ELECTRONIC WORKSHOPS IN COMPUTING

Series edited by Professor C.J. van Rijsbergen

C.R. Roast and J.I. Siddiqi, Sheffield Hallam University, UK (Eds)

BCS-FACS Workshop on Formal Aspects of the Human Computer Interface

Proceedings of the BCS-FACS Workshop on Formal Aspects of the Human Computer Interface, Sheffield Hallam University, 10-12 September 1996

Formal Methods and Human Communication

K. Stenning and C. Gurr

Published in Collaboration with the
British Computer Society

Formal Methods and Human Communication

Keith Stenning

Corin Gurr

Human Communication Research Centre, University of Edinburgh*

Edinburgh, UK

Abstract

“Which picture is worth which 1000 words?” The best known theories approach this question through explanations in terms of human computational architecture. So, for Larkin & Simon [7], diagrams are efficacious because they allow the parallel computational power of human vision to be brought to bear on diagrammatic representations (at least in some cases).

A more fundamental approach separates the problem into two parts—issues about computational complexity arising from the nature of the semantic interpretation (issues which are abstract with regard to architecture); and issues about how human computational architecture in particular can be brought to bear on different representations. On this view, diagrams are often logically inexpressive and this is why they lead to efficient inference. Indeed, this is part of the reason why parallel computational mechanisms may be applicable to them.

This paper reviews experiences of applying this semantic approach to the empirical study of modality assignment in disparate domains (logic teaching, safety critical software engineering and the teaching of formality) and draws together some conclusions concerning the multifarious implications of formality for HCI.

1 Introduction

For several years now, a group of researchers at HCRC have been exploring a program of applying the concept of expressiveness, as developed from logic and computational complexity theory, to the problem of understanding human reasoning and communication, especially the contrasting properties of different modalities of representation. We have been rather eclectic in our choice of problems to study because we think there are advantages in pursuing the same theoretical idea across quite different problems. The aim of this paper is to describe some examples drawn from this program to give what may be a rather different perspective on possible roles for formal approaches in HCI. Under a narrow interpretation, these studies might not be thought of as HCI at all. But they all have implications for the symbiosis between humans and their computer systems.

Mackinlay [8] has popularised a distinction between *expressiveness* and *effectiveness*. The expressiveness of a representation is given by its semantic interpretation; its effectiveness is determined by users’ abilities to exploit the information it expresses. On this view expressiveness is very much the stuff of the computer science of representations and effectiveness is the concern of the psychologist. HCI clearly straddles both of these concerns.

A different way of interpreting the distinction is that expressiveness (as spelt out in computational complexity theory) determines the complexity of inference without regard to the particularities of the processor which makes the inferences. Effectiveness is taken to be determined by the specifics of human computational architecture (among other determinants). But this way of putting it has the advantage that it is clear that expressiveness will also have its effect on the human processor, because its generalisations apply to *any* architecture. In fact, we will argue, expressiveness is a rather neglected but substantial determinant of the usability of representations by people.

The baldest expression of this hypothesis is that graphical representations are generally (though not necessarily) inexpressive (in the technical sense) and that this is what plays a major role in determining their usability. Generally

*The support of the Economic and Social Research Council UK (ESRC) is gratefully acknowledged. The work was partly supported by the ESRC funded Human Communication Research Centre. We also wish to acknowledge partial support from the following grants: Grant # 910954 from the NATO; and from Grant # GR/J586129 from the SERC/DTI.

speaking, the most weakly expressive system of representations which is nevertheless expressively adequate for the task at hand will be the best.

