;
        {primary target hit now present secondary}
    if mcount1 <> nt then begin
      mx := tp[trials[mcount1+1].ft1,x];
      my := tp[trials[mcount1+1].ft1,y];
DrawSecondary(mx+trials[mcount1].fposx,my+trials[mcount1].
fposy,
                                trials[mcount1].ft1);
       showpointer(oldx,oldy);screenPointer := true;
       hitTrue := false;
       While ((NOT hittrue) and (not liftup))  do begin
       ButtonPushed := false;
       repeat until buttonPushed;
       buttonPushed := false;
       if
hit(mx+trials[mcount1].fcposx,my+trials[mcount1].fcposy,8)
        then begin
          hitTrue := true;toldx := oldx; toldy := oldy;
          sound(1000);
        end
       else  begin
        hittrue := false;
        smissed := true;
        sound(100);
       end;
       delay(200);
       nosound;
     end;{while}
       screenpointer := false;hidePointer;
       clearviewport;
       showPointer(Oldx,Oldy);screenpointer:=true;
      end;{if}
      {what if lifted}
      while liftup do begin
          ScreenPointer := false; HidePointer;
           liftup := false;
           clearViewPort;
           button := true;
           OuttextXY(10,10,'You lifted the mouse! Move
mouse to pad centre and ask to start');
           OuttextXY(60,50,'then locate at the centre of
the circle');
           buttonPushed := false;
           readln(az);
           readln(az);liftup := false;
           oldx := csx; oldy := csy; mposx := cmx;mposy :=
cmy;
           clearViewport;
           {now to locate for start of next trial}
```

```
           HitPrimary := false; error := false;
Circle(mx+trials[mcount1].fcposx,my+trials[mcount1].fcposy
,4);
           showpointer(oldx,oldy);screenPointer := true;
           hitTrue := false;
           While ((NOT hittrue) and (not liftup))  do begin
            ButtonPushed := false;
            repeat until buttonPushed;
            buttonPushed := false;
            if
hit(mx+trials[mcount1].fcposx,my+trials[mcount1].fcposy,4)
             then begin
             hitTrue := true;toldx := oldx; toldy := oldy;
             sound(1000);
            end
            else  begin
             hittrue := false;
             sound(100);
            end;
           delay(200);
           nosound;
          end;{while}
        end;{while}
       if smissed then serror := serror + 1;
       if smissed then rerun;
        screenpointer := false;hidePointer;
       clearviewport;
       showPointer(Oldx,Oldy);screenpointer:=true;
      mcount1 := mcount1 + 1;
      lifted := false;
       until NOT(smissed);
      end;{if last}
     end;{loop2}
   end;{loop1}
    {TimerOff;}
    close (outfile);
    RestoreRS232;
    writeln(inf,'Sub ',subno,' Order ',torder,' Sec err
',serror);
    close(inf);
  END.
```

Appendix A4 Analysis Program (x positional data)

```
Program conind;
{Converts stream of data into single file trials}
Type
      DataRec = Record
       Period : LongInt;
       Xvalue : integer;
       Yvalue : integer;
       Mxvalue : integer;
       Myvalue : Integer;
       Bvalue : byte;
      End;{record}
Var
 tr,xr : real;
 lifted,
 missed,
 moved  : integer;
 infile : file of datarec;
 outfile :TEXT;
 trial : integer;
 name    : string[18];
 item : datarec;
 oldtime : real;
 oldxr : real;
BEGIN
 trial := 1;
 lifted := 0;moved := 0;missed := 0;
 oldtime := 0;
 assign(infile,'c:\subdata.exp');reset(infile);
 read(infile,item);
 if item.bvalue = 0 then begin
 str(trial,name);
 name := 'c:\temp\mx' + name + '.res';end
 else
 begin
  name := '0';
  trial := 0;
  if ((item.bvalue and 2) = 2) then lifted := lifted + 1;
  if ((item.bvalue and 8) = 8) then moved := moved + 1;
  if ((item.bvalue and 15) = 1) then begin
   missed := missed + 1;
   write('1',' ');
  end;
 end;
 assign(outfile,name);
 rewrite(outfile);
 oldxr := item.Xvalue;
 if item.period = -1 then begin
   {Write the header}
   {writeln(outfile,item.Xvalue);
   writeln(outfile,item.Yvalue);
   writeln(outfile,item.Bvalue);}
 while not eof(infile) do begin
   read(infile,item);
   while item.period > 1000000 do begin read(infile,item);
end;
   if item.period = -1 then begin
    close(outfile);
    if item.bvalue = 0 then begin;
```

```
     trial := item.yvalue;
    str(trial,name);
    name := 'c:\temp\mx' + name + '.res';end
    else
    begin
     name := '0';
     if ((item.bvalue and 2) = 2) then lifted := lifted +
1;
     if ((item.bvalue and 8) = 8) then moved := moved + 1;
     if ((item.bvalue and 15) = 1) then begin
      missed := missed + 1;
      write(trial,' ');
     end;
    end;
    assign(outfile,name);
    rewrite(outfile);
   {Write the header}
  { writeln(outfile,item.Xvalue);
   writeln(outfile,item.Yvalue);
  writeln(outfile,item.Bvalue); }
  end{if}
  ELSE
   Begin
   tr := item.period; xr := item.Xvalue;
    while tr - oldtime > 70  do begin
     oldtime := oldtime + 28;
     writeln(outfile,oldtime);
     writeln(outfile,oldxr);
    End;{while};
   if ((oldtime <> tr) AND (tr <> 0)) then begin
   writeln(outfile,tr);
   writeln(outfile,xr);
   {writeln(outfile,item.Myvalue);}
   end;
    oldtime := tr; oldxr := xr;
  end;{if/else}
 end;{while}
 end;{if}
 close(infile);
 close(outfile);
 writeln(' Lifted =  ',lifted,'  ','moved = ',moved,'
','missed = ',missed);
 readln; END.
```

Appendix A5 Curve Differentiating Program (x Velocity)

```
PROGRAM curvefit;
uses dos,crt;
CONST
     trials = 36;
     Eqns = 3;
     nps  = 3;
     a0   = 1;
     a1   = 2;
     a2   = 3;
     a3   = 4;
TYPE
    index = 1 .. Eqns;
    MatrixArray = array [index, 1 .. Eqns+1] of real;
VAR
   name1 : string[20];
   name2 : string[20];
   matrix  : matrixarray;
   numeqns : index;
   done,
   first,
   nosol,
   start,stop    : boolean;
   loop2,
   z,
   loop     : integer;
   np       : integer;
   xpoints : array[1 .. nps] of real;
   ypoints : array[1 .. nps] of real;
   apoints : array[1 .. nps] of integer;
   vp,vel,vpo : real;
   xi : longint;
   oldy : integer;
   sx1,sx2,
   sx3,sx4,
   sy1,sx0,
   sr1,sr2,
   sr3       : real;
   xt,yt : real;
   ipfile,
   opfile2,
   opfile1    : TEXT;
   trialLoop : integer;
 Procedure Swap(i,j : index);
  Var
     count : integer;
     temp  : real;
  Begin
   For count := 1 to (eqns + 1) do begin;
    temp := matrix[i,count];
    matrix[i,count] := matrix[j,count];
    matrix[j,count] := temp;
   end{for}
  End;{proc}
 Procedure subtract(computed : index);
 Var
  i,k,count : integer;
  temp : real;
 Begin
```

```
   temp := 1/matrix[computed,computed];
   For count := 1 to eqns+1 do begin
    matrix[computed,count] := temp*matrix[computed,count];
   end;{for}
   For i := 1 to eqns do
    if i <> computed then begin
     temp := (-matrix[i,computed]);
     for k := 1 to (eqns+1) do
      matrix[i,k] := matrix[i,k] + (temp *
matrix[computed,k]);
    end;{if}
  End;{proc}
  Procedure calculate;
  Var
    computed, count : index;
  Begin
   nosol := false;
   computed := 1;
   while (computed <= eqns) AND (not nosol) do begin
    nosol := true;
    count := computed;
    while (count <= eqns) AND (nosol) do
     if matrix[count,computed] <> 0 then
      nosol := false
     else
      count := count + 1;
     if not nosol then begin
      swap(computed,count);
      subtract(computed);
     end;{if}
     computed := computed + 1;
    end;{while}
   End;{proc}
   Procedure GetPoints;
   Var
    y, x : real;
    count : integer; a : string;
    Begin
     sx1 := 0; sx2 := 0; sx3 := 0; sx4 := 0; sy1 := 0;
     sr1 := 0; sr2 := 0; sr3 := 0;
     count := 1;

     while count <= nps do begin
      x:= xpoints[count]-xpoints[1];y := ypoints[count];
{      writeln(x,' ',y);readln(a);}
      sx1 := sx1 + x;
      sx2 := sx2 + (x * x);
      sx3 := sx3 + (x * x * x);
      sx4 := sx4 + (x * x * x * x);
      sy1 := sy1 + y;
      sr1 := sr1 + (1 * y);
      sr2 := sr2 + (x * y);
      sr3 := sr3 + (x * x * y);
      count := count + 1;
     end;{while}
      sx0 := nps;
    End;{proc}

