



Cyclic interaction: a unitary approach to intention, action and the environment

Andrew Monk*

Department of Psychology, University of York, York YO1 5DD, UK

Received 14 October 1997; accepted 5 May 1998

Abstract

The history of psychological explanation in human–computer interaction (HCI) is reviewed in order to illustrate the notion of cyclic interaction. The claim made is that much real behaviour is usefully thought of as a continuous process of cyclic interaction with the environment. According to this account action leads to changes to the state of the world, these are evaluated with respect to, and in a manner conditioned by, the user's current goals. This evaluation leads to the reformulation of goals and further action, this action leads to a new state of the environment, and so on. Cyclic interaction is contrasted with the more commonly adopted view of cognition that may be caricatured as a 'one-shot comprehension' where perception and recognition lead to action but the role of goals and the effects of action on the environment are not primary concerns. It is argued that a change of emphasis in cognitive research is required to make good these omissions, with new kinds of experimental paradigm and new ways of modelling behaviour. © 1998 Elsevier Science B.V. All rights reserved

Keywords: Human–computer interaction; Cyclic interaction

1. Psychological explanation in human–computer interaction

Investigators from very different research traditions have been attracted to the problem of designing more useful computer systems. Stimulated by the funding models adopted by government agencies, computer scientists, psychologists and anthropologists have come together to form inter-disciplinary research teams to

* Tel.: +44 1904 433148; fax: +44 1904 433181; e-mail: aml@york.ac.uk

address this practical problem. The result has been a rare degree of inter-disciplinary debate and comparison (Monk and Gilbert, 1995). The most recent manifestation of this is the debate between ethnographers and anthropologists on the one hand, and psychologists and cognitive scientists on the other. ‘Cognitivism’ is being attacked in favour of accounts of behaviour in terms of ‘situated action’, that is an account of behaviour as meaningful actions contextualised in a history of previous actions and social values. The points being made have been taken seriously enough for the journal *Cognitive Science* (1993) to devote a special issue to a debate between proponents of situated action and their critics.

Cognitive psychology has similarly received criticism from ecological psychologists for not making explicit the role of the environment in explanations of human behaviour. In ecological psychology (Gibson, 1979) the objective is to explain the critical aspects of an organism’s actions on the environment and the way the environment in turn governs action (Turvey and Carello, 1986). Slightly more cognitive, but still critical of conventional approaches, there are the proponents of ‘distributed cognition’ (e.g. Hutchins, 1994) and ‘activity theory’ (e.g. Nardi, 1996). Distributed cognition takes the view that the environment is a cognitive resource that can be used in parallel with internal resources such as long term memory. Activity theory is similarly concerned with the way the environment and tools support action but is additionally concerned with consciousness, and the goals of groups and society.

Some of these critics of conventional cognitive psychology are suggesting that any attempt to explain behaviour taking a process-based cognitive stance should be abandoned. The theme of this paper is that this is not necessary and it is possible to explain many of the phenomena described by proponents of situated action, ecological psychology and distributed cognition in an information processing framework. Rather, the phenomena described suggest a change of emphasis in the way cognitive psychologists think about behaviour and the experiments they perform. This paper illustrates this need for a change of emphasis through the history of psychological explanation in HCI. This began with standard ‘experimental cognitive’ accounts in terms of memory, perception and so on. These were found wanting as much of the variation in user behaviour was found to be due to the formation of inappropriate intentions and these accounts made no reference to the goals of the computer users. There followed a period where an approach based on work on human and machine problem solving was adopted. Recently, this approach has in turn been criticised by various groups who emphasise external activity and the influence of the environment leading to accounts that view behaviour as a continuous process of cyclic interaction with the environment. HCI is chosen as an example because that is the author’s field; it would be surprising if there were not other fields of practical enquiry with similar histories.

1.1. The application of experimental cognitive psychology

Fig. 1 is a caricature of a typical experiment of the kind that might be found in an undergraduate textbook in experimental cognitive psychology. A stimulus, perhaps a word or a sentence, is presented on a computer screen. The experimental subject

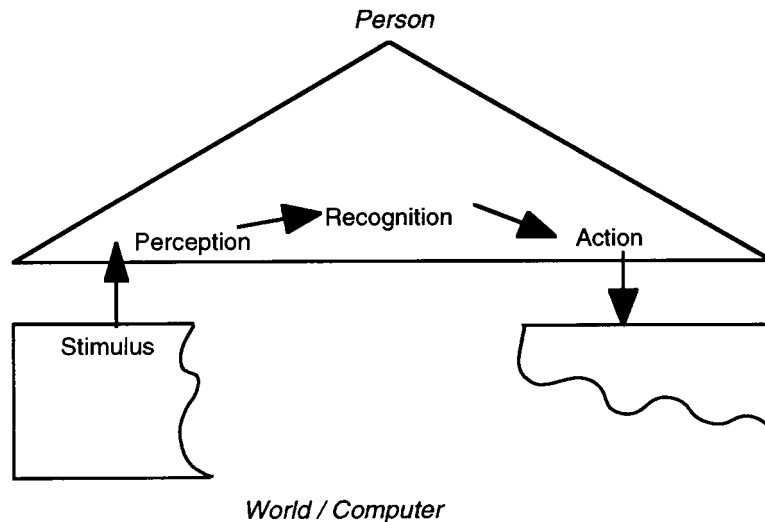


Fig. 1. A caricature of many experiments in cognition – ‘one-shot comprehension’.

viewing this perceives and recognises the stimuli and forms a response. This can be characterised as a view of behaviour as single-shot comprehension. The nearest ‘real-life’ task is understanding the meaning of some sign or symbol and, while this stimulus–response sequence may be repeated many times in an experiment, each of these experimental trials is viewed as essentially a separate event.