This hypothesis may be old and vague, but it nevertheless is well enough specified as an approach to the differences between graphical and linguistic representations to contrast with the best known account of these differences in Larkin & Simon [7]. Their approach sees the advantage of graphics in the greater computational power of the human visual system (its particular architecture) when that power can be applied to the task at hand. Our hypothesis sees the advantage as lying in the fact that in the right circumstances, less computation has to be performed (by a device of any architecture). Our approach stresses the semantic properties rather than the visual ones. In the end these two hypotheses must be synthesised. The reason why, for example, the particular parallel architecture of human spatial processing can be brought to bear on graphical representations may be precisely because they are weakly expressive. Be that as it may, the importance of weak expressiveness for psychological understanding has received too little emphasis.

This paper will illustrate some of the results and the difficulties of applying this program to the study of interactions between people and representations. Many of the representations will have realisations in machines. The plan of the paper is as follows. In the next section, a study of people learning/teaching logic through machines as an illustration of the advantages and problems of this approach is briefly reviewed. This study casts light both on the properties of representations which determine differences between modalities, and also determine differences between peoples' reactions to them. Section 3 will describe a study of related differences between graphical and algebraic specification 'languages' in software engineering. Section 4 views formality from the perspective of the learner asking what it is that a student has to learn when learning elementary logic? Learning formality is both an important issue for users of computers and also for the dissemination of formal methods amongst the communities which design computer systems. Finally Section 5 draws together some conclusions about these multifarious implications of formality for HCI.

2 Graphics and language in teaching logic

Stenning, Oberlander & Cox [14] report a study of Stanford undergraduate learning first order logic from Hyperproof [1], a multi-modal environment, as compared with a control class learning from the natural-deduction-sentential-only part of the Hyperproof program.

Although Hyperproof graphics are rather inexpressive relative to the predicate calculus, some 'abstraction tricks' have to be used in order to enable the program to display a manageable number of graphical representations for classes of possible models. Figure 1 shows the Hyperproof interface. The cylinders visible on the chequerboard stand not for cylinders but for polyhedra from the Hyperproof domain. Their question marks can be replaced by badges indicating shape (but still not indicating size). Alternatively, 'paper bag' icons can mark the position of polyhedra, revealing their size but not their shape.

Our interest was originally drawn to such devices by study of misinterpretations of Euler's Circles. This classical graphical system is of little use in reasoning unless either implicit or explicit conventions for abstraction are introduced [16]. Our interest in how most practical graphical systems use such devices lead us to predict that students' facility with using these graphical conventions in Hyperproof would determine the usefulness of the Hyperproof environment in teaching. We expected that these devices would be well used and that the environment would be rather successful.

As the cited paper shows, we were correct in our prediction that it was the limited graphical abstraction conventions on which usability turned. But we were taken by surprise by the strength of the interactions between students' prior styles of reasoning and the outcomes of different methods of teaching. For those students who were adept at the Graduate Record Exam analytical questions which we used as pretests, Hyperproof was an extremely effective way of teaching logic. For the, in general, equally able students who scored lower on this subscale, Hyperproof teaching was a disaster on some transfer tests. Oberlander, Cox, Tobin, Stenning & Monaghan [12] present some of the results of analysing the proofs produced by these two groups of students revealing underlying reasoning mechanisms. The students who benefit from the multimodal teaching are *not* the ones who *prefer* graphical representations. In fact the strategy of going straight for concrete graphics is more often pursued by the students who do not benefit from HP. The students who do benefit are the ones who are adept at exploiting the graphical abstraction conventions in translating back and forth between modalities. Cox, Stenning & Oberlander [3] show that these differences are not merely special

Figure 1: The Hyperproof Interface. The main window (top left) is divided into an upper graphical pane, and a lower calculus pane. The tool palette is floating on top of the main window, and the other windows reveal a set of goals which have been posed.

effects of the multi-modal program but extend to students' spontaneously produced problem solving representations.

Our theory of the relation between the semantics of graphical representations and their usability turned out to be essential for analysing the effects of Hyperproof teaching. Without a theory about what aspects of graphics to investigate, these effects would have remained buried in megabytes of students' keystroke by keystroke logfiles. But what we found is that individual differences between students actually outweighed the modality effects—or at least the modality effects are not predictable without reference to cognitive styles.

