Two-bit quartering, extended report

Joris Verrips, MD
Paramaribostraat 138”
1058 VP Amsterdam
j.verrips@planet.nl

last changed July, 2009

Abstract

Due to uncommon neurological diseases such as ALS and pure motor aphasia, some people loose the ability to write and to talk but keep the ability to read and to formulate. To create text one may try electronic switches and (passive) scanning, a reliable and simple but generally speaking also an extremely slow process of about ten characters per minute or less. Several scanning techniques are described, as well as two-bit quartering, a new technique for switch input developed by the author. Like Morse Code, once learned it is relatively fast. Also, it combines rather well with text prediction. Three variants were tested with two healthy test-subjects and a copy task of voiced poetry. Input rates were 16.5 characters per minute (cpm) with one switch and 26.9 cpm with two switches and after additional practice. Curve fitting was done to a simplified Michaelis-Menten equation v(t)=at/(t+c) or v(t)=a/(1+c/t) where v=input rate, a=asymptote of input rate, t=time and c=learning time. This leads to expected input rates of 19 and 35 cpm respectively and learning times of about 20 minutes. To compare different scanning-techniques without further experimentation frequency-weighted clicks and pauses per character were computed. Also, possibilities for more research are discussed.

Introduction

Scanning methods are intended for people who cannot access a normal keyboard because of conditions such as Amyotrophic Lateral Sclerosis, locked-in syndrome or spastic paresis. They can but need not be combined with synthetic speech. This section starts with illustrations of several communication techniques. First, photographs 1 to 3 show diverse switches and a communication aid with a scan module.



Photo 1. Several switches to record blinking (with an infrared light), clicking and pressing.

Photo 2. Probably the best-known communication aid is the Lightwriter.

Photo 3. A model with a single switch and a scan unit that supports row column scanning and also stepped scanning with integrated word prediction.

See Figure 1 for low-tech row column scanning.

A	B	C	D	Yes
E	F	G	H	No
I	J	K	L	M
N	O	P	Q	R
S	T	U	V	Why
Space	W	X	Y	Z

Figure 1. Simple communication aid for row-column scanning with a colored piece of paper.

In passive mode the interpreter speaks first colors then characters and may use partner-prediction and imagination to speed up. Patients may use grunting, eye-blink or tiny movements, either passively or actively, and learning time seems minimal. Active mode is also called tapped scanning and has been used in prisons; if 3-2 means three vertical two horizontal we see 3-2=’j’. It is faster than passive mode and closely resembles active row column scanning with switches.

Figure 2 show a matrix with a blue focus in the left upper angle as presented on screen. The first column is empty, therefor the machine can not 'read' characters in those eight fields.

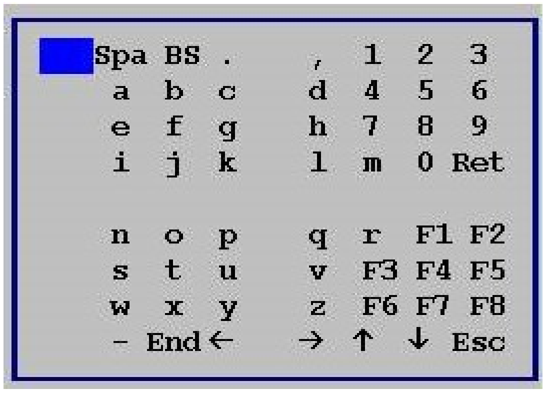


Figure 2. Matrix for row-column scanning.

To select 'e' with active row column scanning we need to move the focus two times downwards (D) and once to the right (R), or DDR. Depending on the algorithm, this character j may then be read by the machine. See figure 0.3 to 0.6.

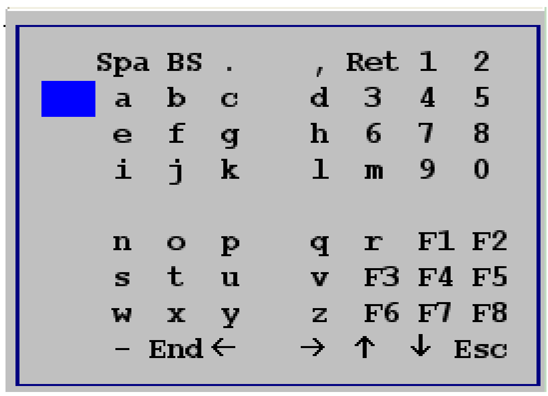


Figure 3. After activating of a switch called 'D' for downwards.

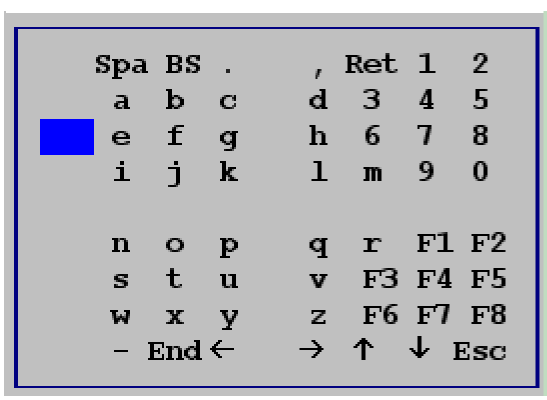


Figure 4. Another 'D' moves the focus again. 'R' now moves the focus to the right.

