A Model of Non-Preferred Hand Mode Switching

Jaime Ruiz* Andrea Bunt† Edward Lank‡

David R. Cheriton School of Computer Science
University of Waterloo
Waterloo, ON, Canada

Abstract

Effective mode-switching techniques provide users of tablet interfaces with access to a rich set of behaviors. While many researchers have studied the relative performance of mode-switching techniques in these interfaces, these metrics tell us little about the behavior of one technique in the absence of a competitor. Differing from past comparison-based research, this paper describes a temporal model of the behavior of a common mode switching technique, non-preferred hand mode switching. Using the Hick-Hyman Law, we claim that the asymptotic cost of adding additional non-preferred hand modes to an interface is a logarithmic function of the number of modes. We validate the model experimentally, and show a strong correlation between experimental data and values predicted by the model. Implications of this research for the design of mode-based interfaces are highlighted.

Index Terms: H5.2 [User Interfaces]: Interaction styles—

1 Introduction

Mathematical models of performance have been valuable in informing the implementation of interfaces, predicting their use, and designing new interaction techniques. Fitts' Law has been used to predict time for pointing tasks [9], to alter control placement in interfaces [6], and to compare different pointing devices [1]. Models of character generation [4] and keystroke input [5] have been used to predict the expected time to enter data. Finally, the design of new widgets such as crossing-based widgets [2] is a direct result of applying and enhancing models of performance [3].

Our research aims to validate a model for mode-switching time in *tablet applications* – applications that receive the majority of input through an electronic stylus or data tablet. Typical examples of tablet applications are the Windows Journal and Microsoft OneNote applications for Tablet PCs. In these applications, the input device, typically an electronic stylus, is overloaded via a set of states or interface modes. The Windows Journal and Microsoft OneNote applications include interface modes accessed using software buttons at the top of the screen. These buttons allow the mapping of stylus gestures to actions such as inking to create content and erasing, highlighting, and lasso selection to edit content.

Research aimed at improving mode-based interaction is, in general, motivated by the goal of advancing pen-tablet interfaces for the Tablet PC. These improvements and investigations, however, have implications that extend to the broader class of applications that make use of interface mode and stylus interaction. For example, applications such as computer-aided design software and character rotoscoping software are geared toward expert users, and the goal of these software applications is to allow the pointer to be mapped to the drawing of tens or hundreds of highly precise image artifacts.

*e-mail: jgruiz@cs.uwaterloo.ca
†e-mail:abunt@cs.uwaterloo.ca
‡e-mail:lank@cs.uwaterloo.ca

Graphics Interface Conference 2008
28-30 May, Windsor, Ontario, Canada

As a result of the prevalence of modes in tablet applications, a number of studies have contrasted techniques for accessing modes (e.g., [18], [14], and [21]). Of the different techniques compared, *non-preferred hand mode switching* has performed particularly well in comparison to alternatives. With this mode-switching technique, users are able to select the desired mode with their non-dominant hand, while using the stylus to gesture with their dominant hand. Despite the promise of this technique, understanding of how it impacts user performance and, in particular, how it scales is limited.

The goal of this paper is to investigate if, for expert users, mode selection using the non-preferred hand becomes a simple decision-response task and, consequently, the technique's scalability as the number of available modes increases. To do so, we develop a mathematical model based on a novel application of the well-established Hick-Hyman Law [6, 12, 15, 23]. We validate our model using data from a controlled experiment in which users performed a series of moded gestures in a tablet interface, where up to eight interface modes are supported using single and chorded keypresses of the non-preferred hand. Experimental measures demonstrate that the Hick-Hyman Law is an accurate predictor of the time taken to switch modes for users accessing between two and eight modes.

This model of non-preferred hand mode switching has implications for the design of mode-based interfaces for tablet applications. Our model shows that the asymptotic cost of adding modes is a logarithmic function of the number of modes for interfaces that use non-preferred hand mode switching. Since we show that the marginal cost of additional modes is small, our work implies that when using non-preferred hand mode-switching, providing a rich set of interface modes is a valid design decision when creating interfaces.

The paper is organized as follows. First, we explore related work, with a focus on non-preferred hand mode switching in stylus interfaces and the Hick-Hyman Law. We then describe a mathematical model of non-preferred hand mode switching and an experiment designed to test this model. This is followed by the presentation of the results. Finally, we discuss the implications of this research to the design of tablet interfaces.

2 Related Work

Many researchers have studied variations in interaction techniques for stylus input systems that aim to allow both command and input in a fluid manner [2, 8, 13, 18, 22, 14]. Past research can be separated into research that seeks alternatives to modes, and research to improve the accessibility of software modes. The goal of this paper is to improve accessibility of modes in interfaces by developing an enhanced understanding of the expected performance of a specific mode-switching technique.

A significant amount of comparative research exists in mode switching. Li et al. [18] studied five different existing mode-switching techniques and concluded that, of the five techniques, non-preferred hand performed best based upon the metrics of speed (fastest), error rate (second lowest), and user preference (most preferred). Hinkley et al. [13] proposed a post-gesture delimiter technique called a "pig-tail" for determining gesture interpretation and compared it to using a handle, a timeout, or a button to alter a gesture's "mode". Grossman et al. [10] proposed "hover widgets"; where the tracking state of a Tablet PC is used to access modes, and compared it to using a software button to switch interface modes. This type of comparative research is invaluable in determining the relative merits of one technique over another.