This is *interface* research but in a rather different sense than is customary. Hyperproof graphics provide a different interface onto first order logic but not at the level of buttons and keyboard functions. Hyperproof teaches a partially interpreted first order logic in which some of the non-logical symbols have a fixed interpretation in terms of spatial relations. Hyperproof teaches a case-based approach to proof at a rather higher strategic level than is common (or we suspect possible) for a natural deduction system. This sounds more like a conceptualisation than an interface, but it powerfully determines cognitive results of interactions between human and computer.

Just to connect these results to more conventionally recognisable HCI issues, Oberlander & Inder [10] have shown what are probably closely related individual differences in using alternative browser interfaces. Interestingly, their gross data on time and errors to perform a retrieval task with two interfaces showed no differences between designs (this time distinguished by their use of discourse structure) until users were divided into groups on independent grounds. The individual differences masked the large equal and opposite differences in performance with the two designs between the groups. The general moral for evaluation must be that without a theory of representations and cognitive processes, evaluations are likely to miss what are important effects.

3 Graphics and algebra in specification

CCS (Calculus of Communicating Systems [9]), an algebraic formal language for modelling concurrent systems, is a highly expressive language. However, while this makes CCS an excellent language in which to model concurrent systems, the lack of constraints to guide the construction of CCS specifications can make it difficult and confusing to use. These difficulties are common to many highly formal languages and manifest themselves primarily as problems of human–human communication: either when learning to use such a language or when attempting to comprehend or communicate to another the consequences of a particular model. Users of CCS will often develop constraining design guidelines when modelling familiar concepts. Graphical representations of CCS may capture such concepts particularly effectively by utilising the inherent constraints of graphics to reflect these, as yet informal, design guidelines.

An illustrative example of the human–human communication difficulties inherent in CCS (in common with other formal languages) and the “design guideline” approach taken to overcome them results from the analysis of a case study into the use of formal methods carried out by industry. A CCS specification of a small multiprocessing system was derived from its natural language specification. Henceforth we shall refer to the producers of the natural language specification as “the specifiers” and the producers of the CCS model as “the designers” (both teams in fact shared a common member). The designers examination of their model uncovered a potential behaviour of the system which they had considered impossible. Further analysis showed that the specifiers had made certain incorrect assumptions concerning the relative timings of concurrent agents within the system. The model and its subsequent analysis were the subject of a further study [5] which uncovered other errors, due in part to misuse of the designers informal (and unspecified) design guidelines. This latter study showed that these guidelines could be captured graphically, and that this could be used to provide a precise explanation of the original incorrect assumption on the part of the designers.

The system in question, referred to as a “Twin Buffer” system, consisted of two buffers which could independently receive data and a third “Voter” component which responded to “vote” requests by reading the contents of each buffer and outputting the results of a computation of these. Figure 2 presents two “flow graphs” for the CCS specification of this system. A flow graph represents the static structure of a CCS specification, showing the potential lines of communication, represented by labelled arcs, between components of the specification, represented by named circles. The flow graph of Figure 2(a) hides the interior details of the specification, showing only the potential lines of communication from the twin buffer system to the outside environment. The flow graph of Figure 2(b) represents the same specification, but shows also the interior components, the two buffers and the voter, and the potential internal communications between them.

Interestingly the above flow graphs illustrate one of the errors in the specification which was uncovered by the subsequent analysis of [5]. The flow graph of Figure 2(a) hides almost all of the detail of this specification, but this serves to emphasise the fact that, unusually, according to this specification the twin buffer system never produces any data. The lines of communication represented in this flow graph show inputs to the two buffers (*writeA* and *writeB* actions), requests for data (*vote* actions) and output warning messages in case of error (*overA*, *overB* and *under*), yet none of these potential communications represents the output of the results of any request for data. While this is certainly an oddity, and illustrates particularly well the efficacy of representing limited aspects of information by diagrams (static structure only, in this case), we note that this error has arisen from an over-abstraction in the construction of the CCS specification, rather than being an error inherent in the twin buffer system. A deliberate decision was made to abstract from a representation of any actual data in the twin-buffer system while maintaining a representation of the situation where the system failed to manipulate the data. Unfortunately the specifiers abstracted too far by failing to capture any explicit representation of the system succeeding in manipulating the data.