    Procedure coeff;
```

```
   var i: string[1];
   Begin
    matrix[1,a0] := sx0; matrix[1,a1] := sx1; matrix[1,a2]
:= sx2;
                                    matrix[1,a3] := sr1;
    matrix[2,a0] := sx1; matrix[2,a1] := sx2; matrix[2,a2]
:= sx3;
                                    matrix[2,a3] := sr2;
    matrix[3,a0] := sx2; matrix[3,a1] := sx3; matrix[3,a2]
:= sx4;
                                    matrix[3,a3] := sr3;

   End;{proc}
   Procedure velacc( pointx:integer);
   VAR acc : real; a : string;loop: integer; loop2 :
longint;
   begin
{writeln('eqn ',matrix[a0,eqns+1],' ',matrix[a1,eqns+1],'
',
   matrix[a2,eqns+1]);}
      vel := matrix[a1,eqns+1]
           + ((xpoints[pointx]-xpoints[1]) * 2 *
matrix[a2,eqns+1]);
{         if ypoints[3] = ypoints[2] then vel := 0;}
    if pointx = 2 then begin
     if ypoints[2] = ypoints[1] then vel := 0;
    end;
    if pointx = 3 then begin
        if ypoints[2] = ypoints[3] then vel := 0;
    end;
      {vp := matrix[a1,eqns+1] + ((xpoints[2]-xpoints[1])
            * 2 * matrix[a2,eqns+1]);}
     { if start then begin vel := vel + vpo; vpo:= vpo +
vp;end;{if}
      {if stop then vel := vel + vpo-vp;}
     acc := 2 * matrix[a2,eqns+1];
 {   writeln(xpoints[pointx],' corx ',xpoints[pointx]-
xpoints[1],'vel ',vel,' xp1 '
              ,' yvalue ', ypoints[pointx]);readln(a);}
     writeln(opfile1,xpoints[pointx]);
writeln(opfile1,vel);
     {writeln(opfile2,xpoints[pointx]);
writeln(opfile2,acc)}
   End;{proc}
BEGIN
For trialLoop := 1 to 1 do begin
 str(trialloop,name1);
 name1 := 'c:\temp\mx' + name1 + '.res';
 assign(ipfile,name1); reset(ipfile);
 str(trialLoop,name2);
 name2 := 'c:\temp\velx' + name2 + '.res';
 assign(opfile1,name2); rewrite(opfile1);
{ assign(opfile2,'c:\diffax.dat');rewrite(opfile2);}
 first := true; done := false; np := nps; start := false;
 stop := false;  {start read first five points}
  for loop := 1 to nps do begin;
   readln(ipfile,xt);readln(ipfile,ypoints[loop]);
   xpoints[loop] := xt;  end;{for}  getpoints;
  coeff;  calculate;  velacc(1);vpo := vel;
```

```
  velacc(2);vpo := vel;{velacc(3);}
  start := false;  while NOT eof(ipfile) do begin
   for loop := 1 to nps-1 do begin
    xpoints[loop] := xpoints[loop+1];
    ypoints[loop] := ypoints[loop+1];
   end;{for}
   readln(ipfile,xt);readln(ipfile,ypoints[nps]);
   xpoints[nps] := xt;
  getpoints;  coeff;calculate;velacc(2); end;{while}
 start := false;
 stop := false;
 {velacc(2);}
 velacc(3); close (opfile1);{ close (opfile2);} close
(ipfile); end;END.
```

Appendix A6 Vector Addition for Velocity

```
program joinxy;
const
     trials = 36;
var ipfilex,ipfiley,opfilev : text; flag : boolean;
    time, timeo,valuex, valuey,tot      : real;
    name1,name2,name3  : string[20];
    tloop : Integer;
 begin
  flag := false;
  For tLoop := 1 to 1 do begin
  str(tloop,name1);
  str(tloop,name2);
  str(tloop,name3);
  name1 := 'c:\temp\velx' + name1 + '.res';
  name2 := 'c:\temp\vely' + name2 + '.res';
  name3 := 'c:\temp\vel' + name3 + '.dat';
  assign(ipfilex,name1);reset(ipfilex);
  assign(ipfiley,name2);reset(ipfiley);
  assign(opfilev,name3);rewrite(opfilev);
  timeo := -32;
  while (NOT eof(ipfilex)) AND (NOT eof(ipfiley)) do begin
        readln(ipfilex,time); readln(ipfiley,time);
        readln(ipfilex,valuex);readln(ipfiley,valuey);
        if time-timeo<20 then begin
        flag := true;
        writeln('time error ',tloop,' t =
',timeo);readln;end;
        timeo := time;
        valuey := valuey*0.76;
        if (eof(ipfilex) and (tot = 0)) then tot := 0
        else
        tot := SQRT((valuex*valuex) + (valuey*valuey));
        writeln(opfilev,time);
        writeln(opfilev,tot);
  end;{while}
  close(opfilev);  close(ipfilex);
  close(ipfiley);
  end;
  if flag then halt;
  end.
```

Appendix A7 Movement Parser Program

```
Program Parser01;
{parses vel and acc}
uses dos,crt,graph;
TYPE
    tps = RECORD
          time : real;
          typetp : string[10]
         end;
    m = record
        typem : string[10];
        start : real;finish: real;dur   : real;
        ptdur : real;dist  : real;epv   : real;
        distr : real;pdistr : real; curvea: real;
        velm  : real;velpk : real;accpk : real;
        extent: real;aim   : real; effect: real
        end;
Const
     j = true;Kjerk = 3;trace = false;lineon = trace;
     step = false;range = 27;zerov = 0.00001;
     zero = 0.00000001;tpmax = 300;practice = false;
     ty1 = 50;ty2 = 300;
     parnm = false; {non movement - movement}
     parsc = true;{simple complex}
     parall = false; {all phases}
     pardec = false; {previous criteria}
     parsm = false;  {only acc - non movement}
Var
   otime : real; maxp,minp,mint,maxt : real;
   lastvel : real;jerk,jump    : real; tplast : string;
   graphdriver, graphmode : integer; lasttp : string;
   lastmov: string;ltp : string; first : boolean;
   xpos, ypos,oxpos,oypos : real; x,y : real;
   name : string[18]; targetx, targety  : real;
   radius   : integer; targetdata : array[1..108,1..6] of
integer;
   trials : integer; tptr : integer;
   curve,aim,extent : real;  xfile,yfile : text;
   ctime : real;tr : real; total : real; mn : integer;
   tp : string[8];  vecfile, accfile, outfile : Text;
   time, lastacc, vel, acc      : Real;  flag, parsed,
   tp0, tp1, tp2, tp3, tp4, tp5, tp6, tp7, tp8 :Boolean;
   loop, loop1, loop2, loop3, ptr : integer; ttime,
   tdist : real; tpoints : array [0 .. tpmax] of tps;
   movement: array [1 .. tpmax] of m;ttpoints : array[0 ..
tpmax] of tps;
Function distance(x1,y1,x2,y2 : real) : real;
Begin
 distance := sqrt((x1-x2)*(x1-x2) + (y1-y2)*(y1-y2));
end;
Function distanceb(x1,y1,x2,y2 : real) : real;
var distance : real;
Begin
 distance := sqrt((x1-x2)*(x1-x2) + (y1-y2)*(y1-y2));
 if distance <= radius then distanceb := 0
 else distanceb := distance-radius;
end;
Procedure getTarget;
```