Using a variety of experimental paradigms of this kind, cognitive psychologists have tested hypotheses concerning perception, recognition and response formation. Cognitive psychologists were the first behavioural scientists to focus their attention on the problem of designing better computer systems. It was naturally assumed that the theories they generated could be applied to this practical problem. The user of a computer is interacting with an information processing machine and so the existing accounts of human information processing ought to be particularly well suited to inform the design of such machines. However, even cognitive psychologists found very little of cognitive psychology to be useful in this regard (Landauer, 1987). While HCI research has led to new understanding and considerable gains in the usability of commercial software, this has not been due to the application of experimental cognitive psychology. The theme of this paper is that conventional cognitive explanation, conceived as a series of essentially independent or one-shot comprehensions, suffers two important omissions. The first is that it takes no account of the way goals condition behaviour. The second is that it takes no account of the effects of actions on the environment. The next two sections take these omissions in turn.

1.2. *Goals and goal generation*

The first omission has been recognised for some time. It became apparent when HCI researchers examined the mistakes people made when using a computer

(Norman, 1981; Wright and Monk, 1989; Reason, 1990). Many of the errors made were best explained as due to the user having an inappropriate intention. Though the processes of perception, recognition and response formation are clearly involved, they do not account for much of the variation in behaviour. What was needed was a model of how a skilled user generated goals and sub-goals appropriate to the task and how a novice user could learn to do this. This omission was addressed by adopting work from studies of human and machine problem solving (e.g. Card et al., 1983).

Colloquial accounts of human behaviour depend very much on attributing intentions, e.g. 'he was trying to...', 'she wanted to...'. This is such a natural way of thinking that whatever the philosophical status of intention it provides a very convenient way of understanding behaviour (Dennett, 1990). Cognitive psychologists have generally shied away from thinking about intention. The exception to this is work on problem solving with its close links to artificial intelligence and cognitive science. Newell (1980) describes the process of finding a solution to a puzzle, such as the tower of Hanoi problem, in terms of goals and sub-goals in a problem space. The top level goal is given by the puzzle 'to get all the discs from the left-hand peg to the right-hand peg'. The rules of the puzzle, the legal moves, define a set of possible states the discs could be in. This is the problem space. The solution to the puzzle is described as a route through this problem space. Procedurally, the puzzle solver encodes this route as an intention to achieve some intermediate state in the problem space. This can be described as a sub-goal of the primary goal to solve the problem. An example of a sub-goal might be, 'to get the small and medium sized disks onto the middle peg'. In the process of learning the solution different sub-goals and sub-sub-goals are tried out. Some become part of the solution, some are rejected as unproductive. This model of problem solving, originally described by (Newell and Simon, 1972) has been successfully simulated in computer programs and similar techniques are commonly used in artificial intelligence. These work by generating a 'goal stack'. Consider Fig. 2. Let us say an initial goal G1 generates a further goals G1.1 which in turn generates a further goal G1.1.1. The goal stack is now G1, G1.1, G1.1.1. When G1.1.1 is achieved it can be 'popped' off the stack and some new goal G1.1.2 'pushed' on in its place. Eventually all the sub-goals are achieved and the primary goal can be popped off the stack.

Card et al. (1983) adapted this model to the 'puzzle' of using a computer. Table 1

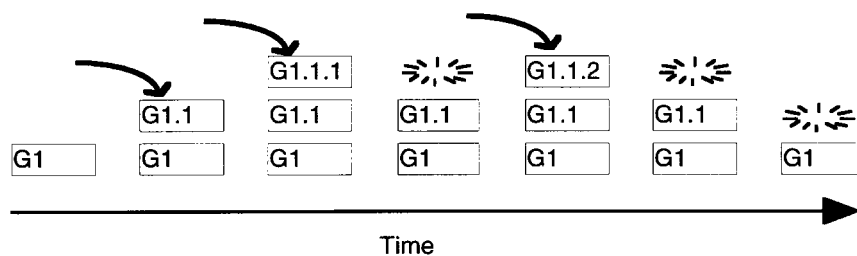


Fig. 2. A goal stack.

