Int. J. Man–Machine Studies (1991) **35**, 379–397

The influence of interface style on problem solving

GUNNVALD B. SVENDSEN

Norwegian Computing Center, Postboks 114 Blindern N-0314 Oslo 3, Norway

(Received 9 November 1988 and accepted in revised form 25 February 1990)

According to a recent theory by Hayes and Broadbent (1988), learning of interactive tasks could proceed in one of two different learning modes. One learning mode, called S-mode, has characteristics not unlike what traditionally has been called "Insight learning". The other mode, called U-mode, is in some respects like trial and error learning. Extending the theory to human–computer interaction, it predicts different problem-solving strategies for subjects (Ss) using command and direct manipulation interfaces. Command interfaces should induce S-mode learning, while direct manipulation should not do this. The theory was supported by two experiments involving the tower of Hanoi problem. Ss with a command interface made the least number of errors, met criterion in the least number of trials and used the most time pr. trial. They were also more able to verbalize principles governing the solution of the problem than Ss using a direct manipulation interface. It is argued that the theory may explain the "feeling of directness" that goes with good direct manipulation interfaces. Further, the results indicate that user friendliness, as this is traditionally measured, in some cases may prove to reduce the users' problem-solving ability.

Introduction

It is well documented that user interfaces differ in user friendliness. Such differences are important, and empirical investigation of the factors contributing to them helps in understanding what makes computers more readily accessible to the general public. In this paper, however, we shall discuss the influence of the user interface from a somewhat different angle. The topic is whether interface style is capable of influencing the users problem-solving or learning strategy.

There is a traditional distinction between two types of learning; learning by trial and error and learning by insight. Trial and error learning is characterized by gradual reduction of the number of errors made during the learning period. Learning by insight on the other hand is characterized by a sudden leap from error prone to error free performance. Insight learning is somewhat like successful hypothesis testing. The subject carries out a conscious evaluation of the information in the problem situation. If he finds the right solution, he is not likely to make further mistakes.

The distinction between trial and error learning and learning by insight has a parallel in recent research. Broadbent has, through a series of experiments, produced evidence of two types of learning (Berry & Broadbent, 1984; Broadbent, FitzGerald & Broadbent, 1986; Hayes & Broadbent, 1988). Hayes and Broadbent (1988) argue that learning of interactive tasks may proceed either in what they call an U-mode or a S-mode. Learning in U-mode takes place outside working memory and is accordingly not verbally reportable. It seems to be "unconscious" in the sense

0020-7373/91/090379 + 19$03.00/0

that subjects (Ss) are not able to state what they have learned. It involves the "unselective and passive aggregation of information about the co-occurrence of environmental events and features" (Hayes & Broadbent 1988, p. 251). S-mode learning takes place in working memory and is available for verbal report. Ss know what they have learned and can state it verbally. It is "selective, effortful, and reportable, and is the type of learning which would normally be referred to as 'problem solving'" (op. cit. p 251).

The learning modes seem to be appropriate to two types of tasks. This is indicated in an experiment by Reber Kassin, Lewis and Cantor (1980). The experiment shows that when the task is characterized by high "saliency", i.e. if the important features of the task are easily perceived by the Ss, Ss instructed to look for rules perform better than Ss given no such instruction. However, they perform worse when the task has low saliency. Since conscious rule-seeking is characteristic of S-mode learning, the experiment indicates that S-mode learning is appropriate if the learning task is relatively simple and characterized by a high degree of saliency. On the other hand, if the task is complex and the saliency is low, U-mode learning is best suited.

An interesting and important feature of S-mode learning is that it can be induced by factors extraneous to the task, also in circumstances where it is not suited. As reported by Reber and Lewis (1977), merely asking Ss to introspect seems to switch them into S-mode learning. The success of the S-mode learning thus seems to depend upon the saliency (and complexity) of the task, while apparently it can be induced simply by a requirement to verbalize during the learning process.

The theory, as presented by Hayes and Broadbent (1988), is a general learning theory, and thus it should be valid in a situation where a user is to solve a problem by means of a computer. The result reported by Reber and Lewis (1977) gives an indication as to how the theory could be applied to human–computer interaction. If verbalizing during learning induces learning in S-mode, it is natural to assume that typing commands should do the same. In other words, using a command-driven interface should induce S-mode learning. In contrast, direct manipulation interfaces could be quite devoid of verbal content. If they are, such interfaces should not induce S-mode learning. Applying the theory in this way, the hypothesis is that command interfaces induce S-mode learning, while direct manipulation interfaces induces U-mode learning.

Saliency might also be given a straightforward interpretation in the domain of human–comuter interaction. The saliency of a user interface could reasonably be operationalized as the amount and type of feedback the interface offers. Interfaces that obscure, or hide, relevant aspects of the problem will have a lower saliency than an interface that does not do this. An interface of the MS-DOS type would consequently have low saliency, while an interface like the one in MacIntosh would have high saliency.

Given a salient presentation of a relatively simple problem, this predicts that Ss using a command-driven interface will learn in S-mode and consequently make fewer errors than Ss using a direct manipulation interface. However, since learning in S-mode is hypothesized to be demanding, Ss should use more time per learning trial than one using a direct manipulation interface. Second, Ss using a command-driven interface should more readily make a verbal account of their behaviour. More specifically they should be more able to state what they have learned and to give reasons for why they behaved in a given way. Third, a reduction in saliency should have a negative effect on the performance of Ss using a command-driven interface, both with respect to time used and errors made, while the performance of those using a direct manipulation interface should be unaffected or improved. Further, since S-mode learning is hypothesized to be effortfull while U-mode learning is effortless, it is predicted that Ss should prefer to use direct manipulation over a command-driven interface.

In summary, Hayes and Broadbents theory gives rise to five predictions when comparing a command-driven interface and direct manipulation interfaces. Ss using the command-driven interface should:

(1) make less errors;
(2) use more time per problem solving trial;
(3) be negatively affected by reduction in saliency; and
(4) be more able to state what they have learned.
(5) But Ss should prefer direct manipulation interface compared with the command interface.

The first three predictions concern problem-solving performance. The fourth prediction involves the Ss ability to state what they have learned, and thus possibly their ability to generalize their problem-solving behaviour to new situations. The last prediction touches upon the "user friendliness" issue. More precisely, it involves one of the traditional measures of user friendliness, namely subjective satisfaction.

