HUMAN–COMPUTER INTERACTION, 2006, Volume 21, pp. 343–392

Experimental Evaluations of the Twiddler One-Handed Chording Mobile Keyboard

Kent Lyons, Thad Starner, and Brian Gane
Georgia Institute of Technology

ABSTRACT

The HandyKey Twiddler™ is a one-handed chording mobile keyboard that employs a 3 × 4 button design, similar to that of a standard mobile telephone. We present a longitudinal study of novice users' learning rates on the Twiddler. Ten participants typed for 20 sessions using 2 different text entry methods. Each session was composed of 20 min of typing with multitap and 20 min of one-handed chording on the Twiddler. We found that users initially had a faster average typing rate with multitap; however, after 4 sessions the difference became negligible, and by the 8th session participants typed faster with chording on the Twiddler. Five participants continued our study and achieved an average rate of 47 words per minute (wpm) after approximately 25 hr of practice in varying conditions. One participant achieved an average rate of 67 wpm, equivalent to the typing rate of the 2nd author, who has been a Twiddler user for 10 years. We analyze the effects of learning on various aspects of chording, provide evidence that lack of visual feedback does not hinder expert typing speed, and examine the potential use of

Kent Lyons is a computer scientist with an interest in human–computer interaction issues of mobile computing; he is a Research Scientist in the College of Computing at the Georgia Institute of Technology. **Thad Starner** is a computer scientist with an interest in wearable computing and intelligent agents; he is an Assistant Professor in the College of Computing at the Georgia Institute of Technology. **Brian Gane** is a psychologist with an interest in cognitive and motor skill acquisition; he is a graduate student in the School of Psychology at the Georgia Institute of Technology.

CONTENTS

multicharacter chords (MCCs) to increase text entry speed. Finally, we explore improving novice user's experience with the Twiddler through the use of a chording tutorial.

1. INTRODUCTION

Mobile computing is becoming one of the most widely adopted computing technologies. There are currently 1.3 billion mobile phone subscribers and could be as many as 2 billion by 2007 (Baker et al., 2004). Wireless text messaging is widespread, and some researchers have predicted that the number of wireless text messages sent per year will soon reach 1 trillion (Lindstom, 2002). These statistics are remarkable considering the inefficiencies and poor design of current text entry methods for mobile devices.

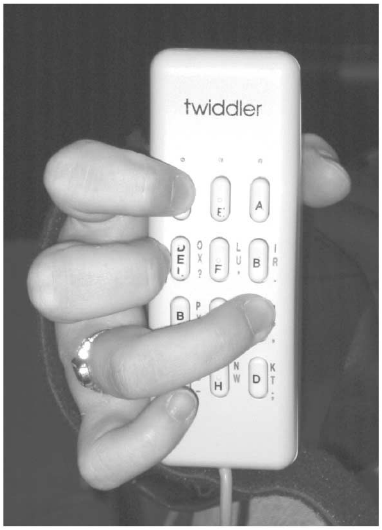
The desire to increase text entry rates has a long history, and recently there has been a resurgence in research exploring how physical keyboards can be used for mobile devices. Improving text entry speed may open new markets for wireless e-mail, which is desired by 81% of consumers, according to one survey ("Women Embracing SMS," 2002), and wireless e-mail is predicted to drive the next stage of the industry's European data revenues. Unexpected segments of the user population may benefit from improved text entry capabilities. For example, the deaf community has adopted wireless texting as a convenient means of communication within the community (Henderson, Grinter, & Starner, 2005).

In this article, we detail our research on the HandyKey Twiddler™ keyboard (see Figure 1; also see Lyons, Gane, Starner, & Catrambone, 2005; Lyons, Plaisted, & Starner, 2004; Lyons, Starner, et al., 2004). First, we describe the Twiddler keyboard and how it compares to typing on similar 3 × 4 keypads of mobile phones. We then present a longitudinal study comparing the learning rates for the Twiddler relative to the de facto standard for mobile phone text entry, multitap. Next, we present a continuation of our study designed to explore expert characteristics of Twiddler typing, and finally, we explore how to improve a novice Twiddler user's typing experience.

The HandyKey Twiddler is a mobile one-handed chording keyboard with a keypad similar to that of a mobile phone (see Figure 2); it has been adopted by many wearable-computer users (Lyons, 2003; Starner, 2000). It has 12 keys arranged in a grid with three columns and four rows on the front. Unlike a mobile phone, the Twiddler is held with the keypad facing away from the user, and each row of keys is operated by one of the user's four fingers (see Figure 3). Instead of pressing keys in sequence to produce a character, multiple keys can be pressed simultaneously to generate a chord. In addition, the Twiddler has several modifier buttons such as Alt, Shift, Control, and so forth on the top–back operated by the user's thumb.

The default keymap for the Twiddler is shown in Figure 4. It consists of single-button and two-button chords that are assigned in an alphabetical order. The nomenclature for our labeling of the chords is derived from the keymap printed on the face of the Twiddler, and this representation is also on the left side

Figure 1. **Chord for the letter "j" on the Twiddler.**



of the Figure 4. The right side of the figure shows smaller versions of the representation that indicate how to type each character. A shaded rectangle represents the corresponding button on the Twiddler that should be depressed to generate that character. Characters a through h each require only a one-button press (single), as indicated by the black rectangles in the first row of Figure 4. The letters i through z are typed with two-button chords. For these letters, two of the buttons on the top row act as Shift keys. The Shift button for i through q is called the red Shift (rectangles in the second row of Figure 4), and the Shift for r through z is the blue Shift (the rectangles on the bottom row).

Below each representation in Figure 4 is the letter that is generated and a four-character code that denotes which keys are pressed in each row. *L* indicates the leftmost button in a row, *M* the middle, and *R* the right button. A *0* means the corresponding finger is not used in the chord. Note that the designation for left and right is from the user's perspective while holding the keypad facing away. As a result, there is a left-to-right mirror between Figure 1 and Figure 4. For example, the chord for *a* is L000, which indicates that a user presses the left button on the top row from his or her perspective. To generate *j* (R0L0), a user presses the right key on the top row and the left key on the third row (see Figure 1).

With traditional keyboards, a character is generated when the corresponding button is pressed. This strategy cannot be used for chording because the user can not press all of the keys for the chord at exactly the same time. In-

Figure 2. **The Twiddler next to the Sony Ericsson T610 mobile phone.**

stead, the Twiddler generates the keycode once the first button of a chord is released. Just before this point, all of the buttons for the chord have been depressed, so the proper keycode can be generated. In Section 6, we explore the relationship between the timings of pressing the buttons and how they relate to learning to chord.

For a chord on the Twiddler, each of the fingers may be in one of four states (pressing one of three buttons on a row, or not pressing anything). Ignoring the "chord" in which no buttons are pressed, there are

$$4^4 - 1 = 255$$

possible chords using the four main fingers. The modifier buttons operated by the thumb allow more chords. HandyKey includes what we have termed *multicharacter chords* (MCCs) in the default keymap: single chords that generate a sequence of several characters. For instance, there are chords for some frequent words and letter combinations such as *and*, *the*, and *ing*. Users can also define their own MCCs. We present an evaluation and analysis of the effects of MCCs on expert typing rates in Section 7.1.

Figure 3. **The Twiddler being held in typing position.**

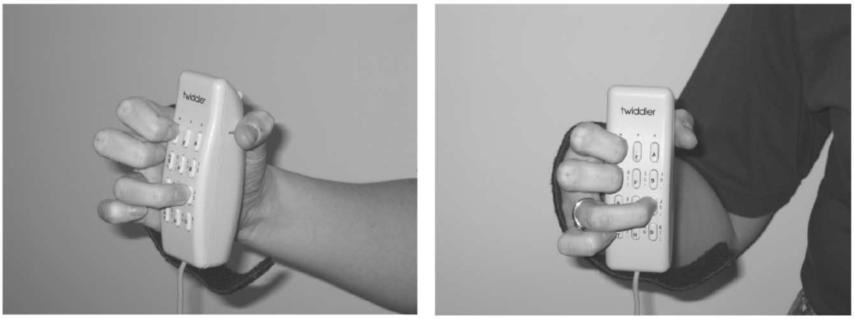
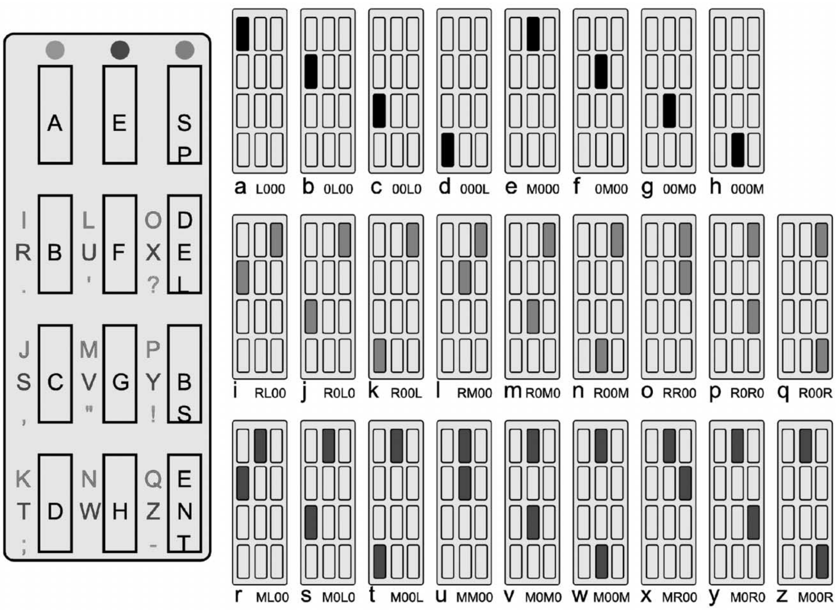


Figure 4. **The keymap for chording on the Twiddler. On the right, each grid of 3 × 4 rectangles represents the keypad from the user's perspective. The shaded rectangles are the buttons that need to be depressed to type the character printed below each keypad. Also displayed is a four-digit textual representation of the chord.**



2. RELATED WORK

2.1. Typing on Mobile Phone Keypads

There are two ways to accommodate the small form-factor keyboards that are resulting from the decrease in size of mobile technology: make the keys very small, like on mini-QWERTY keyboards, or remove the one-to-one mapping between keys and characters. Most phones map more than one character onto a key because they inherited the 12-button keypad of traditional phones. When multiple characters are assigned to one key, a method is needed to disambiguate between the possible options. Wigdor and Balakrishnan (2004) presented a taxonomy with three dimensions for ways to disambiguate: the number of keys used (one or more), the number of presses performed on the key(s), and the possible temporal ordering of key presses (consecutive or concurrent). These methods can be further combined with linguistic models to disambiguate the key presses. Chording on the Twiddler represents a point in this space that uses concurrent presses from multiple buttons.

For mobile phones, multitap is a very common text entry technique. The alphabet is mapped onto 8 of the 12 buttons on the mobile phone keypad resulting in three to four letters per key. To generate a character and disambiguate between the characters on the same key, the user presses a single key multiple times to cycle through the letters until the desired one appears on the screen. Users hold the keypad toward them and can enter text with one or two hands using one or two fingers or thumbs. Once the desired letter appears, users can press the next key to start the process again for the next letter, wait for the automatic timeout, or use a special kill key to bypass the timeout. The timeout is a feature that deactivates the current key after a specified amount of time.

Because multitap is so prevalent on mobile phones, it has become the de facto baseline for which to compare other mobile phone entry techniques. Research has found multitap typing rates for novice users ranging from 7.2 to 8.7 words per minute (wpm) with 15 to 30 min of practice (Lyons, Starner, et al., 2004; MacKenzie, Kober, Smith, Jones, & Skepner, 2001; Wigdor & Balakrishnan, 2003, 2004). James and Reischel (2001) found similar rates for novices. They also reported experts as typing at only 7.93 wpm; however, James and Reischel's only criterion for expertise was that the participants send text messages and they did not indicate how much time these expert participants spent practicing. The other studies showed that as users gain experience, their typing rates can increase to 11.0 to 19.8 wpm (Lyons, Starner, et al., 2004; MacKenzie et al., 2000; Wigdor & Balakrishnan, 2003, 2004). Silfverberg, MacKenzie, and Korhonen (2000) predicted maximum expert typing rates of 20 to 27 wpm. Combined, this research shows that although multitap is a very common typing method, it is also relatively slow.

T9™ is another common mobile phone input method. Like multitap, the T9 method assigns multiple letters to each button on the keypad. However, instead of the user disambiguating every character with multiple button presses, T9 uses language disambiguation. Using a dictionary, T9 presents the most probable string the user is attempting to enter given the input so far. If the presented text is incorrect, the user can press a special key to cycle through possible alternatives. One study found that novice T9 users type 9.1 wpm whereas experts can achieve 20.4 wpm (James & Reischel, 2001). Unfortunately, T9 rates drop drastically once the user needs to enter words that are not in the dictionary, such as proper nouns.

Several new methods have recently been developed for entering text on mobile phone keypads, including LetterWise (MacKenzie et al., 2001), TiltText (Wigdor & Balakrishnan, 2003), and ChordTap (Wigdor & Balakrishnan, 2003). These methods offer novice performance similar to multitap (7.3 wpm, 7.4 wpm, and 8.5 wpm, respectively). In addition, each of these methods produces faster expert typing rates than does multitap given the same amount of practice. LetterWise users achieved a rate of 21 wpm after approximately 550 min of practice. TiltText users reached 13.6 wpm and ChordTap 16.1 wpm with about 160 min of practice.

Figure 5 provides a summary of this work and also includes the results of the studies performed in this work. Where it could be derived, the Experience

Figure 5. **Comparison of mobile text entry rates using 3 × 4 keypads.**

Method	Keyboard	Experience	WPM
Chording	Twiddler	1,500 min	47.1
Chording	Twiddler	400 min	26.2
LetterWise (MacKenzie et al., 2001)	Desktop keypad	550 min	21.0
T9 (James & Reischel, 2001)	Nokia 3210 phone	Expert	20.4
Multitap	Twiddler	400 min	19.8
ChordTap (Wigdor & Balakrishnan, 2004)	Modified Motorola i95cl phone	160 min	16.1
Multitap (MacKenzie et al., 2001)	Desktop keypad	550 min	15.5
TiltText (Wigdor & Balakrishnan, 2003)	Modified Motorola i95cl phone	160 min	13.6
Multitap (Wigdor & Balakrishnan, 2003)	Motorola i95cl phone	160 min	11.0
T9 (James & Reischel, 2001)	Nokia 3210 phone	Novice	9.1
Multitap (James & Reischel, 2001)	Nokia 3210 phone	Novice	8.0
Multitap (James & Reischel, 2001)	Nokia 3210 phone	Expert	8.0
Multitap (Butts & Cockburn, 2002)	Desktop keypad	N/A	7.2
Two key (Butts & Cockburn, 2002)	Desktop keypad	N/A	5.5

Note. WPM = words per minute. N/A = not available.

column shows the approximate number of minutes the novice user spent typing with the given method before the maximum words per minute were calculated. Studies that were not longitudinal but characterized participants as novice or expert, are marked accordingly. In summary, these studies reveal that most text entry methods for mobile devices are relatively slow.