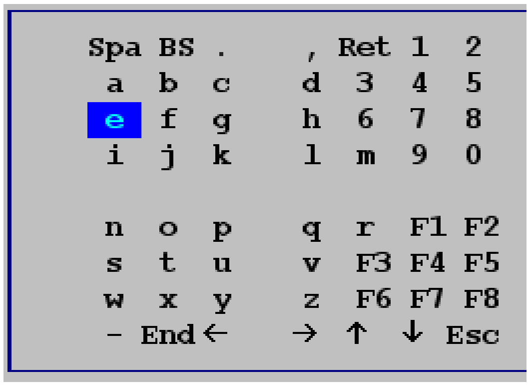


Figure 5. When we take a pause, the character 'e' is chosen.

When the focus changes colour from time to time a single switch may allow to move either vertically or horizontally. Then especially it is pleasant to have an empty row or an empty column so as not to hurry during or just after a 'change of direction'. We would then select 'e' not with DDDR<pause> but with CCC<long pause>C<pause>. C stands for Click and after <long pause> the focus changes direction and the <pause> can be short. Row column scanning can be varied in many ways, Figure 6 shows a frequency optimised matrix. Though harder to learn, for the average character the distance from the origin is shorter than in the alphabetical matrix of Figure 3.

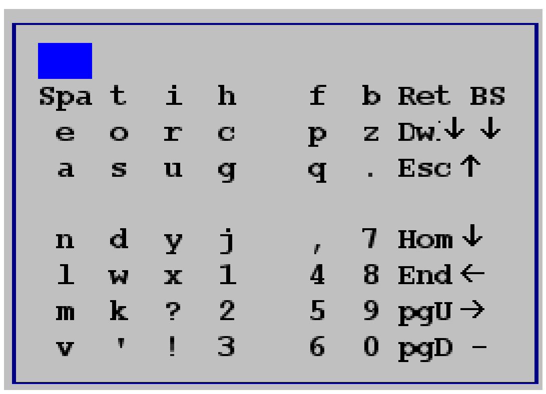


Figure 6. Matrix with frequent characters close to the origin in the left upper corner. We can also see an empty first row, that suggests column-row scanning.

Another interesting variant is stepped scanning, present among others in the Lightwriter of Photo 3. One switch moves the focus from left to right, the machine moves the focus from top to button and the other switch accepts. The matrix is longitudinal, three or four rows and about twelve columns. See Figure 7.

Figure 7. Longitudinal matrix for stepped scanning and integrated wordprediction with Lightwriter SL87.

Figure 8 to 11 show two-bit quartering with two switches.

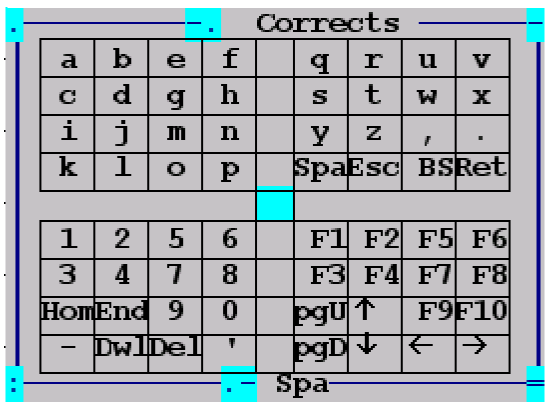


Figure 8. Two-bit quartering. Four different quarters are filled with characters and are displayed on screen surrounded by a blue square. Spa = Space, pgU = Page Upwards and so on.

In the four angles of the blue surrounding square of Figure 8 we see four different codes and two more in the middle above and below for Reset (Corrects), when we select the wrong quarter, and for Space, that requires just a single code. Two switches mean dash ('-') and dit ('.') to select these six codes. The symbol '=' means two dashes separated by a short pause while ':' means two dits with a pause between them. With repeated dashes we will see Figures 9 to 11.

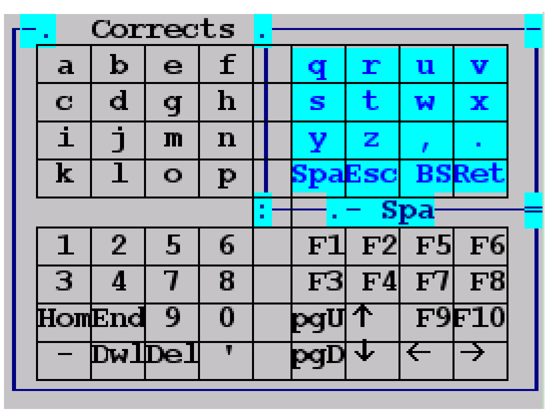


Figure 9. Screen image after one dash in Figure 8, followed of course by a pause. The second quarter is now marked and more codes have effect on this quarter only. The code '-.' corrects and brings back Figure 8. Code '-' leads to Figure 10.