The tower of Hanoi (ToH) puzzle is appropriate for testing this hypothesis, because it has been shown (Stinesen, 1975) that Ss may use both S-mode and U-mode strategies in solving the problem. Besides, the problem has been extensively studied in the problem-solving literature. The factors influencing Ss performance in solving the problem are therefore well known, making our study less susceptible to unknown factors.

Comparing preferences between a command and direct manipulation interface, poses some special problems in a study that concerns problem solving. In order to give Ss an opportunity to describe their preference, Ss should be acquainted with both interface types. Naturally, it is possible to use independent groups, where each group solves the ToH problem with one interface and thereafter rates the interface in terms of subjective satisfaction. Comparing those ratings, however, assume the highly unlikely event that Ss use the same subjective reference point and unit (see Torgerson, 1958). On the other hand, it is hard to find an experimental design that both assesses the influence of user interface on problem solving, and allows the Ss to use both interface types in the same situation. Two experiments have therefore been conducted. The first experiment was designed to test the first four predictions mentioned above, while the second was designed to assess the Ss preference for the command-driven *vs* the direct manipulation interface.

Experiment I

METHOD

The ToH problem involves a number of disks of graduating size and three positions on which the disks can be placed. At the outset all disks are arranged pyramidally on one of the positions, with the largest disk on the bottom. Call this position the "start" position. The task is to move all the disks to another position, the "goal" position, under the constraints that: (1) only one disk can be moved at a time; (2) a disk is not allowed to be placed on top of a disk smaller than itself, and (3) a disk may only be moved to one of the three positions. The minimum number of moves needed to solve the problem with n disks is given by $2^n - 1$. In the experiment five disks were used, thus the minimum solution required 31 moves.

The ToH problem was implemented with four different interfaces. Two interfaces being command-driven and two of the direct manipulation type. One of the interfaces of both types gave continuous feedback, (high saliency condition), while the other gave no visual feedback except when a special command was issued (low saliency condition).

This amounts to a 2×2 factorial design, with the factors: (1) Interface style, with levels (a) command interface, and (b) direct manipulation; and (2) saliency, with levels (a) high—with visual feedback, and (b) low—no visual feedback.

SUBJECTS

Thirty seven undergraduate psychology students were used. The subjects were assigned randomly to the four conditions, balancing sex over the conditions. Ss were paid an equivalent of approximately US $15. Only Ss without any experience or knowledge of the ToH problem were used. No explicit procedure was followed in order to establish Ss experience with use of mouse and keyboard. However, none of the Ss had any problems using the input mechanism they were assigned to.

MATERIAL AND APPARATUS

Figure 1(a) and (b) shows the displays in the two interface conditions with high saliency. The positions of the ToH problem were displayed as horizontal lines at the bottom of the screen. The lines were marked with the Norwegian equivalence to "1 (start)", "2 (help)", "3 (goal)", and they were displayed continuously. The disks were displayed as horizontal bars.

In the command interfaces the word "Command:" was placed at the bottom of the screen. To move a disk with the command interfaces, the Ss typed in the Norwegian equivalence to "from $\langle n1 \rangle$ to $\langle n2 \rangle$" and then hit the retun key, $n1$ and $n2$ being either 1, 2 or 3. For example, if Ss typed "from 1 to 3", the top disk in position 1 would move to position 3, and position itself centrally on top of any disks there. If the interface gave visual feedback, the Ss could observe these movements. If the Ss wrote a command that was not understood by the program, the program responded with an error message telling the Ss that the command was not admissible. No disk was moved in that event.

In order to move a disk in the direct manipulation interface with visual feedback, Ss pointed to the disk to be moved with an arrow, by moving a mouse. When the arrow was inside the disk, Ss depressed a mouse button. Holding the button down, Ss could drag the disk to the position of his choice. When the mouse button was released the disk dropped and placed itself centralized on top of any other disks on that position. It was not possible to move disks other than those on the top.

To move a disk with the mouse without visual feedback, the Ss pointed at the screen above the area from which the disk was to be moved, pressed the button and

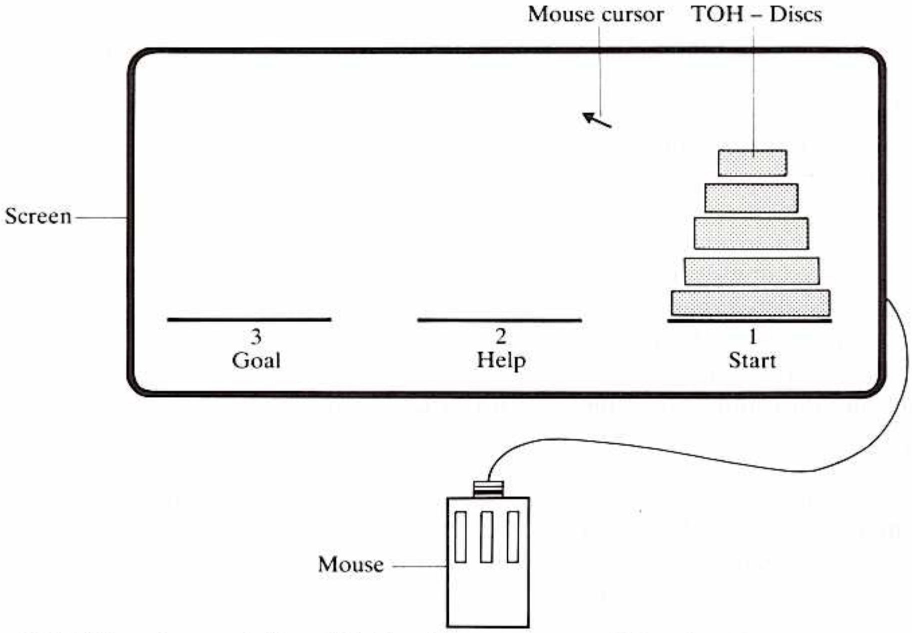

FIGURE 1(a). Schematic presentation of the interface to the tower of Hanoi problem when Ss uses direct manipulation with visual feedback.