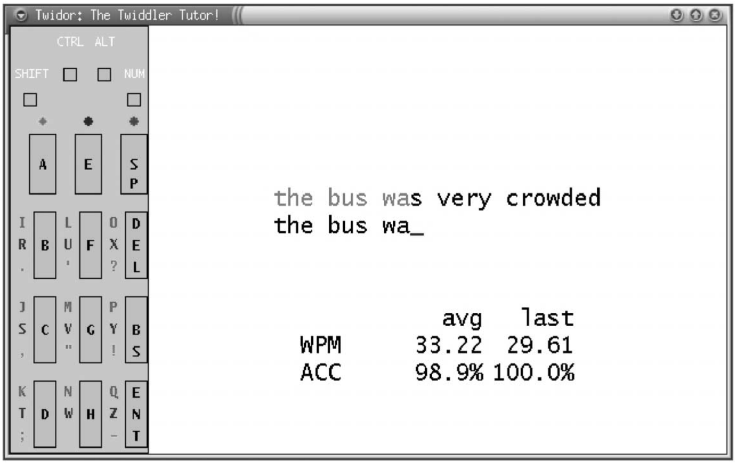
2.2. Chording Keyboards

Although chording keyboards have not been widely adopted, they have a long history. Achille Colombo filed a patent for a mechanical chording typewriter in 1942, and many of the first chording keyboards were used in the postal services in the 1950s, with the first formal evaluations occurring in the 1960s (Noyes, 1983). One common design criteria for many of these keyboards was the minimization of hand movement around the keyboard. As such, many designs used one key per finger. Ratz and Ritchie (1961) reported on an evaluation of the response time of novice users for typing the 31 different possible chords on a five-key, one-handed keyboard. Seibel (1962) continued evaluation on a similar keyboard and found significant performance increases with practice. Conrad and Longman (1965) investigated the potential of using a chording keyboard to enter postal codes. They found that their chording participants reached a functional level (being able to enter all of the needed postal codes) after less training than the participants that learned on a traditional typewriter. Furthermore, the participants in the chording group typed faster than did the typewriter group.

Gopher and Raij (1988) evaluated the learning rates and performance of one- and two-handed chording keyboards relative to traditional desktop keyboards. Their participants typed 30 to 35 wpm after 20 hr of practice, and after 60 hr reached nearly 60 wpm. In addition, their participants learned to type more quickly on the chording keyboard. After 35 hr of practice, the two-handed chording group typed nearly twice as fast as the QWERTY group (42 wpm vs. 24 wpm, respectively). It is also interesting to note that the participants using a one-handed version of the chording keyboard performed equally as well as the two-handed group for the first 25 hr of typing practice.

More recently, Matias, MacKenzie, and Buxton (1993, 1996) developed the Matias Half-QWERTY™ keyboard. This keyboard leverages the symmetry of a traditional desktop QWERTY keyboard by mirroring the keys from one half of the keyboard onto the other. Thus, instead of using two hands to type on each half of a keyboard, the user types with one hand only on one half, using a chord to select keys from the alternate half of the keyboard. Upon evaluation, Matias et al. (1996) found their participants reached 50% of their two-handed typing speed after 8 hr of practice and were typing between 23.8 and 42.8 wpm after 10 hr of typing.

Figure 6. **Experimental software showing the keyboard representation, phrase, and statistics.**



3. CHORDING VERSUS MULTITAP

Our longitudinal experiment comparing chording to multitap was a 2 × 20 within-subjects factorial design. We tested two typing conditions, chording and multitap, for 20 typing sessions. This design is similar to previous longitudinal text entry research experiments (MacKenzie et al., 2001; MacKenzie & Zhang, 1999). During each typing session, we presented the participants with our two typing conditions, one at a time. Depending on the condition, the testing software presented the participants with the keyboard layout for either multitap or chording and a phrase to be transcribed by the participant (see Figure 6).

3.1. Participants

We recruited 12 participants from the Georgia Institute of Technology. All participants were informed of the significant time commitment required for the study and were compensated for their participation at the rate of $1 × wpm × accuracy over the entire session, with a minimum of $8 per session.[1]

1. Compensation can be used to motivate participants to perform at their highest level of capability (e.g., Schumacher et al., 1999). Our formula encouraged participants to maximize their speed while minimizing erroneous key presses. We do not believe it encouraged participants to type faster at the expense of accuracy or vice versa. As discussed in Section 4.2, error rates were comparable to previous studies that did not directly compensate for performance.

Two participants dropped out within eight sessions due to time constraints. Of the 10 participants that completed the study, 8 were men and 9 were right-handed. Eight of the participants reported that they owned or used a mobile phone on a regular basis. Most participants reported that they did not send text messages on their phones, whereas a few said they sent a few messages per week. None of the participants had used a Twiddler before this study. We chose only native English speakers as our test phrases were in English. We also recruited participants without long fingernails that might have impeded typing speed.

3.2. Equipment and Software

We used the Twiddler keypad for both conditions. Although the Twiddler is not a typical multitap keypad, this decision ensured that our study was similar to past work using desktop keypads for experimentation (Butts & Cockburn, 2002; MacKenzie et al., 2001). Furthermore, it reduced the possible confound of introducing a second device that might or might not have been optimal for multitap text entry. The faceplates of three Twiddlers were modified with labels for multitap (see Figure 7). Labels were appropriate because multitap is designed to be used while the keypad is facing the user;

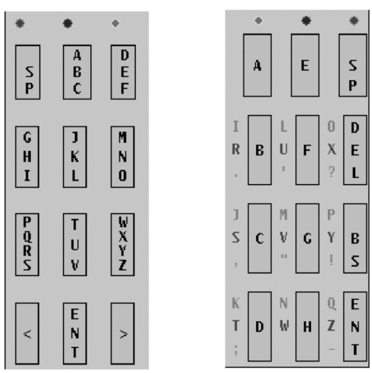
Figure 7. **On the left, typing using multitap on the Twiddler. On the right, chording with the Twiddler.**

however, the Twiddler keypad is designed to face away from the user when the user is chording. To prevent participants from turning the chording keypad to look at the keys, we covered the chording labels on the Twiddler. The labels posed another potential problem due to left and right mappings (as discussed in Section 1.1). The test software displayed key presses to the user as if the Twiddler were held as intended. If the participants turned the keypad around for the chording condition, they would have to mirror the image in their heads.

Depending on the condition, the testing software presented the participants with the key layout for either multitap or chording (see Figure 8) on the left-hand portion of the display (see Figure 6). We included this on-screen key layout as a potential aid for the participants who were novice typists because similar on-screen help is often found in software used to teach desktop typing. In pilots, we found this aid helped novices overcome their reluctance to try the keyboard, and we believe displaying this aid was a reasonable accommodation for mobile phone input given the Twiddler phone design described in Section 12. Each phrase was presented on the same screen along with a transcription of the participants' key presses. The software also displayed performance statistics for both the last phrase typed and the mean for all of the phrases typed up to that point.

The experiment was conducted in our usability laboratory. This was a stationary test for which participants sat at a desktop computer running our test software developed in Java™. The computer stations were Intel Pentium™ III–based PCs. The Twiddler was attached to the computer via a serial cable and continually sent the state of all of its buttons to the computer at 2400 baud, resulting in a key sample rate of approximately 45Hz. The software parsed the serial stream as text input.

Figure 8. **Layout for multitap (left) and chording (right).**



The software collected data at the level of button presses. Every key press and release was recorded to a log file. When a button was pressed or released, the system logged the time-stamp (obtained with Java's System.currentTimeMillis() system call), the character generated (if any), and the state of all of the Twiddler's buttons. The current text entry method condition was logged as well as the phrases presented to the participant. With this data we could determine when each key was pressed and released, the duration each button was held, the time between releasing one button and pressing the next, and the resulting transcribed text. When calculating our statistics, we disregarded the first character entered, thereby ignoring any delay between when the text was presented and when the participant began to type.

3.3. Procedure

The 20 sessions of the experiment were scheduled Monday through Friday over the course of 3 weeks. Each session was separated by at least 2 hr and no more than 2 days and lasted approximately 45 min. The session was split into two 20-min parts based on condition and separated be a 5-min typing break.

At the beginning of the experiment, each participant was given written, verbal, and visual instructions explaining the task and goal of the experiment. The researcher explained how to type for both methods on the Twiddler and demonstrated how to hold the device for each condition. He also explained that the key layout mimics a mobile phone, mapping number keys to Twiddler keys. Finally, he showed the participants how to press each letter of the alphabet for both methods. For multitap, he explained that the keypad is held facing the participants. The participants were informed they could wait for the timeout or utilize the kill button, and they could use one or two index fingers or thumbs to type. For chording, the researcher showed the participants how to strap the Twiddler onto their hand. He also showed how to press each key with the tip of the finger and how to press multiple keys simultaneously to generate chords. At this time, participants were randomly assigned to a condition (balanced across participants) for the 1st session. This condition was tested first, followed by the second condition, and the order of presentation alternated from session to session.

The software was self-administered (under researcher supervision), and participants had unique anonymous log-in IDs. The participants were asked to copy a presented phrase by typing on the Twiddler keyboard and instructed to type as quickly as possible while minimizing errors. The program provided statistics as feedback so the participants could monitor their progress.

Each condition consisted of several blocks of trials, and each block contained 10 text phrases. The phrases for each block were selected according to a uniform random distribution with the additional constraint that no phrase appeared twice in the same block. Phrases could appear multiple times across blocks. The software presented blocks until 20 min had expired. We used the phrase set developed by MacKenzie and Soukoreff (2003), which consists of 500 phrases specifically designed to be a representative sample of the English language. The phrases consist of approximately 28 characters each and contain only letters and spaces. We altered the phrases to use only lowercase and American English spellings.

Each condition began with a warm-up round that consisted of typing the two phrases *abcd efgh ijkl* and *mnop qrst uvwx yz* twice. This data was not used in measuring performance. During this phase, the program also highlighted the correct buttons to press to type the next letter in the phrase. Once the warm-up ended, the highlighting was turned off, but the key layout remained. The participants were then instructed to begin typing, and data recording began. After each block of 10 phrases, the program paused to show the participant's typing rate and accuracy for that block. After 20 min, the program displayed the statistics for that condition and instructed the participant to take a 5-min break. After the break, the participant changed grip on the Twiddler, the program switched to the second input method, and the participant proceeded with the second condition.

Prior to the first session and after the last session, each participant also typed a total of 40 phrases using a standard desktop QWERTY keyboard. We collected this data as a baseline typing rate for each participant, and it was not included in the compensation for those two sessions.

4. RESULTS

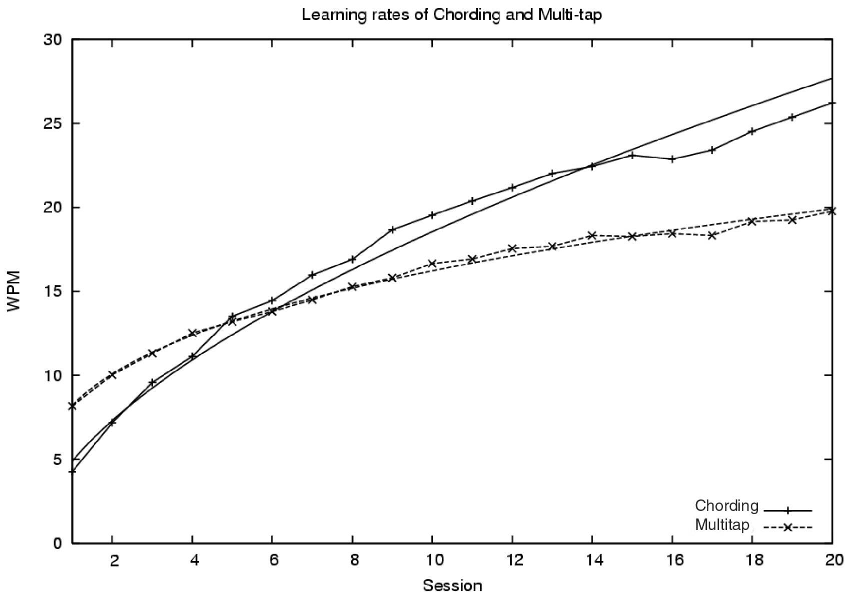
For each of our 10 participants, we collected approximately 2,100 transcribed phrases. In total for both conditions over all 20 sessions and 10 users, we collected 600,000 transcribed characters.

4.1. Text Entry Rates and Learning Curves

The mean entry rates for Session 1 were 8.2 wpm for multitap and 4.3 wpm for chording. By Session 20, the means reached 19.8 wpm for multitap and 26.2 wpm for chording. Although the performance scores for both conditions showed improvement, the scores for the chording condition rapidly surpassed those of multitap (see Figure 9).

An analysis of variance (ANOVA) of text entry speed showed a main effect for typing method, $F(1, 9) = 45.2$, $p < .0001$, and for session, $F(19, 171) = 36.8$, $p < .0001$. There was also a significant Method × Session interaction, $F(19, 171) = 3.6$, $p < .0001$. The main effect of session was expected as was the significant

Figure 9. **Learning rates and exponential regression curves for multitap and chording for twenty 20-min sessions.**



Method × Session interaction. The participants learned to type faster over the course of the 20 sessions. Initially, participants typed faster with multitap, but after a few sessions the difference eroded. By the 8th session, chording became faster, $t(9) = 3.1$, $p < .05$. The magnitude of the differences also increased as the sessions continued.