Of greater interest are the design guidelines used in an informal manner by the specifiers and the efficacy of graphical representations both in capturing these guidelines, and in elucidating the incorrect assumption concerning timing made by the specifiers.

Figure 3 presents the CCS specification of the voter component of the twin buffer system. This specification indicates that the voter responds to a *vote* action by outputting a *voteA* and *voteB* action and then moving into the *Votelock* state. *Votelock* was designed to “provide a mechanism for deadlocking the *Voter* process when either of the buffers deadlocked” (which would be indicated by an *underS* warning). In CCS the summation connective “+” represents non-deterministic choice, indicating that *Votelock* can either respond to an *underS* action by outputting an *under* warning and then moving into the *Trip* state, or it can return to the *Voter* state and await further *vote* actions.

Figure 2: “Opaque” (a) and “Exploded” (b) views of Flow Graph for CCS Specification

$$\begin{aligned}
 \text{Voter} &\stackrel{\text{def}}{=} \text{vote}.\overline{\text{voteA}}.\overline{\text{voteB}}.\text{Votelock} \\
 \text{Votelock} &\stackrel{\text{def}}{=} \text{underS}.\overline{\text{under}}.\text{Trip} + \text{Voter}
 \end{aligned}$$

Figure 3: CCS Specification of the Voter Component

Figure 4: Statechart of the CCS Specification of the Voter Component

The above CCS definition of the voter component is indicative of the design guidelines informally adopted by the specifiers. These guidelines were captured graphically in the subsequent study [5] where the CCS specification was translated into a “statechart”. Statecharts [6] are a graphical language for representing reactive (concurrent) systems and Figure 4 is the statechart derived from the CCS agent *Voter*. In statecharts states are represented as labelled blocks and transitions as arrowed arcs with non-empty labels of the form a/t , which denotes a transition triggered by the event a that generates the sequence of actions t . The unanchored arrow pointing to the *Voter* state is used to indicate that this is the initial state for this component.

The two most significant tactics adopted by the specifiers in constructing their CCS specification were that:

1. only those states of the system deemed somehow significant would be represented by named agents (e.g. *Voter*, *Votelock* and *Trip*)
2. sequences of actions connecting such significant states should always be of the form *action/reaction*.

We are able to capture these two tactics precisely in the graphical language of statecharts, as indicated by Figure 4. In that statechart only the significant states (those named by the specifiers) are explicitly represented and transitions along arcs are of the form *action/reaction*, in accordance with both the syntax of statecharts and the specifiers second tactic.

The second tactic above is a highly significant constraint of standard CCS. In the semantics of CCS there is no particular difference between input and output actions. They are in fact symmetrical and could be completely interchanged throughout a specification without affecting the semantics. Furthermore, in CCS there is no requirement for actions (whether input or output) to be “triggered” in a reactive manner. Actions, and thus the behaviour of a CCS specification, may be completely spontaneous, requiring no external prompting. This is a feature which considerably contributes to the expressiveness of CCS. However, experience indicates that this is a feature prone to cause confusion among certain users and so it is not especially surprising that the specifiers in this case have sought to disable it. In fact they have not been entirely successful at this, as the statechart of Figure 4 indicates. A peculiarity of this statechart is that one of the two arcs between the *Voter* and *Votelock* agents has no label. This is in fact an error in this specification as it indicates that the *Votelock* agent is able to move into the *Voter* state even if an *underS* warning is received. This error is connected with the previously mentioned error of no output data being represented in the CCS specification, as the specifiers intention had been that the unlabelled arc should represent the transition in states to be made if data was successfully read from both buffers. Had this data been explicitly represented then this would have given a label to the arc in question and thus given the CCS specification the correct behaviour (successful retrieval of data and the warning message being mutually exclusive).