While a significant amount of comparative research exists, we are not alone in our desire to classify the scalability of techniques, or to combine the specification of models of performance with the study of mode switching. For example, when studying pressure widgets, Ramos et al. [20] found that a maximum of six unique states can be controlled using pressure, thus determining the limits on scalability of pressure widgets. As well, Accot and Zhai [2] proposed "crossing-based" interfaces, where instead of pointing, the user moves the cursor beyond the boundary of the targeted item. Accot and Zhai then extended Fitts' Law to take into account the bivariate pointing tasks (e.g., directional vs. amplitude constraints) that underly goal crossing [3]. Hinkley et al. [14] developed and used keystroke-level model (KLM) analysis for their Springboard technique to demonstrate that the temporal efficiency of local marking menus is often minimized by other ancillary costs.

Despite the benefits of comparing mode-switching techniques, it remains difficult to compare previous techniques with new ones. It also remains difficult to predict changes in performance of techniques as interfaces increase in complexity, i.e., as interfaces add more modes. Our goal is to overcome these challenges by developing a model that characterizes the performance of non-preferred hand mode switching. To do this, we first present some background in psychology research that underlies non-preferred hand mode switching. We then introduce the Hick-Hyman Law and our model of non-preferred hand mode switching.

2.1 Non-Preferred Hand Mode Switching

As we discussed briefly in the introduction, non-preferred hand mode switching is an asymmetric bimanual task – a two-handed task where each hand has a different role. The non-preferred or non-dominant hand controls the state, or mode, of the interface, while the preferred hand performs the moded action. For example, given the task of drawing a coloured line on a Tablet PC, the non-preferred hand would perform the colour selection (i.e., the mode selection), while the preferred hand would draw the line (i.e., the moded action). The rationale behind the design of asymmetric bimanual tasks is based upon a bimanual coordination model called the kinematic chain model [11]. The kinematic chain model states that the efficiency of bimanual interaction is achieved by allowing the action of the non-preferred hand to both precede and set the frame of reference for the action of the preferred hand.

As a result of these exceptions, recent work in non-preferred hand mode switching has explored bimanual parallelism in mode switching. First, Lank et al. [17] explored allowing bimanual parallelism in two-mode interfaces. They designed three variants of non-preferred hand mode switching, one that followed the kinematic chain model and two others that violated it by allowing the tasks of the two hands to overlap. They observed that, if the mode selection could partially overlap the pen gesture, users performed faster. Further, they observed that for the two-mode case, there was no statistically significant difference in the time taken to draw a moded or unmoded gesture, a phenomenon they dubbed '*cost-free*' mode switching.

More recently, we [21] have compared two different techniques for non-preferred hand mode switching to control three and four modes in a pen-tablet interface: one that followed the kinematic chain model and another that allowed parallelism. We examined three time intervals in our study: the time until mode was activated, the time between mode activation and pen down, and the time to draw a stroke. The main results of our previous work were as follows. First, we demonstrated that the temporal benefits of parallelism extends to three- and four-mode interfaces. Second, we found the temporal benefits of parallelism are the result of a reduction in the time to activate a mode and the time between the mode switch and pen down. Finally, the results showed that the temporal cost of mode switching increases as the number of available modes exceeds two.

The purpose of our previous work was to examine if the temporal benefits of parallelism in non-preferred hand mode switching could be extended to four-mode interaction and whether parallelism would continue to result in a '*cost-free*' mode switch. The purpose of this study is to present and validate a temporal model for non-preferred hand mode switching that describes the asymptotic cost of increasing modes to an pen-tablet interface.

2.2 The Hick-Hyman Law

While the results in the above section further our understanding of which variant of non-preferred hand mode switching is best, it is not clear whether the technique can or should be scaled to more than four modes. The goal of this paper is understand whether scalability can be modeled by the Hick-Hyman Law.

The Hick-Hyman Law [12, 15] describes the time to respond to a set of alternative choices. In Hick's [12] original experiment, subjects were seated in front of a set of lights. Based on the location of the light, subjects pressed a corresponding button with one of their ten fingers. In Hyman's [12] experiments, subjects responded to lights with vocalizations. Hick found that response time was a logarithmic function of the number of equally probable choices. Hyman, allowing variation in the probability of any one choice, found a linear correlation between response time and information entropy. Information entropy, measured in bits, is calculated using the formula:

$$H = \sum_{i=1}^{n} p_i \log_2 \left(\frac{1}{p_i} \right) \tag{1}$$

where n is the number of alternatives and p_i is the probability of the ith alternative. If the probability of the alternatives are equal, equation 1 can be expressed as:

$$H = log_2(n) \tag{2}$$

The time taken to respond to n alternatives is therefore:

$$RT = a + b(H), \tag{3}$$

where a and b are empirically determined constants. When a user responds to n equiprobable alternatives, the response time can be expressed as:

$$RT = a + blog_2\,(n)\,. \tag{4}$$

We are unaware of any research using the Hick-Hyman Law to describe the choice reaction time required to plan and activate mode switches in tablet interfaces, however, the law has been used to predict performance in hierarchical menus and in the design of virtual keyboards. Landauer and Nachbar[16] showed that decision times for menu-item selection in hierarchical menus correlated with the times predicted by the Hick-Hyman Law. Cockburn et al. [7] extended these results to accommodate the transition from novice to expert behavior. In the evaluation and design of virtual keyboards, Smith and Zhai [24] used the Hick-Hyman Law as a tool to inform the placement of virtual keyboard keys to optimize performance.