Table 1
GOMS analysis of editing task adapted from Card et al., 1983, p. 142

Goal: edit-manuscript	
Goal: edit unit task	<i>Repeat until no more tasks</i>
Goal: acquire unit task	
get-next-page	<i>If at end of manuscript</i>
get-next task	
Goal: execute-unit-task	
Goal: locate-line	
[select:use-qs-method	
use-lf-method]	
Goal: modify-text	
[select:use-s-command	
use-m-command]	
verify edit.	

shows their analysis of someone editing a document using the POET word processor. The top-level goal is to make all the changes marked on a paper version of the document. This is to be achieved by doing the next task ‘Goal: edit unit task (repeat until no more tasks)’. This sub-goal generates two sub-sub-goals: (i) finding and reading the next change on the printed version (‘Goal: acquire unit task’) and (ii) making this change to the electronic version (‘Goal: execute unit task’). Eventually a goal leads to an *operation*, ‘get-next-page’. Sometimes a *selection rule* is needed to choose between operations, for example, with this word processor there are two ways to locate the correct line in the electronic document. The user must select between ‘use-qs-method’ and ‘use-lf-method’. A *method* is a complex operation. For example, lf-method involves pressing the line-feed button until the text is visible.

This notation, and the method Card et al. describe for deriving the analysis, is known as GOMS: Goals, Operations, Methods and Selection. In the GOMS model behaviour is controlled by a constantly changing goal stack and the main determinant of detailed behaviour are the rules for adding and removing goals from this stack. Card et al. present GOMS as a practical engineering technique for comparing alternative design solutions, and it has been used successfully in this way. Gray et al. (1990), for example, claim to have saved a phone company \$3 million a year by their painstaking GOMS analysis of the work of toll assistance operators. A GOMS analysis is an idealised picture of how a skilled user should think and behave, i.e. it presents a model of the solution to the puzzle. Newell’s later work was to model the development of such a solution in the context of a unified theory of cognition. Soar (see Newell, 1990) is an integrated cognitive architecture that learns by exploring the problem space. Soar models are now finding increasing use for modelling in HCI (e.g. Howes and Young, 1996).

The notion of a goal, and some notation for describing how goals are formed, would seem to be essential ingredients in an explanation of how people interact with computers. However, the GOMS approach, in particular, has been severely criticised as putting too much emphasis on what goes on in the user’s head at the expense of

developing a proper understanding of action and the context of behaviour (Suchman, 1987). These criticisms are discussed in Section 1.3.

1.3. *Situated action versus planning*

Fig. 3 develops the caricature of cognition from Fig. 1 by adding the missing ingredient ‘Goals’. Fig. 3 can be described as a planning model of cognition. It is the kind of model that might be specified using GOMS. Behaviour is viewed as arising from some initial goal and the sub-goals are generated in response to that initial goal in advance of action.

This view of behaviour as a plan generated in response to a clear initial goal does not accord with many people’s experience of using computers, particularly computers with graphical user interfaces. Graphical user interfaces, such as the Apple Macintosh and Microsoft Windows, are perhaps the single most influential application of HCI research. They have made complex functionality available to users who do not have the time to learn how to use the same facilities using command-driven interfaces such as UNIX or DOS. The power of graphical user interfaces comes from the iterative behaviour they engender. Rather than formulating a complete command that is either correct or incorrect, the user makes piecemeal adjustments to screen objects. These may be selected text in a word processor, an element in a drawing or whatever. The adjustments can easily be undone and re-done in different ways so that the user moves towards the desired outcome in a series of iterations. Where complex commands are inevitable they are specified in a similar piecemeal fashion by manipulating a command object, e.g. a dialogue box. This way of working has been dubbed ‘direct manipulation’. It depends on users moving through several recognise–act cycles where the effect of previous actions on the computer display

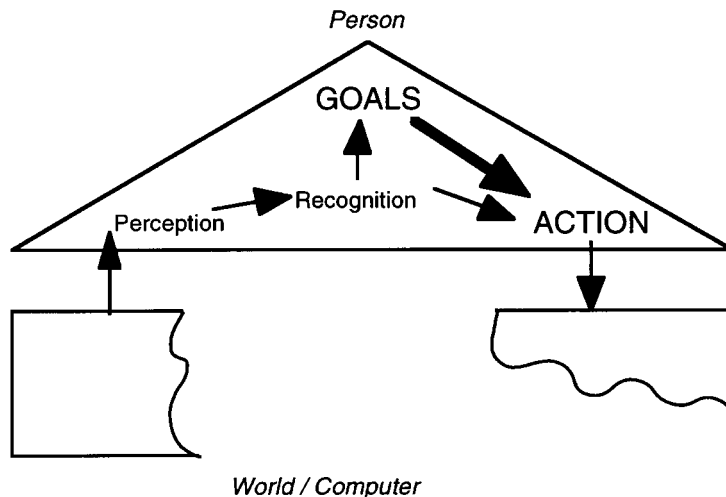


Fig. 3. Adding ‘goals’ to the caricature – a ‘planning model’. Action arises from some initial goal and the sub-goals generated in response to it. Perception and recognition may still be presumed to be involved but their role is not made explicit, hence the small type face.

is as important in governing their behaviour as anything ‘cognitive’ going on in their heads. The cyclic nature of behaviour is the second omission that results from focusing on cognition as one-shot comprehension.