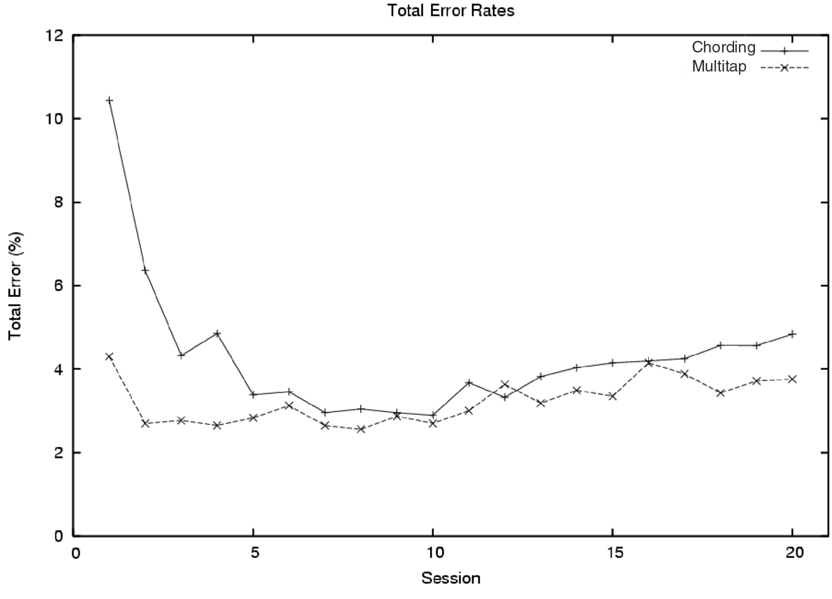
For each typing method, we derived exponential regression curves to model the power law of practice (see Figure 9; see also Card, Moran, & Newell, 1983). The equation for the Twiddler curve was $y = 4.8987x^{0.5781}$; the equation for multitap was $y = 8.2235x^{0.2950}$. The x values were the number of 20-min sessions and the y values were the predicted rate in words per minute for that session. The curves were well fitted to the data, accounting for over 98% of the variance (Twiddler $R^2 = 0.9849$; multitap $R^2 = 0.9961$). As can be seen, multitap rates began to plateau whereas the chording method showed steadily increasing typing speeds. The crossover point in the curves indicates where one condition's typing rate surpassed the other. In our study, the crossover occurred after the 5th session (100 min) of practice. The regressions are interesting because they suggest that the faster typists could reach 60 wpm, the rate of our expert, after approximately 80 sessions (27 hr) whereas the slower typists could achieve 45 wpm.

4.2. Error Rates

We used Soukoreff and Mackenzie's (2003) total error rate metric, which combines corrected and uncorrected errors. For this metric, all of the keystrokes are assigned to one of four categories: correct (C), incorrect and not fixed (INF), incorrect but fixed (IF), and fixed (F). The INF category of keystrokes represent mistakes that appear in the final transcript. The IF category are mistakes that are made but subsequently corrected with the F category of keystrokes. The total error rate then is the number of errors (INF + IF) divided by the number of characters in the text entered (C + INF + IF). This metric accounts for both the errors that are left in the transcript as well as the errors that were made while entering text but were corrected in the final text.

Figure 10 shows the average total error rates per session for both conditions. The error rates in our study were comparable to those of other studies (MacKenzie et al., 2001), and all of the error rates were less than 5% after the second session. The chording method error rates started at 10.4% but quickly decreased. We believe the high initial rate was due to the fact that the participants had no experience with chording on the Twiddler.

Figure 10. **Total error rates for chording and multitap.**



5. TOWARD EXPERTISE

Next, we extended our previous study to confirm the prediction of expert rates from our previous experiment. We continued with a very similar procedure, and 5 of our original 10 participants agreed to participate. The 5 that declined participation did so because of the large additional time commitment. The remaining 5 spanned the range of typing rates with both the slowest and fastest participants remaining in the study. The procedure was modified to focus on chording; we replaced the multitap condition from our original experiment with a second chording session. For this experiment, we compensated each participant at the rate of \$0.33 × wpm × accuracy. The pay rate was reduced because our participants were typing faster than we initially anticipated.

We collected data for approximately 20 additional sessions resulting in a total of 40 sessions (about 13 hr of practice per participant). The average typing rate for our participants increased to 37.3 wpm. Figure 11 shows the typing speeds for each of the participants by session. Also plotted are individual regression curves, which had correlations of at least .96, indicating the data was well fit. They predicted that after 60 sessions, even the slowest participants

Figure 11. **Per-user typing rates and regressions.**

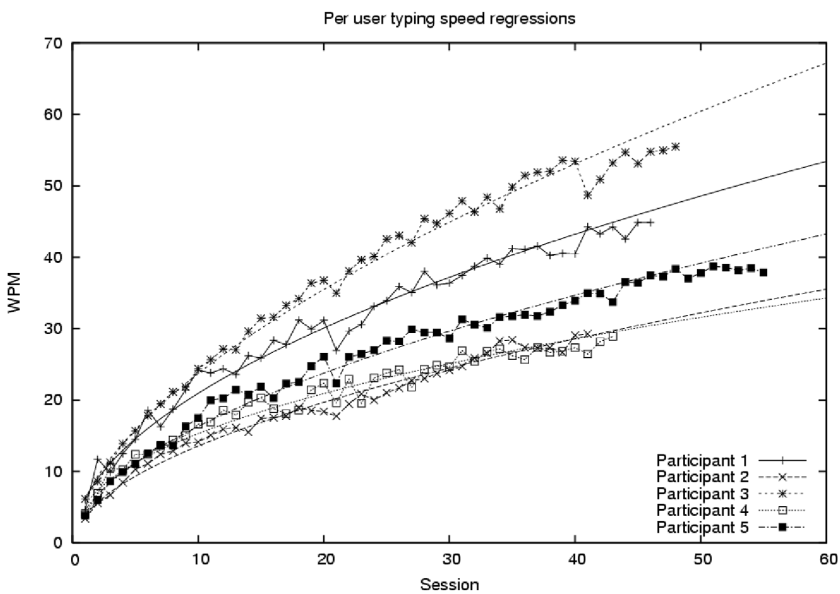
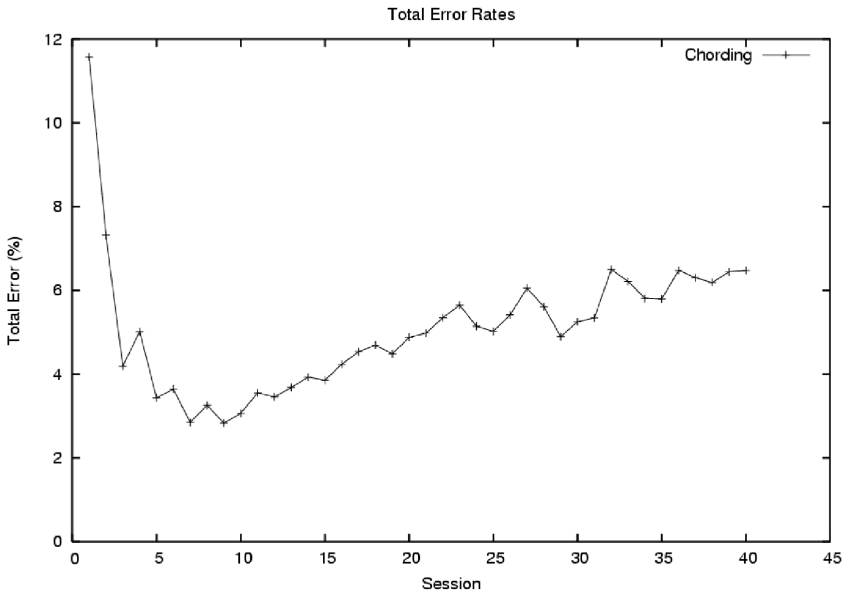


Figure 12. **Mean error rate across participants.**



would be able to type at 35 wpm whereas the fastest would achieve rates in excess of 65 wpm.

Figure 12 shows the average error rate across participants using Soukoreff and Mackenzie's (2003) total error rate metric. The final mean error was 6.2% and was slightly above that of other typing studies with a similar experimental design (Soukoreff & MacKenzie, 2003). As shown, participants rapidly reduced their error rates as they initially learned to chord. As they learned to type faster, their error rate gradually increased. A similar effect, where error rates gradually increased as participants became experts, was shown with the Half-QWERTY keyboard (Matias et al., 1996).

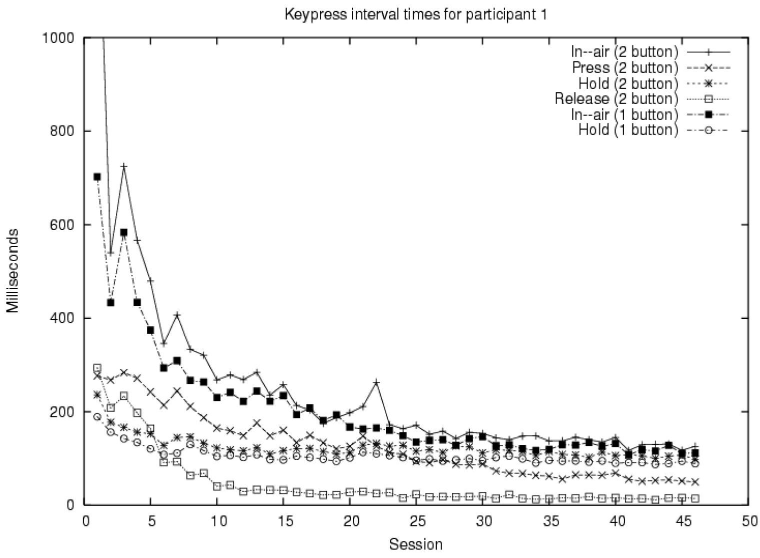
6. ANALYSIS OF LEARNING RATES

In addition to confirming the learning rate for the Twiddler, our additional data allowed us to examine how users type on the Twiddler and to study the nature of the learning involved with chording. With a traditional keyboard, a character is generated by pressing and releasing a single key. Chord typing, however, may involve pressing and releasing two or more buttons to generate a character. We instrumented our experimental software to record the time each button was pressed and released for every chord. By examining the time intervals between each button press and release, we were able to gain insight into how novice users spend most of their time while learning and what optimizations we might make to aid performance.

Typing a degenerate chord involving only a single button has one press and one release. This keypress has two intervals associated with it: in-air and hold. The first interval, in-air, is the time from when the last chord was completed (all of the buttons were released) to when the button for the current chord is depressed, in other words, the time when no keys are held down. The other interval is the hold time and represents the interval between the press of the button and its release. We extended this notion of intervals to two-button chords as well. The interval during which no buttons are pressed down is the in-air time, and the time during which all of the buttons are depressed is the hold time. However, the buttons in the chord may not be pressed or released at exactly the same moment in time. This introduces two additional intervals. The time between the press of the first and second buttons of a chord is the press interval whereas the time between releasing the first and second buttons of a chord is the release interval. Thus, the sequence of two-button chord time intervals is in-air, press, hold, and release, whereas single buttons have only in-air and hold intervals.

Figure 13 shows per-session averages of these intervals for a representative participant. This graph highlights where participants spent their time in chording and suggests where the improvements of learning had the most ef-

Figure 13. **Keypress interval times for a single participant.**



fect. These values were computed by taking the intervals for each chord typed in sentences without any errors and then averaged for the whole session on a per-user basis. We did not include sentences with errors as we did not want to confound our data. Mistyping one chord can impact several others, and it would not have been straightforward to incorporate the error data with our individual time intervals.

6.1. In-Air Interval

All of the participants' average in-air intervals for single- and two-button chords are shown in Figure 14 and Figure 15, respectively. These time intervals exhibited the largest effects of learning. For novices, it is likely that this interval was dominated by the cognitive effort associated with remembering how to type each character and how to move their fingers to the correct position to type the letter. For experts, the delay became dominated by the time it took to move the fingers from one chord to another. Comparing the in-air interval for single- and two-button chords reveals that, on a per-user basis, the single-button times were slightly faster and showed better rates of learning. However, the two-button in-air interval tracked the single-button interval rather well. By the end of the study, the difference between the times on a

Figure 14. **In-air interval times for single-button chords.**

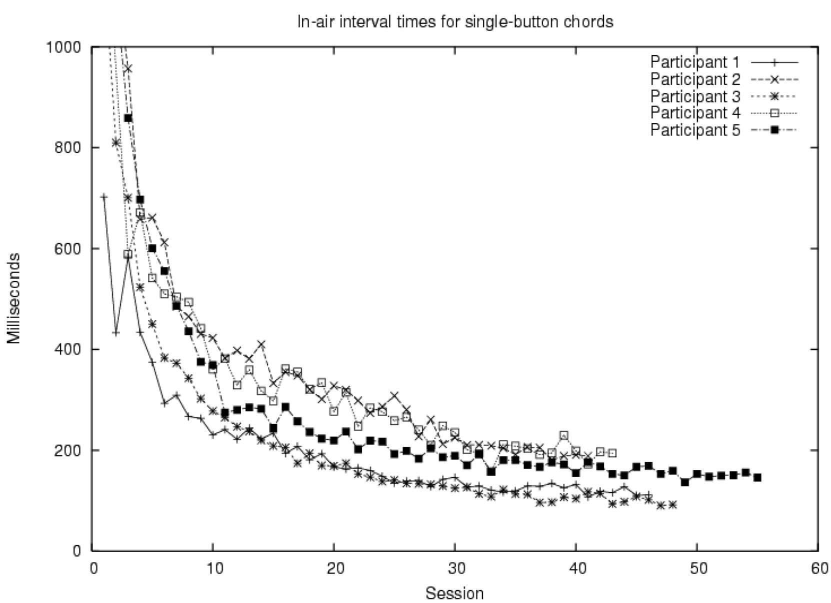
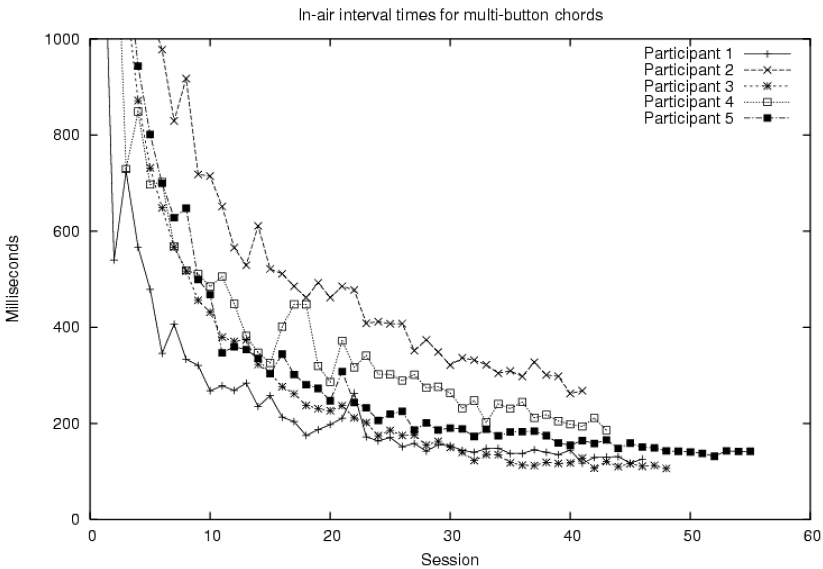