A further advantage of the statechart of Figure 4 is that it supplies a readily understandable and formal account for the underlying, undocumented, assumption which led the designers to their erroneous expectations of system behaviour. Their assumption can be explained as being a belief that an action, such as *vote*, and all of the actions associated with it (that is, $\overline{voteA}$ and $\overline{voteB}$) would be processed sufficiently quickly that no intervening actions could occur. As described above, the syntax of their CCS specification was indicative of which were considered the associated sequences of actions. Examining the statechart of Figure 4 we see that this assumption corresponds precisely to the assumption that transition along an arc, such as that labelled $vote/\overline{voteA}.\overline{voteB}$, will happen sufficiently quickly that no intervening transitions can occur. This assumption is formally expressed in the semantics of statecharts and is known as the *Synchrony Hypothesis* [11]. However, it is an assumption which is not matched by the semantics of CCS, in which it is possible for intervening actions to be processed in between the processing of actions such as *vote* and $\overline{voteA}$. This capability meant that the CCS specification did not always behave as the specifiers had expected. On review it was decided that the behaviour of the CCS specification was correct, and that it was the specifiers initial assumption which was at fault.

Work in progress at the HCRC is formalising design guidelines for CCS and ensuring that these guidelines, both the above two tactics and guidelines suggested by further study of the use of CCS, can be captured graphically. One thread of this research, for example, is examining how design guidelines can guarantee properties such as causal dependencies and the persistence of dynamic concurrent objects. This is of particular interest when flow graphs are concerned, such as those of Figure 2. In general it is possible for the static structure of CCS specifications to change over time. As a CCS component evolves it is possible for certain actions to cease to be possible, for certain subcomponents to disappear, and for other subcomponents to be spontaneously created. For example, while the flow graph of Figure 2(b)

shows the three components of the initial state of the twin buffer system's CCS specification and the potential lines of communication between them, it is conceivable that as this system responds to inputs some of these lines may disappear, some components may vanish and other components may suddenly appear. The work in progress has shown that by using design guidelines for CCS, which enforce properties to prevent such unconstrained behaviour, a flow graph representing some CCS component can be guaranteed to represent its static structure for *every* state to which that component may evolve. Further properties similarly enforced, such as causality and determinacy, and this implies that we may greatly increase the expressiveness (information content) of flow diagrams without having to alter or enhance their conveniently simple language.

4 Learning formality

Although our investigations into logic learning began from an interest in contrasting the use of graphical and linguistic modalities in learning, they raise more fundamental issues about just what it is that is being learnt in elementary logic classes. These issues have led to investigations of students' naive interpretation of English quantifiers. These studies reveal aspects of students' explicit models of language in communication which have implications for the learning of formality in other domains.

On one story about the relation between logic and natural language, it is paradoxical that students have to take logic classes at all—English just *is* a logical language, and students have a fluent command of this language already. Grice [4] was inspired to his celebrated theories of how logic is embedded in communication from observations of his tutorial students' interpretations of quantifiers. He noted that it is not unusual for students to indulge in inferences which might be verbalised roughly as follows:

Teacher says *Some A are B*. If *All A are B* he would have said so. So it must be that *Some A are not B*.

From this and other observations Grice based his pragmatic injunction to be informative, and his maxims of quantity, quality, etc. We would argue that Grice just assumed that logic already gave a good account of how language functioned in *derivation*. Of course in derivation, one absolutely must *not* be informative (at least in the sense of providing information not given in the premises), since to be informative is to draw an *invalid* inference. Derivation is a discourse in which participants (if indeed there are more than one) share a common set of premises. The only thing anyone needs to be 'informed' about is how to get from premises to some target conclusion.