```
Var digit,count,loop1 : integer; infile : text;dummy,d :
string;
    e,target : integer;rad1,rad2,rad3 : integer;
Begin
 if practice then begin
  trials := 24;
  assign(infile,'c:\textp.out')
 end
 Else
  Begin
  trials := 36;
  assign(infile,'c:\text.out');
 End;
 reset(infile);
 readln(infile,dummy);
 DUMMY := copy(dummy,43,2);
 val(dummy,rad1,e);
 readln(infile,dummy);
 DUMMY := copy(dummy,43,2);
 val(dummy,rad2,e);
 readln(infile,dummy);
 DUMMY := copy(dummy,43,2);
 val(dummy,rad3,e);
 count := 2;
 for loop1 := 1 to trials do begin
  if loop1 < 10 then digit := 1;
  if (loop1 >=10) AND (loop1 < 100) then digit := 2;
  if loop1 >= 100 then digit := 3;
  repeat
   readln(infile,dummy);
   count := count + 1;
  until count = 7;
  count := 0;
  d := copy(dummy,26+digit,1);
  val(d,target,e);
  targetData[loop1,4] := target;
  d := copy(dummy,38+digit,1);
  val(d,targetData[loop1,5],e);
  d := copy(dummy,49+digit,1);
  val(d,targetData[loop1,6],e);
  dummy := d;
  if target = 1 then begin
   targetdata[loop1,1] := 100; targetdata[loop1,2] := ty1;
  end;
  if target = 2 then begin
   targetdata[loop1,1] := 540; targetdata[loop1,2] := ty1;
  end;
 if target = 3 then begin
   targetdata[loop1,1] := 100; targetdata[loop1,2] := ty2;
  end;
 if target = 4 then begin
   targetdata[loop1,1] := 540; targetdata[loop1,2] := ty2;
  end;
  if practice then begin
   if loop1<=8 then targetdata[loop1,3] := rad1;
   if (loop1>8) and (loop1 <=16) then targetdata[loop1,3]
 := rad2;
   if loop1>16 then targetdata[loop1,3] := rad3;
  end
```

```
 else
  Begin
  if loop1<=12 then targetdata[loop1,3] := rad1;
  if (loop1>12) and (loop1 <=24) then targetdata[loop1,3]
:= rad2;
  if loop1>24 then targetdata[loop1,3] := rad3;
 end;
write(target,' rad ',targetdata[loop1,3],'    ');
end;{loop1}
close(infile);
end;{proc}
Function error(dt,d2,dm,d3: real; flag1,flag2 : integer) :
real;
VAR ref,move,tol,e : real;
Begin
 if dt<> 0 then begin;
 if d2 = 0 then ref :=  pi/2 else
 ref := ARCTAN(sqrt(dt*dt-d2*d2)/d2);
 if d3 = 0 then move := pi/2 else
 move := ARCTAN(sqrt(dm*dm-d3*d3)/d3);
 if dt = radius then tol := pi/2 else
 if dt<radius then tol :=
ARCTAN(radius/sqrt(radius*radius-dt*dt)) else
 tol := ARCTAN(radius/sqrt(dt*dt-radius*radius));
 {correct for quadrants}
 if flag1 = 2 then ref := pi - ref;
 if flag1 = 3 then ref := pi + ref;
 if flag1 = 4 then ref := 2*pi - ref;
 if flag2 = 2 then move := pi - move;
 if flag2 = 3 then move := pi + move;
 if flag2 = 4 then  move := 2*pi - move;
{find size of difference between move and ref}
 e := sqrt((ref-move)*(ref-move));
 if e > pi then e := sqrt((e-2*pi)*(e-2*pi));
  e := tol - e;
  if e > 0 then error := 0 else error := sqrt(e*e);
  end
  else error := -1;
End;
Procedure errors(t1,t2 : real; mn : integer);
var d2,d3,startx,starty,endx,endy,time,x1,x2,y1,y2 : real;
     flag1,flag2 : integer;
Begin
 time := ctime;
 if Ctime > t1 then begin writeln('Error in
time');halt;end;
  while time <> t1 do begin
   readln(xfile,time);readln(yfile,time);
   readln(xfile,xpos);readln(yfile,ypos);
   if trace then begin
   setcolor(white);
   circle(round(xpos),round(ypos),2);
   setcolor(white);
   end;
   if first then begin
                  if trace then begin
                   setcolor(white);
                   circle(round(xpos),round(ypos),7);
                   setcolor(white);
```

```
                  end;
                  tdist :=
distanceb(xpos,ypos,targetx,targety);
                  first := false;
   end;
   if time <> t1 then begin oxpos := xpos;oypos :=
ypos;end;
  end;
  {x and y are appropreate}
  startx := oxpos;starty := oypos;{here}
  total := 0;
  x1 := oxpos; y1 := oypos;   {here oxpos,oypos}
  repeat
    readln(xfile,time);readln(yfile,time);
    readln(xfile,xpos);readln(yfile,ypos);
  if trace then begin
   setcolor(white);
   circle(round(xpos),round(ypos),2);
   setcolor(white);
  end;
    x2 := xpos; y2 := ypos;
    total := total + distance(x1,y1,x2,y2);
    x1 := x2; y1 := y2;
  until t2 = time; oxpos := xpos;oypos := ypos;
  ctime := time;
  endx := x2;endy := y2;
  extent := distance(startx,starty,endx,endy);  {here}
  curve := ( extent / total );
  if trace then begin
  setcolor(white);
    if lineon then begin
  rectangle(round(startx-4),round(starty-4),
            round(startx+4),round(starty+4));{here}
Line(round(startx),round(starty),round(endx),round(endy));
{here}
  {Circle(round(endx),round(endy),5);}
  rectangle(round(endx-4),round(endy-4),
            round(endx+4),round(endy+4));
        end;
  setcolor(white);
  end;
  {writeln(startx,' ',starty,' ',endx,' ',endy);}
  {having startx,starty,endx,endy how about accuracy?}
  if (targetx >= startx) AND (targety <= starty) then
flag1 := 1;
  if (targetx < startx ) AND (targety <= starty) then
flag1 := 2;
  if (targetx < startx ) AND (targety > starty) then flag1
:= 3;
  if (targetx >= startx) AND (targety > starty) then flag1
:= 4;
  if (endx >= startx) AND (endy <= starty) then flag2 :=
1;
  if (endx < startx ) AND (endy <= starty) then flag2 :=
2;
  if (endx < startx ) AND (endy >  starty) then flag2 :=
3;
  if (endx >= startx) AND (endy >  starty) then flag2 :=
4;
```

```
  aim := error(distance(startx,starty,targetx,targety)
                 ,distance(startx,starty,targetx,starty),
  extent,distance(startx,starty,endx,starty),flag1,flag2);
 {set up the movements reults}
 movement[mn].epv := distance(endx,endy,targetx,targety);
 movement[mn].aim := aim*57.3;
 movement[mn].curvea := curve;
 movement[mn].dist := extent;
 movement[mn].distr :=
(distanceb(startx,starty,targetx,targety))
                   - distanceb(endx,endy,targetx,targety);

 if distanceb(startx,starty,targetx,targety) <> 0 then
BEGIN
 movement[mn].extent := (extent -
distanceb(startx,starty,targetx,targety));
 if movement[mn].extent >= 0 then begin
  if 2*radius >= movement[mn].extent then
movement[mn].extent := 0
  else movement[mn].extent := movement[mn].extent -
2*radius;
 end;
 movement[mn].extent := movement[mn].extent

/distanceb(startx,starty,targetx,targety);
 end
 else movement[mn].extent := -1;
  if distanceb(startx,starty,targetx,targety) <> 0 then
BEGIN
   movement[mn].effect :=
(((distanceb(startx,starty,targetx,targety))
                          -
(distanceb(endx,endy,targetx,targety)))
            /(distanceb(startx,starty,targetx,targety)));
  end
  else movement[mn].effect := -1;
 end;
Function GetPoint : boolean;
Begin
 getpoint := false;
 tplast := tp;
 repeat
  time := tpoints[ptr].time;
  tp   := tpoints[ptr].typetp;
  ptr := ptr + 1;
until NOT((tp = 'max') or (tp = 'min') or  ( (time < 200)
and (time > 0) ) );
  if tp <> 'eot' then begin
  getPoint := true;
 end;
end;{proc}
Procedure dumptps;
 var loop1 : integer;
 Begin
  for loop1 := 0 to ptr do begin
   writeln(tpoints[loop1].time,'
',tpoints[loop1].typetp);
  end;{for}
 end;{proc}
```