Figure 15. **In-air interval times for two-button chords.**

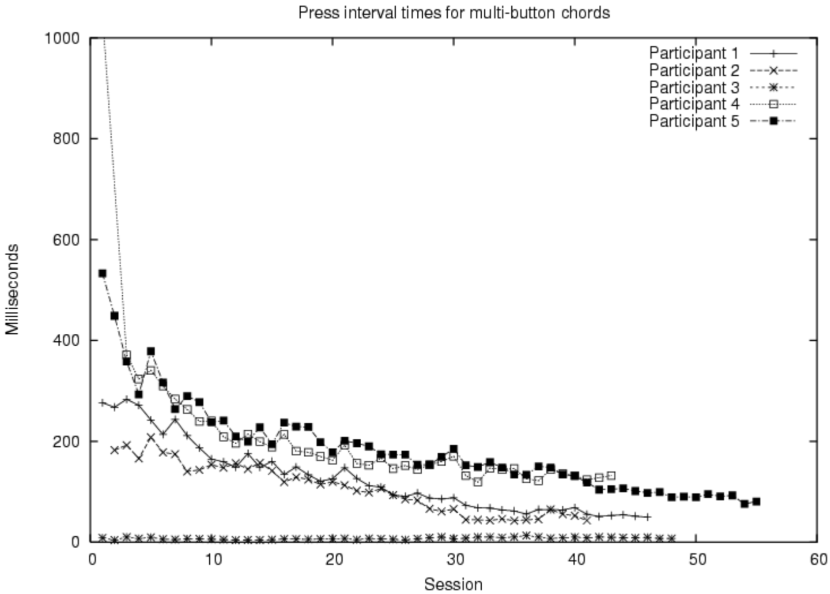


per-user basis became much smaller. On average our participants used 244 ms to type a single-button chord and 354 ms for a two-button chord. The discrepancy was mostly due to one individual (Participant 2 in Figure 15) who lagged behind on learning the two-button chords. With additional practice his rates would approach that of the others, and the difference between the in-air times for single and two-button chords would decrease.

6.2. Press Interval

Figure 16 presents the press interval, which is the time between the first and second buttons of a chord being pressed. This interval was particularly interesting because it revealed different typing strategies between users. As shown in Figure 16, Participant 3 always pushed both of the buttons in a chord at nearly the exact same time. The average delay between the first and second button press was only 7.25 ms, indicating that he always pressed both buttons as one action. The other participants showed a larger delay between these button presses, indicating that they pressed the buttons sequentially and likely learned how to press the chords in a different way than did Participant 3. The delay could have resulted from planning and executing the two button presses in the chord separately. The slower users may also have initially waited for haptic feedback after pressing the first button. For these partici-

Figure 16. **Press interval times (two-button chords).**



pants there was some learning associated with this interval; however, the in-air interval was more pronounced. This interval may also have had implications for expert typing rates. At a typing rate of 60 wpm, the average time to type one character is 200 ms. Because the press interval times varied up to 100 ms by the end of this phase of our experiments and applied to more than 66% of the alphabet, pressing both buttons of a chord at the same time should have significantly increased typing rates.

6.3. Hold and Release Intervals

Our last two time intervals were the release interval (see Figure 17) and the hold interval (see Figures 18 and 19). The average hold interval showed slight improvement with practice, and in general single-button chords were held for slightly less time. At the end of this phase of the experiment, the single-button chords were held 98 ms whereas two-button chords were held 107 ms. Perhaps participants spent the extra time to ensure that they avoided releasing the first finger before the second one was depressed. Finally, although only one participant pressed both keys of a chord simultaneously, all of the participants rapidly learned to release both buttons of a chord at approximately the same time. After about 10 sessions, most of the users released both keys in less than 25 ms.

Figure 17. **Release interval times (two-button chords).**

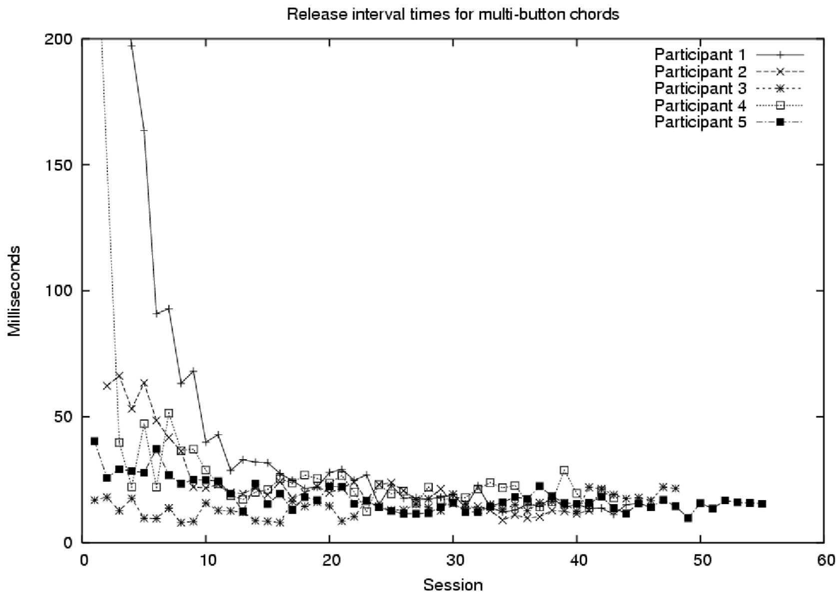


Figure 18. **Hold intervals for single-button chords.**

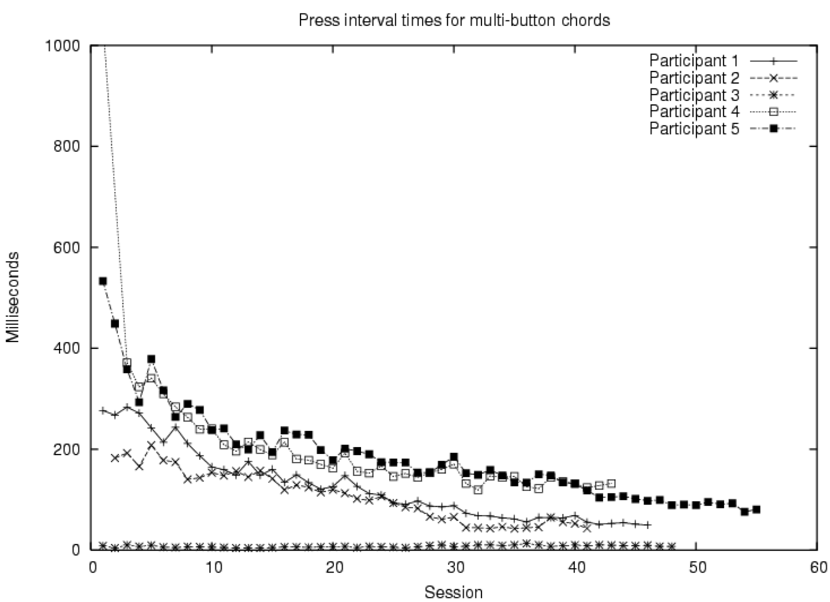
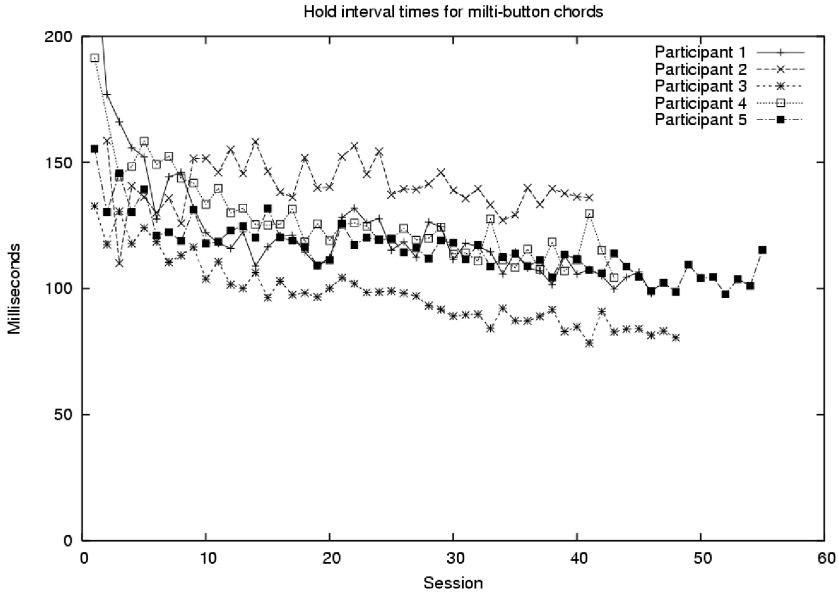


Figure 19. **Hold intervals for two-button chords.**



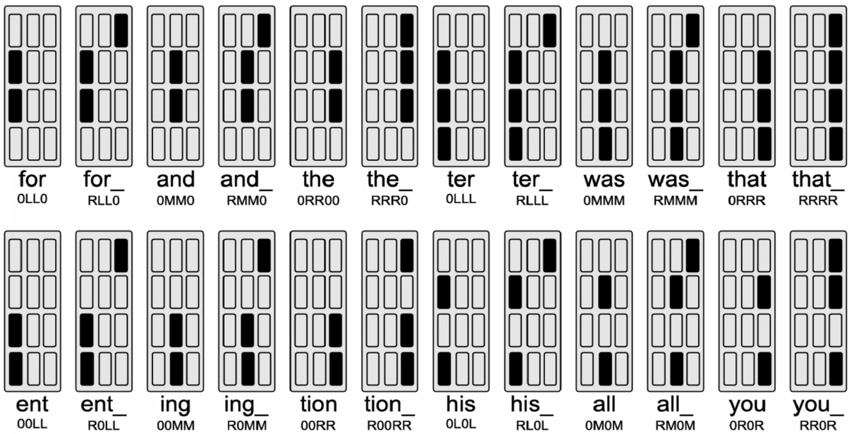
7. EXPERT USAGE

After approximately 45 sessions, we had collected enough data that we could be confident of our regressions' predictions and we considered our participants to be expert typists. At this point we continued our experimentation with two additional small studies designed to investigate various aspects of expert typing. In particular, we examined the possible benefits of MCCs and the effects of typing with reduced visual feedback (blind typing).

7.1. Multicharacter Chords

As mentioned previously, there are 255 possible chords that can be typed on the Twiddler using the four fingers. Of these, only a small subset are allocated to the alphabet and punctuation needed to type English text. Some of the unused chords can be employed as MCCs that can generate any sequence of characters. In the next phase of our experiment we wanted to determine if MCCs for short common words and suffixes would improve our participants' typing rates. Our hypothesis was that MCCs would have a positive impact on typing rate because the number of button presses needed to type any given

Figure 20. **Keymap for multicharacter chords (MCCs) with and without trailing space (represented by "_").**



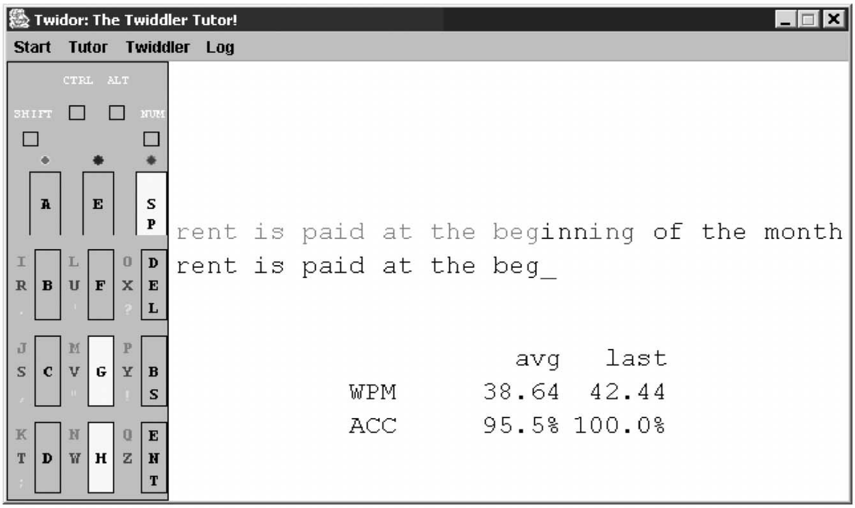
MCC string, such as *the*, would be reduced to one chord. Using a MCC would reduce the overall number of keystrokes per character (KSPC) because fewer keystrokes (button presses) would be needed to generate the same text (MacKenzie, 2002).

Using word frequency data from the commonly used text corpus, the British National Corpus (Leech, Rayson, & Wilson, 2001, pp. xvi & 304), we selected 12 strings of at least three letters that are prevalent in written English. For this experiment we selected *for, and, the, ent, ing, tion, ter, was, that, his, all,* and *you* to be typed as MCCs. We assigned these strings to unused chords that did not involve the index finger. As many of these strings are normally followed by a space character, this assignment enabled us to add 12 extra MCCs that had a trailing space, such as for *the.* The buttons used for these chords were the same as for the normal version, only the user also would depress the button used for Space (the right button operated by the index finger). Figure 20 shows the keymap for the additional MCCs.

Our experimental software has a diagram of the Twiddler keypad that was designed to act as a guide to help the users learn the basic alphabet keymap. We modified the diagram so that the keys needed for the MCC were also highlighted (see Figure 21). To encourage the use of MCCs, we modified the error calculation so that typing the MCC string letter by letter counted against the participant's accuracy.

The effect of MCCs on our participants' typing rates was mixed. Initially, our participants typed more slowly when using MCCs as they were novices for those chords. For the 1st session, the average typing speed dropped to

Figure 21. **Our experimental software showing the use of multicharacter chords (MCCs); *ing* is the MCC to be typed ("R0MM") and is highlighted.**



83.5% of what it had been. However, on the 5th session, the average speed was 97.1% of the pre-MCC speed, and by the 10th session it was 104.5% and continued to improve. Even though the rate increased beyond the typing speed just before the introduction of MCCs, the participants were still slowly learning. If we had not introduced MCCs and just had our participants continue to practice, we would have expected the rate to increase to approximately 112% based upon our regressions. As a result, we cannot attribute the overall increase in typing rate solely to the effects of MCCs.