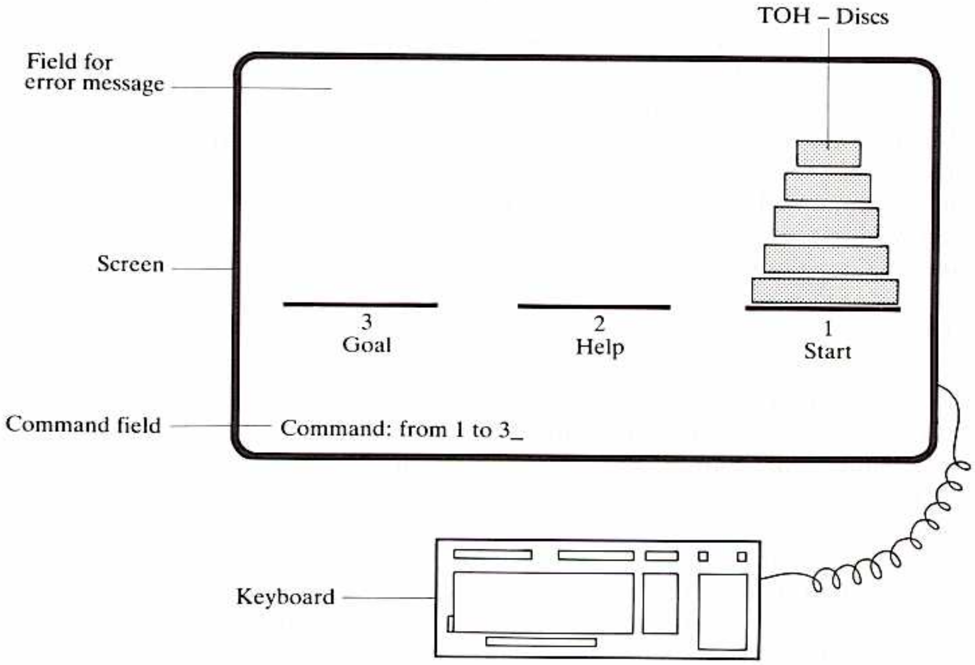

FIGURE 1(b). Schematic presentation of the command-driven interface to the tower of Hanoi problem.

moved the arrow to the position where the disk was to be placed, then released the button. To indicate that a disk was "attached" to the arrow, the computer gave a low sound while the disk was being moved.

In the two non-feedback interfaces a display of the disks could be produced by typing the command "show" (command interface) or pressing the rightmost button on the mouse (direct manipulation interface). Then the display screen was active until a character was typed or the button was released.

If the Ss tried to make a unlawful move, for example placing a larger disk on a smaller, the computer gave a short sound and moved the disk back to its last position. This also happened if the Ss tried to make a move that deviated from the shortest path to the solution.

The program was implemented on a IBM compatible AT computer with EGA graphics monitor and optical mouse with three buttons.

PROCEDURE:

An instruction was read to the subjects stating that they were to participate in an experiment concerning human–computer interaction. They were told approximately how long the experiment would take and that they, in the course of the experiment, were to solve a problem. They were then shown a ToH setup consisting of five wooden disks on a cardboard with three positions. The positions were marked as displayed on the computer screen. The task and the conditions under which the disks were permitted to be moved was explained. They were then presented with the computer implementation of the problem, and shown how to move a disk. They were told that the problem was to be solved more than once, and with as few errors as possible. In the two non-feedback groups the Ss were told not to issue the show command (or press the right button) unless they had lost track of the disks.

When all the disks were moved from the start position to the goal position, the problem was repeated if any of two conditions were not met: the problem was solved twice without any errors, or when S had made 20 trials.

After criterion was reached Ss were asked to state if they had discovered any rules that applied to solving the problem. They were also asked how they would instruct a person not aquainted with the problem so as to help him solve it in the most efficient way. Ss were classified as being able to verbalize rules if they were able to state a solution involving break down of the problem into subgoals.

RESULTS

Figure 2 shows the mean number of trials to criterion in the four conditions. ANOVA gave a significant effect of interface style ($F(1,33)=13{\cdot}744$, $p<0{\cdot}001$), but no effect of saliency and no interaction effect. Figure 3 shows the mean number of errors made in the four conditions. The pattern was the same; a significant effect of interface style ($F(1,33)=11{\cdot}477$, $p<0{\cdot}05$), but no effect of saliency and no interaction effect.

These results comply with the prediction that Ss, using the command-driven interface, should make the least errors and the least number of trials. Thus corraborating the hypothesis that command-driven interfaces induce S-mode learning while direct manipulation does not.

An interaction between interface style and saliency was predicted: there should

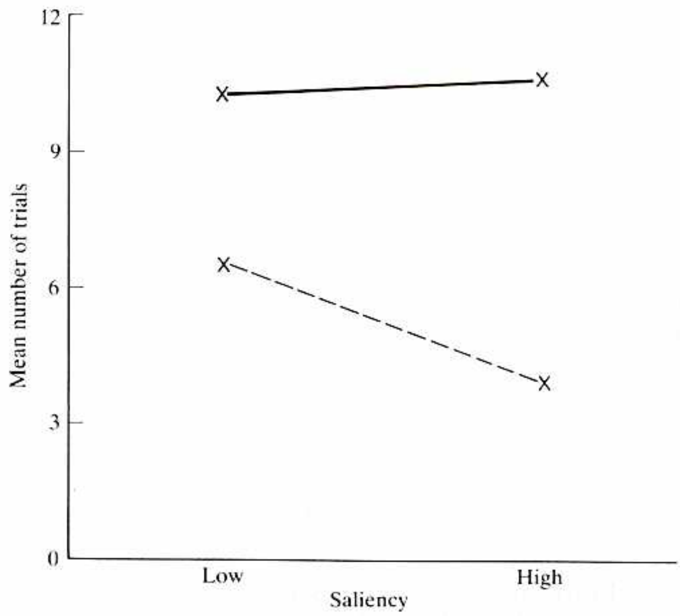

FIGURE 2. Mean number of trials to criterion in the four conditions. Key: —, direct manipulation; – – –, command.

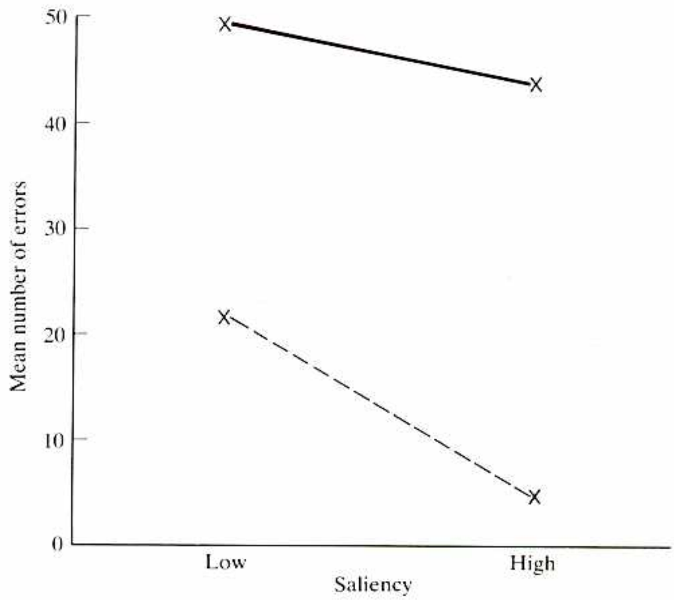

FIGURE 3. Mean number of errors over all trials in the four conditions. Key: —, direct manipulation; – – –, command.

be a deterioration of the performance in the command × low saliency condition compared with the command × high saliency condition, without the same deterioration in the direct manipulation groups. This interaction failed to reach significance. The trend was, however, as predicted in both figures.

