Oriented Scanning compared to Alternative Code

Joris Verrips

Some literate people need to use switches to create text that is voiced by synthetic speech to communicate because they can neither speak nor type due to rare neurological conditions like locked in syndrome, ALS and MSA and a rare literate but aphonic patient with spastic palsy. Row-column scanning is inherently slow, therefore text input with switches can be quite frustrating. Two alternative techniques were tested with a copy task by three able bodied test subjects. Oriented Scanning requires few clicks but significant visual attention. In Alternative Code eleven groups of consecutive characters have related codes. This assignment is displayed on screen and appears easy to learn.

The problem

How to input text at a high rate with a single switch and by a method that is rapidly learned has been called 'the single key challenge', see MacKenzie, 2009. The ‘input rate problem' as by Alm, Todman, Elder and Newell, 1993, is a bit more general, includes other access methods, and has been studied for a long time in Augmentative and Alternative Communication, or AAC. Morse Code, defined in 1836 by Vail, and widely applied until about 1960, *almost* solves this problem because it is fast[1]. Usually a single switch is employed and short switch closures called 'dits' and long switch closures called 'dahs' are combined with pauses into codes of characters. Frequent characters have short codes and numerals, clicking and moving a mouse pointer or entering all other characters are possible too.

Morse Code however is neither learned easily nor is it error tolerant, as many a former boy scout can testify. One may throw out codes that are hard to learn, this is the approach of Oriented Scanning that presents the result in a scanning matrix. One may also maintain the codes but change their assignment to the alphabet. This is the approach of Alternative Code, displayed on screen as well. Both techniques might be helpful for reading handicapped subjects and are described below.

Oriented Scanning

Oriented Scanning can be operated with a single switch, clicked for vertical and sustained for horizontal movement, but using two switches feels more natural. One switch effects a movement Down and the other effects a movement Right. In the scanning matrix of Figure 1, ten cells have *two* distinct meanings each. The cell labeled 'd\e' reads 'd' when accessed from the left with Down Right and Pause. The same cell means 'e' when accessed from above with Right Down and Pause. One might say that these cells are *oriented*, whence the name Oriented Scanning, a variant of active row-column scanning[2]. This trick appears to be new and is combined with stored words and with selective

1 In the past, professional telegraphers achieved rates over 40 *words* per minute. Such rates seem impossible to achieve in AAC, but do motivate research into Morse Code.

2 With active scanning, sometimes named reverse scanning, is meant that a click of a switch moves a focus, unlike passive scanning where the focus is moved by the software and clicks make it change direction or stop, read a character and start anew. As a rule active scanning is faster than passive scanning.

delays.

Figure 1. Oriented Scanning with a blue focus that is moved by switches. Bs = Back Space, Sp = Space, F1 to F3 speak and edit a line of text.

To select 'm' requires Down Down Right and a Pause of, say, 240 milliseconds. In the same circumstances, to select 's' will require Down Down Down and a Pause of 400 milliseconds. Pauses are 66% longer on the margins, where the selection direction may change. Long codes such as 'j' or 'k' are more error prone and have a longer pause as well. When text prediction is activated Figure 2 is displayed.

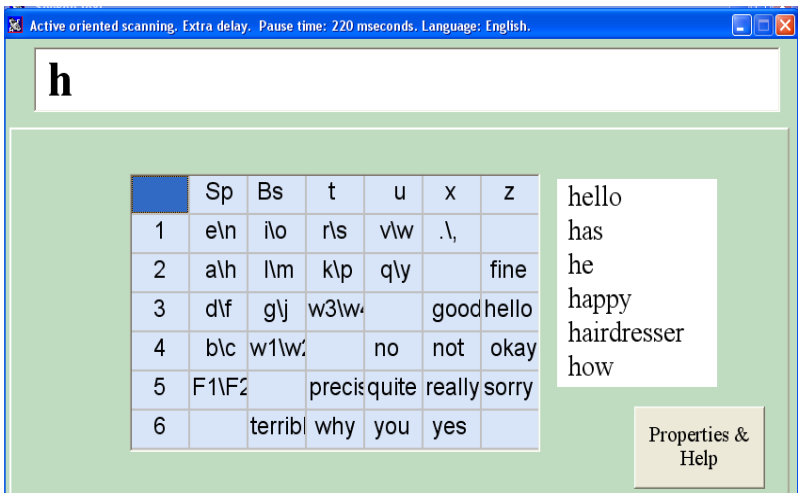

Figure 2. Oriented Scanning with text prediction after 'h', selected by Right Down Down. Down Down Down Down will now select 'happy ', the fourth word in the prediction list.