To better understand the effects of MCCs, we compared the amount of time participants needed to type the MCC strings letter by letter just before the introduction of MCCs and the time needed to type the new chord. On average, participants typed the MCC strings using the MCC in 58.5% of the time it took to type the same characters letter by letter (596 ms vs. 1018 ms).

An analysis of our phrase set revealed that 17.5% of the characters in our phrase set can be typed with our set of MCCs. Weighted by the frequency of MCCs in our phrase set, this would correspond to approximately an 8% increase in average overall typing speed. This effect would likely be more pronounced using a phrase set more representative of English on a word-frequency basis instead of letter frequency (MacKenzie & Soukoreff, 2003) and as our participants mastered the new MCC.

At the end of the MCC phase, our participants required an average of 596 ms to type each MCC and were still showing signs of improvement with

MCCs. Although our MCCs might take longer to type in general because they involve up to four buttons, the chords for the alphabet that require two buttons take only 354 ms on average, which is only 31.3% more time than typing a single-button chord. As a result, we expect MCC rates would have improved once our participants mastered typing the MCC.

7.2. Blind Typing

Our final evaluation with our 5 expert Twiddler typists explored blind typing, the ability to type with limited visual feedback. In a mobile environment, a user's visual attention may be diverted away from his or her display while entering text. For instance, wearable-computer users may use a head-up display to monitor notes being taken during a conversation (Lyons, 2003). However, while doing so, the user will often try and maintain eye contact with his or her conversational partner. This task would likely require many different resources from the user, but one key factor is the availability and effectiveness of visual feedback while typing. Silfverberg (2003) examined the effect of visual and tactile feedback when using a mobile phone keypad. Overall, he found that limited visual feedback increased errors. Furthermore, having a keypad with poor tactile feedback resulted in even higher error rates relative to a keypad with good tactile feedback.

Inspired by Silfverberg (2003), our expert case study (Lyons, 2003), and our own anecdotal experience of typing with limited visual feedback, we designed the last phase of this experiment to evaluate blind typing on the Twiddler. We designed three conditions (normal feedback, dots feedback, and blind) over five sessions of typing. Each condition required 15 min and the order of conditions was randomized across the participants to minimize ordering effects. Our normal feedback condition displayed the text typed under the phrase presented to the participant as shown in Figure 21, but without highlighting of the buttons or letters in the presented phrase. As the Twiddler is held with the keypad facing away from the user, this condition corresponds most closely to Silfverberg's indirect visual feedback condition. For our dots condition, we displayed periods for each character typed instead of the transcribed text. Thus, participants saw their position in the supplied phrase, but not specifically what they had typed. This condition was designed to simulate monitoring text typed without being able to actually read the letters. Specifically, we wanted to simulate seeing the text using only peripheral vision such as when using a heads-up display. Finally, the blind condition did not show any on-screen indication of what had been typed and mimicked Silfverberg's no visual feedback condition. For both the dots and blind conditions, participants were shown their transcribed text and error statistics when they pressed enter at the end of the phrase. We predicted that, as in Silfverberg's study, reducing the visual feedback would limit our participants' typing rate and accuracy.

Figure 22. **Per participant typing and error rates for the three conditions.**

	Participant				
Feedback	**1**	**2**	**3**	**4**	**5**
			Typing rate (wpm)		
Normal	51.8	37.6	64.2	36.2	41.8
Dots	51.7	37.5	67.2*	36.0	43.1*
Blind	53.7*	37.5	67.7*	36.6	41.7
			Percent error		
Normal	5.61	5.62	7.01	9.83	6.64
Dots	4.82*	5.02	5.75*	9.26	5.83
Blind	5.03	4.63	5.90*	8.89	5.44*

Note. wpm = words per minute.
$^*p < .05$

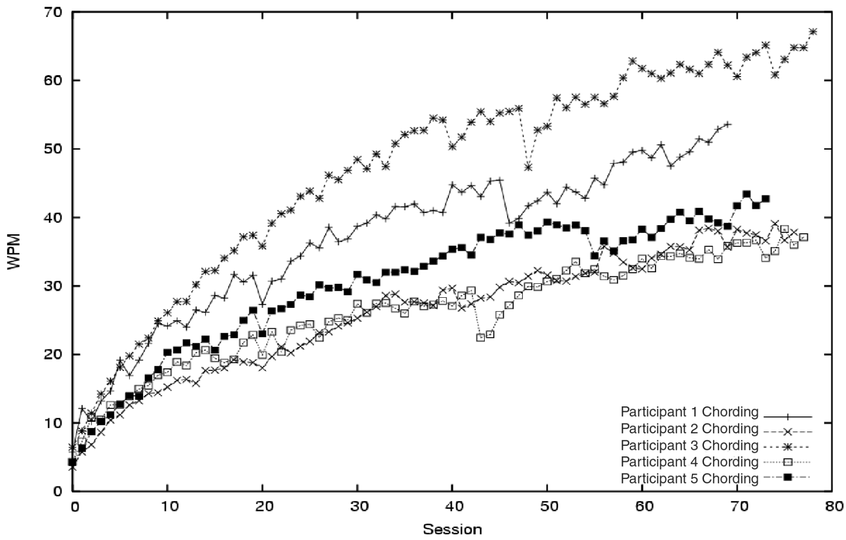
We were surprised to find that changing the visual feedback did not hinder the participants in their typing as expected. In some cases, both typing and error rates improved with the reduced feedback. Figure 22 shows the change in typing speeds and error rates for the three typing conditions. Values where a two-tailed t test showed a statistically significant difference ($p < .05$) from the normal condition are marked. Whenever there was a significant difference between normal typing and one of the reduced feedback conditions, the reduced feedback condition showed an improved typing rate or a reduced error rate. One possible explanation for this trend is that participants were operating with open-loop motor control in the blind conditions. When there was visual feedback, users switched to a closed-loop mode and incorporated the visual feedback into their typing process, thus requiring slightly more time.

7.3. Expert Typing Rates

By the end of all of our experiments, our 5 participants completed an average of 75 sessions each, which corresponds to approximately 25 total hr of practice. Figure 23 shows the typing rates for our participants across all of our experimental conditions, including the initial comparison to multitap, training toward expertise, MCCs, and finally, blind typing. The final average typing rate reached 47 wpm, and unexpectedly, our fastest participant achieved a rate of 67.1 wpm, which is as fast as the second author, a Twiddler user of 10 years.

We have analyzed various aspects of expert chording on the Twiddler keyboard, including text entry speed, the use of MCCs, and the effects of visual feedback. We found that our participants reached an average typing rate of 47 wpm, with our fastest participant reaching 67 wpm. Our data on MCCs indicate that they could provide even higher typing rates. We examined how our

Figure 23. **Data across all phases of experiment for all 5 participants.**



participants learned to chord, showing most of the speed increase associated with learning occurs during the in-air time interval. We also found a difference in strategy of how our participants press the buttons of a chord. The blind-typing data shows that the Twiddler can be used effectively with limited visual feedback, which is important in a mobile environment. Given the expert users' high text entry speeds and ability to touch-type, chording seems to be a viable mechanism for text entry on future mobile devices.

8. AIDING NOVICE TWIDDLER TYPING

Our final work on the Twiddler focused on the novice user. Although the Twiddler shows great potential for permitting rapid text entry in a mobile environment, our studies showed that the initial novice typing rate is about half that of multitap (Section 4.1). Such initial typing rates or even a perception of difficulty may discourage a potential Twiddler user (e.g., while shopping for a new mobile phone or mobile e-mail device). Our goal was to design a tutor that would reduce the perceived difficulty of learning the keyboard. We present a study examining the effects two potential tutor aids had on novice typing rates and user perceptions of the typing tasks. One potential barrier for novice Twiddler users is the difficulty of "hunt-and-peck" typing. The orientation of the hand while typing on the Twiddler is more like that used for playing a musical instrument such as a guitar than that used for typing on a computer keyboard (see Figure 3). To look at which key to press, a user must rotate the Twiddler out of typing position to bring the keypad into view. Another barrier is chording, pressing multiple buttons simultaneously to generate a character. For the Twiddler, the majority of the characters in the alphabet require the use of chording. To address these potential problems, we explored two aids for novice users: a structured phrase set and software highlighting the keys to be pressed.

8.1. Phrase Set

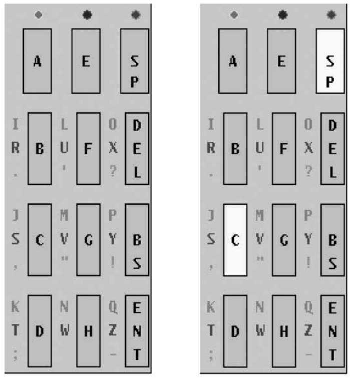
The first aid we developed employs a phrase set tailored to the Twiddler keymap. One common practice with tutors for desktop keyboards is to subdivide the alphabet based on the physical layout of the keyboard. For instance, the software starts by teaching the user the “home row” and gradually adds more letters to be learned based on the position of the keys on the keyboard. We extended this analogy to the Twiddler keymap and different phrases that exercise different categories of chords. Our new phrase set is initially restricted so that the user types only letters requiring a single-button press (a–h). Next the phrase set changes so the participant types just the chords that involve the red Shift (i–q). Then, the phrase set uses the combination of single-button presses and red Shift (a–q), followed by just blue Shift (r–z), single and blue Shift (a–h and r–z), and finally all of the letters.

In addition, simplifying a complex task into smaller tasks can reduce the workload associated with learning the complex task and can reduce error rates (Catrambone & Carroll, 1987; Schneider, 1985). Our new phrase set can be ordered so that the task of learning all 26 letters of the alphabet is simplified into several subtasks. Each task focuses on learning subsets of the alphabet where each subset is associated with a critical gross physical movement. By segmenting the phrase set based on the different types of chords, we can help the user focus on the different types of physical movements needed to type. The phrases that use only a single button let the user explore the keyboard. The “red” and “blue” phrases give practice for the motions needed to type the different chords involving the two Shift keys. Finally, the phrases that use combinations transition the user to more realistic text and the associated movements required.

8.2. Highlighting

The second aid we developed supplements an on-screen keyboard representation that provides the user with the mapping between buttons and characters (see Figure 24). The representation is shown to the user on the left-hand

Figure 24. **Graphical representation of Twiddler chording keymap. Shown without highlighting (left) and with highlighting (right).**



portion of the display (see Figure 6) and is the same as the representation that is printed on the faceplate of the Twiddler.

We provided this on-screen representation in our previous studies so that our participants could use it as a reference while learning to type. Although informative, it is visually busy and requires some experience to understand and use. To facilitate the use of the Twiddler representation, our software can highlight the next set of buttons the user is to press (see Figure 24, right). The highlighting is designed to reduce the amount of time the user spends visually scanning the representation. When highlighting is turned on, the buttons to be pressed for the next character change color. For this study we determined the highlighting based on the position in the text; however, more sophisticated methods that account for errors could also be used.

Next, we present our study designed to explore these two aids. Our goal was to determine if either aid could improve novice Twiddler typing and to see what combinations could lead to the best novice typing rates and lowest workload.

9. COMPARING NOVICE AIDS

Our experiment retained the same core design from our previous Twiddler studies (Lyons, Plaisted, et al., 2004; Lyons, Starner, et al., 2004), which were based on other text entry research (MacKenzie et al., 2001; MacKenzie & Zhang, 1999). For these studies, experimental software presented a sequence of phrases one at a time and the participants were asked to type the displayed text. Phrases were grouped into 20-min typing sessions and the experimental variables could be manipulated per session.

Figure 25. **Example phrases exercising different portions of the Twiddler keymap.**

Characters	Example phrase
Single	dad added a facade
Red	i look ill in pink
Single + red	a feminine chief in old age
Blue	suzy trusts wussy russ
Single + blue	the greatest war there ever was

9.1. Design

The experiment was composed of two 20-min sessions: practice and evaluation. We manipulated our independent variables (i.e., the typing aids) during the practice session only. The typing aids were not used during the evaluation session. The first independent variable was the phrase-set aid. Our Twiddler phrase set had 14 phrases that required only single-button presses, 14 phrases that required only the red Shift, and 14 for the blue Shift. We had 26 phrases that used single and red characters, and 25 that used single plus blue. Figure 25 shows some example phrases from each of our categories. In total, our 93 phrases had an average length of approximately 25 characters and the correlation with the frequency of characters in English was 89% (MacKenzie & Soukoreff, 2003).

Manipulating the phrase set yielded two conditions: ordered and unordered. The ordered condition presented the phrases in a structured order. Initially, our software randomly selected phrases that required single-button presses. Next it used only red phrases, then single plus red, blue, and single plus blue. In contrast, the unordered condition randomly displayed any of the phrases for the whole period. This condition allowed us to control the content of phrase set but did not offer the aid of learning in sequence. The evaluation session used the phrase set developed by MacKenzie and Soukoreff (2003). These phrases averaged approximately 28 characters each and were selected randomly from the set of 500 total phrases. The phrases contained only letters and spaces, and we altered the phrases to use only lowercase and American English spellings. These were phrases specifically designed as representative samples of the English language and had a correlation with English of 95%.

Our other independent variable, highlighting, yielded three conditions: off, on, and delayed. In the highlighting-off condition, the on-screen representation was shown but did not change. In the highlighting-on condition, the buttons for the next character to be typed were highlighted (see Figure 24). In the delayed-highlighting condition, no buttons were initially highlighted; if there was no activity after a short delay, then the keys to press were highlighted. After pilot testing, a 1.5-sec delay was chosen. This value was large enough to allow the pilot participants to type many of the characters they had already learned without the highlight appearing. This value also corresponded to typing at 8 wpm, which, as discussed previously, is the rate at which many novices type with other mobile phone methods. For the practice session, each participant was assigned to one of the three highlighting categories. For the evaluation session, highlighting was turned off for all participants. As a result, our experiment was a 3×2 design. We had three highlighting and two phrase-set possibilities resulting in a total of six between-subject conditions.

9.2. Participants

We recruited 60 students from the Georgia Institute of Technology. The majority participated in return for credit in their respective courses and a few students volunteered. As in our previous experiments, all of our participants had no experience with the Twiddler. Each participant was assigned randomly to one of the six conditions resulting in 10 participants per condition. Our participants ranged in age from 18 to 37 years old (M = 20.9, SD = 3.7). Thirty-one participants were women and four were left-handed. Twelve participants were non-native English speakers. The non-native speakers had been speaking English on average 8.9 years ($SD = 6.4$). Fifty-two of our participants were mobile phone owners. The owners made an average of 6.6 calls per day ($SD = 5.4$) and sent an average of 2.3 text messages each day ($SD = 4.5$).