We suggest that the outlines of a resolution of the paradox of logic learning can be found in distinguishing these two kinds of discourse: *exposition* and *derivation*. We propose that many students come with their explicit model of communication attuned to expository discourse. What they have to learn is to make equally explicit their implicit grasp of derivation.

The evidence that students' come with an expositional model of communication comes from collecting data on their immediate sentential and graphical inferences. They are presented with a premiss sentence such as *All A are B*, and then asked whether it follows that say *All B are A*? They are asked whether this immediate inference is valid, whether the negation of the conclusion follows, or whether the two sentences are logically independent. Alternatively they can be asked which of the five Euler diagrams of the sets A and B make the premiss sentence true.

Psychologists have studied both this 'interpretation' data, and the data of syllogistic reasoning for about 70 years now. Logic tutors (like Grice) have been studying them for nearer 2000 years. Certain common errors have been named fallacies since classical antiquity. But the attention of tutors tends to be focussed on errors of *commission*—like saying the example inference is valid. What tutors tend to miss is what their students don't do. Collection of systematic data reveals that just as often as students 'illicitly convert' *All A are B* into *All B are A*, students *fail* to validly convert *Some A are B* into *Some B are A*. What is also revealed is that, though both errors are equally common, they are almost never made by the same student [13].

The most striking pattern in the data is that for questions which like these ones 'convert' the subject and predicate terms of the premiss, students who make errors are bimodally distributed. They tend either to accept invalid inferences for *all* the quantifiers tested, or to reject valid inferences for *all* the quantifiers. Responses to inferences where subject and predicate terms are not converted also show some polarisation but not nearly so strongly, and for these questions (e.g. *Some A are not B*, does it follow that *No A are B*?) there are essentially no students who invariably claim logical

independence for all the logically dependent cases. Grice only scratched the tip of the iceberg. In particular, he ignored the problem of how to give a pragmatics which covered both derivational and expository discourse, and he paid no attention to the possibility that there might be quite systematic differences between people in their explicit models of kinds of communication.

What have these ‘styles of quantifier interpretation’ got to do with exposition and derivation? Briefly, exposition is the discourse of asymmetrical authority in which someone who knows something is telling it to someone who doesn’t (see [12](in press) for development of this concept of authority). The subject/predicate structure of these simple sentences coincides with the information packaging into *given* and *new* (see [18] for an account of information packaging). The speaker puts common knowledge into the ‘given’ subject term, and what is new information for the hearer into ‘old’ predicate term. In exposition, *Some A are B* is not at all equivalent to *Some B are A*. What the new logic student has to grasp explicitly is that in derivation, these two really are equivalent. Indeed, logical calculi replace subject/predicate structure with function/argument structures, and their extensional semantics maps terms and predicates onto sets. Learning that *Some A are B* is equivalent to *Some B are A in derivation*, is learning what might be called the *extensionalist* stance.

On this account, learning a ‘formal language’ is learning that certain natural language structures which have functions in exposition, lose these functions in derivation. Graphical methods like Euler’s Circles or Venn diagrams may assist at least some students in this learning because graphics has no systematic information packaging. The characteristic diagram for *Some A are B* is, unlike the sentence, symmetrical. However, what we do not yet know is what underlies the difference between students who ‘rashly’ draw invalid conversion inferences, and students who ‘hesitantly’ fail to draw valid conversion inferences. They appear to both have problems with the conversion of subject/predicate structures, but what distinguishes them? We can show that this behaviour *does* affect their reasoning behaviour in solving syllogisms [15]. It is not merely superficial behaviour brought on by the apparent simplicity of the interpretation task. But we cannot yet fully relate these individual differences in style of interpretation to the ones observed in Hyperproof teaching. That is an issue for further research.