Suchman’s book *Plans and Situated Actions* (Suchman, 1987) has been very influential in this attack on one-shot cognitivism. A large part of this book is a study of two users trying to understand the interface to an advanced photocopier with the aid of an expert help system. This study is in the ethnomethodological tradition (see e.g. Greatbatch et al., 1995) and uses a well defined notation to make a precise transcription of all the interaction that goes on. This records what was said by the two users, their non-verbal behaviour such as pointing and working the machine as well as the behaviour of the machine itself. The notation allows the precise temporal relationships of these events to be recorded. In her interpretation of these records Suchman seeks to explain the meaning of these events to the users.

Through this analysis Suchman shows how the sequence of events that unfolds is governed largely by the way that the users interpret their own actions and those of the machine, and that this interpretation is in turn dependent on the particular social and physical circumstances pertaining at the time. By way of analogy she considers a canoeist about to navigate some rapids. After the event the canoeist might recount planning his route through the rapids; however, close observation of his behaviour would reveal that he is really only reacting to environmental signals provided by his orientation in the canoe, the current and so on. The ‘plan’ may have served to find an advantageous initial position and perhaps to identify important physical cues to attend to, but his moment to moment behaviour was governed by his understanding of the momentary state of the environment. The goal ‘get through the rapids without capsizing and avoiding that boulder on the right’ is only the briefest sketch of his behaviour. The canoeist may generate an account of what happened in terms of intentions, after the event, but really his behaviour was an ‘emergent property’ of the interaction between his actions and the environment.

In Suchman’s case the intellectual task was learning to work a photocopier. Another influential author, Lave (1988), studied the everyday use of mathematics in shopping and cookery. She showed that faced with everyday problems of measurement or calculation people do not activate a general plan for abstract arithmetical manipulation as might be taught at school. Rather, they use the constraints provided by the physical environment. So, for example, a dieter faced with the problem of measuring three-quarters of two-thirds of a cup of cottage cheese was observed to divide the cheese into thirds, take two of these portions, spread them out and divide them into quarters. Two-thirds of three-quarters is, of course, one half but using the cheese to do the calculation in this way was seen as simpler than the corresponding symbolic manipulation.

Fig. 4 is a caricature of the position adopted by Suchman (1987) and Lave (1988). The oval arrow signifies their point that action is a continuous process. Action leads to effects that leads to new actions and so on. In this sense action is ‘situated’ in a history of the effects of previous actions. This notion of behaviour as cyclic interaction is central to this paper.

The model caricatured in Fig. 4 lays emphasis on observable events, the actions taken and the effect these actions have in the world. The role of abstracted a priori goals in generating behaviour is denied. To quote Suchman (1987)

‘The alternative view is that plans are resources for situated action, but do not in any strong sense determine its course.’ (page 52) ‘To return to Mead’s point, rather than direct situated action, rationality anticipates action before the fact, and reconstructs it afterwards.’ (p. 53)

This may seem a very extreme position to take. It seems uncontroversial to this author that at the lowest, most detailed, level of analysis behaviour is governed by environmental contingencies; this is the nature of motor skills (see e.g. Fitts and Posner, 1968). Similarly, at the highest, least detailed, level of analysis behaviour is motivated by plans, in the case of the canoeist say ‘to reach the campsite by noon’. A less extreme statement of what Suchman and other authors coming from an ethnomethodological tradition are saying is that many of the behaviours we would normally describe as cognitive or intellectual fall into the former rather than the latter category.

1.4. *Display-based reasoning*

Further examples of data that go against the notion of behaviour as planned activity, that have been interpreted in this less extreme way, can be found in studies of human–computer interaction. Mayes et al. (1988) asked users of MacWrite to recall details from the screens they viewed each time they used this word processor. A typical question was:

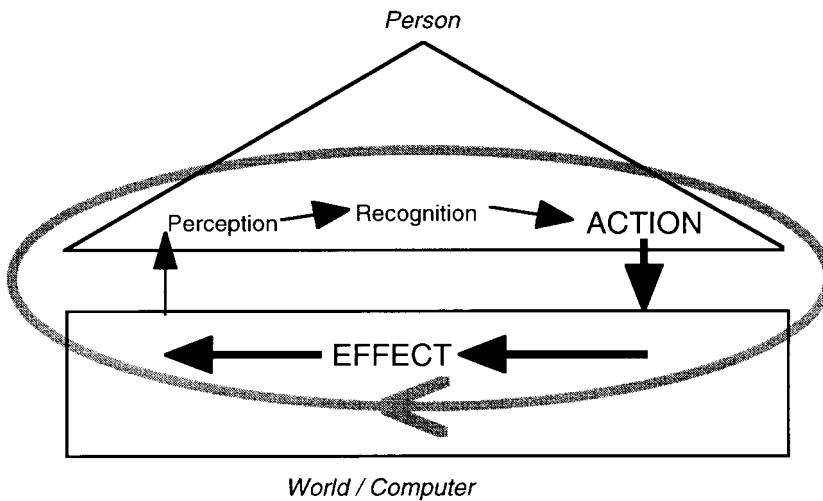


Fig. 4. A caricature of the situated action position. behaviour arises through a continuous cycle of actions and effects. The roles of perception and recognition are not specified, hence the small type face.