Figure 4 shows the mean time spent in solving the task in the four conditions. Here, both interface style, saliency and the interaction effects were significant ($F(1, 33) = 5{\cdot}497$, $p < 0{\cdot}05$; $F(1, 33) = 10{\cdot}662$, $p < 0{\cdot}05$; $F(1, 33) = 6{\cdot}023$, $p < 0{\cdot}05$, respectively).

While both main effects were significant, it is reasonable to assume that this results from the high score of the command × low saliency condition. The expected deterioration is clear in this figure.

The last prediction regarding performance concerned the time used per trial in the four conditions. Since S-mode learning demands more effort than U-mode learning, it was expected that Ss in the command conditions would use more time per trial than those in the direct manipulation condition. A comparison between the time used per trial would, however, tend to confirm this hypothesis since the two interface styles differ greatly in the amount of time they require the Ss to use in order to carry out a move. In the command condition, Ss had to push 10 keys to make a move. On average Ss used 6·2 s each move to make a command. In the direct manipulation condition Ss used 1·1 s to make a move. In calculating average time used per trial (see Figure 5), the time used to move a disk was therefore subtracted from the time spent on making a move.

ANOVA revealed that both main effects were significant ($F(1, 33) = 16{\cdot}658$, $p < 0{\cdot}001$; $F(1, 33) = 8{\cdot}580$, $p < 0{\cdot}05$), for interface style and saliency, respectively.

In summarizing the results regarding performance (trials, number of errors and

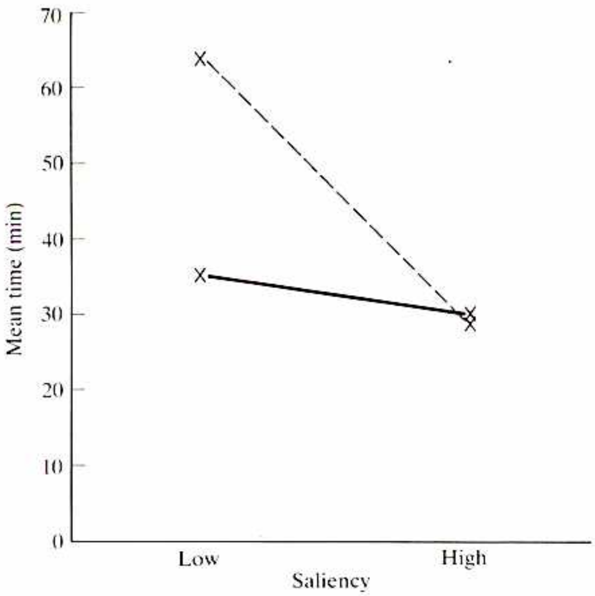

FIGURE 4. Mean time (in min) to reach criterion in the four conditions. Key: —, direct manipulation; – – –, command.

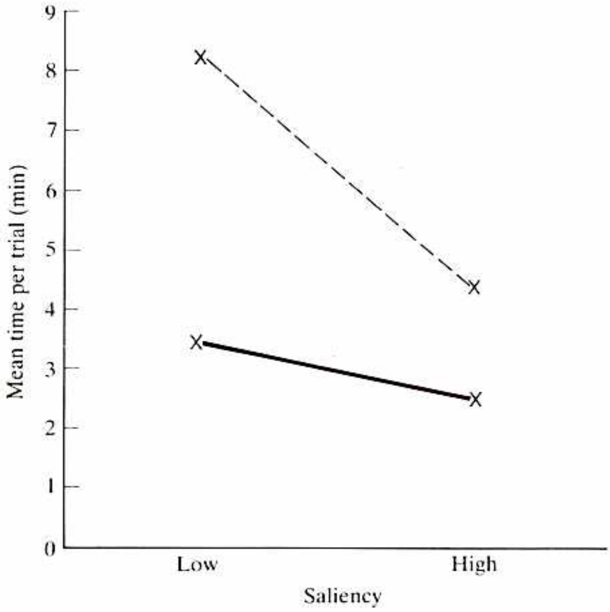

FIGURE 5. Mean time to perform one trial. The time is corrected for difference in time used to make a move. Key: —, direct manipulation; – – –, command.

time used), support was found for the hypothesis that command-driven interfaces induce S-mode learning, while direct manipulation interfaces do not. In the command × high saliency condition, Ss reached criterion with the fewest number of trials, made fewer errors and used least total time. Further, while there was only a marginal difference between high and low saliency conditions when using direct manipulation, this difference was clear given a command-driven interface. Last, clear evidence was found that Ss in the command conditions used the most time per trial, indicating a more demanding problem-solving strategy.

The hypothesis predicts that Ss in the command conditions would show a greater ability to verbalize principles that lead to a solution of the problem than those in the direct manipulation conditions. Figure 6 shows the proportion of Ss able to state principles for solving the problem in the four conditions.

As the figure shows, there was evidently no interaction effect between the conditions. The effect of interface style was therefore tested by Fisher's Chi^2 test for independence of samples, giving $\chi^2 = 4{\cdot}659$, $p < 0{\cdot}05$. By the same procedure, saliency failed to reach significanse at the 0·05 level. These results further support the hypothesis that command-driven interfaces induce S-mode learning, while direct manipulation interfaces do not.

DISCUSSION

An alternative explanation of the present results could be that there is a trade-off between subjective cost of making a move and the time used on a move. It is reasonable to assume that Ss in the command conditions would experience a higher cost associated with making a move than Ss in the direct manipulation conditions, thus explaining the greater amount of time used per move in these conditions.