What are the implications of these observations for learning formal subjects? If we accept this model of what is happening in logic classes, then students start with a dominantly expository model of communication in which many of the structures of their natural language, particularly the sequencing of information in sentences, function to control the interpersonal patterns of authority for knowledge present in the communication situation. To learn logic is to learn to make explicit a model of derivational discourse in which patterns of authority for knowledge are quite distinct, and which the function of controlling asymmetrical patterns of authority no longer has any place.

Is this a model of what constitutes learning formality which can be applied to learning in other domains? We think so. We can take another example domain, learning to reason in uncertainty, and see that it has many of the same features as the learning of elementary deduction. Learning the concept of *independence* in statistical reasoning plays a role rather like learning what we have here called the extensionalist stance. As psychologists Tversky & Kahneman [17] have demonstrated many times, untutored intuitions about joint probabilities display many of the sensitivities to context that we have noted in students’ interpretations of quantifiers. Asked to imagine a person fitting a brief personality description (of a rather nerdy character), subjects were asked to judge the likelihood of them: being an accountant; playing jazz for a hobby; or both being an accountant and playing jazz for a hobby. Tversky and Kahneman’s subjects rated the last as more likely than the first. What is strange is that the judgement asserts that the intersection of two properties (playing jazz, and being an accountant) is *more* probable than one of the properties alone. Whatever is the basis of subjects’ reasoning, their representation of properties clearly diverges from that required by the extensionalist stance.

Tversky and Kahneman’s observations of judgements of sequences of events such as tosses of a coin betray a belief in what is often called the Gambler’s Fallacy, and which they generalise to ‘the law of small numbers’. At root, the Gambler’s Fallacy is the belief that tosses in a sequence are non-independent. Because the long-run expectation is half heads, it is assumed that a run of heads will have to be followed by an increase in the number of tails *in order to make the end average come out right*. Learning the concept of statistical independence is learning that if events can be classified into causally insulated categories, their frequencies in populations will obey certain algebraic laws.

Of course there are notable disanalogies between these two domains of formalisation. But perhaps there are enough commonalities to reward study.

5 Conclusions

At the most superficial level, what each of these studies have in common is that they have something to say about formality and the study of human behaviour. In each case an account of the semantics of representations in (very simple) formal terms, enables the analysis and modelling of what would otherwise be incomprehensibly complicated behavioural phenomena. Without a view about the semantics of graphics, the mental processes underlying learning from Hyperproof remain, well, underlying. Without a view of the relation between graphical and formal algebraic methods in specification, we cannot understand the different kinds of misunderstanding which the alternative approaches are suited to uncover. Without asking for a cognitive model of students' explicit ideas about communication, we would have uncovered the broad patterns of response in interpreting quantifiers. In all these areas, it is formal theories of semantics and the question how they fit to behavioural phenomena which drive investigations.

It is curious that these apparently diverse studies have come against striking and highly systematic individual differences between people. The specification work reveals differences in the efficacy of different media for presenting different kinds of information and for finding different kinds of error. None of the studies set out to find such differences. Cognitive science has generally been rather embarrassed by findings of individual differences, with partial justification given the dubious ends to which their study is often put. Whatever this avoidance is justified by, it is not, in principle, justified. People do differ in how they go about reasoning. It is encouraging that it is exactly when we start to apply formal methods to psychological study that these differences emerge most clearly. These differences are not just due to the inherently ineffable individuality of human beings. In fact they constitute an amazing opportunity to analyse underlying mental processes. Without them, it would have been extremely hard to break into the complexities of any of our data.

Individual differences are clearly of great significance to HCI. HCI is accustomed to differences in level of expertise. Some of the differences mentioned here are probably related to expertise, though some are more to do with what is better termed cognitive style. One exciting prospect is that these differences offer a way to relate the cognitive and the social sides of HCI more analytically. On the face of it, differences in reactions to modality of presentation are paradigmatically cognitive variables. But we hope we have at least made a case for interest in the hypothesis that these differences may also be related to students' understandings of social patterns of authority for knowledge. Using completely different methodology, Bloor [2] has shown that the choice between algebraic and geometric approaches in the development of some 19C mathematics were correlated with social attitudes to professionalisation and to more general political and religious attitudes to the issues of the day. Less distantly, viewing educational communication of formal subjects (whether mediated by person or by machine) as the establishment of symmetrical authority for derivation makes it possible to relate many issues in HCI to familiar issues in education.