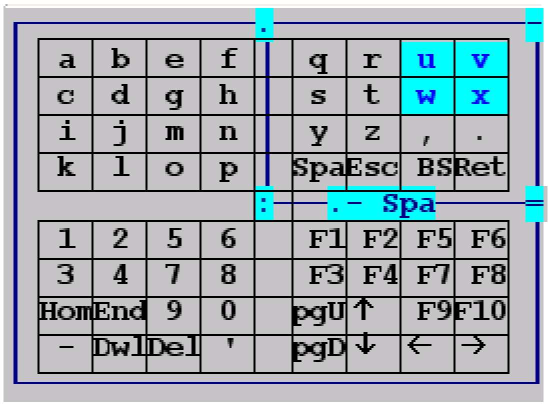


Figure 10. Again the machine waits on an action by the user. Another '-' will choose 'v' and will display Figure 11, followed by Figure 8.

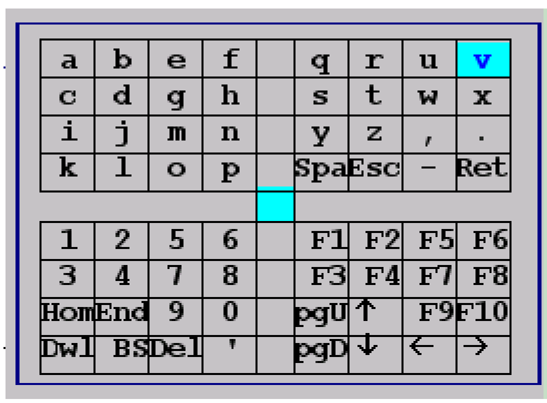


Figure 11. For clarity the character 'v' is marked, then Figure 8 is shown.

Two-bit quartering with a single switch

Commercial distributors prefer portable communication aids accessed by a single switch. When we choose for access with variable click time codes have to be redesigned, dashes are about three times longer than dits so two consecutive dashes does not feel right. In Figure 12 the matrix is also adapted on the basis of character frequencies with a prominent place for BS, or BackSpace.

Figure 12. Redesign for a single switch with a frequency-optimised matrix.

Above the matrix variables can be set and the last few characters are displayed, here the word 'Dag '. We see codes for dit, twice dit, three times dit, dit dash and, for Space, dash. The shortcut dash now chooses Space but changes meaning as soon as a quadrant is chosen. This meaning is shown in the middle, 'a', 'i' and 's'. Twice dit chooses the second quadrant. To select 'i' a user gives dit dit pause dash pause, three clicks and two pauses. Other characters are visible after '..-', the code for Reset[1].

How to compare several scanning techniques

Two-bit quartering was discovered in 2003 as a new text-entry method. It seemed rapidly learnable and useful even with eyes closed. In experiments with the author as test-subject active two-switch row-column scanning was significantly faster than passive row-column scanning and two-bit quartering was a little bit faster yet[2].

Passive scanning means that users have to wait for the machine to suggest first the correct row and then the correct column. With active scanning users moves the focus by clicking and the machine 'reads' when one does not click. As a click takes less time than a pause active scanning is faster than passive scanning. According to the literature with passive scanning input rate is about 5 to 10 characters per minute (cpm). No reliable data were found on stepped scanning or on active row-column scanning[3]. See Table 1 for an overview.

---

[1] Reset has several functions and is called 'Corrects' in Figure 12.

[2] Verrips, 2003.

[3] See Simpson, Koester en Lopresti, 2006.

	Focus moved	Character accepted	Layout
Passive scanning	By machine	By user	Rows and columns
Active scanning	By user	By machine	Rows and columns
Stepped scanning	User horizontal, machine vertical	By user (2nd switch)	Rows and columns
Two bit quartering	By users	By machine[4]	Shrinking quarters

Table 1.Several scanning methods that can be combined with diverse scan matrices and switches.

To compare scanning mechanisms reliably, with members of the target groups, with realistical tasks (such as a copy task) and in a reproducible manner and to also record ease of learning is not an easy task, one may wonder if it is feasible. This article describes reproducible experiments with relevant tasks by healthy test-subjects to loosely compare three different types of two-bit quartering. Though the author is convinced the results do reliably reflect differences between those three setups, the test-design does not systematically exclude effect of learning or inter-individual differences. Also it offers some modelling, explains how word prediction fits in and discusses possible further research.

Design

Because the target group is highly variable reproducible experiments with reliable measurements request cooperation with paid healthy test subjects. The first one, a personal friend, was demonstrated Figure 12. His short and long click times were measured, the pause time was set at 1.75 times the short click time (1.75*205 milliseconds = 360 milliseconds) and the conversion time from dit to dash likewise. The test subject clicked his name and played with the system for a while. Next a copy task was done with two-bit quartering and input through the right mouse button. Almost four times ten minutes per hour Dutch poetry was read with five minutes pause to relax.

Selected characters and words were spoken and the ‘conversion moment’ was made audible with a click. Upper case characters were ignored as well as punctuation. Errors could be corrected but this was not stressed. After thirty minutes characters no longer were audible and after forty-five minutes a second appointment was made. Pause time and conversion moment were both changed from 360 to 300 milliseconds, shorter times were not accepted[5]. The test subject copied the matrix twice and practised for another 45 minutes. See Photo 4 for an impression.