3 A Model of Non-Preferred Hand Mode Switching

As in the menu-selection and typing tasks modeled in previous work, mode switching can also be formulated as a decision problem: Given n possible modes for any gesture, the user must select the appropriate mode. In this section, we describe the individual components of mode-selection tasks and how these different components combine to produce a model of mode-switching performance based on the Hick-Hyman Law.

A task requiring a mode switch can be broken down into four time periods: perception and planning time, T_c; mode activation time, T_m; the time interval between mode switch and start of the gesture, T_{int}; and the time to complete the gesture, T_s. Since separating perception and planning time (T_c) from the beginning of mode activation (T_m) is difficult, it is typical in prompted experimental tasks to measure the interval to initial response [7, 12, 15, 17, 21]. As a result, our model describes three salient time intervals: the time until mode switch occurs, $T_c + T_m$; the time interval between mode switch and pen action, T_{int}; and the gesture time T_s.

Given these time intervals, we define the task time for non-preferred hand mode switching as:

$$T_{Task} = (T_c + T_m) + T_{int} + T_s \tag{5}$$

We hypothesize that the perception, planning and mode activation time, T_c+T_m, can be considered a response to a set of alternatives. Thus, $T_c + T_m$ is described by the Hick-Hyman Law. Substituting $T_c + T_m$ with the Hick-Hyman Law results in the following equation:

$$T_{Task} = a + b\sum_{i=1}^{n} p_i \log_2\left(\frac{1}{p_i}\right) + T_{int} + T_s \tag{6}$$

where a and b are empirically determined constants. As we have already noted in the previous section, n is the number of available modes and p_i is the probability of the ith alternative. If we assume that all modes are equally probable, the model becomes:

$$T_{Task} = a + blog_2\,(n) + T_{int} + T_s \tag{7}$$

Note that the model claims that the time to initiate the gesture after the mode switch (T_{int}) and the time to perform the gesture (T_s) remain constant regardless of the number of modes in the interface. Therefore, for the model to be an accurate predictor of task efficiency as the number of modes increases, the following must be valid: the number of modes must affect the time required to complete a task; the time required for T_{int} and T_s should remain constant; and the time required to complete the task must increase logarithmically.

4 Experiment

The goal of our experiment was to validate our model of non-preferred hand mode switching formulated in the previous section. To do so, we measured performance of moded gestures in a tablet interface with an application that supported between two and eight modes.

4.1 Participants

Eight people, seven male and one female, all right-handed, participated in the experiment. Participants ranged in age from 21 to 26 years and were recruited from the local university.

4.2 Task

The task given to our participants was similar to the simple line cutting task described by Lank et al. [17]. Participants were shown a line at the top of the screen to indicate which color and thickness of line to draw. Their goal was to set the appropriate mode via keypresses with their non-preferred hand and then to draw a line bisecting two vertical bars. The experimental interface is shown in Figure 1. Also shown are the desired output (the black bar at the top of the screen) and a user's gesture (the black line bisecting the two vertical bars in the center of the screen).

Line bisection in our experimental task controls what is drawn by requiring a minimum gesture length of 1000 pixels. We have proposed a model of task time that includes mode-switching time, $T_c + T_m$, the time interval between mode switching and drawing, T_{int}, and the drawing time, T_s. Our model claims that $T_c + T_m$ are correlated with information entropy contained in the mode decision and that T_{int} and T_s are independent of number of modes. The line bisection ensures within-condition deviations in T_s are not a result of drawing a shorter gesture.

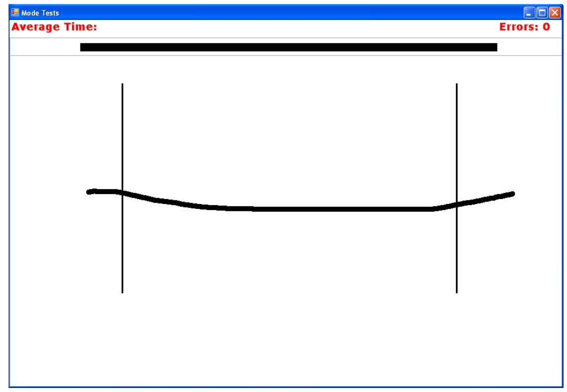

Figure 1: The experimental task. Subjects were asked to draw a line bisecting two vertical bars in the mode indicated at the top of the screen.

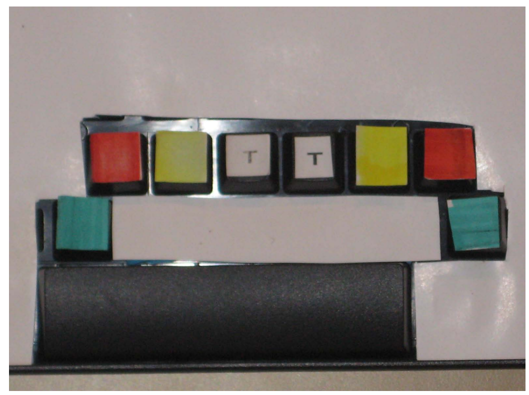

Figure 2: Modified keyboard used for input by the non-preferred hand.

Keypresses with a participant's non-preferred hand were performed on a modified USB keyboard, which is depicted in Figure 2. All keys except for the eight keys shown and the space bar were removed. The space bar was disabled but not removed to give participants a place to rest their palms. Participants positioned their non-preferred hand's index, middle, ring, and pinky fingers above either the left four keys (for right-handed subjects) or the right four keys (for left-handed subjects).[1] Labels were placed on the keys to indicate the mapping of keys to modes.