```
Procedure dumpm;
var loop1 : integer;
begin
 for loop1 := 1 to mn do begin
  writeln(movement[loop1].typem,'
',movement[loop1].start,' ',movement[loop1].finish);
 end;
end;
Begin
     clrscr;
     gettarget;
     for loop := range to trials do begin
     clrscr;writeln('Processing trial ',loop);
     ptr := 0;
    str(loop,name);
     name := 'c:\temp\acc' + name + '.dat';
     assign(accfile,name);
     str(loop,name);
     name := 'c:\temp\vel' + name + '.dat';
     assign(vecfile,name);
     reset(accfile);reset(vecfile);
     readln(accfile,time);readln(vecfile,time);
     readln(accfile,lastacc);readln(vecfile,vel);
     lastvel:= vel;

     tpoints[ptr].time := 0;tpoints[ptr].typetp := 'nm';
     ptr := ptr + 1;
     ltp := 'nm';
        if  lastacc < 0 then begin
              if lastacc * -1 < zero then lastacc := 0;end
        else
         if lastacc < zero then lastacc := 0;

        if lastacc <> 0 then begin writeln('jerk at
start');lastacc := 0; end;
        lasttp:='';
     mint := time;maxt := time;
     maxp := lastacc;
     minp := lastacc;

     While NOT EOF(accfile) do
     Begin
        tp1 := false;tp2 := false;tp3 := false;tp4 :=
false;
        tp5 := false;tp6 := false;tp7 := false;tp8 :=
false;
        readln(accfile,time);readln(vecfile,time);
        readln(accfile,acc);readln(vecfile,vel);
        {get rid of noise}
        if acc < 0 then begin
         if acc * -1 < zero then acc := 0;end
        else
         if acc < zero then acc := 0;
         if vel < zerov then vel := 0;
         if lastvel < zerov then lastvel := 0;
         {turning points}
if acc > 0 then begin
 if acc >= maxp then begin
  maxp := acc;
```

```
  maxt := time;
 end;
end;
if acc < 0 then begin
 if acc <= minp then begin
  minp := acc;
  mint := time;
 end;
end;
{acc}   if (((lastacc = 0) OR (lastacc < 0)) AND (acc > 0
)
           and (ltp <> 'acc')) then tp1 := true;

{dec}   if ((((lastacc = 0) OR (lastacc > 0)) ANd (acc < 0
))
                      and (ltp <> 'dec')) then tp2 :=
true;

{conV  if ((lastacc < 0) AND (acc = 0 ) AND (vel <> 0))
then tp3 := true;
        if ((lastacc > 0) AND (acc = 0 ) AND (vel <> 0))
then tp3 := true;}

{nm}    if ((lastacc < 0) AND (acc = 0 ) AND (vel = 0) )
then tp4 := true;
        if ((lastacc > 0) AND (acc = 0 ) AND (vel = 0) )
then tp4 := true;

{ca    if ((lastacc > 0) AND (acc = lastacc)) then tp5 :=
true;}

{cd    if ((lastacc < 0) AND (acc = lastacc)) then tp6 :=
true;}
        if (tp1 or tp2 or tp3 or tp4 or tp5 or tp6 or tp7
or tp8) then
        begin
        if (lasttp = 'acc') then begin
           tpoints[ptr].typetp := 'max';
           tpoints[ptr].time := maxt;
           maxp := 0;
           ptr := ptr + 1;
        end;
        if (lasttp = 'dec') then begin
           tpoints[ptr].typetp := 'min';
           tpoints[ptr].time := mint;
           minp := 0;
           ptr := ptr + 1;
        end;
         tpoints[ptr].time := time;
         if tp1 then tpoints[ptr].typetp := 'acc';
         if tp1 then ltp := 'acc';
         if tp2 then ltp := 'dec';
         if tp2 then tpoints[ptr].typetp:='dec';
         if tp3 then tpoints[ptr].typetp := 'con';
         if tp4 then ltp := 'nm';
         if tp4 then tpoints[ptr].typetp:='nm';
         if tp5 then tpoints[ptr].typetp := 'ca';
         if tp6 then tpoints[ptr].typetp := 'cd';
         if tp7 then tpoints[ptr].typetp := 'jump';
```

```
        lasttp :=tpoints[ptr].typetp;
        ptr := ptr + 1;
        if ptr > tpmax then begin
         clrscr;
         writeln('Error! not enough tps allocated');
         halt;
        end;{if}
       end;{if}
       lastvel := vel;
       lastacc := acc;
    end;{while}
           if (lasttp = 'acc') then begin
         tpoints[ptr].typetp := 'max';
         tpoints[ptr].time := maxt;
         maxp := 0;
         ptr := ptr + 1;
       end;
       if (lasttp = 'dec') then begin
         tpoints[ptr].typetp := 'min';
         tpoints[ptr].time := mint;
         minp := 0;
         ptr := ptr + 1;
       end;
    tpoints[ptr].time := time;tpoints[ptr].typetp :=
'eot';
 close(accfile);close(vecfile);
 {dumptps;}
 {test and create tps with jerks}
  if j then begin
    {reopen velocity and acc files}
    str(loop,name);
    name := 'c:\temp\acc' + name + '.dat';
    assign(accfile,name);
    reset(accfile);
    readln(accfile,time);
    readln(accfile,lastacc);
       {get rid of noise}
       if acc < 0 then begin
        if acc * -1 < zero then acc := 0;end
       else
        if acc < zero then acc := 0;
       {get rid of noise}
       if lastacc < 0 then begin
        if lastacc * -1 < zero then lastacc := 0;end
       else
        if lastacc < zero then lastacc := 0;
    ptr := 0; tptr := 0;
    while tpoints[ptr].typetp <> 'eot' do begin
     while NOT( (tpoints[ptr].typetp = 'dec')
                 or (tpoints[ptr].typetp = 'acc')
                 or (tpoints[ptr].typetp = 'eot') )
     do begin

      ttpoints[tptr] := tpoints[ptr];
      ptr := ptr + 1; tptr := tptr + 1;
     end;
      ttpoints[tptr] := tpoints[ptr];
  if tpoints[ptr].typetp <> 'eot' then begin
     {now see if jerk}
```

```
while time <> tpoints[ptr].time do begin
 readln(accfile,time); readln(accfile,acc);
  {get rid of noise}
  if acc < 0 then begin
   if acc * -1 < zero then acc := 0;end
  else
   if acc < zero then acc := 0;
end;
{now at correct time for start of movement}
ttpoints[tptr] := tpoints[ptr];
if tpoints[ptr].typetp = 'acc' then begin
ptr := ptr + 1;{get time for max/min}
if time = tpoints[ptr].time then lastacc := acc;
  while time < tpoints[ptr].time do begin
   lastacc := acc;
   readln(accfile,time);readln(accfile,acc);
  {get rid of noise}
  if acc < 0 then begin
   if acc * -1 < zero then acc := 0;end
  else
   if acc < zero then acc := 0;
   if acc < lastacc then begin {jerk}
    tptr := tptr + 1;
    ttpoints[tptr].typetp := 'IA(a)';
    ttpoints[tptr].time := time;
    repeat
     lastacc := acc;
     readln(accfile,time);readln(accfile,acc);
  {get rid of noise}
  if acc < 0 then begin
   if acc * -1 < zero then acc := 0;end
  else
   if acc < zero then acc := 0;
    until acc >= lastacc;
   end;
  end;{while}
  if tpoints[ptr].typetp<> 'eot' then begin
  ptr := ptr + 1; {move to next phase start}
  if time = tpoints[ptr].time then lastacc := acc;
  while time < tpoints[ptr].time do begin
   lastacc := acc;
   readln(accfile,time);readln(accfile,acc);
   if time < tpoints[ptr].time then begin
   if acc > lastacc then begin {jerk}
    tptr := tptr + 1;
    ttpoints[tptr].typetp := 'IA(b)';
    ttpoints[tptr].time := time;
    repeat
     lastacc := acc;
     readln(accfile,time);readln(accfile,acc);
  {get rid of noise}
  if acc < 0 then begin
   if acc * -1 < zero then acc := 0;end
  else
   if acc < zero then acc := 0;
    until acc <= lastacc;
   end;
   end;{if}
   end;{while}
```