References

- [1] Barwise, J. and Etchemendy, J. Hyperproof. CSLI Publications, Stanford, Ca., 1994.
- [2] Bloor, D. Hamilton and Peacock on the essence of algebra. In Mehrrens, H., Bos, H., and Schneider, I. (Eds), Social History of Nineteenth Century Mathematics. Birkhauser, Basel, 1981.
- [3] Cox, R., Stenning, K., and Oberlander, J. The effect of graphical and sentential logic teaching on spontaneous external representation. Cognitive Studies: The Bulletin of the Japanese Cognitive Science Society, 2(4):56–75, 1995.
- [4] Grice, H. Logic and conversation. In Cole, P. and Morgan, J. (Eds), Syntax and Semantics: Vol 3, Speech Acts, pages 43–58. New York: Academic Press, 1975.
- [5] Gurr, C. Supporting formal reasoning for safety critical systems. High Integrity Systems, 1(4):385–396, 1995. Oxford University Press.
- [6] Harel, D. Statecharts: A Visual Approach to Complex Systems. Technical Report CS 84-05, Department of Applied Mathematics. The Weizmann Institute of Science, 1986.

- [7] Larkin, J. H. and Simon, H. A. Why a diagram is (sometimes) worth ten thousand words. *Cognitive Science*, 11:65–99, 1987.
- [8] Mackinlay, J. Automating the design of graphical presentations of relational information. *ACM Transactions on Graphics*, 2:110–141, 5.
- [9] Milner, R. *Communication and Concurrency*. Prentice Hall, 1989.
- [10] Oberlander, J., Cox, R., Tobin, R., Stenning, K., and (submitted), P. M. Individual differences in proof development strategy following multimodal logic teaching. In *Proceedings of the Annual Meeting of the Cognitive Science Society of America*. UCSD, 1996.
- [11] Pnueli, A. and Shalev, M. What is in a Step: On the Semantics of Statecharts. In *Theoretical Aspects of Computer Science*. Springer Verlag, 1992. *Lecture Notes in Computer Science* No. 526.
- [12] Stenning, K. Embedding logic in communication: lessons from the logic classroom. In van Benthem, J. (ed), *Proceedings of Colloquium on Logic and Argumentation*. Royal Netherlands Academy of Arts and Sciences, 1996 (in press).
- [13] Stenning, K., Cox, R., and Oberlander, J. Attitudes to logical independence: traits in quantifier interpretation. In *Proceedings of the Seventeenth Annual Conference of the Cognitive Science Society*, pages 742–747, Hillsdale, N.J, 1995. Distributed by Lawrence Erlbaum Associates.
- [14] Stenning, K., Cox, R., and Oberlander, J. Contrasting the cognitive effects of graphical and sentential logic teaching: reasoning, representation and individual differences. *Language and Cognitive Processes*, 10(3/4):333–354, 1995.
- [15] Stenning, K., Cox, R., and Yule, P. Quantifier interpretation and syllogistic reasoning. In *Proceedings of the Annual Meeting of the Cognitive Science Society of America*. UCSD, 1996.
- [16] Stenning, K. and Oberlander, J. A cognitive theory of graphical and linguistic reasoning: logic and implementation. *Cognitive Science*, 19:97–140, 1995.
- [17] Tversky, J. and Kahneman, D. *Judgement under Uncertainty: Heuristics and Biases*. Cambridge University Press, 1983.
- [18] Vallduví, E. *The Informational Component*. Garland Publishing, New York, 1992.