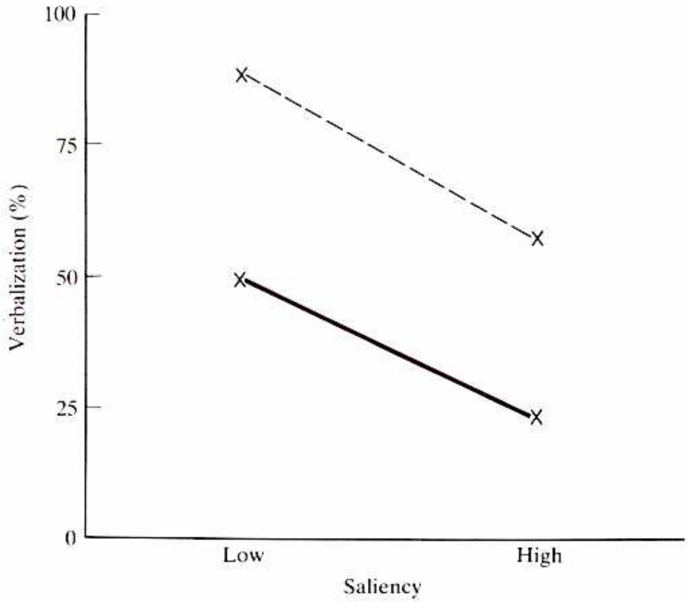

FIGURE 6. The percentage in each condition being able to verbalize a rule governing the solution of the problem. Key: —, direct manipulation; – – –, command.

Further, given that the probability of making a correct move is proportional with the time used on that move, the trade-off hypothesis would also explain the differences between command and direct manipulation conditions with respect to number of errors made.

The trade-off hypothesis is, however, insufficient in one respect. It cannot explain why the Ss in the command × low saliency condition, who use by far the most time, make about five-times as many errors as the Ss in the command × high saliency condition. The learning mode hypothesis, on the other hand, predicts these results. It thus seems reasonable to retain the more complex of the two hypothesis.

Experiment II

METHOD

The hypothesis to be examined in this experiment was whether the direct manipulation interface was prefered to the command-driven interface. Consequently, the saliency factor was dropped from the design, leaving two interfaces: direct manipulation and command, both with visual feedback. To give Ss experience with the interfaces, Ss solved the ToH problem three-times with each interface, changing interface after each trial. Since it is not possible to rule out bias due to recency and/or primacy effects, two groups were employed. One group ("Com–Dir") started with the command interface and ended with the direct manipulation interface, thus following a c–d–c–d–c–d pattern. The other group ("Dir–Com") followed the reversed pattern: d–c–d–c–d–c.

Preferences were assessed twice during the experiment: after trial two, and after the last trial (trial six).

SUBJECTS

Fifteen undergraduate psychology students were used. Ss were selected, screened, paid and assigned to groups in the same way as in experiment I.

MATERIAL AND APPARATUS

In order to assess preference for command or direct manipulation interface, a short interview guide was constructed. The guide consisted of six questions. The first three questions concerned different aspects of subjective satisfaction:

(1) What interface would Ss prefer if they were to use the interface over a longer period of time?
(2) Which interface would Ss recommend a friend to use if he were to participate in the experiment?
(3) Which of the interfaces was most easy to start using?

It was assumed that the responses to these questions would be highly correlated. The hypothesis predicts a high rating in favour of the direct manipulation interface on all the questions.

The last three questions were of a exploratory nature, and concerned how the Ss regarded their own performance:

(4) Which interface thought the Ss were the fastest to use?
(5) Which interface the Ss thought they made most errors in using?

The learning mode hypothesis predicts that Ss using the command interface should be most aware of their performance. However, it is difficult to predict how this would influence their rating of the two interfaces. No predictions were therefore given.

The last question was also somewhat exploratory. The Ss were asked to consider whether they had approached the task, or thought about the task, in a different manner when they used the direct manipulation and the command interfaces. It would be suprising if Ss could not detect a difference between learning in S- and U-mode, especially since S mode is characterized by a conscious effort, while U mode is not. Given that Ss switch between learning modes as they change interface, we would therefore predict that Ss report a difference in the way they think about, or approach, the problem when they use the two interfaces. However, it was difficult to predict how Ss would describe this difference. No effort was therefore made to predict what Ss verbal reports would contain. Nor was there any strong reason to believe that the verbal reports would fall into categories. This question was consequently asked open-ended.

Apart from the interview guide, the material was the same as in Experiment I.

PROCEDURE

The procedure deviated slightly from that of Experiment I. Ss were given the same general introduction to the ToH problem and the experiment. Ss were also given the same introduction to the computer implementation of the problem, but in this experiment Ss were given this instruction twice, once for each of the two different interfaces. This instruction was given the first time they used the relevant interface. Ss were not told that they would be interviewed during and after the experiment.

After trial two and the last trial (trial six), the experimenter administered the interview guide. The Ss answers were written down by the experimenter.

RESULTS

No significant differences were observed between the two groups, "Com–Dir" and "Dir–Com", neither regarding preferences nor on performance as measured by number of errors made. All Ss were therefore treated as coming from the same population.

Figure 7 and 8 show the results from the first and second assessment of preference. The figures indicate that most Ss preferred the direct manipulation interface. They would recommend this interface to others, they consider it the most easy to use, and also found it the fastest. In order to reach significance on the 5% level in a one-tailed test, 12 out of 15 Ss have to choose one of the alternatives ($p(n \geq 12) = 0{\cdot}02$; $p(n \geq 11) = 0{\cdot}06$). For "prefer", "recommend" and "ease", the tendency was therefore not significant in the first assessment. On the second assessment "prefer" and "ease" reached significance, while "recommend" did not. However, the overall picture was clear—Ss tended to prefer the direct manipulation interface, and the results were consequently taken to comply with the prediction that Ss would prefer the direct manipulation interface to the command-driven interface.