```
  end;{eot}
end{acc now for dec}
else
begin
ptr := ptr + 1;{get time for max/min}
  if time = tpoints[ptr].time then lastacc := acc;
  while time < tpoints[ptr].time do begin
   lastacc := acc;
   readln(accfile,time);readln(accfile,acc);
  {get rid of noise}
  if acc < 0 then begin
   if acc * -1 < zero then acc := 0;end
  else
   if acc < zero then acc := 0;
   if acc > lastacc then begin {jerk}
    tptr := tptr + 1;
    ttpoints[tptr].typetp := 'IA(c)';
    ttpoints[tptr].time := time;
    repeat
     lastacc := acc;
     readln(accfile,time);readln(accfile,acc);
  {get rid of noise}
  if acc < 0 then begin
   if acc * -1 < zero then acc := 0;end
  else
   if acc < zero then acc := 0;
    until acc <= lastacc;
   end;
  end;{while}
  if tpoints[ptr].typetp<>'eot' then begin
  ptr := ptr + 1; {move to next phase start}
  if time = tpoints[ptr].time then lastacc := acc;
  while time < tpoints[ptr].time do begin
   lastacc := acc;
   readln(accfile,time);readln(accfile,acc);
  {get rid of noise}
   if acc < 0 then begin
   if acc * -1 < zero then acc := 0;end
  else
   if acc < zero then acc := 0;
   if time < tpoints[ptr].time then begin
   if acc < lastacc then begin {jerk}
    tptr := tptr + 1;
    ttpoints[tptr].typetp := 'IA(d)';
    ttpoints[tptr].time := time;
    repeat
     lastacc := acc;
     readln(accfile,time);readln(accfile,acc);

  {get rid of noise}
  if acc < 0 then begin
   if acc * -1 < zero then acc := 0;end
  else
   if acc < zero then acc := 0;
    until acc >= lastacc;
   end;
   end;{if}
  end;{while}
  end;{eot}
```

```
        end;{acc now for dec}
        tptr := tptr + 1;
        end;{while not eot}
        end;{if not eot}
        {close(accfile);}
        ttpoints[tptr] := tpoints[ptr];
      for loop1 := 0 to tptr do begin
       tpoints[loop1] := ttpoints[loop1];
      end;{for}
      {take out reset and close}
      reset(accfile);
      time := 0;
      for loop1 := 2 to tptr-1 do begin;
       while time <> tpoints[loop1].time do begin
        otime := time;
        readln(accfile,time);readln(accfile,acc);
       end;
       tpoints[loop1].time := otime;
      end;
      close(accfile);
    end;
{lets construct movements}
   {1. find start delay}
   ptr := 0;      flag := false;
             {rules for parsing}
   if getpoint then begin
    if tp <> 'nm' then begin writeln('error in non
movement');halt;end;
    mn := 1;
    movement[mn].typem := 'nm';
    movement[mn].start := time;
   end;{if}
   parsed := false;
   while not(parsed) do begin
    if getpoint then begin
    {if NOT((movement[mn].start < 80)  and (mn <> 1))
     then begin}
     movement[mn].finish := time;
     mn := mn + 1;
     movement[mn].start := time;
     {end;}
  if parall then begin
      if tp = 'acc' then begin
     if getpoint then begin
      if tp <> 'dec' then begin
         ptr := ptr -1;
         movement[mn].typem := 'acc';
      end
      else movement[mn].typem := 'sm';
     end;
     flag := true;
    end;

    if ((tp = 'dec') AND (NOT flag)) then begin
     if (movement[mn-1].typem = 'acc') then
      begin writeln('Parser error ');halt;end
     else
      movement[mn].typem := 'dec';
    end;
```

```
      if ((tp = 'IA(a)') and (Not flag)) then begin
      if movement[mn-1].typem = 'nm' then begin
writeln('error');halt;end;
       movement[mn].typem := 'IA(a)';
      end;
      if ((tp = 'IA(b)') and (Not flag)) then begin
      if movement[mn-1].typem = 'nm' then begin writeln
('error');halt;end;
       movement[mn].typem := 'IA(b)';
      end;
      if ((tp = 'IA(c)') and (Not flag)) then begin
      if movement[mn-1].typem = 'nm' then begin writeln
('error');halt;end;
       movement[mn].typem := 'IA(c)';
      end;
      if ((tp = 'IA(d)') and (Not flag)) then begin
      if movement[mn-1].typem = 'nm' then begin writeln
('error');halt;end;
       movement[mn].typem := 'IA(d)';
      end;
     end;
   if parsc then begin
       if (tp = 'acc') then begin
        if getpoint then begin
         if (tp = 'dec') then begin
          if getpoint then begin
           if ((tp = 'acc') or (tp = 'nm')) then
            movement[mn].typem := 'Smovement'
           else
           movement[mn].typem := 'Cmovement';
          end;
         end
         else  movement[mn].typem := 'Cmovement';
        end;
        while ((tp <> 'acc') and (tp <> 'nm') and (tp <>
'eot')) do begin
         if getpoint then jerk := 1;
        end;
        ptr := ptr -1;
        flag := true;
       end;
   end;
   if parsm then begin
     if (tp = 'acc') then begin
        movement[mn].typem := 'Smovement';
        while  ((getpoint) and (tp <> 'nm') and (tp <>
'acc')
                 and (tp <> 'eot'))  do begin
        end;
       ptr := ptr-1;
       flag := true;
       end;
   end;
   if parnm then begin
       if (tp = 'acc') then begin
        movement[mn].typem := 'movement';
        while  ((getpoint) and (tp <> 'nm') and (tp <>
'eot'))  do begin
        end;
```

```
      ptr := ptr-1;
      flag := true;
      end;
  end;
  if pardec then begin
      if (tp = 'acc') then begin
       movement[mn].typem := 'movement';
       while  ((getpoint) and (tp <> 'nm') and
                (tp <> 'acc') and (tp <> 'IA(d)') and (tp
<> 'eot'))
                do begin
                end;
               ptr := ptr-1;
               flag := true;
     end;
      if ((tp = 'IA(d)') and (NOT flag)) then begin
       movement[mn].typem := 'corr';
       while  ((getpoint) and (tp <> 'nm') and (tp <>
'eot')
                 and (tp <> 'acc') )  do begin
       end;
       ptr := ptr-1;
       flag := true;
      end;
    end;
      if ((tp = 'nm') and (NOT flag)) then
movement[mn].typem := 'nm';
     flag := false
    end
    else
     parsed := true;
   end;{while}
   movement[mn].finish := time;
   {dumpm;readln};
     {lets work out distance for a movement}
  {need to know radius for trial - this will be a loop for
trial and moves}
  {for each movement need to know start and end}
  {need to know dt - distance to target}
  str(loop,name);
  name := 'c:\temp\mx' + name + '.res';
  assign(xfile,name);reset(xfile);
  str(loop,name);
  name := 'c:\temp\my' + name + '.res';
  assign(yfile,name);reset(yfile);
  {readln(xfile,ctime);readln(yfile,ctime);}
  ctime := 0;
  if trace then begin
  graphdriver := detect;
  InitGraph(graphDriver,GraphMode,'');
  end;
  first := true;
   str(loop,name);
   name := 'c:\move' + name + '.par';
   assign(outfile,name);rewrite(outfile);
   ttime := movement[mn].finish;
   writeln(loop);
   writeln(outfile,loop);{trial}
   writeln(outfile,targetData[loop,4]);{target no}
   writeln(outfile,
```

```
   writeln(outfile,targetData[loop,3]);{radius}
   writeln(outfile,targetData[loop,5]); {direction}
   writeln(outfile,targetData[loop,6]); {distance}
   if trace then begin
    setcolor(white);
circle(targetdata[loop,1],targetdata[loop,2],targetdata[lo
op,3]);
   end;
   writeln(outfile,mn);{number of phases}
   For loop1 := 1 to mn do begin
    writeln(outfile,movement[loop1].typem);
    if loop1 = 1 then
writeln(outfile,movement[loop1].start)
    else writeln(outfile,movement[loop1].start-8.334);
    writeln(outfile,movement[loop1].finish-8.334);
    movement[loop1].dur := movement[loop1].finish -
movement[loop1].start;
    if loop1 = 1 then movement[loop1].dur :=
movement[loop1].dur - 8.334;
    writeln(outfile,movement[loop1].dur);
    writeln(outfile,movement[loop1].dur/ttime);
    targetx := targetdata[loop,1];
    targety := targetdata[loop,2];
    radius := targetdata[loop,3];
    if (movement[loop1].typem = 'nm') then
    begin
      if lastmov = 'nm' then begin writeln('error - 2 non
moves');halt;end;
     if loop1 = 1 then movement[loop1].extent := 0;
     if loop1 > 1 then begin if trace then begin
       setcolor(white);circle(round(xpos),round(ypos),7);
       setcolor(white); end;
      if distance(xpos,ypos,targetx,targety) < radius then
        movement[loop1].extent := -1 else
movement[loop1].extent := 0;  end;
     writeln(outfile,movement[loop1].extent);
    end;
    if ((movement[loop1].typem <> 'nm') AND
         (movement[loop1].dur > 0)) then begin
errors(movement[loop1].start,movement[loop1].finish,loop1)
;
     writeln(outfile,movement[loop1].dist);
     writeln(outfile,movement[loop1].curvea);
     writeln(outfile,movement[loop1].distr/tdist);
     writeln(outfile,movement[loop1].aim);
     writeln(outfile,movement[loop1].extent);
writeln(outfile,movement[loop1].dist/movement[loop1].dur);
     writeln(outfile,movement[loop1].effect);
     writeln(outfile,movement[loop1].epv);
    end; if step then readln;
   lastmov := movement[loop1].typem;
   end;{loop1}close(xfile);close(yfile);
    tr := distance(xpos,ypos,targetx,targety)/radius;
    if (tr >1) and (tr<1.3) then tr := 1;
   writeln(outfile,tr);
   if tr > 1 then begin
      writeln('Error in tr',tr);readln;end;
   close(outfile);  if trace then readln;
  lastmov := '';  end;{loop}END.
```

Appendix A8 Sample Parser Output