9.3. Procedure

The experiment took approximately 90 min to complete; it began with a demographic questionnaire and QWERTY typing test (for 3 min). Next, the participants were given written instructions explaining how to hold and type with the Twiddler and how the typing software worked. As appropriate, the instructions explained the breakdown of the phrase set and the highlighting. For each segment of the study, we instructed the participants to type "as quickly and accurately as possible."

The practice session of Twiddler typing started next, beginning with a warm-up round that consisted of typing the two phrases *abcd efgh ijkl* and *mnop qrst uvwx yz* twice. This warm-up data was not used in measuring performance. After the warm-up, the participants began the practice session. At this point the 20-min timer started and data recording began. The practice session was divided into six blocks (see Figure 26). Once the 20-min practice session was over, the participants took a 5-min typing break. During the break, they

Figure 26. **Practice session block durations and character set exercised for ordered and unordered conditions.**

Block	Duration	Ordered Characters	Unordered Characters
p1	4 min	Single	Single + red + blue
p2	4 min	Red	Single + red + blue
p3	2 min	Single + red	Single + red + blue
p4	4 min	Blue	Single + red + blue
p5	2 min	Single + blue	Single + red + blue
p6	4 min	Single + blue + red	Single + red + blue

filled out a NASA Task Load Index (NASA–TLX) questionnaire (Hart & Staveland, 1988). The evaluation session started once the questionnaire was completed and the break was over.

At the beginning of the evaluation session, the participants were instructed that the highlighting would be turned off for the upcoming session (for those who had highlighting in the practice session). At this point, the software switched to using the MacKenzie phrase set for all participants. After typing the alphabet twice, participants resumed the experiment. The evaluation session was divided into four blocks of 5 min. At the end of the 20-min session, participants filled out a second NASA–TLX questionnaire based on the evaluation session only.

9.4. Equipment and Software

As in our previous experiments, the testing software was self-administered under researcher supervision. It presented the participants with the key layout for chording (see Figure 24) and statistics of performance so participants could monitor their progress. A phrase was displayed on the screen, and the participant's typed text appeared immediately below the presented text (see Figure 6). The software was modified to include a built-in scripting engine used to configure and control the experimental conditions. Six scripts (one for each of our conditions) were used by the software to run the participants through our procedure.

10. RESULTS

Across our 60 participants we collected approximately 3,500 phrases of chording data, which resulted in 84,000 transcribed characters. Using this data, we examined the effects of our experimental manipulations on participants' typing speed, error rate, and workload. We collapsed the 10 experimental blocks into a session variable with two levels: practice and evaluation. This session variable allowed us to compare the mean typing speed and error rates for the six practice blocks to the mean typing speed and error rates for the four evaluation blocks. We performed a $3 \times 2 \times 2$ (Highlighting × Phrase Set × Session) ANOVA on each measure (typing speed, error rate, and workload). Where appropriate, we also examined the individual two-way interactions and simple effects of each manipulation. Finally, we explored each condition's typing speed and error rate trends across all sessions. All results are interpreted using $\alpha = 0.05$.

10.1. Text Entry Rates

First, we examined the effect our conditions had on typing speed, measured in words per minute. For each participant, we calculated the cumulative words per minute value across an entire session by taking the sum of the total number of words and dividing by the total time spent typing in the phrases for the session. Figure 27 displays each group's mean words per minute and standard deviation for the practice and evaluation sessions.

The change in typing rates from the practice to the evaluation session depended on the highlighting condition, $F(2, 54) = 8.43$, $p = .001$. A simple effects analysis demonstrated that the highlighting-off group typed slower during the practice session than the evaluation session, $F(1, 54) = 9.32$, $p < .01$. In contrast, the highlighting-on group typed faster during the practice session than during the evaluation session, $F(1, 54) = 7.02$, $p = .01$. The delay group exhibited no reliable difference in typing rate between the practice and evaluation sessions, $F(1, 54) = 1.33$, $p = .25$. A simple effects analysis of highlighting for each session did not reveal a significant difference in mean words per minute between the three highlighting groups during the practice session, $F(2, 54) = 0.46$, $p = .64$, or during the evaluation session, $F(2, 54) = 2.53$, $p = .09$.

There was also a significant interaction between phrase set and session, $F(1, 54) = 4.26$, $p = .04$. Simple effects analysis of phrase set in the practice ses-

Figure 27. **Mean typing rates in words per minute (with standard deviations) for the practice and evaluations sessions for all six groups.**

	Highlighting			
Phrase Set	**Off**	**Delay**	**On**	***M* with *M*(*SD*)**
		Practice session		
Ordered	6.61 (2.58)	6.73 (1.20)	6.21 (1.16)	6.52 (1.72)
Unordered	5.17 (1.74)	4.88 (1.15)	6.34 (1.69)	5.46 (1.63)
M(*SD*)	5.89 (2.26)	5.81 (1.49)	6.28 (1.41)	5.99 (1.75)
		Evaluation session		
Ordered	6.92 (2.15)	6.61 (1.50)	5.42 (1.80)	6.32 (1.89)
Unordered	6.69 (1.87)	5.69 (1.48)	5.55 (2.20)	5.98 (1.88)
M(*SD*)	6.80 (1.96)	6.15 (1.53)	5.48 (1.96)	6.15 (1.88)

sion revealed that the ordered phrase-set group typed faster than the unordered phrase-set group, $F(1, 54) = 6.01$, $p = .02$. In contrast, during the evaluation session there was no significant difference between phrase-set groups, $F(1, 54) = 0.50$, $p = .48$. These results indicate that the ordered group typed faster than the unordered group in the practice session, but not in the evaluation session.

There was not a significant interaction between highlighting and phrase set, $F(2, 54) = 1.11$, $p = .34$, nor was there a three-way interaction between highlighting, phrase set, and session, $F(2, 54) = 1.12$, $p = .34$.

In addition to analyzing the average typing rates for the practice and evaluation sessions, we also examined the typing speed trends that occurred across the learning trials. Figures 28 and 29 show the typing rate across the 10 blocks, for both the unordered and ordered groups. Recall from our procedure (Section 9.3) that for both groups the first six blocks (practice; p1–p6) were unequal intervals. Furthermore, the ordered group's phrases changed between the practice blocks (see Figure 26). For all groups the evaluation sessions (e1–e4) lasted for 5 min and drew from all letters in the alphabet.

All groups showed a drop in typing rates from the last practice block to the first evaluation block. Presumably this was due to the removal of the typing aids, switching to the MacKenzie phrase set, and the 5-min break between the

Figure 28. **Typing rates across practice and evaluation blocks (unordered condition).**

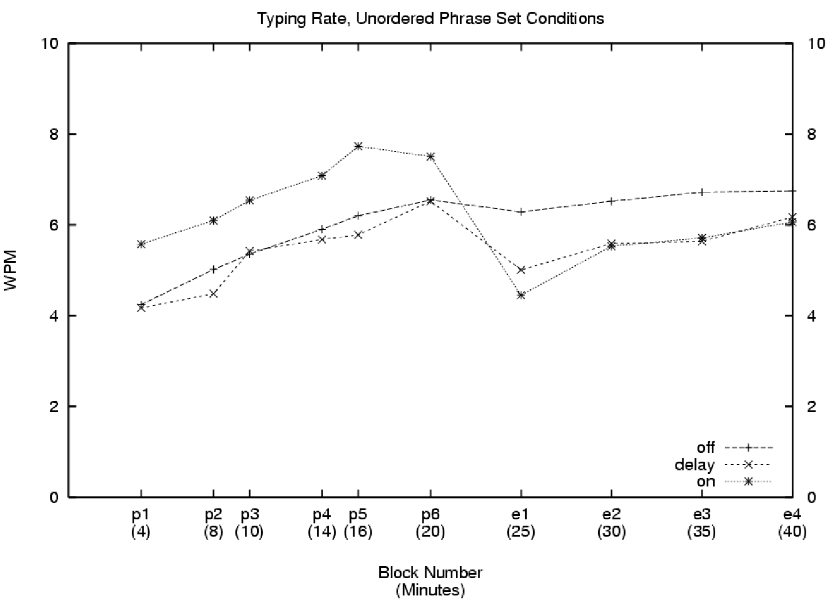
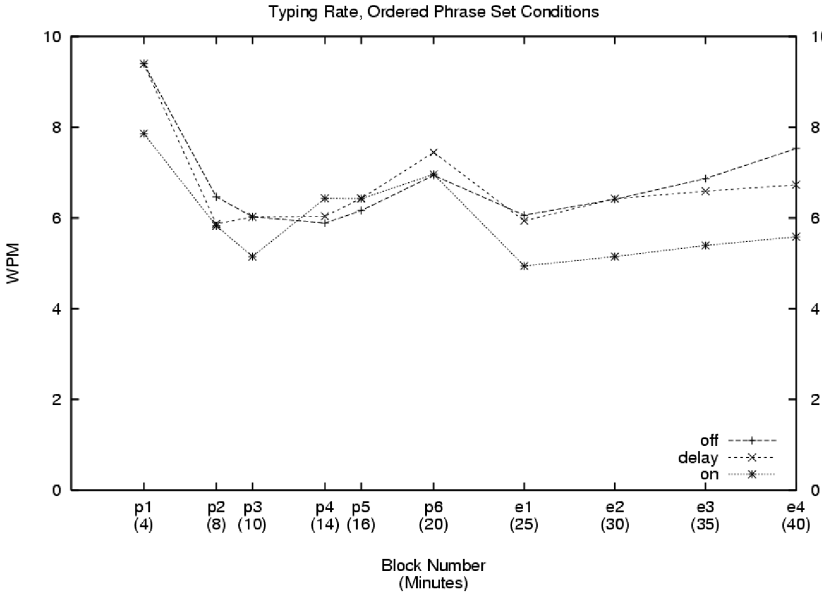


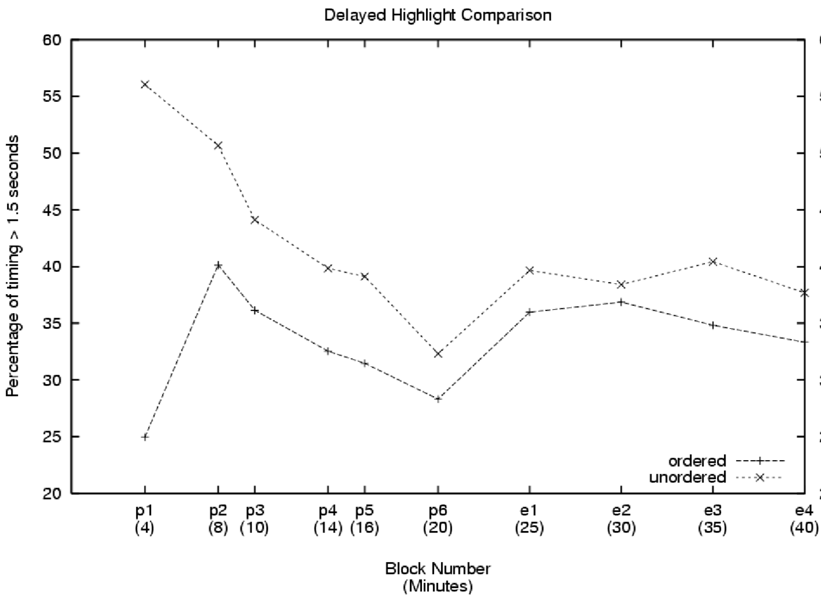
Figure 29. **Typing rates across practice and evaluation blocks (ordered condition).**



sessions. Although all groups' typing rate dropped, the unordered, highlighting-off group did not show as dramatic a decrease. This group did not have any aids to help them learn to use the Twiddler during the practice phase; not surprisingly, this group followed a power learning curve. The drop in typing rates from p6 to e1 was sharpest for the highlighting-on groups (both ordered and unordered). Furthermore, these groups consistently typed slower after highlighting was removed (e1–e4). This result suggests that participants were relying heavily on the highlighting to type during the practice session. It seems the highlighting-off group's effort to find and remember keys and key combinations improved their learning.

The trends for the delayed-highlighting groups illustrate an interesting performance finding. The typing rates suggest that the delayed-highlighting aid interacted with the phrase-set manipulation. For the unordered group, the removal of highlighting lowered the typing rate of both the highlighting-on and the delayed-highlighting group. In contrast, in the ordered group, removing highlighting during the evaluation session lowered the typing rate of the highlighting-on but not the delay group. We believe that the phrase-set manipulation changed the utility of the delayed-highlighting manipulation because the ordered group had a reduced number of letters to type in any given session (except p6). The delayed-highlighting groups had keys highlighted after 1.5

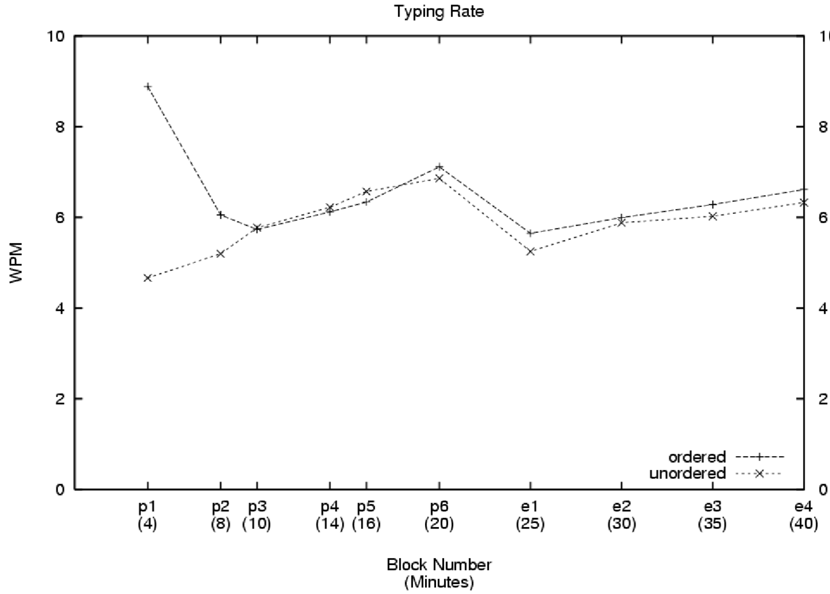
Figure 30. **Proportion of characters in which more than 1.5 sec elapsed without generating a character (delayed conditions).**



sec. Because the ordered group had less characters to search for and fewer key mappings to remember, the ordered group likely typed a greater proportion of characters before 1.5 sec elapsed. To explore this hypothesis, we examined the frequency that participants in the ordered and unordered group had more than 1.5 sec elapse before typing a character (thus the delay threshold would have been exceeded, and the keys highlighted). As predicted, participants in the unordered group had a higher percentage of keys highlighted (see Figure 30). This finding suggests that the unordered group was relying on delayed highlighting more than the ordered group was. The greater reliance on highlighting explains why the shift from practice to evaluation caused corresponding drops in typing rates for both the highlighting on and delayed highlighting for the unordered group but not the ordered group.