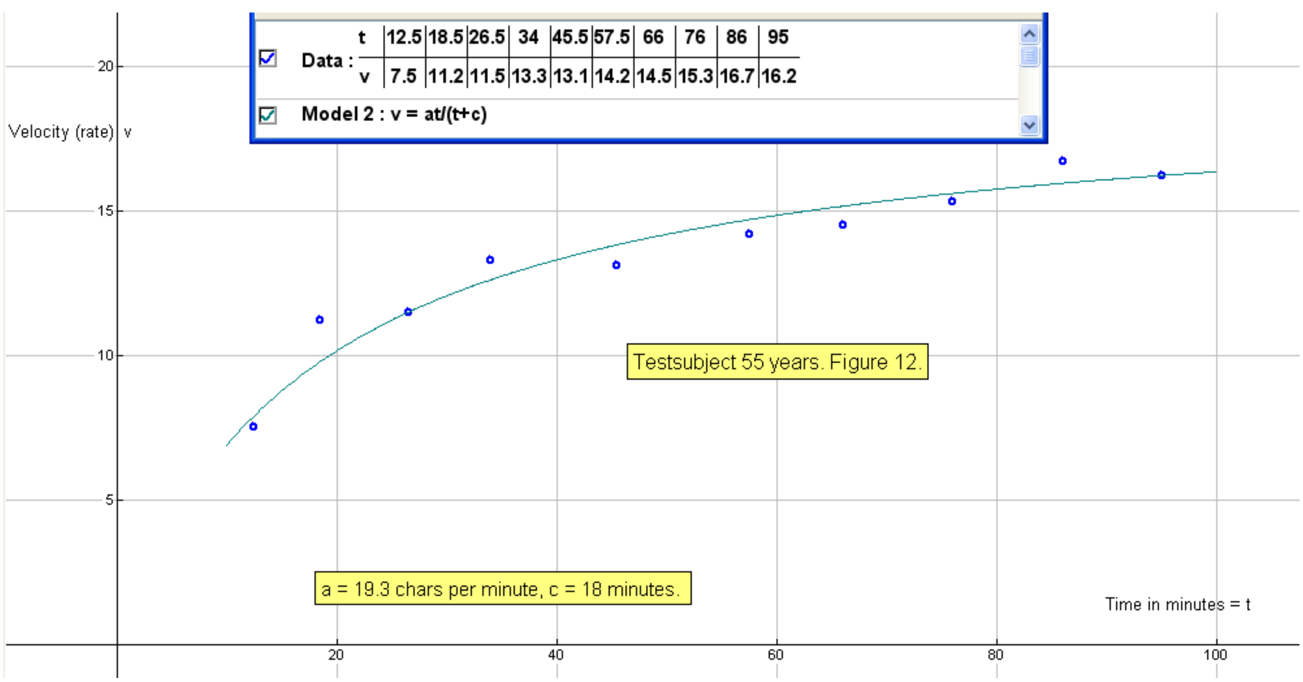
[4]Though if a technique called pause reduction is used, the distinction can not be made.

[5] Long clicks take about three times as long as short clicks. See King 1998 for references.

Photo 4. Copy task with poetry and clicks entered through the keyboard.

Data

See Figure 13 for an overview of results. The green curve is a model of the input-rate as measured by the software and as a function of the time, v(t)=at/(t+c) with a and v in characters per minute, t and c in minutes. A represents the limit of the input rate and c represents learning time. We took a=19.3 cpm and c=18 minutes[6]. The average value of v over the last two sessions was 16.5 characters per minute.



[6] For the curve fit Raaijmakers, 1987, was combined with inspection.

Figure 13. Input rate with a copy task and with a single switch, variable click time and Figure 12. The first task took five minutes and started after ten minutes of practice. Measured input rate was 7.5 characters per minute, we see a data point in (t=12.5, v=7.5). The second task took seven minutes so we compute t=(15+22)/2=18.5.

The fastest variant of active scanning is probably optimised two-bit quartering with two switches[7] and extreme pause reduction[8].

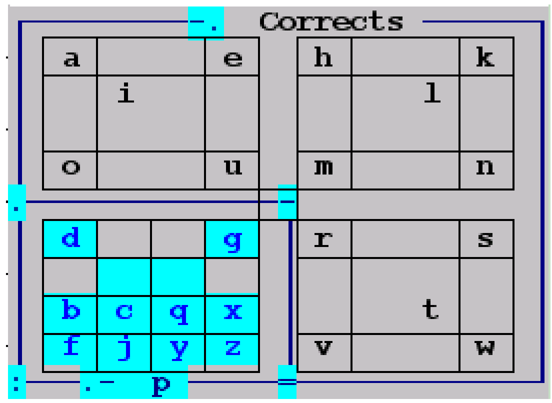


Figure 14. Optimised matrix with vowels in the first quarter.

In Figure 14 eighteen characters can be selected with just two codes and space with just one. To access BackSpace and Return one must first choose ‘-.’, labelled Corrects. The second test subject entered a few more spoken poems with this technique. Figure 15 shows results, a gradually increasing speed and slowly diminishing variance, both signals that learning occurs. Curve fitting leads to values of 34.9 cpm for a and an amazingly low value of 21 minutes for c[9]. The average of the last three sessions was 26.9 cpm.

