Scoping analytical usability evaluation methods: a case study

Ann E. Blandford[1], Joanne K. Hyde[2], Iain Connell[1] & Thomas R. G. Green[3]

[1] UCL Interaction Centre
University College London
Remax House, 31-32 Alfred Place
London WC1E 7DP
A.Blandford@ucl.ac.uk

[2] Dept of Computer Science
University of Bath
Claverton Down
Bath BA2 7AY

[1] Dept of Computer Science
University of Leeds
Leeds LS2 9JT

Abstract

Analytical usability evaluation methods (UEMs) can complement empirical evaluation of systems: for example, they can often be used earlier in design, and can provoke reflection and insight not readily available from user observations. Here, we report on a case study on the use of seven analytical UEMs, focusing particularly on the scope of each. These seven techniques were applied to a robotic arm interface, and the findings were systematically compared against video data of the arm in use. The usability issues that were identified could be grouped into five categories: system design; user knowledge; conceptual fit between user and system; physical issues; and contextual ones. Other possible categories such as user experience did not emerge in this particular study. Each analytical technique was found to focus attention on just one or two categories of issues. This approach has identified commonalities and contrasts between techniques and provided accounts of *why* a particular technique yielded the insights it did. As well as assessing the scope, strengths and limitations of each technique, we discuss the nature of expertise in applying analytical UEMs. Expertise has been found to embrace the ability to apply a particular notation or method, including identifying an appropriate level of abstraction for generating the system description; more general expertise in human–computer interaction; and knowledge of the particular kind of system being evaluated and its context of use.

Keywords

Usability evaluation; UEMs; expertise; craft skill; GOMS; Cognitive Walkthrough; STN; Z; PUM; EMU; ESDA; CASSM.

Introduction

Over the years, many analytical usability evaluation methods (UEMs) have been developed, each with a different theoretical basis, or addressing a particular class of usability problems. For example, TAG (Payne & Green, 1986) focuses on the consistency of command structures – most directly relevant to command line interfaces – while FKS (Johnson & Hyde, 2003) focuses on task knowledge structures for collaborative working. Although UEMs have been developed from different theoretical perspectives, studies which have attempted to compare UEMs have tended to rely on usability problem count as the main dependent variable, rather than articulating in detail the different methods' scope and applicability. While some general trends can be deduced, it is difficult to extract from these studies any firm conclusions as to how, and in what circumstances, the different methods might be applied, and at what issues they might be directed. It is therefore difficult to assess

the extent to which the various methods are complementary, contradictory, or overlapping.

For example, while studies which have compared Heuristic Evaluation (Nielsen, 1994) with other methods are in general agreement that heuristic evaluation is good at finding a wide spread of general usability problems (Virzi *et al*. 1993, Cuomo & Bowen 1994) at comparatively low cost (Jeffries et al. 1991, Nielsen & Phillips 1993), other methods such as cognitive walkthrough may be necessary in order to focus on specific, task-related problems or re-design issues (Cuomo & Bowen 1994, Desurvire 1994, Dutt *et al*, 1994). However, the analyst must decide whether a problem is general, specific or task-related. There is stronger agreement that inspection methods (including heuristic evaluation and cognitive walkthrough) and user testing identify usability issues of different sorts and scope (Bailey *et al*. 1992, Karat 1994, Desurvire 1994, Karat 1997), with inspection being more effective in the earlier stages of the development cycle (Jeffries & Desurvire 1992, Karat *et al*. 1992, Desurvire 1994, Karat 1997). However, the precise nature of this difference is not well understood (Karat 1997). Moreover, studies which have attempted to compare the predictive potential of heuristic evaluation and cognitive walkthrough (e.g. Desurvire *et al*. (1992), Sears (1997), Cuomo & Bowen (1994)) are in little agreement as to the proportion of empirical problems which might be successfully identified by the two methods.

A notable exception to the lack of attention given to method scope is the work of John and Kieras (1996a; 1996b), who present a clear account of what particular usability questions each of four variants of GOMS is suitable for addressing. In the work reported here, we take a similar perspective to that of John and Kieras, namely that one central consideration in selecting a UEM is what *kinds* of insights it will yield. As well as exploring the scope of selected UEMs, the study has yielded findings about the nature of expertise in applying UEMs.

As discussed more fully below, Gray and Salzman (1998a) have criticised the UEM literature on two counts of validity. They specifically criticise the use of problem count as a measure of the effectiveness of a UEM, recommending that researchers limit both their expectations and their claims for UEM studies. Gray and Salzman (1998b) express a belief in triangulation and pluralisation of methods, and the view that "nonexperimental forms of empirical enquiry ... are not simply sloppy experiments, but have their own requirements of methodological rigour" (p329). In this paper we adopt one form of nonexperimental enquiry, namely a case study of applying seven analytical techniques to the same portion of a real-world interface, in order to assess technique scope.

Background

Evaluating UEMs

Evaluation of UEMs can take many different forms and address various questions. Ultimately, what matters is what the costs and benefits of applying any particular UEM are. Costs include the time and effort it takes to learn a particular UEM, and then to apply it to a particular system; benefits include the insights obtained from applying a particular UEM. Other considerations might include how well a UEM fits within ongoing design practice and how easy it is for different evaluators to apply the same technique consistently.

Largely as a result of the critique on grounds of validity mounted by Gray & Salzman (1998a), UEM practice has moved beyond simple head-to-head comparisons employing usability problem count as the sole measure, to consider criteria such as the following.

- Internal validity (reliability) – the extent to which different analyses of the same system, using the same UEM, yield the same insights. Hertzum and Jacobsen (2001) report on studies of the evaluator effect, showing that different evaluators typically identify broadly different sets of problems, whether the technique under study is the comparatively loose Heuristic Evaluation (Nielsen, 1994) or the more constrained Cognitive Walkthrough (Wharton *et al*, 1994) or even think-aloud protocols. Jacobsen *et al* (1998) focus particularly on how analysts working with the same UEM assessed the severity of problems and again found very little agreement between analysts.
- External validity – the extent to which the findings from analyses conform to those identified when the system is used in the 'real world'. Gray & Salzman (1998a) and Lavery *et al* (1997) adopt a distinction between what Lavery *et al* term 'validity' (whether the UEM suggests observed problems or, conversely, "false positives") and 'effectiveness' (the proportion of problems predicted that are also revealed during user testing). Cockton *et al* (2003) report that encouraging analysts to reflect on their judgements when using Heuristic Evaluation can result in greatly improved validity of results, although the paper presents little detail of the empirical results against which the analytical findings are assessed for coming to this conclusion. Sears (1997) uses three ratio measures of UEM effectiveness ('thoroughness', 'validity' and 'reliability') to assess the differences between observed problems and predictions.
- Productivity – the number of problems a UEM identifies. This measure is probably the most widely discussed; for example, John and Marks (1997) present counts of the number of problems identified by each of six UEMs, each used by a single analyst, when assessing the same interface. Although the authors state clearly that this is a case study, not an experiment, the simple presentation of these figures in a table strongly suggests comparability of the UEMs on this dimension.
- The practicalities – what is needed to integrate techniques within design practice. This is the focus of work by, for example, Karat (1994).
- Persuasive power – the ability of an analyst working with the UEM to persuade a developer to change the system as a consequence of problem identification. This was one focus of the John and Marks (1997) study. They went further to consider whether any resulting changes were ultimately beneficial to usability, although as a case study the findings were somewhat inconclusive, serving more to point to directions for further work than to give definitive answers to such complex questions.
- Analyst activities – what analysts do when applying a UEM. To the best of our knowledge, no thorough treatment of this question has yet been conducted, but John and Packer (1995), John and Marks (1997) and Jacobsen and John (2000) present case studies that contribute to the picture of how people work with UEMs, with a particular focus on Cognitive Walkthrough. These studies

include a consideration of how techniques are effectively learnt. Of particular relevance to the study reported here is the finding of Jacobsen and John (2000) that the participant who had access to multiple descriptions of CW fared better with it than the participant who only had access to one publication on the technique – although a comparison of just two individuals is not reliable.

- Scope – what kinds of problems a technique is and is not good for finding. As discussed above, we are aware of only one study which addresses this aspect of UEM effectiveness in detail, namely that of John and Kieras (1996a; 1996b) on the scope of four GOMS variants. Even Gray and Salzman (1998a) appear to believe that UEMs should ideally have total coverage of the space of possible problems, stating that they are seeking "evidence that various analytic- and empirical-UEMs do indeed converge upon the same set of usability problems" (p243).

One other point worthy of note from the study of John and Marks (1997) is that one of their subjects simply read the design specification several times, and identified many problems that way – provoking the authors to conclude that "Just reading a prose specification many times seems to be as effective as more structured UEMs at detecting and fixing novice-user problems." (pp199-200)

Methodologically, the comparison of UEMs is rife with traps. There are so many variables – from evaluator experience to the systems used in case studies – and so many possible questions that the landscape of possibilities is enormous, and any one study can only hope to map out a very small portion of the territory. This has its disadvantages: as Gray and Salzman (1998a) note, the small, more carefully designed studies tend to have less impact than the ones that make stronger claims.

The study reported here minimises the pitfalls identified by Gray and Salzman (1998a) by adopting a clear focus on the *types* of usability problems and issues identified by the seven methods, rather than comparing different varieties of problems counts. The reanalyses are also inspectable (Blandford & Hyde, 2004), so that others can see how the conclusions are derived. Other ways in which we believe we have minimised pitfalls are discussed below (see Discussion).