As is typical in tasks where performance is being measured (e.g., [7, 12, 15, 17]), participants were told to draw as quickly as possible without errors in a verbal orientation to the experiment. This is similar to directives in typical Fitts' Law tasks.

4.2.1 Modes

In our experimental task, mode switching with the non-preferred hand supported up to eight unique modes. In default mode, with

[1] While in the end we did not have any left-handed participants, we had originally designed the experimental task to accommodate both right-handed and left-handed participants.

no keys pressed, the line drawn with the stylus was a thin black line. Pressing the index finger onto its button created a thick line. Colours yellow, red, and green were mapped to the middle, ring, and pinky finger respectively. Using the index finger and another finger simultaneously, a chorded gesture, resulted in a thick coloured line. For example, pressing the ring and index fingers would allow the participant to draw a thick red line.

4.3 Apparatus

All experiments were conducted on two identically configured Toshiba R15-S822 Tablet PC's with an attached USB numeric keyboard. The tablets ran custom software written in C# using Microsoft's Tablet SDK and Visual Studio .NET.

4.4 Procedure and Design

In our experiment, the independent variable is the number of modes. Each participant performed the experimental task with two, four, six, and eight-mode conditions. For the two-mode condition, the participant chose between a default black line and a moded thick black line. Each condition added one color. For example, the four-mode condition added yellow and thick yellow line modes to the two modes in the previous condition.

Within each condition, the participants performed two blocks. They were informed that the first block was a practice block and the second was an experimental block. We also informed them that we were recording timing information for both practice and experimental blocks. The goal of the experimental design was to measure expert mode switching. Therefore, conditions were presented in increasing order of available modes. Progressing from the two-mode to eight-mode condition leveraged learning from earlier experimental conditions in later conditions. While counterbalancing would have permitted a wider range of analysis, we were concerned that introducing an eight-mode application immediately to some participants would create too high an experimental load and/or require too much training to be feasible in a single-session experiment. By having participants complete conditions in order of increasing modes, we were able to manage the amount of new information presented to the user in each condition.

A challenge of this experiment, and other similar experiments that try to measure skilled performance, is balancing the need for training against the need to control the length of the experiment. To obtain expert performance, two options present themselves. The first option is to assume that participants can learn a new technique during a prescribed practice block. The second is to allow subjects to continue practicing until their performance converges in some way. To control the length of the experiment and create homogeneity across participants, we chose the first option. We split the number of gestures performed by a participant equally between practice and experimental gestures.

The experiment still consumed a significant amount of time. The average time for our participants to draw one gesture from the presentation of the desired mode to the pen up event was about one second. There was a five-second interval between each pen-up event and the presentation of the next gesture in order to provide the following feedback to the user: the current task time, their overall best time, and if the last task was completed without errors. As a result, each gesture consumed six seconds. Furthermore, each user drew 210 practice gestures and 210 experimental gestures, a total 420 gestures. We also allowed three minute rest breaks between the practice and experimental block within each condition and between the conditions. The minimum total time for the experiment was 3600 seconds, or one hour, and subjects typically took seventy minutes.

The gestures were split between conditions as follows. In the two-mode condition, the participant performed a practice block of 30 gestures and an experimental block of 30 gestures. Within each block, the gestures were split equally between the two modes, black and thick black. In the four-, six-, and eight-mode conditions, practice block gestures were weighted toward the new modes, while experimental block gestures were split equally between all modes. Therefore, while the participant would perform an identical number of practice and experimental gestures, there were 15 practice gestures in each of the new modes and the remaining practice gestures were distributed among the other modes. Following the practice block, the participant would complete the experimental block, where he or she was asked to draw ten gestures in each mode. For example, in the four-mode condition the participant would draw 80 gestures: 15 yellow, 15 thick yellow, 5 black, and 5 thick black lines in the practice block; and 10 gestures in each of the four modes in the experimental block. Within each block the order of the gestures was randomized.

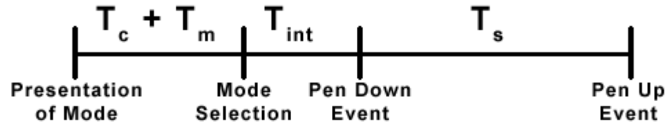

Figure 3: Time intervals of interest in prompted mode switching.

Over 8 participants, timing information for 3360 gestures was collected, as follows:
8 participants
X (30 + 40 + 60 + 80) gestures during the four conditions
X 2 blocks per condition
= 3360 gestures

Of the 3360 total gestures, 1680 were experimental gestures (from the experimental blocks), and 1344 of the experimental gestures required mode switching.

4.5 Measurements

The interface measured the time taken for each line cutting task and recorded errors made by the user. Timing started after presentation of the desired mode and concluded when the participant lifted the pen from the tablet surface after drawing. Subjects were given five seconds to reposition in preparation for the next line cutting task.

Timing information was calculated by obtaining the current value of the hardware's high-resolution performance counter. While the use of the performance counter on the Toshiba R15-S822 Tablet PC allowed 100 nanosecond precision in the timing information, timing values reported here are rounded to the nearest 10ms interval. Given within- and between- participant variances in timing, rounding to 10ms intervals preserves all significant information contained in our measurements.