```
107
1
40
2
3
7
nm
 0.0000000000E+00
 3.9200000000E+02
 3.9200000000E+02
 3.8393731636E-01
 0.0000000000E+00
acc
 3.9200000000E+02
 4.7500000000E+02
 8.3000000000E+01
 8.1292850147E-02
 3.1016124839E+01
 9.2238235356E-01
 1.2169133632E-01
 9.6842590544E+00
-8.6975984886E-01
 3.7368825107E-01
IA(a)
 4.7500000000E+02
 5.5800000000E+02
 8.3000000000E+01
 8.1292850147E-02
 7.1196910045E+01
 6.7936407994E-01
 2.8723986608E-01
 2.2298992360E+00
-6.5905656234E-01
 8.5779409693E-01
dec
 5.5800000000E+02
 6.0000000000E+02
 4.2000000000E+01
 4.1136141038E-02
 3.9824615503E+01
 2.7676678830E-01
 1.6388511756E-01
 0.0000000000E+00
-7.1474522851E-01
 9.4820513103E-01
IA(c)
 6.0000000000E+02
 7.6600000000E+02
 1.6600000000E+02
 1.6258570029E-01
 1.2457929202E+02
 4.6458382069E-01
 4.1551502788E-01
 0.0000000000E+00
 0.0000000000E+00
 7.5047766277E-01
IA(c)
```