Expertise in applying UEMs

As far as we can ascertain, no studies that explicitly address the question of what constitutes expertise in evaluating interactive systems have been conducted to date. Similarly, little work has been done on what it takes to apply a particular evaluation technique other than the studies reported above. John and co-workers (e.g. John & Packer, 1995; John & Marks, 1997) have started to investigate, through diary studies, what it takes to be able to learn and apply particular UEMs effectively. Similarly, Blandford *et al* (1998) studied students learning PUM; they found that students often had difficulty distinguishing between appropriate and inappropriate representations (e.g. when simplifying their description of a design), and that students appeared to get so focused on producing an appropriate representation that they sometimes lost sight of the fact that the representation was simply a tool to support reasoning. These issues of appropriate representation and loss of focus emerged also in the study reported here, so we revisit these issues below (see Discussion).

Within HCI more generally, there has been some consideration of the nature of expertise – focusing on design rather than evaluation, and considering expertise in

relation to craft skill. Long and Dowell (1990) propose that HCI might be a craft, science or engineering discipline, and consider the consequences of each of these views. They consider craft knowledge to be experiential; scientific knowledge to be explicit, formal, testable and generalisable; and engineering knowledge to be formulated in terms of engineering principles. This suggests that craft knowledge lacks rigour or accountability. Rauterberg (2003) makes a similar point, arguing that HCI is still in an "explorative phase", in which the connections between inputs and outputs are largely mysterious. Although the emergence of design and evaluation methods indicates a maturing of the discipline, and possibly a shift towards a stronger engineering base, there is still a poor understanding of when, why or how particular techniques support design or evaluation.

Having craft skill in applying a particular UEM implies having extended experience of doing so – probably to the extent that the skills involved have become automatised, creating a degree of expertise in applying that UEM to designs. Dreyfus and Dreyfus (1985) propose five stages in the development of expertise, from a novice stage where rules are learnt and applied for manipulating context-free elements, to advanced beginner who begins to understand the domain and see meaningful aspects, to competent performer, who learns to set goals and interpret the current situation in terms of what is relevant to achieving those goals, to proficient performer who views a situation as having a certain significance tending towards a certain outcome such that aspects of the situation stand out as salient in relation to that outcome, to the expert, who is able not only to perceive the situation but also to rapidly generate appropriate solutions. Taking a less analytical approach, Klein (1998) has examined the development of experience and expertise, and suggests that experienced people rely to a large degree on pattern-matching, where they identify familiar elements in a situation, and that expertise can be developed by having many experiences, as well as quick and accurate feedback, and time to reflect and learn from the experiences – further reinforcing the notion that expertise is strongly linked to craft skill.

Dreyfus (1992) suggests that experts structure their understanding of their experiences so as to focus on events and objects that are relevant to their current purpose. This suggests that the value of a technique is not only in what it is able to represent, but also in what it can contribute to an experienced analyst. In the Discussion (below), we revisit this idea in the context of the findings of this study.

Setting the scene

Context of the work and methodology

The work reported here was not initially conceived as a single structured study, but evolved into its current form, which we present briefly for the sake of completeness and transparency. The initial aim of the work was to develop and test a rigorous, analytical approach to usability evaluation (EMU) that extended existing approaches to address multimodal usability issues such as modality clashes – e.g. a user being expected to read text while speaking different text. Part of the preparation for this involved reviewing existing analytical evaluation techniques that might form a basis for the new technique. (Another part involved developing a taxonomy of modalities, but that is outside the scope of this paper.)

Five techniques were selected as a starting point; they were all formal or semi-formal, with a theoretical and/or representational basis. Some (most notably Z) focus

primarily on the use of a notation to describe a system clearly; others (most notably Cognitive Walkthrough) focus primarily on method, with a relatively informal description language. The techniques were chosen as representing a range of formality, and having different base-line assumptions about users; for example, GOMS assumes experts, while Cognitive Walkthrough assumes novices learning through exploration. There were two system-oriented description techniques (Z (Spivey, 1989) and STN (Dix *et al*, 1993)), two established user-oriented techniques (GOMS (Card, Moran & Newell, 1983) and Cognitive Walkthrough (Wharton *et al*, 1994)) and one user-oriented technique that had been developed locally (PUM: Young, Green and Simon, 1989; Blandford and Young, 1996). Z and STN are not standard usability modelling techniques, being generally used in software engineering to describe the specification and functionality of a system. They were included to see what leverage well-known techniques with no explicit usability analysis support could give to the understanding of the interface, against which other usability-specific techniques could be compared. These approaches represent some of the more formal modelling techniques, but are not intended to be definitive. Indeed, other approaches, for instance Petri Nets (e.g. Bastide & Palanque, 1990), UAN (Hartson, Siochi & Hix, 1992) or Task-Action Grammar (e.g. Payne & Green, 1986), would have been equally applicable. Other techniques such as syndetics (Duke *et al*, 1998) and ICS Cognitive Task Analysis (Barnard & May, 1999) were not used because there is little published information on their application to interface analysis. The five selected techniques were all applied to the interface for a robotic arm, as described below. Following the STN analysis, feedback was given to the arm developer, who then implemented backtracking (correcting usability issue 2) and consolidated "continue" and "go" into one operation (issue 5); for consistency, all subsequent techniques were assessed to establish whether they would have identified these issues or not.

The results of the work on a modality taxonomy and the experience of applying the five analysis techniques formed the basis for the design of EMU (Evaluating Multimodal Usability: Hyde, 2002). EMU was itself then subjected to evaluation, by teaching it to novice users and by applying it to the same interface as the five earlier techniques.

As further validation of EMU, we compared the findings of all six UEMs (EMU plus the five applied earlier) to empirical data of the robotic arm in use. Unfortunately, during the development of EMU, the robotic arm system had been destroyed in a flood and development was abandoned, so for this we had to rely on video data of the prototype system in use that had been collected before the flood.

By this point, it was very clear that some of the usability findings identified using each technique could be attributed to the method, but that others were fortuitous, due to the general craft skill of the analysts or our growing understanding of the interface. Therefore, a rational reanalysis was conducted to systematically identify which insights could be attributed to the technique, which to craft skill, etc., and also to identify which usability difficulties *should* have been identified using each technique but were not.

Shortly after the completion of this relatively structured study, we were developing a further evaluation technique, CASSM (Concept-based Analysis of Surface and Structural Misfits, formerly known as OSM: Connell, Green & Blandford, 2003), and again the question of scoping arose. Therefore, a further analysis of the same robotic arm, based on the description presented below, was conducted. All the earlier data and

analyses were revisited and expanded to include the new insights derived from the CASSM analysis. Although this final analysis was conducted retrospectively, every effort has been made to apply the same degree of rigour to this final analysis as to earlier ones.

Case study: the robotic arm

The system chosen for analysis was a robotic manipulator for use by wheelchair-bound people (Parsons *et al*, 1995; 1997). This was chosen because the interface was multimodal, the system was relatively simple (so that learning and applying several evaluation techniques was a tractable proposition) and the system was still under development (so that the analyses could actually inform design). The manipulator was intended to be used in a domestic context for everyday tasks such as feeding and grooming, and was developed primarily to prove that a sophisticated manipulator could be produced at a reasonable cost: usability issues were considered informally, if at all. The arm consisted of eight joints, powered by motors, which could move either individual joints or the whole arm at once, via the input devices. The user could either move joints explicitly (selecting the joint and direction of movement) or make use of pre-taught positions that were programmed in; in this study we focus on explicit movement.

The input devices interfaced to a Windows-based application which in turn sent motor control commands to a dedicated microprocessor that controlled the movement of the arm. The interface was based on menu selection. Three different devices could be used: a standard mouse; voice recognition; and a gesture-based interface. The voice recognition system allowed direct menu option selection simply by saying the menu option out loud. It was designed to be trained to individual voices. The gesture input system was based on a baseball cap with two sensors: one detecting movement forwards and backwards, the other detecting movement left and right. This allowed a variety of distinct gestures to form the gesture vocabulary. The gesture system was implemented so that a cursor moved along underneath the menu options cyclically, and an option was selected by making the correct gesture when the cursor was underneath that option. Another gesture acted as a toggle between high and low speed of the cursor. A final gesture was an escape option, which automatically stopped the arm and returned the user to the main menu. A high-level description of the menu options and possible transitions is provided in Figure 1. (This version includes backtracking and omits the "continue" option due to changes in the interface design following our initial analyses, as discussed above.)

For the purpose of analysis, only one task was considered, which exercised only part of the interface. However, the task is one that would be very common to all users, and would therefore give valuable information on the usability of the interface – namely, to move the robotic arm to a certain position, without making use of any pre-taught positions, as though it were to be used to (for example) turn on a light switch. It is this kind of task that the developers of the arm consider to be a basic task, and that should be part of the core functionality of the interface. From the main menu of the application this covers the options *move* and *movearm*. *Move* allows the user to specify a particular arm joint and in what direction it can be moved, as well as controlling its speed. *Movearm* allows the user to move the arm as a whole in a particular direction. At the time of analysis, there was no feedback to the user about settings other than that provided by the visual feedback of the arm's position.