Recall from our model that the total time required to complete the line crossing task can be divided into three measurable components: the temporal cost associated with perception, planning, and mode activation ($T_c + T_m$); the time between the mode switch and the initiation of the pen gesture (T_{int}); and the time to perform the gesture (T_s). Figure 3 summarizes the timing intervals and the corresponding interface events.

Errors were grouped into two classes: mode errors and drawing errors. Mode errors occurred when participants were not in the desired mode. Drawing errors occurred when the drawn line did not bisect the two lines, as required by the task.

4.6 Hypotheses

Our primary experimental hypothesis is that our model of mode switching holds. In other words:

- There exists a linear correlation between information entropy, H, and $T_c + T_m$.

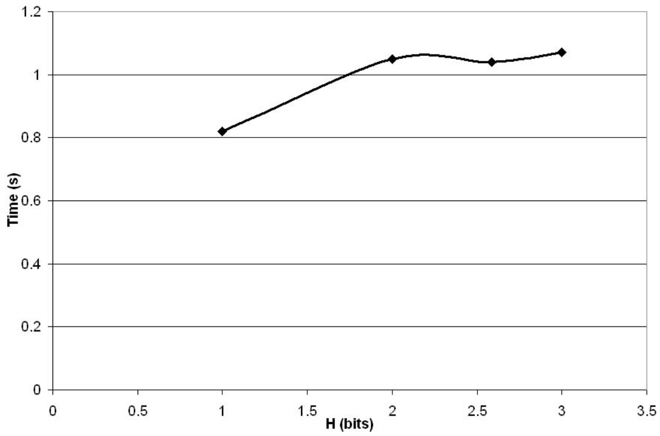

Figure 4: Total time to complete the task by $\log_2{(numberofmodes)}$ (H).

Prior to testing this hypothesis, we first examine the impact of modes on: (1) total task time (T_{Task}); (2) $T_c + T_m$; and (3) T_{int} and T_s. Recall our model of task time, specifically:

$$T_{Task} = a + b\log_2{(n)} + T_{int} + T_s \tag{8}$$

We analyze (T_{Task}) to determine whether increasing modes does affect task time in tablet interfaces. The remaining two analyses allow us to understand whether the effect of increasing the number of modes is restricted to $T_c + T_m$, the perception, planning and activation component of task time. If T_{int} and/or T_s are also affected by the number of modes, our model of task in pen interfaces needs refinement.

5 Results

The results section is organized as follows. First, we discuss the correlation between our model and our experimental data. Next, we perform some exploratory data analysis, where we use our training data to examine situations where modes are not equiprobable.

5.1 Analyzing Experimental Data

Data were analyzed using repeated-measures ANOVA with the number of modes as the within-subjects factor. In all post-hoc pairwise comparisons, we used the Bonferroni adjustment to protect against Type I error.

Table 1 displays the means of the planning and activation time ($T_c + T_m$), the time between mode switch and start of the pen gesture (T_{int}), the time to complete the gesture (T_s), and the average total time to complete a gesture for each condition. Means are calculated by averaging the times for each individual participant and then averaging across participants. As anticipated, we observe an increase in $T_c + T_m$ as the number of modes increases.

In our experimental data, all modes have equal probability. Recall that the information entropy in bits for equiprobable choices is defined as:

$$H = log_2(n) \tag{9}$$

Figure 4 shows a graph of total time, T_{Task}, as a function of information entropy.

Prior to testing model fit, we examined whether or not the number of modes available impacts total task time. Analysis of variance shows that there is a significant main effect of the number of modes on this dependent variable ($F_{3,5} = 12.593, p < .001$). Post-hoc pairwise comparisons revealed that the two-mode condition is significantly different from both the six-mode ($t_7 = -5.225, p = .001$) and eight-mode conditions ($t_7 = -6.860, p < .001$). The remaining comparisons did not reach significance.[2]

We then verified that the number of modes impacts only the perception, planning, and motor activation time, and not the time to initiate and complete the gesture. Analysis of variance for condition on $T_c + T_m$ shows a significant effect of condition ($F_{3,5} = 22.826, p < .001$). Pairwise comparisons indicate that the two-mode condition is significantly different from all other conditions, while the remaining comparisons did not reach significance. There was no significant effect of condition on either T_{int} ($F_{3,5} = 1.460, p = 0.269$) or T_s ($F_{3,5} = 2.360, p = 0.101$). Qualitatively we do note a slight jump in T_{int} from the two-mode to the four-mode condition. We address this point in our discussion of these results.

We now focus on the fit between our model and the experimental data. Figure 5 plots $T_c + T_m$ against information entropy. We see a strong linear correlation between the observed $T_c + T_m$ and the Hick-Hyman Law ($R^2 = 0.908$). With one exception we also see strong correlations for each individual participant. Figure 6 shows each participant's curve, while Table 2 indicates the correlations between each participant's data and the model. The only participant who does not correlate strongly with the model is User 2, who appears to be an outlier with respect to the remaining participants, particularly in the four-mode case.