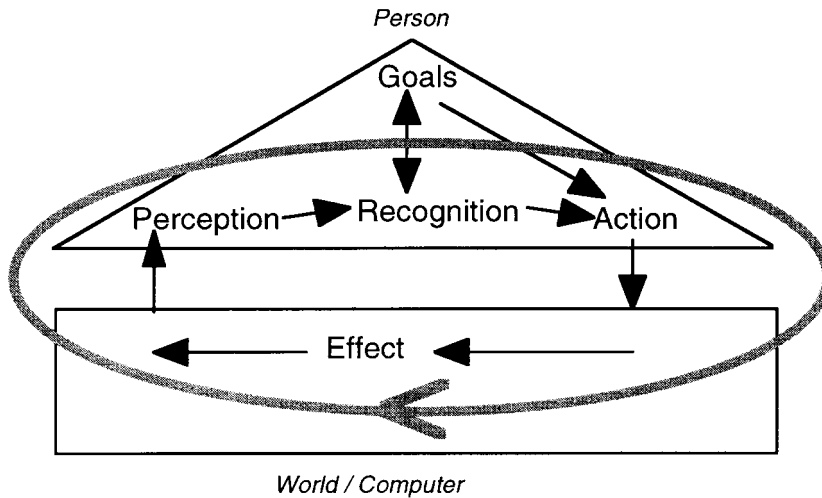


Fig. 5. A full model of cyclic interaction. There is a recognition–action–effect cycle, similar to that in Fig. 4, but action is also seen to arise directly from goals and through the interaction between goals and recognition. The two-headed arrow signifies that goals can condition recognition just as recognition can change goals.

*‘The menu bar is now as follows: **‘Apple’ File Edit Search Format Font Style**. Now please list the choices you have from each menu when it is pulled down’*

The average performance on this task was very poor. Even frequent users were only able to remember the names and locations of less than half of the menu items. This contrasts with their performance in a follow-up study. Here users had access to a working system. When asked to find a menu item by manipulating the mouse over the same prompts users had little difficulty. Users who had recalled menu items in the wrong place generally found those items on the first attempt and without hesitation. So, asked to do a task with the user interface they had no problem, when asked to reconstruct the form of the user interface from memory they had considerable difficulty.

Howes and Payne (1990) interpret these results as an example of display-based reasoning (Larkin, 1989). They suggest that menu selections are made by semantic matching. Their D-TAG model works as follows. Let us say one wants to save one’s work. One looks through the menu headers (these are the single words ‘File’, ‘Menu’, etc. at the top of every Macintosh screen that reveal a whole menu when clicked on). An approximate match is obtained between the meaning of this goal and the meaning of the item named ‘File’. Selecting this reveals another set of items containing the word ‘Save’; this is similarly matched with the goal and selected. This is the sort of process that one might expect of someone exploring a new word processor. Howes and Payne (1990) are proposing that the learning that occurs with continued experience with these menus is simply to automate this matching process. One does not learn a plan: ‘to save one’s work select File then Save’. One simply

learns to do the semantic matching with what appears on the screen more effectively. A detailed model of such display-based interaction has been developed by Kitajima and Polson (1995). This exhibits many of the characteristics of expert users including expert slips.

The Mayes et al. (1988) data are open to a number of interpretations and one can criticise the empirical support for this model. However, the experiment is interesting for suggesting that the behaviour of the users of this kind of menu system could be driven by the appearance of the display and the way it reacts to actions from the user rather than some learned representation of the procedure for working it. Howes (1994) has gone on to build a cognitive model that learns how to format text into double columns in Microsoft Word. The important thing to note about all these models is that, like Mayes et al.'s users, it can only do the task by reacting against an accurate model of the displays generated by the computer. That is, the abstract description of the computer system generated by Howes to depict the behaviour of Microsoft Word is as important in determining the behaviour of the simulated user as his model of what goes on in the user's head. Making explicit the effects of actions allows cyclic interaction and so provides an account of action in a process-based model.

2. Cyclic interaction

The cyclic model exemplified in the display-driven models of Howes and Payne (1990) as well as Kitajima and Polson (1995) is caricatured in Fig. 5. This supplements the model in Fig. 4 with goals. The user's goals condition perception and action. They also affect, and are affected by, recognition of situations and events in the world. To illustrate the different components included in this model consider the account of iterative interaction with a graphical user interface given in Section 1.3. Let us say a user formulates some goal to format a document with a word processor. The goal may be ill defined but this is not a problem because it is still sufficient to condition the processes of perception and recognition to evaluate certain features of the display. This evaluation in combination with the goal leads to the user taking some action with the word processor. This action has certain effects and the document changes its appearance. Assessment of the new display results in new goals being formulated, that in turn leads to new perceptions and new actions. This cycle is repeated until the user is satisfied with the document.

The model has the emergent properties required by Suchman. Consider the user of a word processor who wants to save a document. A GOMS analysis of this task might be specified as in Table 2. There are four separate goals involved. To implement such a model one would have to specify control structures for placing and removing each of these goals.