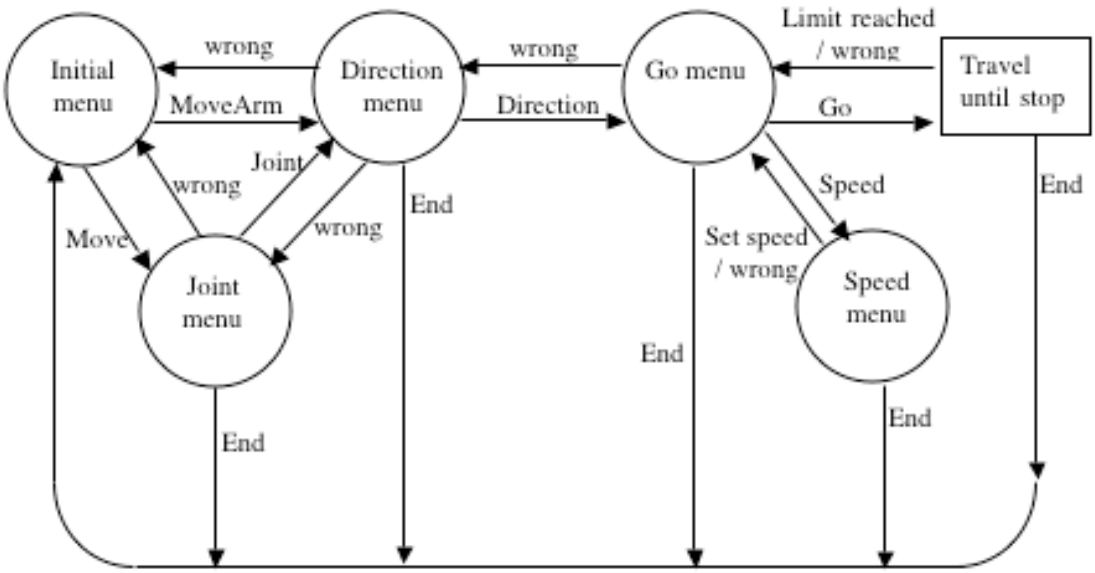


Figure 1: The second STN diagram produced (including error correction).

A selection of techniques

As discussed above, seven semi-formal and formal analytical evaluation techniques were applied in the course of this study. The seven techniques are summarised in Table 1. Most techniques focus attention on either the device or the user. The two most recently developed methods (EMU and CASSM) aim to consider user and device equally, focusing on features of the interaction between these system components.

Technique	Primary source of description	Focus	Developed locally?	Key features
State Transition Networks (STN)	Dix *et al* (1993)	Device	No	Diagrammatic
Z	Spivey (1989)	Device	No	Formal notation based on set theory and first-order predicate logic
Cognitive Walkthrough (CW)	Wharton *et al* (1994)	User	No	Clearly defined method; natural language; goal-based
GOMS	John & Kieras (1996b)	User	No	Highly structured; hierarchical; goal-based
Programmable User Modelling (PUM)	Blandford, Good & Young (1998)	User	Yes	Highly structured; based on means-ends planning
EMU	Hyde (2002)	Interaction	Yes	Clearly defined method, focusing on multimodal issues
CASSM	Blandford, Connell & Green (2003)	Interaction	Yes	Semi-formal, focusing on conceptual misfits between user and device

Table 1: overview of techniques applied in this analysis

Each technique is described briefly, including illustrative extracts from the analysis of the robotic arm. These descriptions are, of necessity, presented at a high level, to give a flavour of each approach, rather than giving the detail that would be needed for learning to apply the UEMs.

STN

The State Transition Network representation of the interface was originally created to clarify understanding of the robotic arm in terms of the structure of the interaction rather than for any usability assessment. STNs are a popular and well-established way of diagrammatically representing an interface (Dix et al, 1993) and can take various forms. For simple interaction sequences, STNs can clearly illustrate the flow of interaction and allow redundant cycles to be identified. The simplest type, as used here, has each state of the system represented by a circle, linked by lines, or transitions, which correspond to the actions necessary to move from that state to another. Figure 1 shows an STN diagram for the latest prototype of the arm controller, as discussed above.

Z

While also being system-oriented, Z (Spivey, 1989) contrasts with STN in being a formal specification notation based on set theory and first order predicate logic. It makes use of schemas, which are collections of named objects with relationships specified by axioms. These schemas can be built up to define large specifications. The mathematical base of Z means that it can be considered to be unambiguous, which makes it a powerful notation for communicating ideas and concepts, and can allow the designer to gain an insight into the structures and relationships that are of importance, and to manipulate those relationships and examine the implications of change. Figure 2 shows an extract from the Z analysis, presenting the schemas defining the joint ('armpart') to be moved and the direction to move it in.

In the following two schemas, *armPartToBeMoved* and *dir* are, effectively, state variables. However, they have not been included in the state schema, because it is possible for the robotic arm to exist in a state in which neither of these variables is defined.

SelectArmPart

Δ RoboticArm
armPartToBeMoved' : ARMPART
armPartToBeMoved? : ARMPART
---
armPartToBeMoved' = armPartToBeMoved?

SelectArmDirection

Δ RoboticArm
dir' : DIRECTION
dir? : DIRECTION
---
dir' = dir?

Figure 2: Extract from Z description showing specifications of which arm part to move, and specifying its direction of motion

CW

In contrast to the first two techniques, Cognitive Walkthrough (Wharton *et al*, 1994) is a cognitively based method. It is designed to uncover usability issues by following the sequence of actions a user would take to perform a set of tasks agreed by the analysts, and by analysing at each stage how successful the user would be in performing the action correctly. The method takes a task-oriented perspective, in that it considers the goal structure and the ways goals are addressed in completing the task. At every stage the interface is evaluated by answering set questions to determine whether or not it provides the necessary information for the user to successfully continue with the task, and what feedback the interface provides to the user. The analysis of user actions is done in terms of success and failure stories. Cognitive Walkthroughs concentrate on ease of learning; this perspective is justified by the fact

that users tend to learn features of an interface as they need to, rather than all at once. Therefore, ease of learning is seen as essential to interface usability. An extract from the CW analysis is presented in Figure 3, including both success and failure stories.

1. Choose option **MoveArm** on the main menu.
Feedback: The interface gives a list of options: **Left, Right, Up, Down, Forward, Back, Wrong, End**
This action is used when the user wishes to move the arm as a whole. The usability issues uncovered in response to the questions are shown below.
Will the users try to achieve the right effect?
Success story: the user knows how to interact with the system using the voice and gesture input devices, because they have been trained beforehand. They can see the arm and visually assess what movement is needed.
Possible failure story: the user may not know the difference between the options **MoveArm** and **Move**, and may be confused when trying to decide between them.
The user may not know (although this is less likely) that the arm can be moved as a whole instead of individually.
Will the user notice that the correct action is available?
Success story: the user will know that they can select an option by nodding their head in a particular manner or by vocalising the word, because they will have been trained beforehand. The cursor will prompt the use of the gesture, the menu names will prompt the use of the voice input device.
Will the user associate the correct action with the effect trying to be achieved?
Success story: the user will know how to select an option to move on to the next menu because of previous training.
Possible failure story: there may be a problem if the user moves their head to look from the interface to the arm and back, in that according to how the gesture system is implemented it may be interpreted as a command. There may be a similar problem if the user is engaging in a conversation while the voice input system is operational, although this is unlikely due to the small number of possible menu names. There may be a possible implementation problem if the user pauses in middle of saying "Move arm".
If the correct action is performed, will the user see that progress is being made toward solution of the task?
Success story: the user will know that progress has been made because the next menu, detailing the next stage of interaction will appear, and the user will know from training that this new menu relates to the next stage of interaction.
Possible failure story: the menu that appears after the **MoveArm** option is chosen gives no indication that the whole arm is going to be moved. Feedback to the user will need to be considered at this point so that the user will know what options have been chosen.

Figure 3: example extract from CW analysis, focusing on the step where the user should select 'MoveArm' from the main menu, illustrating both success and failure stories.

GOMS

GOMS (Card *et al*, 1983) is also a cognitively based method, but more formal than CW. It is based on the idea of the human as an information processor. GOMS stands for Goals, Operators, Methods, and Selection rules, and is based on the premise that a user achieves goals by breaking them down into sub-goals which can then be separately achieved. The Operators are the ways available to accomplish the goals, Methods are defined sequences of operators and goals, and Selection rules determine how to choose between more than one method (John and Kieras, 1996b). The emphasis is not just on the physical aspects of interaction, but also on mental processes — for example, what the user has to know or remember. Varieties of GOMS address goal hierarchies, working memory load, schedule tasks, lists of operators, and production systems.

The interface to the robotic arm was first analysed using CMN-GOMS (John and Kieras, 1996b), which is based on the Keystroke Level Model developed by Card, Moran and Newell (1983). This version of GOMS was chosen as being comparatively easy to learn. It has a strict goal hierarchy, with each method represented as a series of steps that are performed in sequence. The analysis was then taken down to a CPM-GOMS (John and Kieras, 1996b) level in order to examine more fully the cognitive, motor and perceptual aspects of the interaction. CPM stands for either Cognitive-

Perceptual-Motor or Critical Path Method, and is based on the assumption that tasks needing different processes within the Model Human Processor as put forward by Card *et al* (1983) can be performed in parallel. To use CPM-GOMS, a CMN-GOMS analysis is first done to determine the goal hierarchy and methods in order to obtain the basic perceptual, cognitive and motor operators, which are then expressed using schedule charts (e.g. Figure 4).