[7] Pause time can be short, no time is spent on sustained clicks that, also, do not request extra attention.
[8] After two clicks the machine does <u>not</u> take a pause.
[9] That this test subject had some experience with two-bit quartering no doubt influenced the value of c.

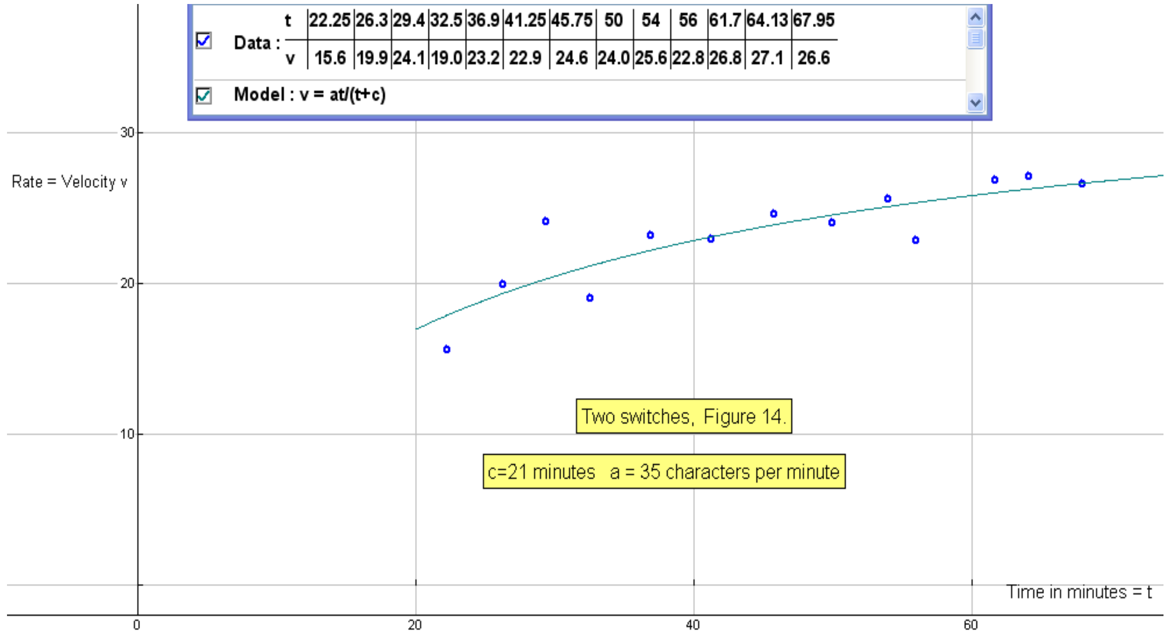


Figure 15. Data with Figure 14 and two switches.

The alphabetically ordered matrix of Figure 8 is probably learned rapidly. To document a test subject of 64 years repeated the exercise during 108 minutes with input by two keys, Figure 8 and ordinary pause reduction[10]. Pause time was 300 milliseconds, see Figure 16 for results. As little time was spent to practice or to demonstrate, the horizontal axis starts close to zero.

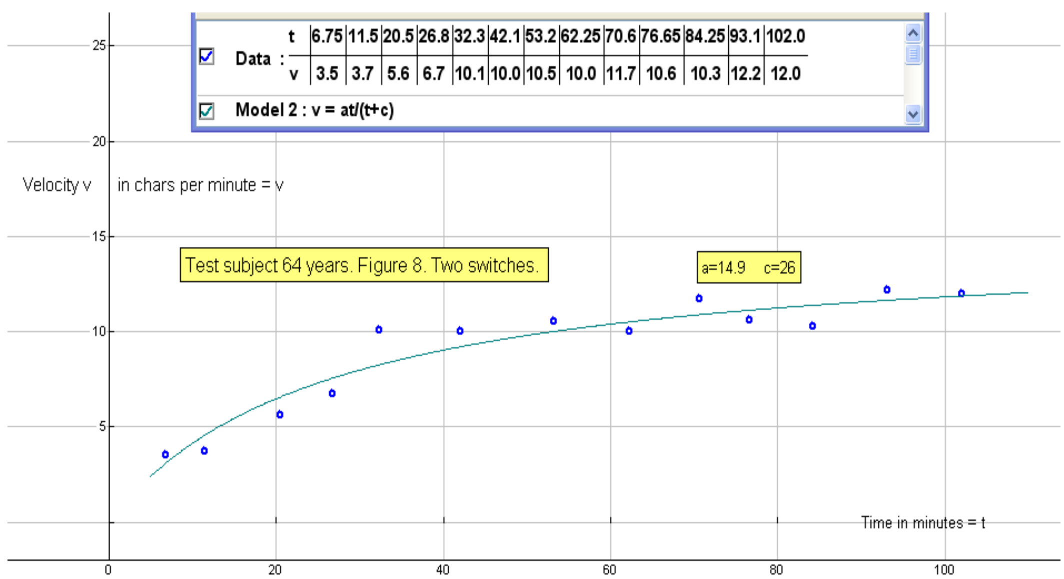


Figure 16. Data of a copy task with Figure 8 and two switches.