After one trial with each of the interfaces, 11 out of 15 Ss reported that they made more errors with the command interface than with the direct manipulation interface (Figure 7). In the second assessment, some Ss revised their judgement: eight out of 15 Ss indicated that they made most errors using the command interface (see Figure

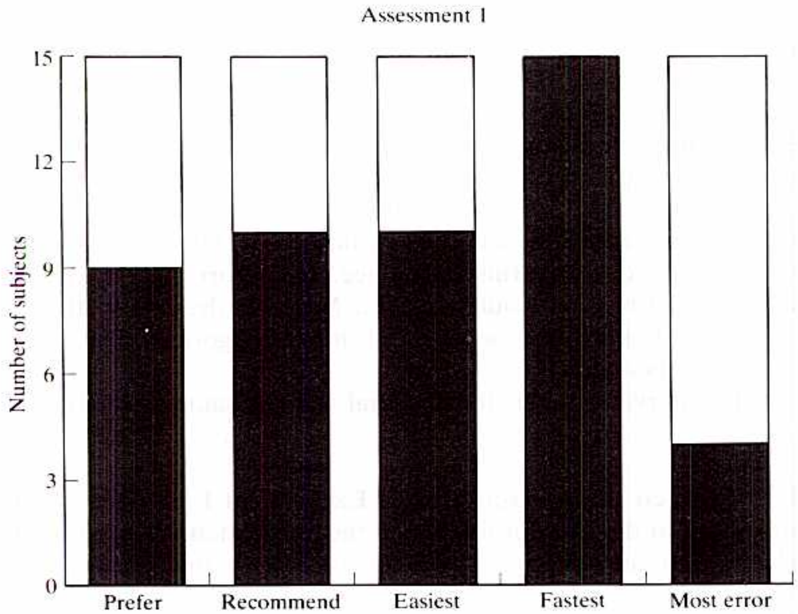

FIGURE 7. Subject's evaluation of the interfaces after one trial with each of the interfaces. Key: ■, direct manipulation; □, command.

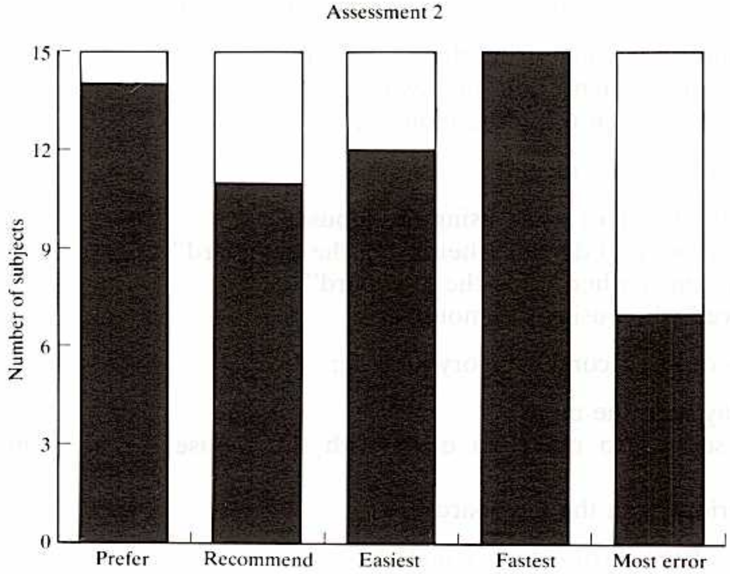

FIGURE 8. Subject's evaluation after all six trials. Key: ■, direct manipulation; □, command.

8). Taken at face value this result throws doubt on the results from Experiment I. However, quite another picture emerged when the errors Ss actually made were analysed. Comparing the number of errors a S made on trial one and two revealed that only one S made most errors with the command interface, three ties were observed, and 11 Ss made most errors with the direct manipulation interface. Comparing the total number of errors a S made on the three trials using the command interface with the total number of errors he made the three times he used the direct manipulation interface gave exactly the same result. Excluding ties from the analysis, a sign test reveals that Ss made significantly more errors when they used the direct manipulation interface ($p < 0{\cdot}01$). Figure 9 shows the mean number of errors made in the groups "Dir–Com" and "Com–Dir". The figure clearly indicates that Ss made the most errors when they used the direct manipulation interface.

Given this result is is difficult to disregard the conclusion that Ss are very poor judges of their own performance when it comes to errors. However, the results also indicate that given prolonged exposure to an interface, the true state of affairs slowly dawns on them. This stands in contrast to Ss ability to judge the speed of their performance with the two interfaces. All Ss indicated that the direct manipulation interface were the fastest to use. This proves to be correct when compared with the log from the experiment.

The last question posed to the Ss concerned whether they thought about, or approached, the ToH problem differently when they used the two interfaces. Only three Ss indicated no difference between the two interfaces in both the first and the second assessment. The proportion of Ss indicating a difference deviates significantly from chance ($p < 0{\cdot}05$), thus supporting the hypothesis.

A clear pattern was found in the verbal reports from the 12 Ss that indicated a difference. Apart from one S, the Ss reports could be classified in three categories:

(1) "Thinking more when using the keyboard";
(2) "More serious when using the keyboard";
(3) "More visual when using the mouse".

Examples of the first include:

- "More trail and error when using the mouse"
- "Thought more and deeper when using the keyboard"
- "Thought longer when using the keyboard"
- "Just moved when using the mouse"

Examples of the second category include:

- "More play with the mouse"
- "Not so serious to make an error with the mouse, to hit "enter" gives the judgement"
- "More serious with the keyboard".

Examples of the third category include:

- "More visual when using the mouse"
- "Don't look so much at the screen when using the keyboard"
- "More abstract when using the keyboard"

One S deviated from this pattern, he indicated that he thought more when he used the mouse. It should be mentioned that this S was identical to the one that made most errors using the command interface.

Although verbal reports should be interpreted with caution, it is evident that the majority of Ss perceived that they approached or thought about the ToH problem differently, dependent upon which interface they used. Eleven of the twelve Ss reporting such a difference characterized their problem solving with the command interface as serious, abstract and thoughtful. When characterizing their problem-solving with the direct manipulation interface, they used terms like playful, visual and trial and error-like. The Ss verbal reports thus coincide with the hypothesis that the command interface induces S-mode learning while direct manipulation induces U-mode learning. Although no predictions were made concerning the content of the verbal reports, this result was taken as independent support for the hypothesis.

DISCUSSION

Although the results from Experiment II, comply with the learning mode hypothesis, two aspects of the results need further discussion: (1) why were the Ss more in favour of the direct manipulation interface after assessment two, than after assessment one? (2) Why were the Ss unable to give a correct assessment of their performance regarding errors, when they were able to judge their performance in relation to speed of performance?