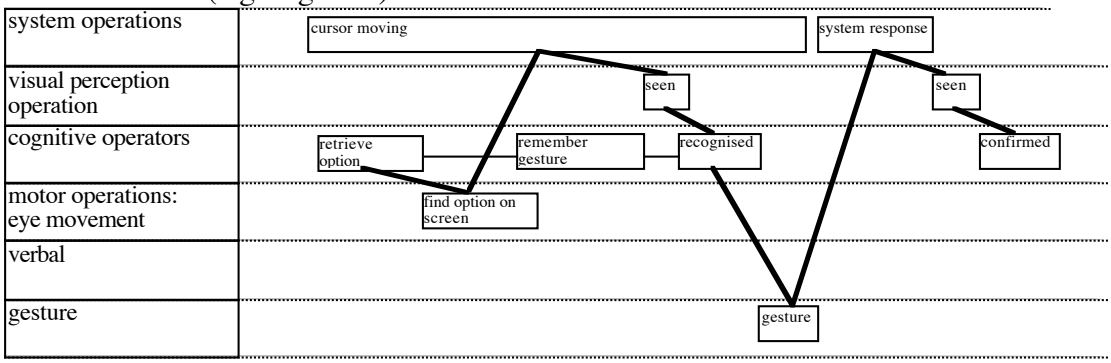


Figure 4: Example schedule chart for CPM-GOMS, showing the sequence of system and user operations for gestural input.

PUM

PUM is the final 'established' method used in the study. Like CW and GOMS, it has a cognitive basis. It focuses on user knowledge, and how that knowledge is used in the interaction to effect changes to the system state. A description of the knowledge that the user needs to operate the interface successfully is written in an Instruction Language (IL), which is then optionally compiled by a cognitive model to simulate predicted user behaviour (Blandford, Buckingham Shum and Young, 1998). Potential user difficulties can be identified both in the ease (or otherwise) with which the analyst can specify the required user knowledge in the IL and in observing the behaviour of the running model (if analysis is taken that far, which is was not in this case). The IL description is relatively formal, consisting of: the conceptual objects that the user manipulates; relations between defined object types; a device description including commands, the initial state, and information displayed to the user; and user knowledge in terms of conceptual operators, initial knowledge, and user task. Figure 5 shows a short extract from the PUM analysis; this extract defines the mental operation 'stop-at', which says that if the user wants to stop the arm with joint J at position L then they need to wait (until the joint is in the right place – or thereabouts, to accommodate reaction times) and then press 'stop', but that they can only do this when the joint in question is moving in the necessary direction (aspects set up by the user, defined by *precondition*s) and if this is technically possible (as defined by *filter*s).

The task goal is achieved by setting the arm moving in the correct direction, then waiting until it gets to the right place (or near it) and pressing "stop". The description is at a level such that "wait then press STOP" is considered as one conceptual operation:

```
operation stop-at (joint: J, location: L)
user-purpose:          joint-at(J,L)
precondition:          direction-specified()=D
                       is-moving(J)
                       joint-specified()=J
filter:                is-within-range(J, L)
                       can-move(J, D)
                       L is in direction D from current position
action:                wait then press STOP
```

Figure 5: short extract from the initial PUM analysis showing the definition of the operation 'stop at'

EMU

EMU (Evaluating Multimodal Usability: Hyde, 2002) was specifically developed to build on the strengths of existing techniques while focusing particularly on multi-modal usability issues. It takes a task-based, procedural form, examining the interaction stage by stage, concentrating on the flow of modalities, and the conflicts and clashes between them. The task is defined, and the modalities are listed. The user, system and environment are profiled and compared to the modality listings in order to find any potential problems. The interaction sequence listing is completed using a notation that describes every step of the interaction in terms of the modalities expressed and received by user and system, and is examined for modality properties and clashes. Each modality is described in terms of three components: whether it is visual, auditory or haptic (currently ignoring the possibilities of it being through taste, smell, etc.); whether the information is expressed in lexical, symbolic or concrete terms; and whether the communication is discrete, continuous or dynamic. For definitions of these terms, see Hyde (2002). The representation of the modalities allows the analyst to keep track of the interaction. Figure 6 illustrates an extract from the EMU analysis of the robotic arm, focusing on the point where the system is displaying a menu with a flashing cursor moving from one option to the next and the user may be looking at the arm or the display (presumably waiting for the desired option to be highlighted so that it can be selected). At this point, the system is expressing (SE) a visual, lexical, continuously displayed modality (the menu display), a visual, symbolic, dynamic modality (the moving cursor) and a haptic, concrete continuous modality (the position of the arm); meanwhile, the user is receiving (UR) visual information that corresponds to either the information on the display or the observable position of the arm.

1.
[SE vis-lex-cont] and [SE vis-sym-dyn] and [SE hap-con-cont]
menu display **moving cursor** **position of arm**

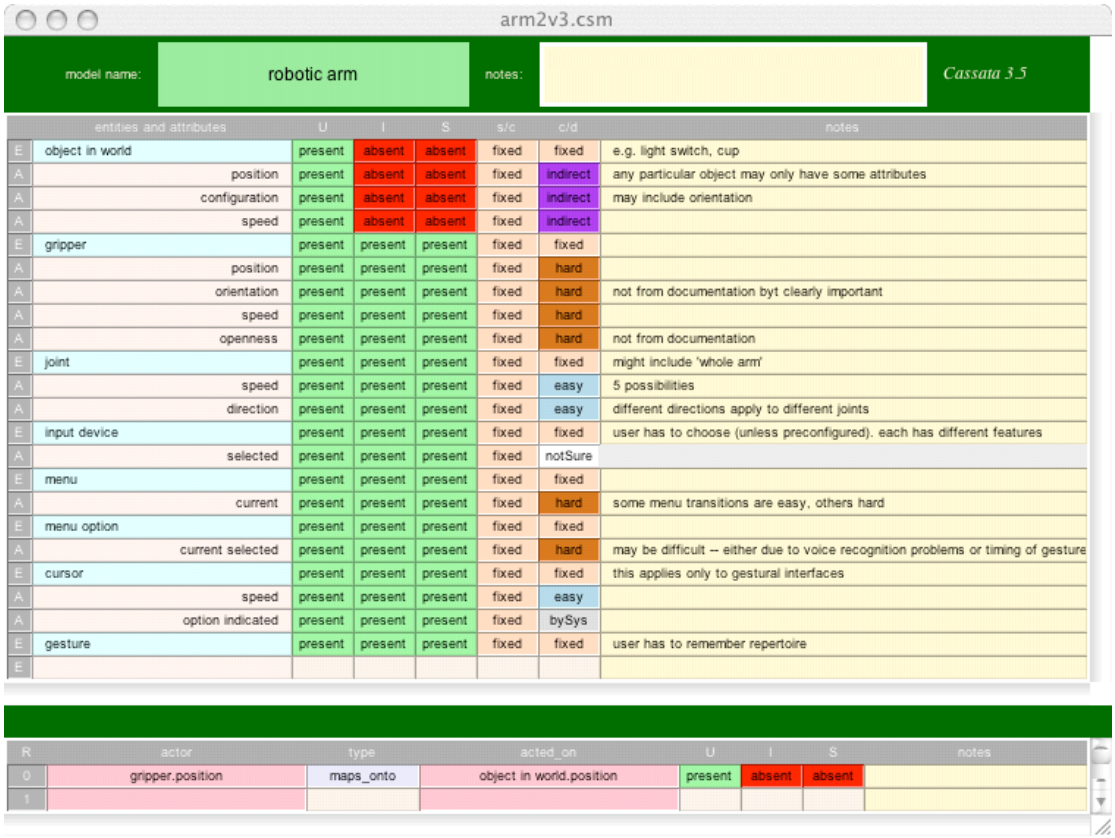
[UR vis-lex-cont] and [UR vis-sym-dyn]
menu display **moving cursor**
precon: [SE vis-lex-cont] precon: [SE vis-sym-dyn]
menu display *moving cursor*
precon: looking at display precon: looking at display
or
[UR vis-con-cont]
position of arm
precon: [SE hap-con-cont]
position of arm
precon: looking at arm

The system is displaying a menu, underneath which is a flashing cursor moving from one option in turn to another. The arm not moving and is at rest. The user is either looking at the menu with the cursor, or at the arm.