Contrast this with the account in Table 3. The User model describes the process of recognition and action carried out by the user (the triangle in Fig. 5). The System model describes the effects of actions on the computer display (the rectangle in Fig. 5). Following Howes and Payne's D-TAG model described above (Howes and

Table 2
GOMS analysis of saving a document using the Macintosh interface

Goal: save document
Goal: use 'Save' menu item
Goal: reveal 'Save'
move mouse cursor into 'File' menu tab
press mouse button
Goal: select 'Save'
drag mouse cursor into 'Save'
release mouse button

Payne, 1990), there is just one general cognitive mechanism that assesses the semantic match between the current goal and the available actions. Because the available actions change as the interaction proceeds, as specified in the System model, there is no need to invoke a detailed plan as in Table 2 and there is only one goal which is instated for the duration of the interaction. The detailed behaviour emerges from the interaction between the System and User models. While it would be potentially possible to use the GOMS notation to describe display-driven behaviour of this kind using the 'selection' mechanism, the point is that Card et al. (1983) provide no complementary abstraction for describing the behaviour of the system. This is what forces them into a plan based account of behaviour.

3. Implications for experimental work

The aim of this paper is to promote approaches to cognitive psychology that treat behaviour as cyclic interaction rather than as 'one-shot comprehension'. Theories of cyclic interaction must explain how the effects of one action lead to the next action, and how goals condition this process. The need for this way of thinking was explained mainly by considering work in HCI, but the points made have much more general applicability. It can be argued that most of the actions we take in our everyday behaviour depend on the effects of our previous actions and the objectives we currently hold. This is rarely true in the bulk of experiments reported

Table 3
A partially specified cyclic model of saving a document using the Macintosh interface

User model:
Goal: save document
Repeat
1. Match goal to visible menu items
2. Select best match
System model:
mouse cursor in 'File' menu tab and mouse down →
file menu displayed, 'File' highlighted
mouse cursor in 'Save' menu item and mouse down →
'Save' highlighted

in psychological texts and journals. There are experimental paradigms in cognitive psychology that go against this general rule but they are few and far between. New empirical paradigms are required that allow hypotheses to be tested about action–effect cycles and the formation of goals.

The first step in doing this is to identify some everyday tasks that involve cyclic interaction and that are suitable to bring into the laboratory. Having found a task that can be abstracted and transferred to the laboratory there are broadly two approaches that can be taken. One is to construct a model that predicts detailed performance. The model might predict certain classes of error as being more frequent than others or that a certain subtask might take longer to perform than others. Actual performance is then observed, in approximately naturalistic conditions, to test those predictions. The other approach is to perturb the process of cyclic interaction in some way to test a specific hypothesis. Some examples of these approaches are given below.

3.1. *HCI*

Both approaches to generating experiments on cyclic interaction have been taken in HCI. Kieras and Polson (1985) built a detailed model of the knowledge needed to work a particular word processor using a production system. At the same time they built a model of the word processor, so making it possible to simulate cyclic interaction between the model of the user and the model of the system. With these models Polson and Kieras (1984) were able to predict the detailed performance of users learning the word processor by counting the number of productions that have to be learned at each stage of their training procedure. In a more recent study, Kitajima and Polson (1995) have built a detailed cyclic model of the task of drawing a graph with Cricket Graph. The simulation studies of this model were to demonstrate the sufficiency of the model, that is to show that it could demonstrate the right action sequences. Serendipitously, it demonstrated a mechanism to explain the commonly observed phenomenon of high error rates in expert performance (e.g. Hanson et al., 1984). An example of an experiments that perturbed the cycle of interaction is provided by Trudel and Payne (1995). They manipulated the goals and actions of participants in experiments on cyclic interaction with a simulated digital watch. One experimental manipulation was to limit the number of actions the participant was allowed to take in exploring this interactive device, another was to provide a list of goals to supplement those they would generate spontaneously. Both manipulations changed the effectiveness with which they learned the action–effect mappings built into the device.

3.2. *Motor skills*

Tasks such as catching a ball have been studied by action psychologists and modelled in terms of how one part of a skilled movement changes the stimulus properties controlling the next part of the movement. For example, Beek et al. (1995) modelled the effects of the actions of a juggler when combined with the

laws of motion using non-linear dynamics. Using this model they were able to predict the most informative stimulus for a juggler wishing to time his or her actions, a prediction later tested and confirmed by experiments in which cyclic interaction was perturbed by only allowing jugglers to see the objects juggled at different points in their trajectory.

3.3. *Eye movements in reading*

Some of the tasks previously abstracted for the laboratory as purely cognitive single-shot tasks may also be viewed as cyclic interaction. For example, reading text involves moving one's eyes over the page in a series of fast 'saccades' interspersed with pauses or 'fixations'. These eye movements have no effect on the text itself, however, because of the relatively small area of foveal vision, they do affect the information available to guide the next eye movement, this affects the information available to make the next eye movement and so on. Models of eye movement control of the kind constructed by Rayner (see Rayner and Pollatsek, 1989) can thus be thought of as describing a form of cyclic interaction. Rayner and his colleagues test their models by perturbing the cycle of interaction by using an eye movement contingent display. Sophisticated eye movement monitoring and very fast CRT displays are used to change the text within a saccade or in some experiments within a fixation, with often surprising and illuminating results.