```
 7.6600000000E+02
 8.6300000000E+02
 9.7000000000E+01
 9.5004897160E-02
 1.6643316977E+01
 5.8727285826E-02
 0.0000000000E+00
 0.0000000000E+00
-1.0000000000E+00
 1.7158058739E-01
nm
 8.6300000000E+02
 1.0210000000E+03
 1.5800000000E+02
 1.5475024486E-01
-1.0000000000E+00
 1.9525624190E-01
```

Appendix B1 One-way Anova Tables

Target size

Figure B1.0 shows the mean completion time for each target size, and table B1.0 shows the ANOVA summary.

One-way ANOVA Table for Target Size

Source of Variation	Sums of squares	DF	MS	F	P<
within cells	4661574.24	94	49591.22		
Size	16195224.86	2	8097612.4	163.29	0.001

Table B1.0

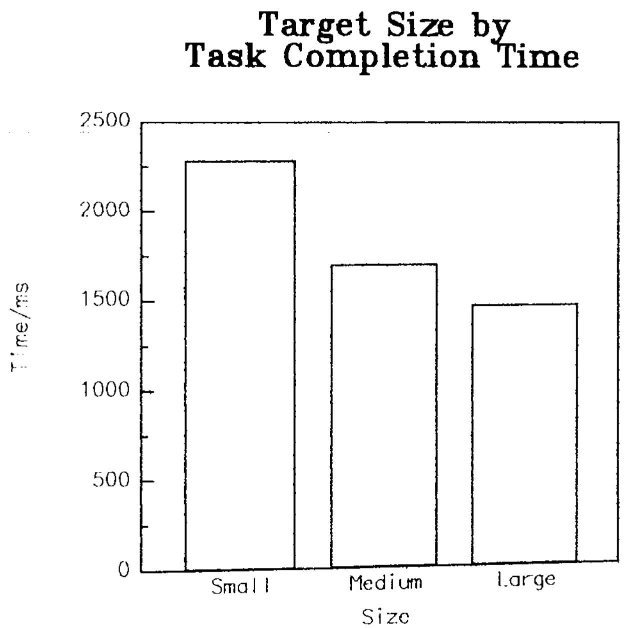


Figure B1.0

Using the Tukey Post-Hoc means test, the means for each target size were found to significantly differ from each other (P<0.01, T 137.57).

Movement amplitude

Figure B1.1 shows the mean completion time for each movement amplitude, and table B1.1 the ANOVA summary table.

One-way ANOVA Table for Movement Amplitude					
Source of Variation	Sums of squares	DF	MS	F	P<
within cells	1664855.43	94	17711.23		
Amplitude	10838633.73	2	5419316.9	305.98	0.001

Table B1.1

Using the Tukey Post-Hoc means test, the means for each movement amplitude were found to significantly differ from each other (P<0.01, T 82.21).

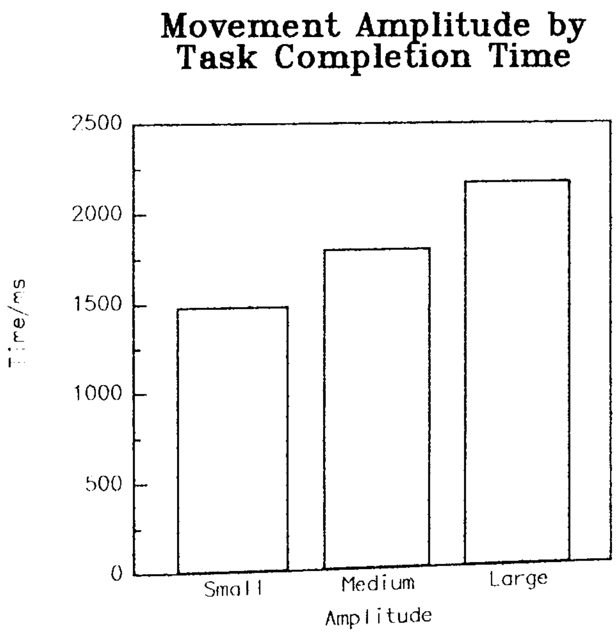


Figure B1.1

Target angle

Figure B1.2 shows the mean completion time for each target angle, and table B1.2 the ANOVA summary.

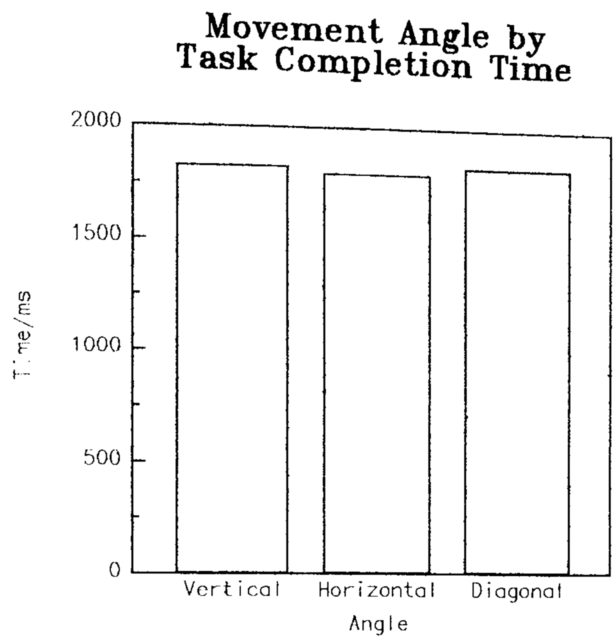


Figure B1.2

One-way ANOVA Table for Movement Angle

Source of Variation	Sums of squares	DF	MS	F	P<
within cells	550644.70	94	5857.92		
Movement Angle	36441.82	2	18220.91	3.11	0.05

Table B1.2

Using the Tukey Post-Hoc means test, the means for each target angle were not found to significantly differ from each other (P<0.05, T 37.56). This is due to the post-hoc tests being more conservative than the Anova tests of significance.

Target location

Figure B1.3 shows the mean completion time for each target location, and table B1.3 the ANOVA summary.

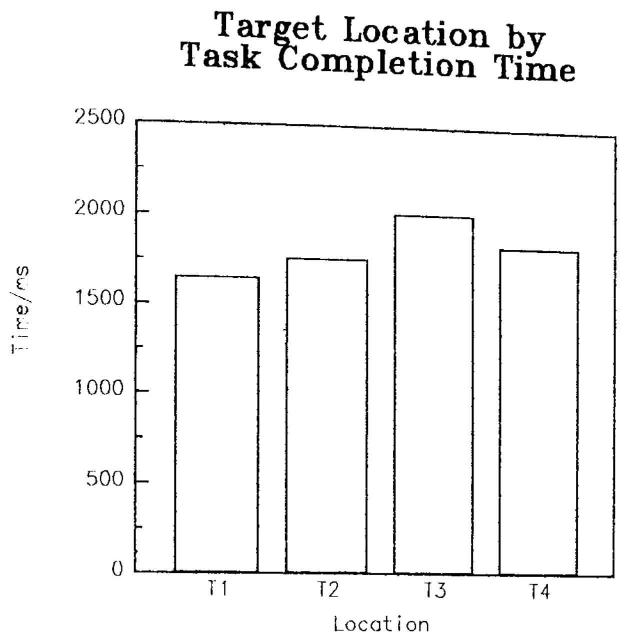


Figure B1.3

One-way ANOVA Table for Target Location

Source of Variation	Sums of squares	DF	MS	F	P<
within cells	3323013.53	141	23567.47		
location	3488368.14	3	1162789.4	49.34	0.001

Table B1.3

Using the Tukey Post-Hoc means test, the means for each target location were found to significantly differ from each other, (P<0.01, T 99.71) for all differences except the difference between target two and four (P<0.05, T 81.54).

Learning

No significant 'learning effect' was found. This was determined by partitioning the trial sequence into three consecutive blocks of 36 trials and performing an ANOVA across these three levels of block. Figure B1.4 shows the three block means.

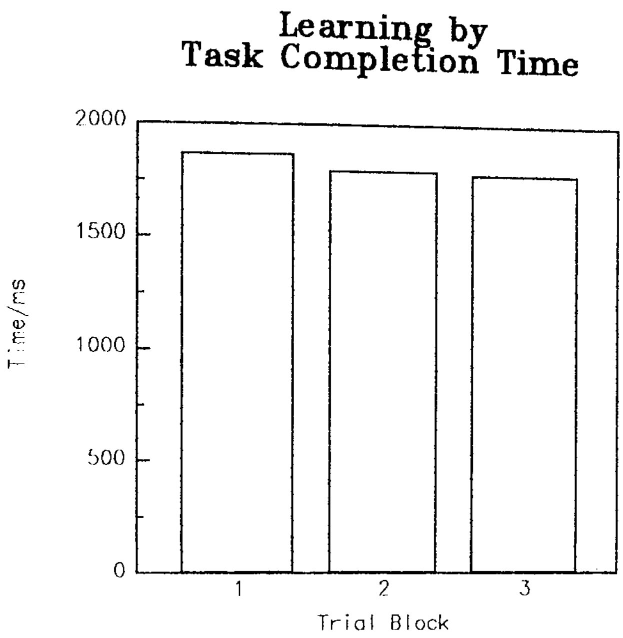


Figure B1.4

Appendix B2 Three-way ANOVA Tables

Three-way ANOVA Table for
Completion Time by
Target Size, Movement Amplitude, and Gain

Source of Variation	Sums of squares	DF	MS	F	P <
within cells	46684164.89	46	1014873.10		
Gain	137992.47	1	137992.47	0.14	0.714
within cells	8063251.37	92	87644.04		
Size	50834527.86	2	25417264.00	290.01	0.001
Gain x Size	3672646.27		1836323.10	20.95	0.001
within cells	4218521.72	92	45853.50		
Amplitude	33961857.72	2	16980929.36	370.33	0.001
Amp. x Gain	43555.77	2	21777.89	0.47	0.623
within cells	4369461.93	184	23747.08		
Size x Amp.	318589.94	4	79647.49	3.35	0.011
(Size x Amp. x Gain)	115454.59	4	28863.65	1.22	0.306
total SS	152420024.53				

Table B2.0

Three-way ANOVA Table for
Reaction Time by
Target Size, Movement Amplitude, and Gain

Source of Variation	Sums of squares	DF	MS	F	P<
within cells	6898766.64	46	149973.19		
Gain	239090.72	1	239090.72	1.59	0.213
within cells	1140156.41	92	12393.00		
Size	133388.35	2	66694.17	5.38	0.006
Gain x Size	40623.13	2	20311.56	1.64	0.200
within cells	574816.01	92	6248.00		
Amplitude	143238.32	2	71619.16	11.46	0.001
Amp. x Gain	81.50	2	40.75	0.01	0.993
within cells	1050239.11	184	5707.82		
Size x Amp.	88732.41	4	22183.10	3.89	0.005
(Size x Amp. x Gain)	6470.56	4	1617.64	0.28	0.888
total SS	10315603.16				

Table B2.1

Three-way ANOVA Table for
Movement Time by
Target Size, Movement Amplitude, and Gain

Source of Variation	Sums of squares	DF	MS	F	P<
within cells	10453171.94	46	227242.87		
Gain	300499.34	1	300499.34	1.32	0.256
within cells	2629663.61	92	28583.30		
Size	14581640.52	2	7290820.30	255.07	0.001
Gain x Size	738446.83	2	369223.42	12.92	0.001
within cells	1982128.69	92	21544.88		
Amplitude	30563518.12	2	15281759.00	709.30	0.001
Amp. x Gain	86876.61	2	43438.31	2.02	0.139
within cells	1881993.32	184	10228.22		
Size x Amp.	667082.38	4	166770.59	16.30	0.001
(Size x Amp. x Gain)	34143.79	4	8535.95	0.83	0.505
total SS	63919165.15				

Table B2.2

Three-way ANOVA Table for
Time on Target by
Target Size, Movement Amplitude, and Gain

Source of Variation	Sums of squares	DF	MS	F	P<
within cells	7223310.25	46	157028.48		
Gain	1984221.56	1	1984221.56	12.64	0.001
within cells	2629439.84	92	28580.87		
Size	8691112.04	2	4345556.00	152.04	0.001
Gain x Size	746685.37	2	373342.68	13.06	0.001
within cells	809555.83	92	8799.52		
Amplitude	132935.66	2	66467.83	7.55	0.001
Amp. x Gain	7008.50	2	3504.25	0.40	0.673
within cells	1292464.53	184	7024.26		
Size x Amp.	70095.94	4	17523.99	2.49	0.044
(Size x Amp. x Gain)	23060.28	4	5765.07	0.82	0.513
total SS	23609889.80				

Table B2.3

Three-way ANOVA Table for
Time on Target (non-movement)
by
Target Size, Movement Amplitude, and Gain

Source of Variation	Sums of squares	DF	MS	F	P<
within cells	4341183.31	46	94373.55		
Gain	1959274.83	1	1959274.80	439.67	0.001
within cells	2065947.02	92	22455.95		
Size	7780774.17	2	3890387.10	173.25	0.001
Gain x Size	960117.10	2	480058.55	21.38	0.001
within cells	464272.18	92	5046.44		
Amplitude	46361.65	2	23180.83	4.59	0.013
Amp. x Gain	13962.44	2	6981.22	1.38	0.256
within cells	724152.60	184	3935.61		
Size x Amp.	20647.04	4	5161.76	1.31	0.267
(Size x Amp. x Gain)	9848.89	4	2462.22	0.63	0.645
total SS	18386541.23				

Table B2.4

Three-way ANOVA Table for
Time on Target (movement)
by
Target Size, Movement Amplitude, and Gain

Source of Variation	Sums of squares	DF	MS	F	P<
within cells	568651.75	46	94373.55		