Figure 6: short extract from the EMU analysis describing the user looking at the display or the arm

CASSM

Finally, Concept-based Analysis of Surface and Structural Misfits (CASSM: Blandford *et al*, 2003) focuses on structures rather than tasks or procedures. The analyst identifies the main concepts that the user works with, those represented at the interface, and those in the underlying system, and reasons about the quality of fit between the user, interface and system concepts. A system can also be assessed against Cognitive Dimensions (Blackwell & Green, 2003), but in this particular case, this did not identify any additional misfits. The Cassata data analysis tool can be used to support analysis, as illustrated in Figure 7. We recognise that this screen shot is difficult to read, but broadly, the items highlighted in red ('absent'), purple ('indirect') or brown ('hard') (all of which appear darker than other cells in grey-scale) highlight potential difficulties for the user of the system. The full CASSM analysis is available from Blandford (2004).



	entities and attributes	U	I	S	s/c	c/d	notes
E	object in world	present	absent	absent	fixed	fixed	e.g. light switch, cup
A	position	present	absent	absent	fixed	indirect	any particular object may only have some attributes
A	configuration	present	absent	absent	fixed	indirect	may include orientation
A	speed	present	absent	absent	fixed	indirect	
E	gripper	present	present	present	fixed	fixed	
A	position	present	present	present	fixed	hard	
A	orientation	present	present	present	fixed	hard	not from documentation byt clearly important
A	speed	present	present	present	fixed	hard	
A	openness	present	present	present	fixed	hard	not from documentation
E	joint	present	present	present	fixed	fixed	might include 'whole arm'
A	speed	present	present	present	fixed	easy	5 possibilities
A	direction	present	present	present	fixed	easy	different directions apply to different joints
E	input device	present	present	present	fixed	fixed	user has to choose (unless preconfigured). each has different features
A	selected	present	present	present	fixed	notSure	
E	menu	present	present	present	fixed	fixed	
A	current	present	present	present	fixed	hard	some menu transitions are easy, others hard
E	menu option	present	present	present	fixed	fixed	
A	current selected	present	present	present	fixed	hard	may be difficult -- either due to voice recognition problems or timing of gesture
E	cursor	present	present	present	fixed	fixed	this applies only to gestural interfaces
A	speed	present	present	present	fixed	easy	
A	option indicated	present	present	present	fixed	bySys	
E	gesture	present	present	present	fixed	fixed	user has to remember repertoire
E							

R	actor	type	acted_on	U	I	S	notes
0	gripper.position	maps_onto	object in world.position	present	absent	absent	
1							

Figure 7: CASSM analysis of robotic arm, presented using Cassata tool

Method

As discussed above, the work reported here was not originally conceived as a single, structured study. However, it can be rationally reconstructed as such. As a single study, the key steps of analysis were as follows:

1. Analysis of the robotic arm using each of the seven analytical evaluation techniques described above.
2. Exploratory Sequential Data Analysis of 6 video extracts of an individual using the robotic arm. This analysis focused on usability issues.
3. Rigorous re-analysis of the arm using each of the seven techniques, taking the full list of usability issues compiled during steps (1) and (2) and constructing a careful account of why each technique did, should have, should not have or did not identify each issue.

1. Initial analyses

Each initial analysis was conducted by one of the first two authors and checked by at least one other author of this paper. At this stage, while every effort was made to conduct each analysis independent of all other analyses, there were inevitably learning and transfer effects. For some techniques, it was necessary to invest substantial time on learning, by reading as many source documents as possible; for others, learning was negligible. There were also unavoidably effects due to the degree of familiarity with the device (familiarity grew throughout the study). These confounds are discussed at length by Gray and Salzman (1998a); subsequent stages of analysis were designed to minimise the effects of confounds on the final analysis.

An 'issue number' was used to index every usability issue identified. Appendix 1 presents a definition of each issue. There is no significance to the ordering of issues.

2. ESDA analysis of video extracts

A form of Exploratory Sequential Data Analysis (ESDA: Sanderson & Fisher, 1994) was used to analyse the only empirical data that was available for the robotic arm – namely 6 short episodes of use of the system by two individuals. ESDA techniques are observational and empirical, and include task analysis, protocol analysis and video analysis. In this particular case, the form of ESDA used was based on analysing the available video evidence in terms of where the user was looking, when the user made a selection (using whichever input device was featured in the episode), whether the arm was moving or not, whether or not there was any audible noise from the arm, and anything the user said. From this data, any perceptible user difficulties were identified.

The video data comprised six excerpts, each one showing a user performing a specific task and using a particular means of input, as shown in table 2. As indicated, most video excerpts used pre-taught positions, in which the user did not have to explicitly select and move individual arm joints.

EXCERPT:	One	Two	Three	Four	Five	Six
SECONDS:	123	83	89	50	89	525
INPUT:	Mouse	Voice	Gesture	Gesture	Voice	Mouse
TASK:	Feeding	Feeding	Feeding	Feeding	Feeding	Drinking
USER:	Expert	Expert	Expert	Novice	Novice	Novice
POSITIONS:	Pre-taught	Pre-taught	Pre-taught	Pre-taught	Pre-taught	Mixture

Table 2: Summary table of video data excerpts

Since the robotic arm was no longer available, so that we could not tailor trials to closely match the rest of the study, ESDA was one of few approaches that could be used. In practice, excerpt 6, which involved a novice user (an individual with the kinds of movement difficulties that the device was intended to support) manipulating the arm at least partly manually was the most informative. However, as shown in Table 2, all excerpts involved feeding or drinking tasks, rather than the light switch task that was used for the analytical evaluations, so it was necessary to take this difference into account in assessing the findings from the UEMs.

Figure 8 shows an excerpt from the analysis of the final video extract. This shows a disabled user performing a drinking task using the robotic arm. A combination of pre-taught and manually selected options was used, selected using a mouse. The video excerpt shows a straw being lowered into a cup, the cup being filled from a dispenser, and the cup being lifted and positioned in front of the user so that the user can drink from the straw. The drink is then returned to the table. The excerpt shown illustrates the movement of the user's eyes, the motion of the arm, and the audible noises for times 70-95 sec through the video.

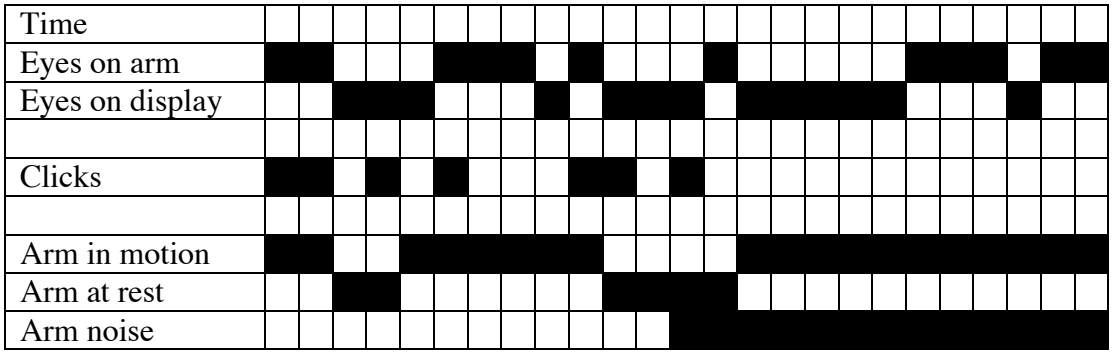


Figure 8: Extract from the modality analysis of excerpt 6 of the video data (showing 70 – 95 sec)

Video evidence was found to corroborate twelve of the usability issues identified, although in some cases the same behavioural phenomenon can be attributed to multiple usability problems, and it is not possible to disambiguate the attribution.

- Issues 12 ('problems of determining left and right, especially when arm contorted') and 13 ('user cannot check direction choice until arm starts to move') could only be assessed through excerpt 6, since the other excerpts used pre-taught positions. The video data shows four instances where all or part of the arm started to move in one direction, only for it to be stopped and moved in the opposite direction.
- Issues concerned with difficulty in positioning the arm (14, 17, 23, 24, 25) were again only applicable to excerpt 6. Video evidence shows various under- and over-shoots where the user had to subsequently correct the position of the arm, implying that an error had occurred. This is at least indicative of user difficulties in judging arm movements and position. On one occasion in excerpt 6, the gripper was poorly oriented for the task, and the user had difficulty seeing it (issue 25).
- One of the users was heard to comment in excerpt 4: "I think it's on slow, innit?", indicating lack of display information about the current speed setting.

There are other issues for which there is inadequate or no video evidence. Of course, the impoverished nature of the available video data makes this somewhat inevitable. These issues are discussed in detail by Blandford and Hyde (2004). There are also a few issues for which it is, in principle, not possible to have video evidence. For example, the redundancy of "continue" (issue 5) would not appear in video data.

One additional usability issue was uncovered in the video data: it was found that the arm itself obscured the user's view at times. Twice in excerpt 6, the user had to move his head substantially to see around the arm.

All the usability issues are summarised in Table 4 (see Results). The video evidence is classified as 'yes' (pretty clear video evidence of issue), 'poor' (some evidence, but not good), 'none' (no video evidence) or 'n/a' (not applicable: video evidence would not make this issue apparent).

3. Re-analyses

As discussed above, the examination of empirical evidence of the robotic arm in use confirmed some of the usability issues identified by the original analyses. However it gave no insight into whether or not those issues were identified in a valid manner, according to the actual claims of the techniques used. Each of the analyses was

therefore systematically re-examined using a single source of description for each particular technique (as shown in Table 1 above), asking the questions: should this technique have supported the identification of this issue, and why (or why not)? This has enabled us to determine whether the usability issues were identified due to the power of the technique, the skill and knowledge of the analyst, or other factors. This, in turn, has enabled us to assess the scope of each approach, at least in relation to the device and task used for this case study. Here, we have instantiated the idea of craft knowledge slightly differently from Long and Dowell (1990), defining craft skill as the analyst using their experiential knowledge *in conjunction with a method* to achieve insights that are informed by the method or notation being used, but not directly derivable from it.