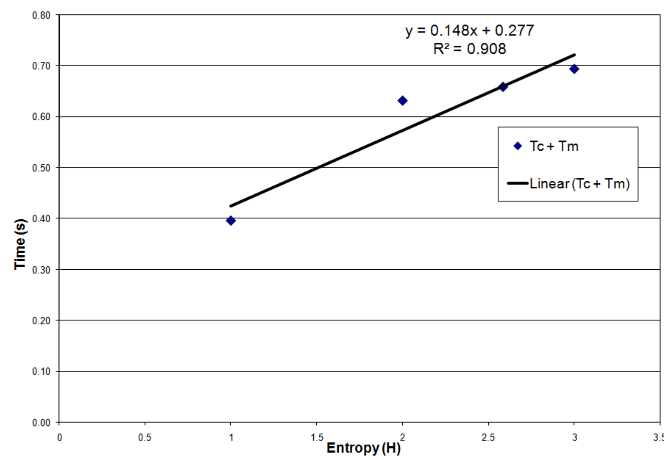

Figure 5: Mean times of $T_c + T_m$ by information entropy.

User	R^2
User 1	0.968
User 2	0.543
User 3	0.902
User 4	0.856
User 5	0.972
User 6	0.832
User 7	0.986
User 8	0.973

Table 2: Model fit for each individual participant.

Finally, we analyzed the number of errors, means and standard deviations for which are displayed in Table 3. There was no significant effect of the number of modes on the percentage of trials with mode errors ($F_{3,5} = 1.446, p = 0.258$) or on the percentage of trials with drawing errors ($F_{3,5} = 2.408, p = 0.096$). We do note from the Table 3, however, that participants tended to make fewer

[2] With the Bonferroni adjustment, the acceptance threshold was p = 0.008

Modes	$T_c + T_m$ (s)		T_{int} (s)		T_s (s)		Total Time (s)	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
2	0.40	0.04	0.10	0.05	0.25	0.03	0.74	0.07
4	0.63	0.15	0.15	0.08	0.26	0.06	1.03	0.21
6	0.66	0.11	0.13	0.07	0.24	0.05	1.01	0.15
8	0.69	0.10	0.13	0.07	0.24	0.04	1.05	0.14

Table 1: Means and standard deviations for the temporal cost of $T_c + T_m$, T_{int}), T_s, and total time, T_{Task}, to complete the task by number of available modes.

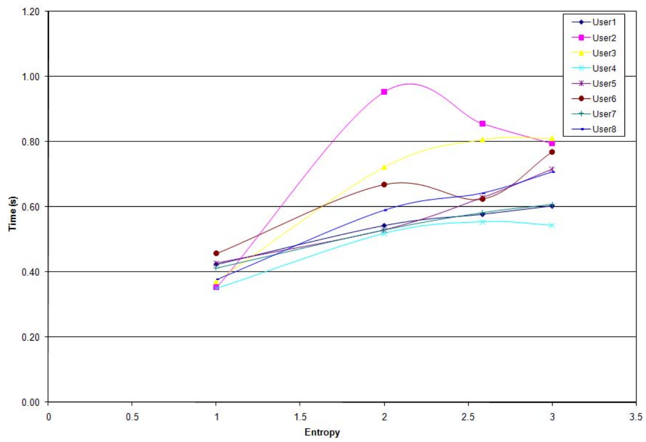

Figure 6: Mean times of $T_c + T_m$ for each participant by information entropy.

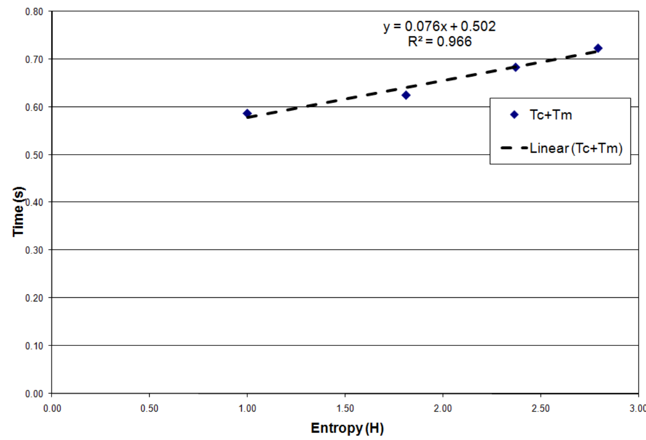

Figure 7: Mean times of $T_c + T_m$ in the training blocks by information entropy. Modes are not equiprobable.

Modes	Drawing Errors (%)		Mode Errors (%	
	Mean	SD	Mean	SD
2	7.1	8.3	3.3	4.4
4	12.8	10.0	5.6	5.9
6	14.6	8.9	5.8	4.5
8	13.1	9.1	7.5	6.9

Table 3: Means and standard deviations for the % of trials that contained drawing errors and the % trials that contained mode errors.

errors, particularly drawing errors, in the two-mode condition as compared to the four-, six- and eight-mode conditions.

5.2 Exploratory Data Analysis: Varying Mode Probabilities

In past work on menu selection [7, 16], the model of menu item access assumed equiprobable choices, as did we in our analysis of experimental gestures. However, as noted in our description of experimental procedure, we recorded times for both the practice and experimental blocks in our experiment. While all modes had equal probabilities in the experimental block, in the practice block we weighted the modes in favour of new modes to encourage practice. Given this unequal weighting of modes, one question that arises is whether our model continues to hold as the probabilities of different modes change. This point is particularly salient in tablet interfaces, where one mode might be more common. For example, in a sketch application such as Windows Journal, the drawing mode might be used more frequently than other modes, and in AutoCAD, the line tool might be used more often than the trimmed corner tool.