In Figure 2, the left upper triangle contains 27 cells with 45 different items that can all be selected with at most six clicks and one pause and include numerals to also access word prediction. Figure 2 is much harder to use than Figure 1 because word prediction easily distracts, because there are more items to choose from and because the ordering of characters is rather complex. In Figure 1 characters are ordered alphabetically and from left to right, but in Figure 2 they

are ordered first on frequency then alphabetically along diagonals rising from left to right. They can be hard to find back in a split second and this ordering is not learned easily.

Alternative Code

Another method for text input with switches may be called Alternative Code. Like Oriented Scanning, and as far as I know, it is a *new* technique and can be used with either one or two switches. Most related codes are assigned to consecutive characters and codes of frequent characters like Sp, Bs, E, A, I and N are short. When we compare the code for 'r' with Figure 1 we find '..-' instead of Right Right Right Down Down, quite a difference, as is the case for character 't'. For other characters the difference is slight and if we use character frequencies of modern printed English[3], we find 2.66 clicks per character with Figure 1 and 2.40 clicks per character with Figure 3.

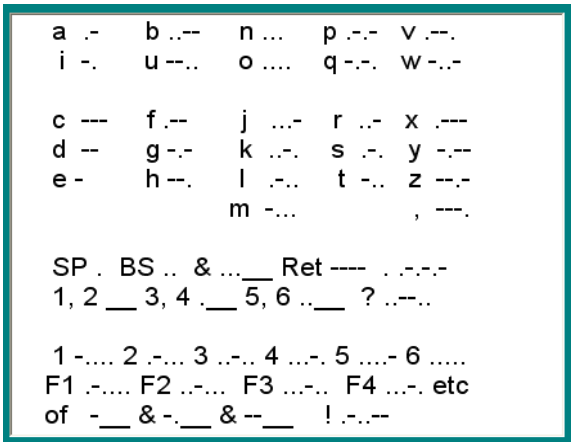

Figure 3. Alternative Code. __ means a sustained dash, of variable length. They can be useful for word prediction and can be preceded by both dits and dashes.

Selection sequences of Oriented Scanning consist of a series of Down, followed by a series of Right, or the reverse, followed by Pause, and including empty series. (Alternative) Morse Code is more compact and therefore error prone. Mathematically speaking, and with R* meaning 'repeat R any number of times including zero' we may say (Alternative) Morse Code = (. OR -)* Pause and Oriented Scanning = (Down*Right*) OR (Right*Down*) Pause.

Test

The following hypothesis was tested: 'input with Oriented Scanning will be as fast as with Alternative Code in the first hour and with a copy task'. Three different test subjects copied English texts[4] from paper with Figure 1, *two* switches, a pause time of two hundred twenty milliseconds, selective delays and for exactly one hour. The same exercise was done with Figure 3, with the same texts, the same pause times and with stretched pauses after codes of length three. All subjects slowly read those texts first to diminish the effort of spelling and the effect of starting with one technique. All subjects tried both systems for ten minutes each to get used to them and to the switches. Pause time was neither varied nor optimized individually[5] and errors had to be corrected. Short breaks were allowed and users were interviewed

3 Using Wolfram's Alpha and adding 18% for Sp and 11% for Bs, based on measurements during development.

4 From MacKenzie, 2009, the first thirty lines with all word lengths below six.

5 Pause time after training is considered optimal at about 1.6 times reaction time, see Simpson, Koester and Lopresti, 2006. They quote evidence

after completing their copy tasks. We measured input rate per minute because this interested us most and compared input rate per minute after equal practice time because this seemed the most relevant fact. See Picture 1 for a first impression.

Picture 1. MB during a copy task.

Results

Both techniques worked rather well. As might have been expected, test subjects verified codes on screen and needed some time to get used to entering text with switches. Table 2 shows average input speeds in two conditions as well as token tests on individual data points, Table 3 shows comments, Figure 4, Figure 5 and Figure 6 show results of the first test subject.