Table 3 shows the possible assessments made in the rational reanalysis. In this table, 'A' issues straddle an ambiguous line between method and craft: had the problem been described differently, these issues would emerge through the method, but selecting the appropriate level of abstraction for the representation is itself a matter of craft skill. The nature of craft skill is discussed in more detail below.

	Was identified	Was not identified
Should have been identified	M: Identified by method	O: Overlooked but should have been identified by method
Could have been found had the problem been described at a different level of abstraction	*[not applicable]*	A: Depends on abstraction level
Could have been identified (through craft skill)	C: Identified through craft skill of analyst	C?: Representation indirectly supports identification, but method does not explicitly
Should not have been identified	*[did not occur]*	[unlabelled]: outside scope of method and representation

Table 3: was an issue identified through the method or craft skill, or could it have been?

The results of this rational re-analysis were then compared with both the original analyses and the results of the video analysis, to determine whether there were substantial differences. Examples of extracts from the re-analyses follow, exemplifying the different cells in Table 3. As noted above, the numbers are the indexes used to label each issue.

Our first example is taken from the re-analysis for Z, and illustrates an issue that is within the scope of the approach and was identified in the initial analysis (M):

> **5. Continue versus Go: Continue seen as redundant**
>
> This issue was identified by the Z specification since both options share the same functionality, and were represented by different schemas with identical contents.

This example from the re-analysis for STN shows an issue that should have been identified but was not (O):

1. Long sequence of operators to move arm

Since the STN shows the number of states that the user has to navigate through before the robotic arm can be moved, this issue should have been identified in the original analysis. That it was not identified shows the extent to which the analysis was dependent on the skill (or lack thereof) of the analyst.

This example from the re-analysis for CASSM illustrates an issue that was identified due to the craft skill of the analyst, rather than directly from the approach (C):

12. Problems of determining left and right, especially when arm contorted

The issue of judging directions when the arm is contorted emerged (with some craft skill) from looking at joints and what the user knows about the directions in which joints can move. It does not emerge directly from the CASSM representation.

The fourth example is taken from the re-analysis for Cognitive Walkthrough, illustrating an issue that is outside the scope of the UEM but might be found by craft skill (C?):

2. Inability to backtrack

CW does not deal with error in terms of its implications, therefore would not find this issue, although it might come out from the craft skill of the analyst through thinking about rectifying errors.

The final example is taken from the re-analysis for PUM, illustrating an issue that would have emerged had the problem been described in more detail (A):

7. Gesture input with twice as many operations as voice because dependent on cursor movement

This did not come out in the original PUM analysis, because the analysis was not written at a low enough level of abstraction for this to be apparent.

The complete rational reanalyses are presented by Blandford and Hyde (2004), and edited highlights are included as Appendix 2 of this paper.

As a further step of analysis, the issues identified were classified into types, reflecting the primary focus of that issue. The classification that emerged comprised five classes:

- (S) System design,
- (K) user Knowledge,
- (C) quality of the Conceptual fit between user and system,
- (P) Physical issues or
- (X) conteXtual ones

This classification is further discussed below. For later reference, the second column of tables 5 and 6 (below) indicates the type of each issue.

Results

A full list of the issues identified by *any* technique (including the video data) is presented as Table 4, using the codes as summarised in Table 3.

	PROBLEM	STN	CW	GOMS	PUM	Z	EMU	CASSM	video
1	Long sequence of operators to move arm	O	**C**	**M**	**C**	C?	**C**		poor
2	Inability to backtrack	**M**	C?	**C**	**C**	**M**			n/a
3	Difficulty of choosing between Move Arm or Move		**M**	**C**	**M**	C?	**M**		none
4	Lack of short cuts	C?	C?	**C**	C?	C?			poor
5	Continue redundant	**M**	C?	**M**	O	**M**			n/a
6	Confusion over joint called Arm		**M**	**C**	**M**	C?	**M**	O	none
7	Gesture input with twice as many operations as voice	A		**M**	A	A			poor
8	Head moved to look at arm while gesture system operational may be interpreted as a command		**C**				**M**	C?	n/a
9	If user pauses in middle of saying "Move arm"...		**C**						none
10	If user engaged in conversation...		**C**				C?	C?	none
11	Lack of feedback about selection		**M**		C?				none
12	Problems of determining left and right, especially when arm contorted		**C**		C?			C?	yes
13	User cannot check direction choice until arm starts to move		**C**						yes
14	Time taken to interact with system to stop arm		**C**	**M**	**C**		**C**		yes
15	Similarity between moving joint and moving whole arm	**M**	C?	**M**	C?	**M**			none
16	Illegal options	C?	C?			**M**			none
17	Mismatch between way that arm works and way that user would move arm		C?		**M**	C?	**C**	**M**	yes
18	Not clear that End returns user to main menu		**M**		**M**		O		none
19	End having two meanings		**M**		**M**		O		none
20	Lighting conditions		C?				**M**		n/a
21	Difficulty for user to move field of vision		C?				**M**		n/a
22	User looking one way, menu options in other direction		C?				**M**	**C**	yes
23	Difficulty of judging arm movements		C?				**M**	**M**	yes
24	Difficulty in judging speed and direction as getting close to target				A			**M**	yes
25	Difficulty in judging position, orientation and aperture of gripper as approaching target							**M**	yes
26	Position and movement of most joints is of limited interest to the user		C?					**M**	none
27	Possible difficulty of timing gesture accurately as cursor moves between options		C?	A				**M**	none
28	Voice recognition problems							**M**	yes
29	Speaking with mouth full…						C?		yes
30	No display of speed				**M**			C?	yes
31	Arm obscuring user's view								yes
32	No arm reversing	C?			**C**				yes
33	Difficult to match names to joints		O		**M**			O	n/a

Table 4: summary table of usability problems
M=found by method; C=found through craft skill
O=overlooked; C?=might have been found by craft skill, but was not;
A=might have been found had problem been described at a different level of abstraction

Tables 5 - 7 present the same data in different formats, to highlight particular issues that are further discussed below.

Table 5 re-structures the data to highlight hits, misses and false positives (FPs) in the analyses. Here a Hit is a usability issue that was identified by using a UEM that was corroborated (in some cases weakly) by the video evidence; a Miss is a usability issue that emerges in the video data but was not identified by a particular method; and a False Positive is an issue that was predicted through analysis, but for which there is no supporting video evidence. This data must be viewed with a fair degree of caution (particularly the false positives) due to the limitations of the empirical data available. However, of particular concern is the number of issues that emerged in the video data (issues 4, 12 13, 29, 31, 32) that were not found through *any* of the methods (although some were identified through the craft skill of the analyst); these issues are highlighted in Table 5. This issue is discussed in more detail below (in the Discussion section).

Table 6 shows an abstraction of the same data, focusing on issues that were or should have been identified through the method. In this table, as in earlier tables, 'A's mean 'should have been identified had the problem been represented at a different level of abstraction'. In this table, the data has been restructured to visually highlight commonalities across methods by clustering. Again, these findings are discussed in more detail below.

Finally, table 7 focuses on craft skill. This table shows the issues that were or could have been readily identified through the craft skill of the analyst when applying each UEM. In contrast to the issues that could be found by the methods, there is no obvious pattern in the issues that might plausibly emerge through craft skill.

		PROBLEM	STN	CW	GOMS	PUM	Z	EMU	CASSM	video	
17	C	Mismatch between way that arm works and way that user would move arm		C?		**M**	C?	C	**M**	yes	Hits/ Misses
23	C	Difficulty of judging arm movements		C?				**M**	**M**	yes	
24	C	Difficulty in judging speed and direction as getting close to target				A			**M**	yes	
25	C	Difficulty in judging position, orientation and aperture of gripper as approaching target							**M**	yes	
28	P	Voice recognition problems							**M**	yes	
22	P	User looking one way, menu options in other direction		C?				**M**	C	yes	
14	P	Time taken to interact with system to stop arm		C	**M**	C		C		yes	
30	K	No display of speed				**M**			C?	yes	
12	P	Problems of determining left and right, especially when arm contorted		C		C?			C?	yes	
29	P	Speaking with mouth full…						C?		yes	
13	K	User cannot check direction choice until arm starts to move		C						yes	
31	X	Arm obscuring user's view								yes	
32	S	No arm reversing	C?			C				yes	
4	S	Lack of short cuts	C?	C?	C	C?	C?			poor	
1	S	Long sequence of operators to move arm	O	C	**M**	C	C?	C		poor	
7	S	Gesture input with twice as many operations as voice	A		**M**	A	A			poor	
3	K	Difficulty of choosing between Move Arm or Move		M	C	M	C?	M		none	FPs
6	K	Confusion over joint called Arm		M	C	M	C?	M	O	none	
11	K	Lack of feedback about selection		M		C?				none	
18	K	Not clear that End returns user to main menu		M		M		O		none	
19	K	End having two meanings		M		M		O		none	
15	S	Similarity between moving joint and moving whole arm	M	C?	M	C?	M			none	
16	S	Illegal options	C?	C?			M			none	
26	C	Position and movement of most joints is of limited interest to the user		C?					M	none	
9	P	If user pauses in middle of saying "Move arm"...		C						none	
10	P	If user engaged in conversation...		C				C?	C?	none	
27	P	Possible difficulty of timing gesture accurately as cursor moves between options		C?	A				M	none	
2	S	Inability to backtrack	M	C?	C	C	M			n/a	
5	S	Continue redundant	M	C?	M	O	M			n/a	
8	P	Head moved to look at arm while gesture system operational may be interpreted as a command		C				M	C?	n/a	
20	X	Lighting conditions		C?				M		n/a	
21	X	Difficulty for user to move field of vision		C?				M		n/a	
33	K	Difficult to match names to joints		O		M			O	n/a	

Table 5: hits, misses and false positives: data from Table 4 rearranged to focus on video evidence. The second column summarises issue type as discussed in text. Shaded rows highlight issues from video data that were not reliably identified by any method. The second column indicates issue type: S=System design; K=user Knowledge; C=Conceptual fit; P=Physical issue; C=contextual.