[10] Pauses are taken after codes that exist of just one click and after all codes when no quarter has yet been chosen.

Interpretation

Figure 13 shows that with just one finger and optimised two-bit quartering spoken poetry can be copied rather efficiently, at least by this test subject. We interprete this as a *compromise* between easy learning and high input rate with a single switch. With two switches higher speeds are possible and Figure 14 is *fast*. The matrix of Figure 8 is *easy to learn* and allows to choose from over sixty-four characters or symbols. This is more or less what one would expect, but represents no proof as only two test subjects were used and as no experiments were done with intended users. Direct experimental comparison with other scanning methods easily introduces some form of bias. However, much of scanning can be modelled, as explained in the next section, and this allows comparisons on paper.

Models

That active scanning is often faster than passive scanning follows from the fact that clicks can be faster than pauses because no reaction time is necessary. How much faster it will be however is not easy to say. In the same spirit two switches must be faster than one switch, because a long click takes more time than a short one. Again how much faster, and how error rate and learning times and user characteristics like reaction times interact with one another, is not easy to say. Due to the enormous number of different scanning techniques such questions seem impossible to answer experimentally for all possible users *and* all techniques *and* all types of switches.

This section will not even try, but will compare different algorithms and scan matrices in terms of expected effort and on paper. First, we compute the expected number of clicks and pauses per character with known relative character frequencies in the language. The number of clicks of a frequent character like ‘a’ counts more than a less frequent character like ‘q’[11], see Table 2.

	Number of clicks per character	Number of pauses per character
Figure 8 with two-switch access and ordinary pause reduction	3.40	2.01
Figure 12 with one-switch access without pause reduction (and with extreme pause reduction)	Short: 2.68 Long: 0.87	1.93 (0.79)
Figure 14 with extreme pause reduction and two-switch access	2.81	0.91
Morse Code with a dual switch[12]	2.62	1.00
Stepped scanning with matrix of photo 3 and without word prediction	4.16	1.36
Stepped scanning with frequency optimised matrix[13] and without word prediction	Circa 3.2	Circa 1.3
Active row-column scanning with frequency optimised matrix and two switch access[14]	3.26	1.5

[11] This resembles the method of Venkatagiri, 1999 who simulated different scanning matrices with the same body of text.
[12] Left = long and Right = short or the inverse. With a single switch long clicks take about three times as long as short ones.
[13] Fewer clicks for frequent characters.
[14] Hard to learn with eyes closed, considerable learning time and no less error-prone than two-bit quartering.

Passive row-column scanning with frequency optimised matrix	3	3.26

Table 2. Weighted values for clicks and pauses per character with Dutch data[15].

Small differences in these numbers need not translate into small differences in input rates as other characteristics are not modelled. For instance, Morse Code automates on the word level, and two-bit quartering can be learned with eyes closed and active row-column scanning seems more error prone than its alternatives. Still, we may estimate the number of seconds per character as expected-number-of-clicks times click-time plus expected-number-of-pauses times pause-time plus overhead. Both pause-time *and* overhead vary per test subject *and* per algorithm because the function of pauses differs between algorithms. We may then construct formulas like 60-seconds-per-minute/seconds-per-character, and though these formulas will not be very reliable the data clearly agree with them. This formula also resembles $v(t)=at/(t+c)=a/(1+c/t)$ where the input speed is expressed as a fraction.

The function c/t gradually decreases with time, represents the variable part of the overhead and suggests a relation between a mechanism being easy to learn and being easy to execute. It is an unusual formula to model learning processes such as where characters are displayed, where to look and how to coordinate. According to the Power Law of Learning, inspired by economics (Wright, 1936), cost per unit = $b\ N^{-x}$ with x the number and both b and N constants determined with curve fitting. If we suppose that different learning processes hardly overlap in time and that their effects diminish we get something like $b=1$, $N=e^2$, $x \approx t-a$ and learning curves g_a: if($t<a$;1;1 $e^{2(a-t)}$). This means that $g_a=1$ if $t<a$ and $g_a= e^{2(a-t)}$ otherwise, x is approached by t-a. With c=3 we see $c/t \approx \sum_{a=1}^{a=6} 1/a\ g_a(t)$, as shown in Figure 17.