The trend data also illustrates why the ordered group typed faster than the unordered group during the practice phase (see Figure 31). The first two practice blocks had the largest difference in typing rates, indicating that typing with only the single keys (p1) and only the red Shift keys (p2) was faster than typing phrases drawn from a pool of all the letters (i.e., unordered group). This finding suggests an alternative approach of introducing the full set to a

Figure 31. **Typing rates across practice and evaluation blocks (collapsed over highlighting conditions).**



user might be useful. Instead of shifting between the different sets of chords, users might benefit from a more gradual introduction to the chords (one at a time) so that their overall typing rate does not suffer.

10.2. Error Rates

Next, we examined the number of errors made. Figure 32 shows the percent error means and standard deviations for each group. We used Soukoreff and Mackenzie's (2003) total error rate metric. This metric accounts for both corrected and uncorrected errors made by the participants and provides a single total error rate.

There was not a significant main effect of phrase set on error rates, $F(1, 54) = 1.32, p = .26$. In addition, there was not a significant interaction between phrase set and session, $F(1, 54) = 1.32, p = .26$, between phrase set and highlighting, $F(2, 54) = 0.74, p = .48$, or between phrase set, highlighting, and session, $F(2, 54) = 0.74, p = .48$. There was a significant interaction between highlighting and session, $F(2, 54) = 4.59, p = .01$. A simple effects analysis of highlighting within each session revealed that highlighting had a significant effect on error rates in the

Figure 32. **Mean percent error (with standard deviations) for the practice and evaluation sessions per group.**

	Highlighting			
Phrase Set	**Off**	**Delay**	**On**	***M* with *M*(*SD*)**
		Practice session		
Ordered	19.9 (11.7)	14.5 (4.8)	12.1 (5.5)	15.5 (8.4)
Unordered	13.6 (6.3)	15.7 (6.7)	10.0 (4.4)	13.1 (6.1)
M(*SD*)	16.8 (9.7)	15.1 (5.7)	11.1 (4.9)	14.3 (7.4)
		Evaluation session		
Ordered	15.2 (10.0)	13.0 (8.4)	15.5 (7.5)	14.6 (8.5)
Unordered	13.0 (6.9)	13.1 (3.4)	13.6 (7.9)	13.2 (6.1)
M(*SD*)	14.1 (8.4)	13.0 (6.3)	14.5 (7.5)	13.9 (7.4)

practice session, $F(2, 54) = 3.50$, $p = .04$, but not in the evaluation session, $F(2, 54) = 0.21$, $p = .82$. A post hoc contrast revealed that in the practice session the highlighting-on group made fewer errors than the other two highlighting groups, $t(57) = 2.50$, $p = .02$.

Next, we examined how the highlighting manipulations impacted error rates as participants moved from the practice to evaluation sessions. The highlighting-on groups' error rates increased between the practice and evaluation sessions, $F(1, 54) = 4.85$, $p = .03$. There was no significant difference in error rates between the practice and evaluation sessions for either the highlighting-off group, $F(1, 54) = 2.88$, $p = .10$, or the delayed-highlighting group, $F(1, 54) = 1.68$, $p = .20$. This result suggests that error rates, which were significantly lower for the group with highlighting on during the practice session, increased to the level of the other highlighting groups during the evaluation session.

As we did for typing speed, we examined the error rate trends. Because there was no interaction between phrase set and highlighting, we averaged across phrase sets to look at the impact of the highlighting manipulation on error rates. The error rate trends demonstrated the same findings as the error rate ANOVAs. The highlighting-on group consistently had error rates lower than those for the other two groups during the majority of the practice session. Once the evaluation session started, the absence of highlighting drove up the error rates for the highlighting-on group. Over the four evaluation blocks, the highlighting-on groups' error rates followed the rates of the other two highlighting groups. Again, this supports the earlier claim that highlighting was effective in reducing error rates in the practice session. The removal of highlighting in the evaluation phase did not increase error rates over and above the delayed-highlighting or highlighting-off groups.

10.3. Workload

The NASA–TLX questionnaire measures subjective workload ratings. Previous studies have indicated that it is a reliable and valid measure of the workload imposed by a task (Hart & Staveland, 1988; Hill, et al., 1992). Subjective workload ratings can be more sensitive to working memory demands than are measures of performance; this is important given the need for the participants to remember the Twiddler key mapping. In addition, subjective ratings can be informative when a task is difficult yet within the individual's capability. For instance, as a task becomes more difficult, the individual can increase his or her effort to maintain the same level of performance. In this case, subjective ratings of workload could capture this increased effort, whereas performance measures could not (Yeh & Wickens, 1988).

The NASA–TLX consists of six scales: mental demand, physical demand, temporal demand, performance, effort, and frustration; each scale has 21 gradations. For each scale, individuals rate the demands imposed by the task. In addition, they rank each scale's contribution to the total workload by completing 15 pairwise comparisons between each combination of scales. This procedure allows an investigation of the task-demands load on each scale, as well as a measure of the global workload.

Interpretation of the mental, physical, and temporal demand scales are straightforward; each scale captures the demand imposed by its title. The performance scale captures how successful participants felt they were at accomplishing the given task. The effort scale captures how hard individuals had to work to achieve their level of performance; both mental and physical effort can contribute to this scale. The frustration scale captures how much the task annoyed or discouraged the participants (Hart & Staveland, 1988).

The overall workload rating is calculated by summing the product of each scale's rating and weight. This calculation results in a score between 0 and 100. It reflects an individual's perception of the amount of workload devoted to each of the scales, along with each scale's contribution to overall workload (Hart & Staveland, 1988). For our study, we analyzed the overall workload ratings in addition to the six individual scale ratings. As with typing and error rates, we used a $3 \times 2 \times 2$ (Highlighting × Phrase Set × Session) ANOVA for each analysis.

Overall Workload

An analysis on the overall workload did not reveal any interesting effects. There was no significant main effect for highlighting, phrase set, or session. In addition, there was no significant interaction between highlighting and phrase set, highlighting and session, and phrase set and session. Finally, there was no three-way interaction between highlighting, phrase set, and session. Although the overall workload score revealed no effects, we felt that an analysis of individual workload scales could still reveal relevant information about how the typing task contributed to different sources of workload (Hart & Staveland, 1988). For each scale, we analyzed the rating (0–20) without regard to the participant's weighting of that scale. On each scale a higher rating reflects more workload or difficulty.

Physical Demand

There was no significant three-way interaction between highlighting, phrase set, and session. Moreover, there was no significant interaction between highlighting and phrase set or highlighting and session. Finally, there was no significant main effect for highlighting. However, there was a significant interaction between phrase set and session, $F(1, 54) = 13.72$, $p < .01$. The ordered group rated physical demand lower in the practice session ($M = 8.42$, $SD = 5.13$) than the evaluation session ($M = 11.27$, $SD = 5.16$), $F(1, 54) = 13.88$, $p < .01$. The unordered group did not rate physical demand differently between the practice and evaluation session. Simple effects were further examined by analyzing the impact of phrase set in the practice session and the evaluation session. In the practice session, the ordered group rated physical demand significantly lower ($M = 8.42$, $SD = 5.13$) than did the unordered group ($M = 12.63$, $SD = 5.22$), $F(1, 54) = 7.56$, $p = .01$. However, in the evaluation session no significant difference in ratings was found between the two phrase-set groups. This suggests that the increase in physical demand between sessions for the ordered group was a result of demand being lowered in the practice session; in the evaluation session the physical demand was not different for either group.

Effort

For the effort scale, there was no significant three-way interaction between highlighting, phrase set, and session. Also, there was no significant interaction between highlighting and phrase set or between phrase set and session. Furthermore, there was no significant main effect of phrase set, indicating that the phrase-set manipulation did not change participants' rating of the effort required to type on the Twiddler. The highlighting manipulation did interact with session, $F(2, 54) = 8.48$, $p = .001$. A simple effects analysis of session at each level of highlighting revealed that the highlighting-off group did not report significantly different amounts of effort between the practice and evaluation sessions. However, the highlighting-on group rated the effort required to type in the practice session lower ($M = 13.38$, $SD = 4.11$) than the effort required in the evaluation session ($M = 14.93$, $SD = 3.61$), $F(1, 54) = 5.64$, $p =$

.02. In contrast, the delayed-highlighting group reported higher effort in the practice session ($M = 13.80$, $SD = 3.30$) compared to the evaluation session ($M = 12.70$, $SD = 3.93$), $F(1, 54) = 11.16$, $p < .01$. Further, simple effects analyses revealed that the three highlighting groups were not significantly different in either the practice or evaluation session.

Mental and Temporal Demand, Performance, and Frustration

The software manipulations did not have any significant effects on mental demand ratings or performance ratings. There was only one significant difference for ratings on the temporal demand scale: a main effect for session. This result indicates that participants rated the evaluation session as more temporally demanding ($M = 10.24$, $SD = 4.19$) than the practice session ($M = 8.28$, $SD = 4.09$), $F(1, 54) = 12.79$, $p < .01$. Ratings of the frustration scale also yielded no significant effects for highlighting, phrase set, or session. It is interesting that there were no effects for session (either a main effect or an interaction with phrase set or highlighting). This result seems to suggest that when the help that was provided in the practice session (such as highlighting on or ordered phrase set) was removed, participants did not feel more discouraged or stressed in the evaluation session.

10.4. Comparison to Original Results

We used data from our first study on Twiddler typing rates (Section 4.1) as a baseline against which to compare our current typing rates. Although many differences existed between the two studies that could account for differences in typing rates (e.g., compensation, instructions, error highlighting, phrase set, etc.), we believed the comparison could still be illuminating. To compare the two studies, we utilized a 2×2 (Session $\times$ Study) ANOVA. The study factor had two levels: previous and current, which corresponded to the original study and the current study. This analysis combined the current study's experimental conditions into one group. There was a significant interaction between the session and study factors, $F(1, 68) = 27.51$, $p < .01$. A simple effects analysis showed that within the practice session, the current study yielded faster typing rates ($M = 5.99$, $SD = 1.75$) than did the previous study ($M = 4.27$, $SD = 1.35$), $F(1, 68) = 8.84$, $p < .01$. However, within the evaluation session there was no significant difference in typing rates between the current study ($M = 6.15$, $SD = 1.89$) and the previous study ($M = 7.18$, $SD = 2.08$), $F(1, 68) = 2.54$, $p = .12$. In the previous study, typing rates increased significantly from the practice session to the evaluation session, $F(1, 68) = 35.81$, $p < .01$. However, in the current study, typing rates did not significantly change between the two sessions, $F(1, 68) = 0.61$, $p = .44$. Together, these results suggest that the current study raised typing rates in the first 20 min.

To investigate the possibility that our Twiddler phrase set (as opposed to the MacKenzie phrase set) was responsible for the difference in typing rates for the first condition, we compared our baseline condition (highlighting off and unordered phrase set) to the previous study's data. If there was a difference between baseline conditions, we could attribute the change to any of the several differences between the two studies, including the phrase set. We used the same 2×2 (Session × Study) ANOVA analysis strategy but limited our data set to the baseline condition in our current study and the previous data's study. As before, we found a significant interaction between study and session, $F(1, 18) = 5.32$, $p = .03$. A simple effects analysis of study at each level of session showed the practice condition did not have a statistically significant difference between the old study ($M = 4.27$, $SD = 1.35$) and the new study ($M = 4.17$, $SD = 1.74$), $F(1, 18) = 1.69$, $p = .21$. Likewise, in the evaluation condition there was no reliable difference between the old study ($M = 7.18$, $SD = 2.08$) and the new study ($M = 6.69$, $SD = 1.87$), $F(1, 18) = 0.31$, $p = .59$. This result suggests that the phrase set by itself was not enough to alter typing rates across studies.

11. DISCUSSION

Taken together, the data on our two aids on novice typing are encouraging. In general, using the ordered phrase set and highlighting helped novice Twiddler typists' performance. The ordered phrase set increased typing rates and lowered the subjective physical demand during the practice session. Although this effect did not persist in the evaluation session, increasing performance while using the aid may help adoption of the Twiddler. Simply presenting the keys to be learned sequentially, in groupings that correspond to the keyboard layout, allows individuals with no experience to type meaningful phrases faster and with less effort. This result is consistent with existing research that has found training beginners on parts of a task, rather than the whole task, is beneficial (Catrambone & Carroll, 1987; Schneider, 1985). With our second aid, we found that using highlighting for the first 20 min of typing increased typing rates, reduced the number of errors, and reduced subjective ratings of effort. However, we believe that the results indicate that this highlighting may have a slight cost. Error rates increased and typing rates decreased once highlighting was turned off. Although error rates increased, the group with highlighting made no more errors than did the groups without; although typing rates decreased, they were not slower than those of the other groups. Using highlighting with a delay did not have an overall positive effect on typing or error rates. It might be that we did not have the correct timing delay to show any meaningful benefit. The failure to find any significant benefits should not rule out future investigations into the utility of delaying highlighting for novices. Comparing the data from this study to the first two sessions of our previous Twiddler evaluation shows that our aids are beneficial for the first 20 min of typing and do not hinder the second 20 min once removed.

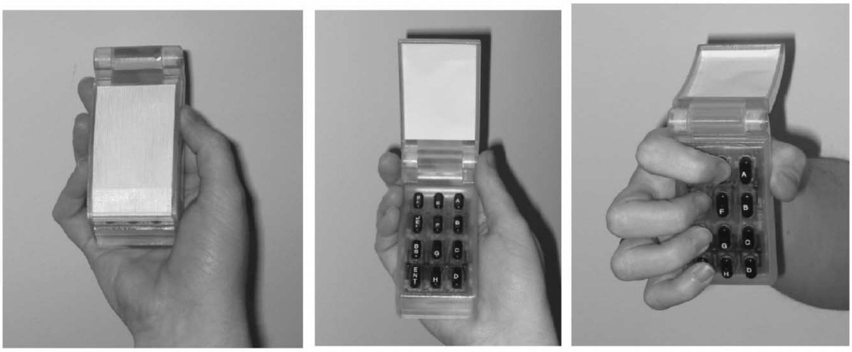
More generally, our first set of studies examining the learning rate of chording on the Twiddler and our relative comparison to multitap indicate that this technique might be useful if incorporated into mobile phones. Previous work on desktop keyboards has shown that chording is learned faster and outperforms traditional desktop typing (Conrad & Longman, 1965; Gopher & Raij, 1988). This trend seems to hold for mobile devices as well; our data show that our participants surpassed their multitap typing rates after approximately 100 min of practice. After 25 hr of training, our participants reached an average typing rate of 47 wpm and our fastest participant reached 67 wpm. These rates are much faster than other mobile keypad entry rates, which seem to peak at approximately 20 wpm (James & Reischel, 2001; MacKenzie et al., 2001).