		PROBLEM	GOMS	STN	Z	PUM	CW	EMU	CASSM	video
1	S	Long sequence of operators to move arm	M	O						poor
5	S	Continue redundant	M	M	M	O				n/a
7	S	Gesture input with twice as many operations as voice	M	A	A	A				poor
15	S	Similarity between moving joint and moving whole arm	M	M	M					none
14	P	Time taken to interact with system to stop arm	M							yes
27	P	Possible difficulty of timing gesture accurately as cursor moves between options	A						M	none
2	S	Inability to backtrack		M	M					n/a
16	S	Illegal options			M					none
3	K	Difficulty of choosing between Move Arm or Move				M	M	M		none
6	K	Confusion over joint called Arm				M	M	M	O	none
18	K	Not clear that End returns user to main menu				M	M	O		none
19	K	End having two meanings				M	M	O		none
33	K	Difficult to match names to joints				M	O		O	n/a
17	C	Mismatch between way that arm works and way that user would move arm				M			M	yes
30	K	No display of speed				M				yes
24	C	Difficulty in judging speed and direction as getting close to target				A			M	yes
11	K	Lack of feedback about selection					M			none
8	P	Head moved to look at arm while gesture system operational may be interpreted as a command						M		n/a
20	X	Lighting conditions						M		n/a
21	X	Difficulty for user to move field of vision						M		n/a
22	P	User looking one way, menu options in other direction						M		yes
23	C	Difficulty of judging arm movements						M	M	yes
25	C	Difficulty in judging position, orientation and aperture of gripper as approaching target							M	yes
26	C	Position and movement of most joints is of limited interest to the user							M	none
28	P	Voice recognition problems							M	yes
12	P	Problems of determining left and right, especially when arm contorted								yes
13	K	User cannot check direction choice until arm starts to move								yes
29	P	Speaking with mouth full…								yes
31	X	Arm obscuring user's view								Yes
32	S	No arm reversing								Yes
4	S	Lack of short cuts								poor
9	P	If user pauses in middle of saying "Move arm"...								none
10	P	If user engaged in conversation...								none

Table 6: Focus on methods: what *should* have been found by each method ('M's, 'O's and 'A's) The second column indicates issue type: S=System; K=user knowledge; C=conceptual misfit; P=physical misfit; X=contextual issue.

	PROBLEM	CW	PUM	CASSM	Z	GOMS	EMU	STN	video
1	Long sequence of operators to move arm	C	C		C?		C		poor
14	Time taken to interact with system to stop arm	C	C				C		yes
2	Inability to backtrack	C?	C			C			n/a
15	Similarity between moving joint and moving whole arm	C?	C?						none
4	Lack of short cuts	C?	C?		C?	C		C?	poor
12	Problems of determining left and right, especially when arm contorted	C	C?	C?					yes
8	Head moved to look at arm while gesture system operational may be interpreted as a command	C		C?					n/a
10	If user engaged in conversation...	C		C?			C?		none
22	User looking one way, menu options in other direction	C?		C					yes
9	If user pauses in middle of saying "Move arm"...	C							none
13	User cannot check direction choice until arm starts to move	C							yes
5	Continue redundant	C?							n/a
16	Illegal options	C?						C?	none
17	Mismatch between way that arm works and way that user would move arm	C?			C?		C		yes
20	Lighting conditions	C?							n/a
21	Difficulty for user to move field of vision	C?							n/a
23	Difficulty of judging arm movements	C?							yes
26	Position and movement of most joints is of limited interest to the user	C?							none
27	Possible difficulty of timing gesture accurately as cursor moves between options	C?				A			none
32	No arm reversing		C					C?	yes
11	Lack of feedback about selection		C?						none
7	Gesture input with twice as many operations as voice		A		A			A	poor
24	Difficulty in judging speed and direction as getting close to target		A						yes
30	No display of speed			C?					yes
33	Difficult to match names to joints			C?					n/a
3	Difficulty of choosing between Move Arm or Move				C?	C			none
6	Confusion over joint called Arm				C?	C			none
29	Speaking with mouth full…						C?		yes
18	Not clear that End returns user to main menu								none
19	End having two meanings								none
25	Difficulty in judging position, orientation and aperture of gripper as approaching target								yes
28	Voice recognition problems								yes
31	Arm obscuring user's view								yes

Table 7: Issues that either were (C) or could have been (C?, A) identified through craft skill

Discussion

The findings presented above have some clear limitations. They are confined to one interface and one task, and the initial analyses were performed mainly by one person. The second of these factors has influenced the results in some particular ways:

- The differences between 'M's (found by method) and 'O's (should have been found by method but were not) relate directly to the skill and experience of the analyst.
- Similarly, the differences between 'C's and 'C?'s (were and could have been identified by craft skill) can be attributed to the analyst effect.
- Finally, the way the problem was represented (including the level of abstraction for the analysis) was chosen by the analyst.

Nevertheless, the controlled nature of this case study – in particular, the rational reanalysis – and the careful avoidance of any numerical comparisons have made it possible for some important qualitative issues to emerge. First, the *scope* of each technique has become apparent, including some unexpected overlaps and disjuncts between the findings of different UEMs, and also some perturbing omissions (usability issues that emerged in the video data that were not found by any analytical technique). Second, issues about the nature of expertise (and craft skill) in usability evaluation have emerged, particularly through the reflective process imposed by the rational reanalysis. Finally, we reflect briefly on the methodology applied.

The scoping of techniques

In looking at the overlaps between the findings of the different techniques (Table 6), the groupings that emerge are as follows.

1. At the levels of abstraction at which these analyses were conducted, STN, Z and GOMS identified very similar issues. Apart from timing information, our GOMS analysis addresses all the kinds of issues outlined as being within scope by John and Kieras (1996a). Under the circumstances in which this study was conducted, it was not possible to include the detailed timing data that would, in other circumstances, enrich the GOMS analysis so that it should deliver more than the strictly device-centred Z and STN analyses. Lindegaard (2003) presents a forceful argument that GOMS timing data is often irrelevant, since the aspects of interaction for which timings can be done are not the most significant in terms of total interaction times. A similar point is made, though less forcefully, by John and Kieras (1996a, p.299). In the current case study, it is likely that the main contributions to total interaction time come from the arm movements rather than the system interactions. Whether or not timing data would be particularly informative, it was a surprise to us that GOMS, as a cognitively based technique, would have so much in common with system-oriented approaches and so little with other user-centred ones. The main explanation for this is likely to be that GOMS assumes users are experts, and therefore does not consider deficiencies in user knowledge, focusing rather on user actions which, of necessity, map directly onto device actions.
2. In contrast, the other user-oriented methods consider user knowledge, leading to another clear grouping of issues, for which PUM, CW, EMU and CASSM all have a high degree of overlap.

3. A third set of issues was identified only by EMU – all concerned with the physical relationship between the user and the device (for example, concerning lighting conditions and hence the user's ability to perceive information correctly from the system).
4. A fourth set of issues was identified only by CASSM; appropriately, these were issues that could be classed as misfits between user and system – for example, that the user might have difficulty judging the position, orientation and aperture of the gripper as it approached a target.
5. Finally, there were issues that emerged in the video data that had not been anticipated by any of the analytical evaluation methods. Some of these had been identified through the craft skill of the analyst while applying a UEM, but others were missed completely. These were mostly issues about use in context – for example, that the physical arm obscured the user's view of the target at some points in the interaction.

As briefly discussed above, using the eight approaches (seven UEMs plus video data), the usability issues can be classed into groups, according to what the primary focus of the issue is.

System design

This concerns the logical design of the system itself, and issues that might make it difficult for the user to work with. In this particular case study, it includes issues such as efficiency of tasks and redundant commands. In other cases, it might include safety and reachability concerns. Broadly, the system-oriented techniques (STN, Z and GOMS) have been found to be the strongest at identifying these kinds of issues. All of these approaches focus on procedural aspects of system design. Other system-oriented approaches such as ERMIA (Green and Benyon, 1996) that focus on structure rather than procedures might complement these techniques, but further investigation lies outside the scope of the present study.

User knowledge

There are a set of issues that all relate to the user's knowledge of the system. In practice, as shown in Table 5, few of these user knowledge issues emerged in the video data. In this particular study, this is partly explained by the fact that only excerpt 6 included non-pretaught moves, and was therefore the only video data that included any requirement on the user to apply their understanding of the system. Thus, the poor quality of the video evidence leaves some unanswered questions about the value of knowledge-oriented analysis techniques that should be the focus of further study (based on a system that has not been destroyed in a flood!).