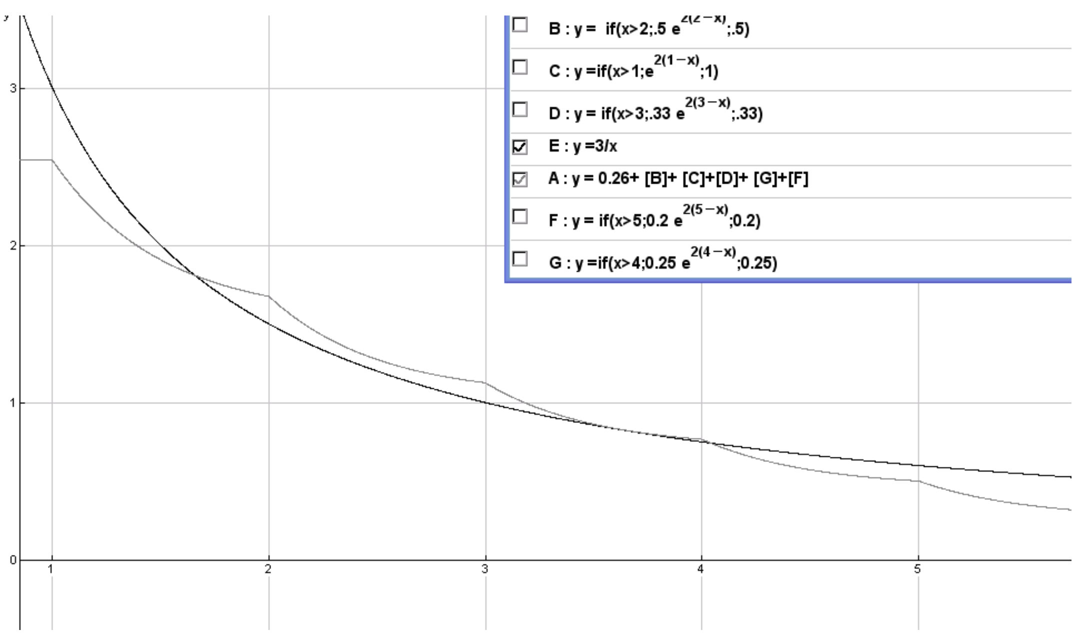


Figure 17. Plot of c/t, graph E, and graph A, a sum of 'ordered learning curves'.

Figure 17 illustrates that curve fitting of $(1+c/t)$ with several functions inspired by the Power law of learning looks feasible. Because actions like clicking and pausing will always require a

[15] Frequencies based on newspaper data and collected by M. Broecke.

minimum of time, any reasonable model will have some constant included as in $a_i+b_ie^{-xc}$. This does not represent a psychologically valid predictive model of the task at hand, it only does not necessarily contradict it.

Finally, we note that all scanning mechanisms considered so far select words character after character. Though certainly word prediction, macro's and stored text can be clinically relevant, it seems that to model input rate a formula like 60-seconds-per-minute/(seconds-per-character+overhead) and compute the expected number of seconds per character with weighted frequencies is rather reasonable and makes one think of a so called keystroke-level model[16].

We conclude that two-bit quartering has interesting properties and that the formula $v(t)=at/(t+c)$ is a reasonable approximation, although it can not predict actual input rate precisely.

Alternatives, extensions and remaining questions

In Figure 18 we see what probably is the fastest input technique with switches, known and used since the 1840s[17] and in part inspired it seems on smoke signals of the American Indians. Morse code was used for ages in (radio-)telegraphy and has been used with eye-blinks and other low-tech systems as well. It then requires a significant learning effort on the part of caregivers and significant others and/or interpreters.

```
   a .-   f ..-.   k -.-   p .--.   u ..-
   b -... g --.    l .-..  q --.-   v ...-
   c -.-  h ....   m ..--  r .-.    w .--
   d -..  i ..     n -.    s ...    x -..-
   e .    j .---   o ---   t -      y -.--
BS ----  SP --  Ret .-.-   1 -....   z --..
```

Figure 18. Good old Morse Code, with minimal changes.

Because two-bit quartering uses but a part of Morse Code functions can be added rather easily. Figure 19 offers an example. After selecting ‘h’ in the middle of the matrix two recently used words starting with ‘h’ are suggested that are bound to two extra codes.

[16]See Card, Moran & Newell, 1982 for more on this branch of human computer interaction. They state, among much else, that this explains just 50% of the time spent.

[17] For quite a while people could not believe that it actually worked. Many others tried to achieve telegraphy with electricity, inspired by optical telegraphs as constructed and used by Napoleon. See Standage, 1998, and King, 2000.

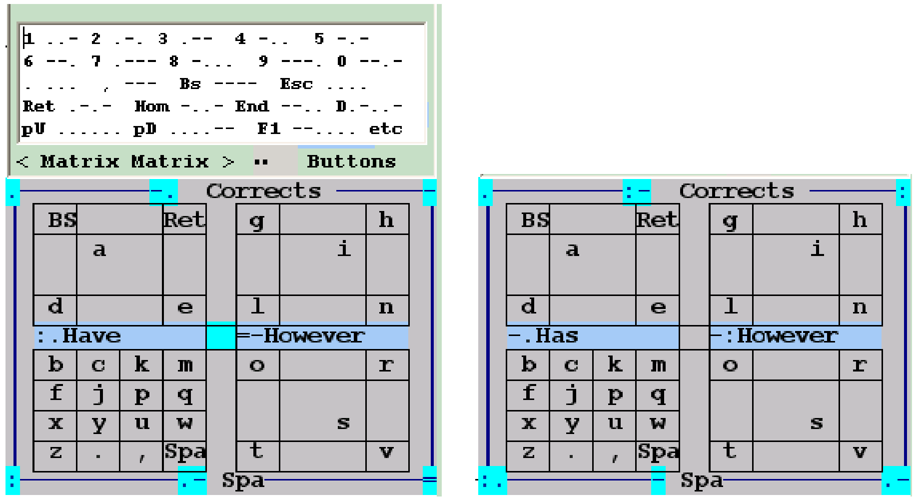


Figure 19. Word prediction can be accessed by extra codes.