The Twiddler also has some other advantages. First, our data indicates that MCCs can provide even higher typing rates for experts by allowing users to enter more characters with fewer key presses. The Twiddler also performed surprisingly well in our blind-typing conditions relative to results found for typing with limited visual feedback on mobile phones or mini-QWERTY keyboards (Clawson, Lyons, Starner, & Clarkson, 2005; Silfverberg, 2003). Most of our expert Twiddler participants were able to type faster or more accurately with limited visual feedback. This effect could be very valuable in a mobile environment.

12. FUTURE WORK

In the future, we are interested in exploring more familiar designs that incorporate similar chording capabilities as the Twiddler that might enable a more widespread adoption. Although we have shown in this work that the Twiddler offers very rapid expert text entry rates, the Twiddler has obtained limited commercial success. There are several potential reasons for this. First, Handykey is a small company with limited recourses. In over a decade they have sold only two models of the Twiddler. The original model used in this work has a serial interface, whereas the Twiddler2 requires either a desktop computer PS/2 port or a USB master. None of these ports are commonly available on mobile devices such as PDAs or mobile phones. One possible way to allow more people to benefit from this technology would be to build a device similar to the Twiddler with better support for mobile computing devices. To that end, we have begun to explore a mobile phone based on the current Twiddler keyboard (see Figure 33). As can be seen in the leftmost and center images, our prototype Twiddler phone resembles current “flip” phones (e.g., the Panasonic model X700). A high-resolution screen could be placed in

Figure 33. **A mobile phone design that incorporates chording capabilities.**



the top half of the phone (in the center image) while the bottom half of the phone could be dedicated to the keyboard. This keyboard placement would allow the same key spacing as on the Twiddler while retaining the same or smaller form factor as current phones. When the screen is flipped over the back of the hand as in the rightmost image, software in the phone would vertically flip the display so that the user could see the screen while typing in a manner similar to that used for the Twiddler. Given the typing speeds allowed by the Twiddler, such a device might enable advanced mobile phone features such as e-mail and other applications currently reserved for the desktop.

We would like to create a model of Twiddler chording that accounts for finger motion and effects between chords. Our analysis of learning rates from Section 6 is a first step. This model would enable us to evaluate different keymaps and optimize them for various tasks such as maximizing expert performance or easing learning. Hopefully the model would also provide more insight into why the Twiddler works so well. Finally, our work on text entry has also uncovered an issue with the current methods used to study text entry. In our study, we compensated our participants to motivate performance, maximizing both typing rate and accuracy. We chose to use the product of these two measures; however, this is an arbitrary combination. It is plausible that allowing more errors enables the participant to type faster, but we do not know by how much. Does a 1% increase in errors enable a participant to type 1 wpm faster or 5? In the future, we would like to quantify the relative trade-offs between text entry rates and accuracy. How much can participants increase their accuracy if they slow down by a given amount; or conversely, how much faster can they type if they sacrifice some accuracy? Although this effect is likely to be input-device specific, it would be useful to understand this trade-off for these types of evaluations. Closely related is the question of how much uncorrected error to allow in these studies. Some work requires error-free text; however, this is often achieved through editing after the text has initially been entered. Requiring the participant to enter 100% correct text during a study would thus minimize external validity. However, if the researcher is to allow errors, how many should he or she permit? The standard practice is to inform the participants to enter text "as quickly and accurately as possible," but the participants determine exactly what that means for themselves. In the future, we would like to examine how much error can be introduced before the meaning of an entered phrase becomes obscured. In particular, we want to use our corpus of phrases we collected from our text entry experiments to determine if a different set of participants can recreate the original phrase when presented with a phrase with a known error rate.

13. CONCLUSION

In this article, we presented a longitudinal study comparing multitap and chording methods on a Twiddler, a mobile one-handed keyboard with a keypad layout similar to that of a mobile phone. Chording outperforms multitap typing speeds, is learned quickly, and appears to have a higher attainable maximum rate. In addition, the chording rates reported here are more than two times faster than those reported in studies on T9 and LetterWise for similar levels of expertise. With the numerous wireless messages sent currently and the predicted increase in wireless e-mail usage, the Twiddler's one-handed chording text entry method should be seriously considered for future mobile phone designs. We have analyzed various aspects of expert chording on the Twiddler keyboard including text entry speed, the effects of visual feedback, and the use of MCCs. We found that our participants reached an average typing rate of 47 wpm and our fastest participant reached 67 wpm. Our data indicate that MCCs could provide even higher typing rates. We examined how our participants learned to chord, showing most of the speed increase associated with learning occurs during the in-air time interval. We also found a difference in strategy of how our participants press the buttons of a chord. The blind-typing data show that the Twiddler can be used effectively with limited visual feedback.

Finally, we presented our evaluation of our two aids, which were designed to help novice typists on the Twiddler mobile one-handed chording keyboard. We found that using an ordered phrase set designed around the Twiddler keymap increases typing rates and reduces physical demand. Using highlighting with our on-screen representation hinders typing rates once turned off. However, highlighting reduces error rates and decreases the subjective physical demand when on.

Our experiments evaluating the Twiddler have shown that an expert Twiddler user can rapidly enter text on a mobile device. In addition, the

Twiddler offers a touch-typing capability and is usable under limited visual feedback conditions. Combined with the ability to help novice typists with our two software aids, chording seems to be a viable mechanism for text entry on future mobile devices.

NOTES

Background. This article is based on part of the Ph.D. thesis of the first author.

Acknowledgments. Thanks to Stephen Griffin for designing the Twiddler phone prototype. This work is funded in part by the National Science Foundation (NSF) and the National Institute on Disability and Rehabilitation Research. This material is based upon work supported by the NSF under Grant No. 0093291. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of NSF. This is a publication of the Rehabilitation Engineering Research Center on Mobile Wireless Technologies for Persons with Disabilities, which is funded by the National Institute on Disability and Rehabilitation Research of the U.S. Department of Education under Grant No. H133E010804. The opinions contained in this publication are those of the grantee and do not necessarily reflect those of the U.S. Department of Education.

Authors' Addresses. Kent Lyons, GVU Center M/C 0760, 85 5th Street NW, Atlanta, GA 30332–0001. E-mail: kent@cc.gatech.edu. Thad Starner, GVU Center M/C 0760, 85 5th Street NW, Atlanta, GA 30332–0001. E-mail: thad@cc.gatech.edu. Brian Gane, School of Psychology, 654 Cherry St., Atlanta, GA, 30332. E-mail: bgane@gatech.edu.

HCI Editorial Record. First manuscript received July 8, 2005. Revision received December 18, 2005. Accepted by Scott MacKenzie. Final manuscript received March 5, 2006. — *Editor*

REFERENCES

Baker, S., Green, H., Einhorn, B., Ihlwan, M., Reinhardt, A., Greene, J., et al. (2004, June 21). Big bang! *BusinessWeek*. Retrieved February 1, 2006, from http://www.businessweek.com/magazine/content/04_25/b3888601.htm

Butts, L., & Cockburn, A. (2002). An evaluation of mobile phone text input methods. *Proceedings of the AUIC 2002*. Melbourne, Australia: ACS.

Card, S. K., Moran, T. P., & Newell, A. F. (1983). *The psychology of human–computer interaction*. Mahwah, NJ: Lawrence Erlbaum Associates, Inc.

Catrambone, R., & Carroll, J. M. (1987). Learning a word processing system with training wheels and guided exploration. *Proceedings of the CHI '87 Conference on Human Factors in Computing Systems*. New York: ACM.

Clawson, J., Lyons, K., Starner, T., & Clarkson, E. (2005). The impacts of limited visual feedback on mobile text entry for the Twiddler and mini-QWERTY keyboards. *Proceedings of the ISWC 2005*. Los Alamitos, CA: IEEE.

Conrad, R., & Longman, D. J. A. (1965). Standard typewriter versus chord keyboard: An experimental comparison. *Ergonomics, 8,* 77–88.

Gopher, D., & Raij, D. (1988). Typing with a two-hand chord keyboard: Will the QWERTY become obsolete? *IEEE Transactions on Systems, Man, and Cybernetics, 18,* 601–609.

Hart, S. G., & Staveland, L. E. (1988). Development of NASA–TLX (Task Load Index): Results of empirical and theoretical research. In P. A. Hancock & N. Meshkati (Eds.), *Human mental workload* (pp. 139–183). Amsterdam: North-Holland.

Henderson, V., Grinter, R., & Starner., T. (2005). Electronic communication by deaf teenagers (Technical Report GIT-GVU-05–34). Atlanta: Georgia Institute of Technology, Graphics Vizualization and Usability Center, College of Computing.

Hill, S. G., Iavecchia, H. P., Byers, J. C., Bittner, A. C., Zaklad, A. L., & Christ, R. E. (1992). Comparison of four subjective workload rating scales. *Human Factors, 34,* 429–439.

James, C. L., & Reischel, K. M. (2001). Text input for mobile devices: Comparing model prediction to actual performance. *Proceedings of the CHI 2001 Conference on Human Factors in Computing Systems*. New York: CHI.

Leech, G., Rayson, P., & Wilson, A. (2001). *Word frequencies in written and spoken English: Based on the British National Corpus*. London: Longman.

Lindstom, M. (2002, February 26). Message madness our big chance. *Sydney Morning Herald*. Retrieved February 1, 2006, from http://www.smh.com.au/articles/2002/02/26/1014471636718.html

Lyons, K. (2003). Everyday wearable computer use: A case study of an expert user. *Proceedings of the Fifth International Symposium on Human–Computer Interaction with Mobile Devices.* Heidelberg, Germany: Springer-Verlag.

Lyons, K., Gane, B. D., Starner, T., & Catrambone, R. (2005). Improving novice performance on the Twiddler one-handed chording keyboard. *Proceedings of the International Forum on Applied Wearable Computing*. Berlin, Germany: VDE VERLAG GMBH.

Lyons, K., Plaisted, D., & Starner, T. (2004). Expert chording text entry on the Twiddler one-handed keyboard. *Proceedings of the ISWC 2004*. Los Alamitos, CA: IEEE.

Lyons, K., Starner, T., Plaisted, D., Fusia, J., Lyons, A., Drew, A., et al. (2004). Twiddler typing: One-handed chording text entry for mobile phones. *Proceedings of the CHI 2004 Conference on Human Factors in Computing Systems*. New York: ACM.

MacKenzie, I. S. (2002). KSPC (keystrokes per character) as a characteristic of text entry techniques. *Proceedings of the Fourth International Symposium on Human–Computer Interaction with Mobile Devices*. Heidelberg, Germany: Springer-Verlag.

MacKenzie, I. S., Kober, H., Smith, D., Jones, T., & Skepner, E. (2001). LetterWise: Prefix-based disambiguation for mobile text input. *Proceedings of the UIST 2001 Symposium*. New York: ACM.

MacKenzie, I. S., & Soukoreff, R. W. (2003). Phrase sets for evaluating text entry techniques. *Extended Abstracts of the CHI 2003 Conference on Human Factors in Computing Systems*. New York: ACM.

MacKenzie, I. S., & Zhang, S. X. (1999) The design and evaluation of a high-performance soft keyboard. *Proceedings of the CHI '99 Conference on Human Factors in Computing Systems*. New York: ACM.

Matias, E., MacKenzie, I. S., & Buxton, W. (1993). Half-QWERTY: A one-handed keyboard facilitating skill transfer from QWERTY. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* (pp. 88–94). New York: ACM Press.

Matias, E., MacKenzie, I. S., & Buxton, W. (1996). One-handed typing with a QWERTY keyboard. *Human–Computer Interaction, 11,* 1–27.

Noyes, J. (1983). Chord keyboards. *Applied Ergonomics, 14,* 55–59.

Ratz, H. C., & Ritchie, D. K. (1961). Operator performance on a chord keyboard. *Journal of Applied Psychology, 45,* 303–308.

Schneider, W. (1985). Training high-performance skills: Fallacies and guidelines. *Human Factors, 27,* 285–300.

Schumacher, E. H., Lauber, E. J., Glass, J. M., Zurbriggen, E. L., Gmeindl, L., Kieras, D. E., et al. (1999). Concurrent response-selection processes in dual-task performance: Evidence for adaptive executive control of task scheduling. *Journal of Experimental Psychology: Human Perception and Performance, 25,* 791–814.

Seibel, R. (1962). Performance on a five-finger chord keyboard. *Journal of Applied Psychology, 46,* 165–169.

Silfverberg, M. (2003). Using mobile keypads with limited visual feedback: Implications to handheld and wearable devices. *Proceedings of Fifth International Symposium on Human–Computer Interaction with Mobile Devices*. Heidelberg, Germany: Springer-Verlag.

Silfverberg, M., MacKenzie, I. S., & Korhonen, P. (2000). Predicting text entry speeds on mobile phones. *Proceedings of the CHI 2000 Conference on Human Factors in Computing Systems.* New York: ACM.

Soukoreff, R. W., & MacKenzie, I. S. (2003). Metrics for text entry research: An evaluation of MSD and KSPC, and a new unified error metric. *Proceedings of the CHI 2003 Conference on Human Factors in Computing Systems.* New York: ACM.

Starner, T. (2000). *Wearable computing and context awareness*. Unpublished doctoral thesis, Massachusetts Institute of Technology Media Laboratory, Cambridge.

Wigdor D., & Balakrishnan, R. (2003). TiltText: Using tilt for text input to mobile phones. *Proceedings of the UIST 2001 Symposium.* New York: ACM.

Wigdor D., & Balakrishnan, R. (2004). A comparison of consecutive and concurrent input text entry techniques for mobile phones. *Proceedings of the CHI 2004 Conference on Human Factors in Computing Systems.* New York: ACM.

Women embracing SMS–study. (2002, October). Retrieved February 2, 2006, from http://www.mobileoffice.co.za/news_2002/100202-women_embracing_sms.htm

Yeh, Y., & Wickens, C. D. (1988). Dissociation of performance and subjective measures of workload. *Human Factors, 30,* 111–120.