3.4. *Reasoning with a pencil*

A more cognitive task with cyclic properties would be reasoning with a pencil. Here the actions and effects are more discrete than in motor skills. To construct an experiment with this task one would select some algebraic or arithmetical manipulation that is difficult to accomplish in the head. People learn to encode such problems by writing them down in a particular way. What has been written acts as a stimulus resulting in other things being written and so on until the problem is eventually solved. The behaviour of people performing such tasks in the laboratory should be reasonably predictable and amenable to measurement. The cyclic interaction could be perturbed if a pen-based interface was used to do the writing. These systems use a stylus to write on a LCD display making it possible to blank different parts of what was written or to introduce delays at critical points in the task.

3.5. *Process control*

The process control simulations used by Berry (1991) in her studies of implicit learning involve cyclic interaction. A participant in one of these experiments might have to control, say, the temperature of the fluid in a tank. The effect of some action depends on the current temperature in the tank and that depends on previous actions. This task was selected because it has action–effect mappings that are difficult to ascertain. This is an interesting variant on the theme developed here as part of the

experimental task is to try and understand these dependencies through cyclic interaction with the system.

3.6. *Everyday transactions*

Finally, there are the many everyday transactions that follow well worn schemata or ‘scripts’. Activities such as going to a restaurant or paying for one’s shopping in a supermarket require one to act appropriately to other peoples reactions to one’s previous actions. Because the effects to be described are the reaction of other people, the rules describing them may contain a degree of uncertainty but there is no reason why this should be difficult to simulate in an experiment, or to model in a theory.

4. Conclusions

The notion of cyclic interaction has a long history in psychology (e.g. Miller et al., 1960). It can be viewed as a discrete version of the standard feedback model from cybernetics and has the same purpose of controlling a system that has both external and internal influences. Thus a central heating thermostat must be able to cope with changes in temperature arising from its own actions (switching on and off the radiators) as well as uncontrollable influences such as the weather or people opening and closing doors. In the same way, cyclic interaction has evolved to control an environment that responds to action from the organism as well as other factors that are less easily predicted. In HCI the idea of cyclic interaction, as expressed in Fig. 5, was introduced by Card et al. (1983) as their ‘recognise–act cycle’. Norman’s seven stages of user activity (Norman, 1986) also envisages a cycle of interaction. Readers may also be interested in parallel developments in research on process control and human reliability analysis (Hollnagel, 1993).

Neither Card et al. (1983) nor Norman (1986) makes explicit how effects of actions on the environment should be abstracted. This is a crucial omission as an abstraction for describing the environment is necessary if one is to describe how one recognise–act cycle leads to another. There are several notations for describing the behaviour of computer systems that may be suitable for this purpose (see Monk and Gilbert, 1995). Existing cyclic interaction models in HCI (Kieras and Polson, 1985; Howes and Payne, 1990; Howes, 1994; Kitajima and Polson, 1995) have separate models for user and computer system. Alternatively, Monk (1998, submitted) demonstrates how one may build an integrated model of system and user using state-transition modelling. It remains to be seen whether these modelling techniques can be generalised to other kind of behaviour.

Cyclic interaction is an account of human information processing stressing the intimate connections between: intention, action and the environment. Intention (goals) in conjunction with some perception of the state of the environment leads to action having some effect on the environment. These effects lead to changes in what is perceived and to new goals leading to new actions and so on. The notion of

intention is at the centre of this account and it is the role of intention that should be the focus of the research agenda that it implies. Under what conditions does the discrepancy between a desired state of the environment and a perceived state of the environment lead to action? How do the effects of an action on the environment lead to new goals? How do goals condition what is perceived about the environment in this process? By asking these questions we can begin to provide an account of the continuous interaction between organisms and environment that is human behaviour.

Acknowledgements

This work was supported by the UK Joint Council Initiative in Cognitive Science and HCI and the ESRC Cognitive Engineering programme. The author would like to thank the numerous people who commented on drafts, particularly Bob Fields, John McCarthy, Jean McKendree, Leon Watts and Peter Wright.