Gain	78.88	1	78.88	0.01	0.937
within cells	257246.94	92	2796.16		
Size	70553.84	2	35276.92	12.62	0.001
Gain x Size	19314.55	2	9657.27	3.45	0.036
within cells	148022.60	92	1608.94		
Amplitude	23765.60	2	11882.80	7.39	0.001
Amp. x Gain	1301.09	2	650.54	0.40	0.669
within cells	237413.62	184	1290.29		
Size x Amp.	17701.50	4	4425.38	3.43	0.010
(Size x Amp. x Gain)	5313.05	4	1328.26	1.03	0.393
total SS	1349363.42				

Table B2.5

Three-way ANOVA Table for
the Number of Movements made by
Target Size, Movement Amplitude, and Gain

Source of Variation	Sums of squares	DF	MS	F	P<
within cells	171.80	46	3.73		
Gain	29.50	1	29.50	7.90	0.007
within cells	51.98	92	0.56		
Size	372.82	2	186.41	329.93	0.001
Gain x Size	13.43	2	6.72	11.89	0.001
within cells	35.44	92	0.39		
Amplitude	234.32	2	117.16	304.11	0.001
Amp. x GAIN	4.62	2	2.31	5.99	0.004
within cells	37.91	184	0.21		
Size x Amp.	15.23	4	3.81	18.48	0.001
(Size x Amp. x Gain)	2.09	4	0.52	2.53	0.042
total SS	969.14				

Table B2.6

Three-way ANOVA Table for
the proportion of Complex Movements
by Target Size, Movement Amplitude and Gain

Source of Variation	Sums of squares	DF	MS	F	P<
within cells	15532.88	46	337.67		
Gain	18693.66	1	18693.66	55.36	0.001
within cells	9587.75	92	104.21		
Size	17016.85	2	8508.43	81.64	0.001
Gain x Size	86.43	2	43.21	0.41	0.662
within cells	7558.73	92	81.07		
Amplitude	11216.26	2	5608.13	69.17	0.001
Amp. x Gain	58.71	2	29.36	0.36	0.697
within cells	13956.35	184	75.85		
Size x Amp.	438.28	4	109.57	1.44	0.221
(Size x Amp. x Gain)	171.82	4	42.96	0.57	0.687
total SS	94317.72				

Table B2.7

Three-way ANOVA Table for
the number of non-movements by Target
Size, Movement Amplitude and Gain

Source of Variation	Sums of squares	DF	MS	F	P<
within cells	26.49	46	0.58		
Gain	19.35	1	19.35	33.61	0.001
within cells	14.11	92	0.15		
Size	115.06	2	57.53	375.24	0.001
Gain x Size	13.24	2	6.62	43.18	0.001
within cells	7.97	92	0.09		
Amplitude	8.34	2	4.17	48.13	0.001
Amp. x Gain	1.20	2	0.60	6.93	0.002
within cells	13.13	184	0.07		
Size x Amp.	1.70	4	0.42	5.95	0.001
(Size x Amp. x Gain)	0.17	4	0.04	0.59	0.669
total SS	220.76				

Table B2.8

Three-way ANOVA Table for
the duration of the 1st movement by Target
Size, Movement Amplitude and Gain

Source of Variation	Sums of squares	DF	MS	F	P<
within cells	566202.95	46	12308.76		
Gain	1059643.83	1	1059643.83	86.09	0.001
within cells	205834.28	92	2237.33		
Size	135933.68	2	67966.84	30.38	0.001
Gain x Size	8342.74	2	4171.37	1.86	0.161
within cells	296120.24	92	3218.70		
Amplitude	2566511.40	2	1283255.7	398.69	0.001
Amp. x Gain	68047.83	2	34023.91	10.57	0.001
within cells	370924.94	184	2015.90		
Size x Amp.	19078.66	4	4769.66	2.37	0.055
(Size x Amp. x Gain)	7434.47	4	1858.62	0.92	0.452
total SS	5304075.02				

Table B2.9

Three-way ANOVA Table for
the velocity of the 1st movement by Target
Size, Movement Amplitude and Gain

Source of Variation	Sums of squares	DF	MS	F	P<
within cells	0.24	46	0.01		
Gain	0.24	1	0.24	795.09	0.001
within cells	0.04	92	<0.00*		
Size	0.04	2	0.02	46.99	0.001
Gain x Size	0.01	2	<0.00	6.40	0.002
within cells	0.08	92	<0.00		
Amplitude	0.72	2	0.36	395.79	0.001
Amp. x Gain	0.04	2	0.02	23.89	0.001
within cells	0.05	184	<0.00		
Size x Amp.	<0.00	4	<0.00	2.37	0.437
(Size x Amp. x Gain)	<0.00	4	<0.00	0.92	0.761
total SS	1.46				

Table B2.10

* SPSSPC displays some numbers to only 2 D.P.

Three-way ANOVA Table for
the effectiveness of the 1st movement by Target
Size, Movement Amplitude and Gain

Source of Variation	Sums of squares	DF	MS	F	P<
within cells	8.08	46	0.18		
Gain	0.34	1	0.34	1.95	0.170
within cells	6.78	92	0.07		
Size	3.53	2	1.77	23.99	0.001
Gain x Size	0.25	2	0.13	1.71	0.187
within cells	5.23	92	0.06		
Amplitude	0.24	2	0.12	2.10	0.128
Amp. x Gain	0.02	2	0.01	0.19	0.827
within cells	10.04	184	0.05		
Size x Amp.	0.75	4	0.19	3.44	0.010
(Size x Amp. x Gain)	0.72	4	0.18	2.29	0.012
total SS	35.98				

Table B2.11

Three-way ANOVA Table for
the error of extent for the 1st movement by Target
Size, Movement Amplitude and Gain

Source of Variation	Sums of squares	DF	MS	F	P<
within cells	5.40	46	0.12		
Gain	0.09	1	0.09	0.75	0.389
within cells	3.25	92	0.04		
Size	2.58	2	1.29	36.53	0.001
Gain x Size	0.10	2	0.05	1.45	0.241
within cells	3.62	92	0.04		
Amplitude	1.52	2	0.76	19.26	0.001
Amp. x Gain	0.05	2	0.03	0.70	0.500
within cells	6.04	184	0.03		
Size x Amp.	0.51	4	0.13	3.89	0.005
(Size x Amp. x Gain)	0.17	4	0.04	1.29	0.277
total SS	23.33				

Table B2.12

Three-way ANOVA Table for
the error of aim for the 1st movement by Target
Size, Movement Amplitude and Gain

Source of Variation	Sums of squares	DF	MS	F	P<
within cells	13377.11	46	290.81		
Gain	163.41	1	163.41	0.56	0.457
within cells	4724.54	92	51.35		
Size	2133.35	2	1066.68	20.77	0.001
Gain x Size	65.23	2	32.62	0.64	0.532
within cells	4473.23	92	48.62		
Amplitude	3382.49	2	1691.25	34.78	0.001
Amp. x Gain	15.01	2	7.51	0.15	0.857
within cells	7256.07	184	39.44		
Size x Amp.	287.11	4	71.78	1.82	0.127
(Size x Amp. x Gain)	251.88	4	62.97	1.60	0.177
total SS	36129.43				

Table B2.13

Three-way ANOVA Table for
the duration of the 2nd movement by Target
Size, Movement Amplitude and Gain

Source of Variation	Sums of squares	DF	MS	F	P<
within cells	328298.74	46	7136.93		
Gain	345879.21	1	345879.21	48.46	0.001
within cells	474925.89	92	5162.24		
Size	13872.03	2	6936.02	1.34	0.266
Gain x Size	3055.12	2	1527.56	0.30	0.745
within cells	331676.08	92	3605.17		
Amplitude	941446.99	2	470723.49	130.57	0.001
Amp. x Gain	18348.66	2	9174.33	2.54	0.084
within cells	636137.75	184	3457.27		
Size x Amp.	29089.54	4	7272.39	2.10	0.082
(Size x Amp. x Gain)	12333.45	4	3083.36	0.89	0.470
total SS	3135063.46				

Table B2.14

Three-way ANOVA Table for the velocity of the 2nd movement by Target Size, Movement Amplitude and Gain

Source of Variation	Sums of squares	DF	MS	F	P<
within cells	0.06	46	<0.00		
Gain	0.15	1	0.15	109.47	0.001
within cells	0.05	92	<0.00		
Size	0.03	2	0.01	28.00	0.001
Gain x Size	0.01	2	<0.00	8.59	0.001
within cells	0.05	92	<0.00		
Amplitude	0.31	2	0.15	268.84	0.001
Amp. x Gain	0.02	2	0.01	18.88	0.001
within cells	0.09	184	<0.00		
Size x Amp.	0.01	4	<0.00	2.55	0.041
(Size x Amp. x Gain)	<0.00	4	<0.00	0.56	0.689
total SS	0.78				

Table B2.15