Test subject	Characters per minute with Oriented Scanning in the first hour; SD; total Back Spaces	Characters per minute with Alternative Code in the first hour; SD; total Back Spaces	Average with Oriented Scanning < Average with Alternative Code	Token test[6], alfa with OS[i]>AS[i]
1 (MN) AS, OS, AS[7]	17.2; 4.1; 162	23.5; 4.2; 159	1	P<0.005
2 (MB) OS, AS	23.4; 4.8; 175	28.3; 3.6; 127	1	P<0.005
3 (JvD) OS, AS, OS, pause time = 260 msec	16.7; 3.1; 156	21.1; 4.66; 109[8]	1	P<0.01

Table 2. Average input speeds, standard deviations, number of BackSpaces and some statistics.

that indicates the fraction (scan rate/reaction time) should be 0.65. Scan rate is not *exactly* the same as pause time.

6 Or Sign test.

7 30 minutes AS, 2 * 30 minutes OS, 30 minutes AS.

8 Did not correct all errors.

Subject	Comments
1 (MN)	Felt pause time too short (adapted to 260 milliseconds after 30 minutes). Prefers Alternative Code because 'feel distracted by the moving focus of Oriented Scanning'.
2 (MB)	Prefers Alternative Code because 'dislike to search correct path before you start to select'.
3 (JvD)	Prefers Alternative Code because 'it automates more easily'.

Table 3. Comments of test subjects.

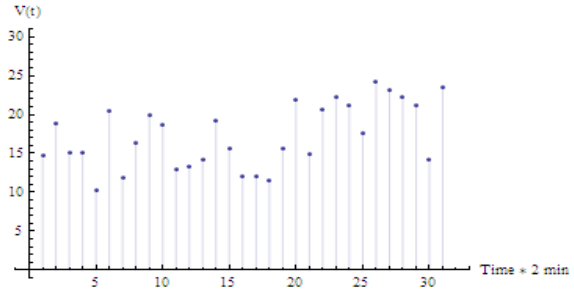

Figure 4. Input rate with Oriented Scanning by the first test subject (MN).

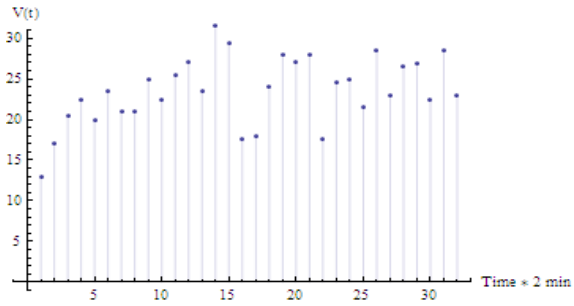

Figure 5. Input rate with Alternative Code by the first test subject (MN).

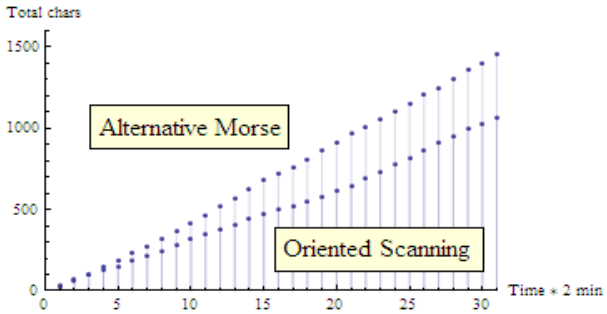

Figure 6. Summed input rates with Alternative Code and Oriented Scanning by the first test subject (MN).

Conclusion

Alternative Code and Oriented Scanning are fascinating techniques for text input with switches. Both appear easy to learn.

Conflict of interest There is a conflict of interest because the author is designer, programmer and distributor of Alternative Code and of Oriented Scanning.

Illustrations Figure 4, Figure 5 and Figure 6 were made with Mathematica, most other Figures are screen images made by the author and Picture 1 was made by Annie Verrips.

Videos On www.depratendecomputer.nl/results.htm several videos are available on communicating with switches.

Www.depratendecomputer.nl/comparisonfulltext.pdf offers more text and some references.

Thanks to test subjects Mente Nauta, quality manager, Marjo Biesboer, speech language pathologist, and Jack van Dillewijn, psychologist. Thanks to Piet van Blokland for the idea of Figure 6and to Denis Anson for criticism.

An earlier version of this text was accepted for publication in Communication Matters, 2012.

Author mail: j.verrips@planet.nl Website www.depratendecomputer.nl.