Figure 9 shows that the greatest difference between the interfaces appear in trial one and two. After the second trial Ss made only slightly more errors with the direct manipulation interface. It could be tempting then to conclude that on Assessment 1 Ss knew that they had made the most errors with the direct manipulation interface,

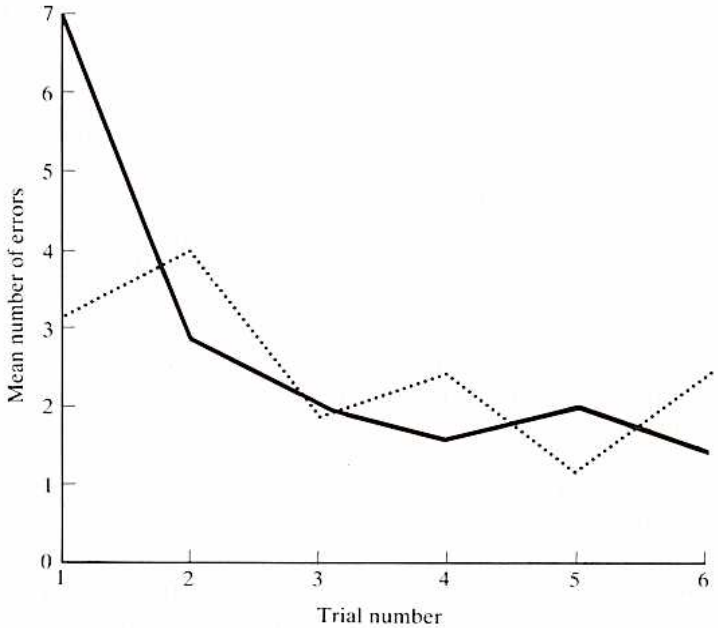

FIGURE 9. Mean number of errors made in each trial for the two groups "Dir–Com" and "Com–Dir". Key: —, Dir–Com; – – –, Com–Dir.

and consequently were reluctant to indicate that they would prefer this interface. By the same token, on Assessment 2, the difference in errors were not apparent to them, and they therefore indicated direct manipulation as the prefered interface. This explanation is, however, contrary to the fact that Ss thought they had made most errors with the command interface, and even more of them thought so in the first assessment than in the second. A more straightforward explanation is that the Ss were inexperienced with the mouse device, and thus experienced some start-up problems. Unfortunately, no independent measure is available to test this explanation, since no test of proficiency with mouse and keyboard were administered before the experiment. This explanation, however, is incompatible with the Ss report that they experienced fewer start-up problems with the direct manipulation interface. Thus, the only viable option left open seems to be to take the Ss report at face value: they liked the direct manipulation interface better than the command interface after some exposure to the interfaces, and the more experience they had with it, the more they liked it.

We could also attempt to answer the second question in a straightforward manner. The command and direct manipulation interfaces differ in one respect when it comes to the possibility for making errors. In the command interface, Ss could write a command that was not understood by the program. If that happened, the program responded with an error message telling the Ss that the command was not admissible. In the direct manipulation interface, no possibility existed for making such an error. Consequently, the error message never appeared. If the Ss equated seeing an error message with making an error, it is natural that they thought they made most errors with the command interface. However tempting this explanation is, it is contrary to the task the Ss were given. When Ss were asked to state with which interface they made the most errors, it was pointed out to them that the question concerned errors in moving disks, not errors in typing. A second explanation could be that the Ss verbal report of the errors they made merely reflected which interface they liked best. However, this explanation is at odds with the fact that the trend in the preference measures is opposite to the trend in the error judgements: more Ss stated that they made the most errors with the command interface in the first assessment rather than the second, while more subjects preferred the direct manipulation interface in the second rather than the first assessment.

A third and more interesting explanation is that Ss were more aware of the errors they made when they used the command interface. According to the learning mode hypothesis it should be expected that Ss using the command interface have a reason for the moves they make. When the move proves to be wrong, it also becomes evident that their hypothesis must be wrong. They must therefore make a conscious effort to revise their hypothesis. Based on the learning mode hypothesis, one should expect that Ss using the direct manipulation interface did not form conscious hypothesis as to what was the correct move. Accordingly, Ss using direct manipulation would not need to stop and consider the problem afresh when a wrong move was made, they just made another move. If this is the case, it is not surprising that Ss become more aware of the errors they make when using the command interface. Thus, the Ss misconception as to when they made most errors is explicable in terms of the learning mode hypothesis.

GENERAL DISCUSSION

ToH tasks are highly unlikely to be of importance in the daily work of any computer user. The question therefore arises as to how the results should be generalized. The differences between the interfaces can probably be generalized to problems sharing the same complexity and saliency as this problem. Further, it is reasonable to generalize the conclusion that command interfaces induce S-mode learning while direct manipulation interfaces induce U-mode learning. Extending the results in this way, they may prove to have both theoretical and practical relevance.

Theoretically, the results tie in with Hutchins, Holland and Norman's (1986) question of what the cognitive basis for the "directness" of direct manipulation systems is. They suggest that the answer is to be found in two aspects of direct manipulation interfaces: "distance" and "engagement". The concept "distance" refers to the ". . . distance between one's thoughts and the physical requirements of the system" (op. cit., p. 93). "Engagement", on the other hand, is concerned with the ". . . feeling that one is directly manipulating the object of interest" (op. cit., p. 94). They go on to suggest that the "directness" of direct manipulation interfaces is a result of small distance and high engagement.

The present results make it possible to suggest an alternative answer to the Hutchins *et al.* (1986) question. The feeling of directness, and the accompanying high degree of user friendliness, may be due to the interfaces permitting the user to learn and perform in U-mode. To put it bluntly, it may be that interaction, without the necessity of concious thought, produces this feeling of directness.

On the practical side, the results pose both a question of the usefulness of direct manipulation interfaces, and, generally, a question of whether the quest for user-friendly software hides some potentially important aspects of the interfaces impact on the user.

It is safe to conclude that direct manipulation with visual feedback was the most user-friendly interface in this study. Experiment II shows that this interface is both preferred and regarded as the most easy to learn by the Ss. Both Experiments I and II show that this interface is the fastest to use. On the other hand, it is also safe to conclude that this interface is not preferable if the goal is effective and error-free problem solving. The results indicate that direct manipulation may hinder effective problem solving in tasks with the same complexity and saliency as a tower of Hanoi puzzle. Hinder, not in the sense that the interface is not user friendly, but to the contrary. The interface is so supportive of thoughtless action that the user neglects to look for rules where these are called for. Thus, the experiments indicate a discrepancy between the user friendliness and the usefulness of direct manipulation interfaces.