To reformulate our model, we replace the logarithmic term by the sum of the inverse of probabilities as described by Equation 1, specifically:

$$T_{Task} = a + b\sum_{i=1}^{n} p_i \log_2\left(\frac{1}{p_i}\right) + T_{int} + T_s \qquad (10)$$

Figure 7 shows the plot of condition against entropy. Despite the unequal probabilities, the weighted entropy model correlates very highly with $T_c + T_m$ ($R^2 = 0.966$).

6 Discussion

In this paper we present a model that describes the temporal cost associated with non-preferred hand mode switching. Using the Hick-Hyman Law, our model defines task time as a logarithmic function of the number of modes available, or information entropy. Our experimental data shows that non-preferred hand mode switching response time does, in fact, correlate strongly with this model ($R^2 = 0.908$). We also found that the mode effects are primarily localized to $T_c + T_m$. Despite the increase in interface complexity, the need to control additional modes appears not to have a significant effect on the cost associated with coordinating hands or with the time taken to draw a gesture.

Based on our training data, we have evidence that the full formulation of the Hick-Hyman Law, which accounts for decision-response tasks where the options are not equiprobable, also holds for non-preferred hand mode switching. While participants' were told that their gestures would be timed in both the training and experimental trials, an additional experiment would be required to verify the strong correlation observed here.

There are three concerns with our data that deserve further investigation, all of which are related, to some extent, to the introduction of chording. First, qualitatively we note a potential order effect with T_{int}. In the four-mode case, when chording was introduced, T_{int} rose and then decreased slightly in the six- and eight-mode conditions. The introduction of chording might have had an adverse effect on a user's ability to coordinate the two hands. However, as users became more familiar with the chording technique, these effects appear to have been reduced. Second, we note that User 2, our participant whose data does not correlate well with our model ($R^2 = 0.543$) seemed to have particular difficulty with the four-mode condition. Finally, error rates in the two-mode condition (without chording) appeared to be lower than in the remaining conditions. While we again did not see a statistically significant difference at the 0.05 level, it might be that chording causes participants to make a larger number of errors.

A follow-up study targeted specifically at understanding what difficulties, if any, users might have in learning and mastering chording is an area of potential future work. Such an experiment would require testing different numbers of modes both with and without chording, and ideally counterbalancing the conditions to rule out any potential order effects. With a counterbalanced experiment, participants could not leverage training from previous conditions. As a result, training would have to be carefully designed so that participants are not overwhelmed when introduced to applications with large numbers of modes. Given concerns with session length, such an experiment might require a between-subjects design.

7 Implications for the Design of Mode-Based Applications

The implications of research in non-preferred hand mode switching are not limited to Tablet PC applications with a small set of modes. Pausch and Leatherby [19] studied the use of keyboard accelerators in mouse-based drawing interfaces. They found that it was common for both novice and experienced users to use their non-preferred hand to switch modes while leaving their preferred hand on the mouse.

More generally, domain-specific drafting software applications such as computer-aided design and character rotoscoping software include a large set of interface modes. In these domains, many users become expert and define their own keyboard shortcuts to access common commands with their non-preferred hand while drawing. As an example of this phenomenon, the AutoDesk knowledge base demonstrates creating a shortcut for the 'line' command.[3]

These examples illustrate the broad applicability of non-preferred hand mode switching research. Specialized hardware, such as our modified keyboard or Tablet PC buttons on the edge of the screen, are one option for accessing modes. However, even in domains where this specialized hardware does not exist, we see the use of the non-preferred hand to control program state. While it was apparent that the use of non-preferred hand speeds the manipulation of interface mode by avoiding homing costs associated with moving between keyboard and mouse, we have also demonstrated in this research the low marginal cost of these non-preferred hand modes.

Alongside efficiency associated with the elimination of homing costs, our implementation of non-preferred hand mode switching makes use of both single keypresses and chording to allow control of up to eight modes with four keys. Our initial assumption was that chording had advantages and disadvantages. One advantage of chording in our experimental design was the ability to provide a one to one mapping of finger to key. This mapping eliminated the cost associated with targeting individual keys, and allowed subjects to rest each finger on its mode key. The disadvantage associated with chording was a hypothesized increase in the complexity of keystroking in the non-preferred hand. We were concerned that chording might introduce additional motor control cost in the action of the non-preferred hand.

While it seems likely that chording is more complex than single keypresses, we observed no significant deviation from our model. In both practice and experimental blocks for four-, six- and eight-mode conditions, a number of chorded operations occurred. It is difficult to make strong inferences about the specific costs associated with chording without a follow-up experiment. However, if chording does not come at a significantly higher cost than single keypresses, the parity of chording might allow its use in form factors with a limited number of hardware buttons, such as the buttons located on the frame of a Tablet PC.

[3] See http://usa.autodesk.com/adsk/servlet/ps/item?siteID=123112&id=2862800&linkID=9240617

8 Summary

In this paper we provide a model describing the temporal cost of non-preferred hand mode switching. Our model is based on the Hick-Hyman Law, a law derived from information theory that describes response time as a linear function of the information entropy. Experiments indicate that this model is an accurate predictor of the time taken to perform a non-preferred hand mode switch for interfaces containing between two and eight modes when modes are equiprobable. Our data indicates that our model is also accurate when modes are not equiprobable. Finally, we discuss implications of this research for expert use of tablet interfaces.