In Figure 19, three more clicks choose either ‘have ’ or ‘however ’ and some more codes are shown on the left. On the right we see that with one switch, codes are slightly different. Whether word prediction integrated in scanning ever pays off is unknown. The mechanism depicted in figure 12 is rather elegant and compares well with alternatives, but in English the average word length is about four characters, plus space. As most predictive algorithms will need two characters to give two reasonable alternatives users then have to look at the screen, make a decision and select with three clicks and one pause. Due to inevitable overhead this probably is not faster than two more characters and a space, about six clicks and two pauses. Perhaps if long words are often repeated or are known to be present and with users who are not easily distracted somebody will show that some form of word prediction does speed up[18]. Likewise for word prediction integrated with on screen keyboards that are accessed by eye-tracking plus a blink switch.

Many more possibilities for (independent) research are present. One might study how low-tech aids as shown in Figure 0.1 compare to high-tech aids[19], what care-givers and family-members think about this, if Morse Code is faster than active scanning[20], how two-bit quartering performs in combination with stored text[21], how cost-effective communication aids are[22], if optimised two switch row column scanning is slower than two-bit quartering[23], how much[24] idem for passive scanning with optimised matrices …. and more.

[18] This would shock careful readers of Koester and Levine, 1996 and 1997. And .. complex aids need not be more usable aids, see Norman, 2002.

[19] Especially so after long training and with appropriate modeling.

[20] Probably so, but it is unclear how many of the target groups could learn Morse Code and if this also applies for two-bit quartering combined with text-prediction. Also, for humans, to ‘read’ Morse Code seems harder than to code it so one would need a computer.

[21] Probably most stored text will be ignored, but not phrases that have been entered in the actual conversation and some significant stories. See Todman, 2000, Verrips 2000 and Verrips 2005.

[22] Faster recovery seems probable in some remittent conditions, as not being able to speak is a rather stressful situation for most people, therefore one would expect Augmentative and Alternative Communication AAC to be highly cost-effective if quality of life is taken into account.

[23] As in Verrips, 2003, with the author as his own test-subject and a single phrase repeated very often, to circumvent many, but not all, types of bias.

[24] Probably very little though it might well be more error prone and more tiresome.

Conclusion

Active two-bit quartering is an interesting alternative for all other forms of scanning.

Conflict of interest

Two-bit quartering is part of Htyp and Etyp, software engineered and distributed by the author.

Note

To download, to see a demo in Flash, or to do some more reading, please visit www.depratendecomputer.nl. Pictures were used with agreement from Toby Churchill Ltd, Cambridge, GB, who manufactures the Lightwriter.

References

Broecke, Marcel P.R. Ter sprake. Spraak als betekenisvol geluid in 36 thematische hoofdstukken, 1988. Reference at www.Onzetaal.nl/advies/letterfreq.php.

Card, S., Moran, T., en Newell, A. The psychology of human computer interaction. 1983. Erlbaum.

King, T.W. Modern morse code in rehabilitation and education. 2000. Allyn and Bacon.

Kopland, R. Geluk is gevaarlijk. Verzamelbundel, 1999, Rainbow pockets.

Koester, H.H. and Levine, S.P. 1996. The Effect of a Word Prediction Feature on User Performance. Augmentative and Alternative Communication, 12:3, 155-168.

Koester, H.H. and Levine, S.P. 1997. Keystroke-level Models for User Performance with Word Prediction. Augmentative and Alternative Communication, 13:4, 239-257.

Norman, D. 2002. The design of everyday things. Basic books.

Raaijmakers, J. Statistical Analysis of the Michaelis-Menten Equation. Biometrics, December 1987, pp 793-803.

Standage, T. 1998. The Victorian Internet. The remarkable story of the telegraph. Phoenix books.

Simpson, R., Koester, H.H., LoPresti, E., 2006. Selecting an Appropriate Scan Rate: The ".65 Rule". Proceedings of RESNA 2006 Annual Conference, Atlanta, GA. Arlington, VA: RESNA Press[25].

[25] Koester Performance Research > News/Publications on www.kpronline.com contains more interesting references.

Todman J. 2000. Rate and Quality of Conversations Using a Text-Storage AAC System: Single Case Training Study. AAC Vol 16, June 2000, 164-179.

Venkatagiri, H. S., 1999. Efficient keyboard layouts for sequential access in Augmentative and Alternative Communication. Augmentative and Alternative Communication, 15, 126–134.

Verrips J., 2000. Test of a communication aid with stored text. Int. J. of Rehabilitation Research. pp 169-175.

Verrips J., 2003. Study on Scanning Techniques. Full report. On www.depratendecomputer.nl > Documentatie > Onderzoek.

Verrips J., 2005. Rollenspelen met opgeslagen tekst. On www.depratendecomputer.nl > Documentatie > Onderzoek.

Wright, T.P., 1936. Learning Curve. Journal of the Aeronautical Sciences, February, 1936.