This conclusion is at odds with the widespread notion that direct manipulation interfaces are unequivocally good. But, if we look at the literature, the conclusion is not contrary to what is reported, (though it is not supported by it either). Some attention has been given to the relative merit of direct manipulation compared with other interface types. (See Whiteside, Jones, Levy & Wixton, 1985; Card, Moran & Newell, 1983; Marongo & Shneiderman, 1987). Whiteside *et al.* (1985) report no effect of interface style, while both Card *et al.* (1983) and Marongo and Sheiderman (1987) report beneficial effects of direct manipulation both when it comes to performance and subjective satisfaction. It is reasonable to assume that the difference between the present study and those mentioned is due to the fact that different things are measured. Those studies investigated learning and performance in relation to a relatively complex program, consequently the task was kept simple. The emphasis in the present study is the opposite: making performance and learning a complex task, while keeping the learning of the interface simple. The difference may therefore be more of appearance than of content.

The mentioned studies investigate commercially available software products, making it difficult to compare those studies with the one reported here. Such products have so much functionality, special features and idiosyncrasies, that it is hard to single out what aspect of the program causes an effect. As Landauer (1987) says: "What one (will) have tested is 'one mess against another'. So many differences will exist between one system and another that the features or factors to blame or credit will be obscure" (op. cit., p. 21). On these grounds it is reasonable to say that the cited studies do not contradict the conclusion that direct manipulation interfaces in some circumstances produce suboptimal problem solving.

It is, however, beyond doubt that direct manipulation interfaces are easy to learn, are often preferred by users and, according to the learning mode hypothesis, are best suited when highly complex tasks are performed. It is therefore natural to speculate on the possibility of tailoring the interface to the users problem space. Some tasks could have a command interface, others a direct manipulation interface, while others again had some kind of hybrid interface. However, such speculations are somewhat premature, since too little is known both when it comes to characterization of tasks and how to induce the different learning modes.

The second question posed above concerns whether the concept of user friendliness makes us overlook important aspects of the interfaces function. "User friendliness" is best described as a conglomerate of measures of how the user learns and performs in relation to a program's user interface. These measures include: time to learn, number of errors made, speed of performance, retention over time and subjective satisfaction (Shneiderman, 1987). How these measures are obtained will differ somewhat from study to study, but usually they stem from a set of benchmark tasks designed to reveal difficulties with the presentation and use of commands and functionality in the program. In short, when a program is characterized as user friendly, it entails that users are able to learn the functionality of a program fairly quickly, are able to use this functionality, and like using the program.

However, based solely upon the measures of user friendliness, is there any reason to believe that a user-friendly interface will make the user more able to solve problems in the domain the program is to be used? Will budgets made with a user-friendly spreadsheet be more accurate than those made with a less user-friendly program? Do we have any reason to believe that reports written with a user-friendly text editor are better structured and more thought provoking? No, based solely on a measure of user friendliness it is not possible to make any conclusions as to the users problem-solving performance. To make such a conclusion, one has to make an assumption regarding the relation between user friendliness and problem-solving performance.

The wisespread assumption that a user-friendly interface makes better problem-solvers than less user-friendly interfaces seems to be based on one of two assumptions: either that using the interface and solving the problem compete for the same cognitive resources, or that a user-friendly interface somehow facilitates problem solving. The first seems to be advocated by Shneiderman (1987), Hutchins *et al.* (1986) and Norman (1987).

The results reported here indicate that neither of these assumptions are warrantable. User friendliness, at least the subjective part of it, seems to be tied to a learning mode (U-mode) that in some situations produces suboptimal problem solving. If this proves to be the case, the pursuit of user-friendly interfaces could have negative consequences in some cases.

It is necessary to exert some caution when the concept of user friendliness is discussed in the light of the present results. One should remember that the application of the learning mode hypothesis is based upon the assumption that verbalization induces S-mode learning. Accordingly, two interfaces that require the same degree of verbalization should not differ with regard to problem solving or user friendliness. On the basis of this hypothesis, it is therefore not possible to explain differences in performance and subjective satisfaction between programs that require the same amount of verbalization. Doubtless, such differences do exist. Consequently, the mode in which the user learns should be looked upon as one of many factors determining the user friendliness of a program.

References

Berry, D. C. & Broadbent, D. E. (1984). On the relationship between task and associated verbalizable knowledge. *The Quarterly Journal of Experimental Psychology*, **36a**, 209–231.

BROADBENT, D. E., FITZGERALD, P. & BROADBENT, M. H. P. (1986). Implicit and explicit knowledge in the control of complex systems. *British Journal of Psychology*, **77**, 33–50.
CARD, S. K., MORAN, T. P. & NEWELL, A. (1983). *The Psychology of Human–Computer Interaction*. Hillsdale, NJ: Lawrence Erlbaum Ass.
HAYES, N. A. & BROADBENT, D. E. (1988). Two models of learning for interactive tasks. *Cognition*, **22**, 249–275.
HUTCHINS, E. L., HOLLAND, J. D. & NORMAN, D. A. (1986). Direct manipulation interfaces. In NORMAN, D. A. & DRAPER, S. W., Eds. *User Centered System Design*, Hillsdale, NJ: Lawrence Erlbaum Ass.
LANDAUER, TH. K. (1987). Relations between cognitive psychology and computer system design. In CARROL, J. M., Ed. *Interfacing Thought: Cognitive Aspects of Human–Computer Interaction*. Cambridge, UK: The MIT press.
MARONGO, S. & SHNEIDERMAN, B. (1987). A study of file manipulation by novices using commands vs. direct manipulation. In *26th Annual Technical Symposium*, 11 June. Gaithersburg, MD: ACM.
NORMAN, D. A. (1987). Cognitive engineering—cognitive science. In CARROL, J. M., Ed. *Interfacing Thought: Cognitive Aspects of Human–Computer Interaction*. Cambridge, UK: The MIT press.
REBER, A. S. & LEWIS, S. (1977). Toward a theory of implicit learning: the analysis of the form and structure of tacit knowledge. *Cognition*, **5**, 333–361.
REBER, A. S., KASSIN, S. M., LEWIS, S. & CANTOR, G. (1980). On the relationship between implicit and explicit modes of learning a complex structure. *Journal of Experimental Psychology: Human Learning and Memory*, **6**, 492–502.
SHNEIDERMAN, B. (1987). *Designing the User Interface. Strategies for Effective Human–Computer Interaction*. Reading, MA: Addision-Wesly.
STINESEN, L. (1975). Conditions which influence acquisition and application of verbal representations in problem solving. *Psychological Reports*, **36**, 35–42.
TORGERSON, W. S. (1958). *Theory and Method of Scaling*. New York: John Wiley & Sons.
WHITESIDE, J., JONES, S., LEVY, P. & WIXTON, D. (1985). User performance with command, menu and iconic interfaces. In *Proceedings of the CHI '85*. New York: ACM.