9 Future Work

In section 6 we discussed short-term directions for future work, specifically, additional experiments to better understand the advantages and disadvantages associated with chording, and model fit when modes are not equiprobable. Our longer-term goals are to investigate asymmetric bimanual interactions in different types of tasks, to understand how such techniques generalize to more complex interactions. In particular, we are interested in exploring tasks that allow for more continuous bimanual interaction than mode switching, such as manipulating sliders in a data-set exploration task or controlling pressure in a drawing application.

10 Acknowledgments

The authors would like to thank everyone who participated in our user trials and the anonymous reiewers for their insightful comments. Funding for this research was provided by the Natural Science and Engineering Research Council or Canada, NSERC.

References

[1] J. Accot and S. Zhai. Performance evaluation of input devices in trajectory-based tasks: an application of the steering law. In *Proceedings of the Conference on Human Factors in Computing Systems, CHI 1999*, pages 466–472, 1999.

[2] J. Accot and S. Zhai. More than dotting the i's – foundations for crossing-based interfaces. In *Proceedings of the Conference on Human Factors in Computing Systems, CHI 2002*, pages 73 – 80, 2002.

[3] J. Accot and S. Zhai. Refining Fitts' law models for bivariate pointing. In *Proceedings of the Conference on Human Factors in Computing Systems, CHI 2003*, pages 193–200, 2003.

[4] X. Cao and S. Zhai. Modeling human performance of pen stroke gestures. In *Proceedings of the Conference on Human Factors in Computing Systems, CHI 2007*, pages 1495–1504, 2007.

[5] S. Card, T. Moran, and A. Newell. The keystroke-level model for user performance time with interactive systems. *Communications of the ACM*, 23(7):396–410, 1980.

[6] S. Card, A. Newell, and T. Moran. *The Psychology of Human-Computer Interaction.* Lawrence Erlbaum Associates, Inc, 1983.

[7] A. Cockburn, C. Gutwin, and S. Greenberg. A predictive model of menu performance. In *Proceedings of the Conference on Human Factors in Computing Systems, CHI 2007*, pages 627–636, 2007.

[8] R. Davis. Sketch understanding in design: Overview of work at the MIT AI lab. In *Proceedings of the AAAI Spring Symposium on Sketch Understanding*, pages 24 – 31, 2002.

[9] P. M. Fitts. The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, 47:381–391, 1954.

[10] T. Grossman, K. Hinckley, P. Baudisch, M. Agrawala, and R. Balakrishnan. Hover widgets: using the tracking state to extend the capabilities of pen-operated devices. In *Proceedings of the Conference on Human Factors in Computing Systems, CHI 2006*, pages 861–870, 2006.

[11] Y. Guiard. Asymmetric division of labor in human skilled bimanual action: The kinematic chain as a model. *Journal of Motor Behaviour*, 19(4):486 – 517, 1987.

[12] W. Hick. On the rate of gain of information. *Journal of Experimental Psychology*, 4:11 – 36, 1952.

[13] K. Hinckley, P. Baudisch, G. Ramos, and F. Guimbretiere. Design and analysis of delimiters for selection-action pen gesture phrases in scriboli. In *Proceedings of the Conference on Human Factors in Computing Systems, CHI 2005*, pages 451 – 460, 2005.

[14] K. Hinckley, F. Guimbretiere, P. Baudisch, R. Sarin, M. Agrawala, and E. Cutrell. The springboard: multiple modes in one spring-loaded control. In *Proceedings of the Conference on Human Factors in Computing Systems, CHI 2006*, pages 181–190, 2006.

[15] R. Hyman. Stimulus information as a determinant of reaction time. *Journal of Experimental Psychology*, 45:188–196, 1953.

[16] T. Landauer and D. Nachbar. Selection from alphabetic and numeric menu trees using a touch screen: breadth, depth, and width. In *Proceedings of the Conference on Human Factors in Computing Systems, CHI 1985*, pages 73–78, 1985.

[17] E. Lank, J. Ruiz, and W. Cowan. Concurrent bimanual stylus interaction: a study of non-preferred hand mode manipulation. In *Proceedings of the Conference on Graphics interface*, pages 17–24, 2006.

[18] Y. Li, K. Hinckley, Z. Guan, and J. Landay. Experimental analysis of mode switching techniques in pen-base user interfaces. In *Proceedings of the Conference on Human Factors in Computing Systems, CHI 2005*, pages 461 – 470, 2005.

[19] R. Pausch and J. H. Leatherby. Voice input vs. keyboard accelerators: A user study. In *Proceedings of AVIOS '91 - Tenth Annual Conference of the American Voice Input/Output Society*, pages 9–14, 1991.

[20] G. Ramos, M. Boulos, and R. Balakrishnan. Pressure widgets. In *Proceedings of the Conference on Human Factors in Computing Systems, CHI 2004*, pages 33–40, 2004.

[21] J. Ruiz and E. Lank. A study of the scalability of non-preferred hand mode switching. In *Proceedings of International Conference On Multimodal Interfaces, ICMI 2007*, 2007.

[22] E. Saund and E. Lank. Stylus input and editing without prior selection of mode. In *Proceedings of the Symposium on User Interface Systems and Technology, UIST 2003*, pages 213–216, 2003.

[23] S. Seow. Information theoretic models of HCI: A comparison of the Hick-Hyman law and Fitts' law. *Human-Computer Interaction*, 20:315–352, 2005.

[24] B. Smith and S. Zhai. Optimised virtual keyboards with and without alphabetical ordering - a novice user study. In *Proceedings of INTERACT 2001: Eight IFIP Conference On Human-Computer Interaction*, pages 92–99, 2001.