References

- Beek, P.J., Peper, C.E., Stegeman, D.F., 1995. Dynamical models of movement coordination. *Human Movement Science* 14, 573–608.
- Berry, D.C., 1991. The role of action in implicit learning. *Quarterly Journal of Experimental Psychology* 43 (A), 881–906.
- Card, S.K., Moran, T.P., Newell, A., 1983. *The Psychology of Human-Computer Interaction*. Lawrence Erlbaum, Hillsdale, NJ.
- Cognitive Science, 1993. Special issue: Situated Action, vol. 17(1), pp. 1–147.
- Dennett, D.C., 1990. True believers: the intentional strategy and why it works. In: Lycan, W.G. (Ed.), *Mind and Cognition: A Reader*. Blackwell, Oxford, pp. 150–167.
- Fitts, P.M., Posner, M.I., 1968. *Human Performance*. Brooks Cole, New York.
- Gibson, J.J., 1979. *The Ecological Approach to Visual Perception*. Houghton-Mifflin, Boston, MA.
- Gray, W.D., John, B.E., Stuart, R., Lawrence, D., Atwood, M.E., 1990. GOMS meets the phone company: analytic modelling applied to real-world problems. In: Diaper, D., Gilmore, G., Cockton, G., Shackel, B. (Eds.), *Human-Computer Interaction – INTERACT '90*. Elsevier Science, Amsterdam, pp. 29–34.
- Greatbatch, D., Heath, C., Luff, P., Champion, P., 1995. Conversation analysis: human-computer interaction and the general practice consultation. In: Monk, A.F., Gilbert, N. (Eds.), *Perspectives on HCI: Diverse Approaches*. Academic Press, London, pp. 199–222.
- Hanson, S.J., Kraut, R.E., Farber, J.M., 1984. Interface design and multivariate analysis of UNIX command use. *ACM Transactions on Office Information Systems* 2, 42–57.
- Hollnagel, E., 1993. *Human Reliability Analysis: Context and Control*. Academic Press, London.
- Howes, A., 1994. A model of the acquisition of menu knowledge by exploration. In: Adelson, B., Dumais, S., Olson, J. (Eds.), *CHI '94 Conference Proceedings: Human Factors in Computing Systems – Celebrating Interdependence*. ACM, New York, pp. 445–451.
- Howes, A., Payne, S.J., 1990. Display-based competence: towards user models for menu-driven interfaces. *International Journal of Man-Machine Studies* 33, 637–655.
- Howes, A., Young, R.M., 1996. Learning consistent, interactive and meaningful device methods: a computational approach. *Cognitive Science* 20, 301–356.
- Hutchins, E., 1994. *Cognition in the Wild*. MIT Press, Cambridge, MA.
- Kieras, D.E., Polson, P.G., 1985. An approach to the formal analysis of user complexity. *International Journal of Man-Machine Studies* 22, 365–394.

- Kitajima, M., Polson, P.G., 1995. A comprehension-based model of correct performance and errors in skilled, display-based, human–computer interaction. *International Journal of Human–Computer Studies* 43, 65–99.
- Landauer, T.K., 1987. Relations between cognitive psychology and computer system design. In: Carroll, J.M. (Ed.), *Interfacing Thought: Cognitive Aspects of Human–Computer Interaction*. MIT Press, Cambridge, MA, pp. 1–25.
- Larkin, J.H., 1989. Display-based problem solving. In: Klahr, D., Kotovsky, K. (Eds.), *Complex Information Processing: The Impact of Herbert A. Simon*. Lawrence Erlbaum, Hillsdale, NJ.
- Lave, J., 1988. *Cognition in Practice*. Cambridge University Press, Cambridge, UK.
- Mayes, T.J., Draper, S.W., McGregor, M.A., Oatley, K., 1988. Information flow in a user interface: the effect of experience and context on the recall of MacWrite screens. In: Jones, D.M., Winder, R. (Eds.), *People and Computers 4*, Cambridge University Press, Cambridge, UK, pp. 275–289.
- Miller, G.A., Galanter, E., Pribram, K.H., 1960. *Plans and the Structure of Behaviour*. Holt, Rinehart and Winston, London.
- Monk, A.F., 1998. Modelling cyclic interaction. *Behaviour and Information Technology*, submitted.
- Monk, A.F., Gilbert, N. (Eds.), 1995. *Perspectives on HCI: Diverse Approaches*. Academic Press, London.
- Nardi, B.A. (Ed.), 1996. *Context and Consciousness: Activity Theory and Human–Computer Interaction*. MIT Press, Cambridge, MA.
- Newell, A., 1980. Reasoning, problem solving, and decision processes: the problem space as a fundamental category. In: Nickerson, R. (Ed.), *Attention and Performance VIII*. Lawrence Erlbaum, Hillsdale, NJ.
- Newell, A., 1990. *Unified Theories of Cognition*. Harvard University Press, Cambridge, MA.
- Newell, A., Simon, H.A., 1972. *Human Problem Solving*. Prentice Hall, Englewood Cliffs, NJ.
- Norman, D.A., 1981. Categorisation of action slips. *Psychological Review* 88, 1–15.
- Norman, D.A., 1986. Cognitive engineering. In: Norman, D.A., Draper, S. (Eds.), *User Centered System Design: New Perspectives on Human–Computer Interaction*. Lawrence Erlbaum, Hillsdale, NJ, pp. 31–61.
- Polson, P.G., Kieras, D.E., 1984. A formal description of users' knowledge of how to operate a device and user complexity. *Behavior Research Methods Instruments and Computers* 16, 249–255.
- Rayner, K., Pollatsek, A., 1989. *The Psychology of Reading*. Prentice Hall, Englewood Cliffs, NJ.
- Reason, J., 1990. *Human Error*. Cambridge University Press, Cambridge, UK.
- Suchman, L.A., 1987. *Plans and Situated Actions: The Problem of Human–Machine Communication*. Cambridge University Press, Cambridge, UK.
- Trudel, C., Payne, S.J., 1995. Reflection and goal management in exploratory learning. *International Journal of Human–Computer Studies* 42, 307–339.
- Turvey, M.T., Carello, C., 1986. The ecological approach to perceiving-acting: a pictorial essay. *Acta Psychologica* 63, 133–155.
- Wright, P.C., Monk, A.F., 1989. Evaluation for design. In: Sutcliffe, A., Macaulay, L. (Eds.), *People and Computers 5*. Cambridge University Press, Cambridge, UK, pp. 345–358.