The knowledge-oriented techniques included in this study are CW, PUM and, to a lesser extent, EMU and CASSM.

Conceptual fit

In contrast to the knowledge issues, and perhaps surprisingly, a relatively high proportion of the usability issues for which there is video evidence relate to the conceptual fit between the user's perspective and the system implementation. This illustrates well the difference between users' conceptions as represented within CASSM and potential user misconceptions, as represented within CW or PUM.

Unsurprisingly (since this was the intention in developing CASSM), CASSM provides the most support in identifying issues regarding conceptual fit.

Physical fit

For a device such as a robotic arm and its interface, physical considerations – for example, concerning timing and the interpretation of multimodal commands – are important. In particular, there is scope for system misinterpretation of user intentions so that the command issued by the user is not that received by the system. Consistent with the motivation for developing EMU to consider multimodal issues, this technique proved the strongest for identifying these issues in the interaction.

Use in context

Finally, there were issues that emerged due to the physical nature of the device and the way it is used in context. Some techniques (particularly the more established approaches that consider system design or user knowledge) encouraged the analyst to focus on the interaction with the system controller and consequently pay little attention to the arm being controlled. Even the newer approaches, which were developed to address broader usability concerns, missed some of the most important issues such as the arm itself obscuring the user's view of the gripper at times.

One might reasonably argue that this is the kind of domain knowledge that is more properly the focus of broader domain analysis techniques, and is therefore outside the legitimate concern of HCI. Nevertheless, this study illustrates that overall usability includes such context-specific factors, and that they need to be accommodated within a total usability analysis.

Discussion

This grouping of usability issues is not exhaustive for all systems – for example, it does not include user experience (Norman, 2004). Nevertheless, it represents the groupings identified for this particular kind of system. Given these groupings, we see that most UEMs have their main strengths within one particular group, and that some important usability issues are missed by all the analytical techniques evaluated.

The nature of expertise

In our reanalysis, we considered whether issues 'should' have been identified by a particular UEM. In practice, it turned out that this was a more complex question than initially anticipated, as illustrated by the extracts from reanalyses included in Appendix 2. Some cases were fairly clear-cut: either the issue was within the scope of the technique or it was not. However, others were less so. Aspects of this were:

1. Task or scenario generation. On several occasions, we could see that had the task been described slightly differently, or had the scenario been embellished more, then the issue would have emerged from the analysis, and been naturally credited to the method rather than craft skill (or being missed completely).
2. Level of abstraction. For four techniques (STN, Z, GOMS and PUM – see table 4), we could see that there were issues that would have emerged had the problem been described at a different (but equally appropriate) level of abstraction or, conversely, that some issues were identified because of the level of abstraction

adopted, which might not have emerged had a different representation been chosen.

3. Source materials. Tutorial and explanatory materials routinely make use of examples to communicate and illustrate points more effectively. Occasionally, it was apparent that the particular example used in tutorial material helped in issue identification, and that the issue might not have emerged otherwise. The most obvious example of this was identifying the 'inability to backtrack' issue of the first version of the system [subsequently changed] using STN.
4. Representation. As is widely recognised (e.g. Cheng, 1999; Cockayne *et al,* 1999), representations can serve an important role in helping the problem solver 'see' the problem in a particular way, which makes particular issues apparent and (conversely) hides others. Thus, even notations such as STN and Z, which are not traditionally used as evaluation techniques, made certain issues apparent, but did not highlight others. This is also true, though not as starkly, of the user-oriented approaches. This matter of representation is the main determinant of whether or not we classified an issue as "findable by craft skill": this was based on our judgement of whether or not the representation made an issue reasonably apparent.
5. Skill with notation. The analyst's skill in working with a notation or applying a method appeared, at least subjectively, to influence the quality of insights obtained through applying that UEM. Although this was more obvious with the more formal representations (such as Z), it was also an issue with the more discursive approaches (such as CW). In some cases, we were aware in conducting the initial analyses that the demands of the notation – requiring that the representation be consistent and complete – dominated the analysis, drawing attention away from the system being analysed towards the notation being used for describing it.

As this list illustrates, there are several important factors that influence the efficacy of applying any UEM to a particular interface. These factors contribute to the 'evaluator effect' (Hertzum & Jacobsen, 2001).

When using a technique, knowledge and skill are needed to use that technique appropriately, just as the analyst will have other knowledge and skill from "the outside world", as illustrated in Table 8.

Technique	General HCI	World
Knowledge (of how technique works and how to apply it correctly)	Knowledge (of potential usability problems and the effect they might have)	Knowledge (of domain and issues that might arise)
Skill (in applying technique properly and identifying issues)	Skill (in identifying issues based on HCI knowledge)	Skill (in identifying issues based on domain knowledge)

Table 8: Knowledge and skill used in technique and from world

In order to successfully apply a technique, an analyst must not only know how the technique works, how to apply it, and what to look for, but also have skill in applying the technique correctly, and in identifying those issues which are relevant. The analyst will also bring knowledge of other usability factors and skill in identifying such

potential issues when using the approach, and may also have domain-relevant knowledge that informs analysis.

Thus, in the hands of different analysts, exactly the same techniques will produce different results, due to novices not being able to apply the technique correctly and identify everything, and experts being able to use the technique as a lever to gain awareness of many other potential issues. This will be at least one factor that creates the 'evaluator effect'. Other factors have been outlined above in our reflection on some of the remaining ambiguities when reflecting on the reanalysis.

The investigation of the re-analyses found that, in line with Dreyfus's (1992) assertion that experts can structure their work to focus on relevant items, and hence use a technique as a prop, many of the issues identified in the initial analyses could be attributed to craft skill rather than emerging directly from the method. In addition, we could see how other issues might also have emerged through craft skill for an appropriately experienced analyst (Table 7).

One proposed advantage of a formal or semi-formal approach is that it provides an accurate representation, and that this accuracy allows many issues to be identified – if only be forcing the analyst to think deeply about the problem. However, obtaining an accurate representation is far from straightforward, and can distract the analyst from the actual task at hand, namely to identify and reason about usability issues. Winograd and Flores (1986) discuss this in relation to Heidegger's work on tools being 'ready to hand' – transparent and unobtrusive in the hands of an expert. As we found with the more demanding (formal) UEMs discussed here, the task of producing a complete and consistent representation could distract from the purpose of gaining usability insights. This issue of achieving an appropriate balance between the demands of a representation and the insights it yields requires further investigation.

Methodology

Finally, we reflect briefly on the methodology applied in this study. We believe that this approach addresses most of the pitfalls identified by Gray and Salzman (1998a) as discussed above (see Background). The advantages can be summed up as follows.

- The evaluator effect (Hertzum and Jacobsen, 2001) is reduced by using only one, very small, team of evaluators to perform all analyses;
- Cause-effect issues are mimimised by presenting an account of why each UEM does or does not support the identification of each usability issue in our set;
- Issues of statistical conclusion validity are avoided by focusing instead on a qualitative comparison of techniques;
- Ambiguities caused by 'method shift' (in which the definitions of UEMs change over time) are removed by basing the reanalyses on single, defined sources of description for each method;
- Finally, we have aimed throughout to distinguish between findings from data and findings from our experience of conducting the work (including factoring out issues identified through the analysts' craft skill).

One of the prices we pay in taking such an approach is that there is no simple 'headline' findings from this study. As reported, this work does not yet have a clear message for practitioners on the costs and benefits of applying different techniques;

however, we believe that it provides a stronger theoretical basis on which further work that takes the study closer to the practitioners' concerns can be built.

Conclusion

Although this study has focused on one interface and task, the findings are certainly not about that particular interface (interesting though it may be). Indeed, the role of the interface and task has been to force the analysis to address a range of issues, many of which have been found to be outside the scope of the techniques tested. The focus on one particular interface and task inevitably means that there are issues that have not emerged in this analysis that might have had a different kind of system been used, or a broader set of tasks and contexts of use considered; nevertheless, we believe that the findings from this study contribute a valuable piece to the jigsaw of understanding the scope and properties of analytical UEMs.

Similarly, although this study has focused on seven UEMs, it is less about the features of those particular techniques than about the nature of analytical evaluation, its strengths and limitations. Yes, the study has had its surprises – the position of GOMS as having more in common with system-oriented techniques than user-oriented ones stands out for us in this regard – but the most important findings, in our view, concern the nature of expertise in analytical evaluation. We have identified several factors that contribute to the quality of an analysis, including the appropriateness of tasks selected, the details of how scenarios of use are described, the level of abstraction used in modelling (applicable to some techniques but not all) and the analyst's expertise in the technique, in general HCI and in the domain of application.

Some of the UEMs included in this study encourage a focus on the control interface rather than on the arm or other aspects of the domain and context of use; others have broader scope; some (notably CW) encourage a focus on local issues (about *this step* in the interaction) so that the broader picture tends to get lost. For a novice analyst, the more difficult techniques encouraged a focus on the notation and getting the representation 'right' rather than using the notation to gain insights about usability. John and Marks (1997) suggest that unstructured consideration of a design description can be just as insightful as the use of a particular analysis technique; however, they say little about the precise skills of the individual doing the inspecting. Ultimately, it may be that UEMs provide structure to help the analyst get going and to ensure coverage of issues within the scope of the approach, but that their limitations also need to be recognised.

This work has presented a novel, rigorous approach to comparing UEMs and validating the findings against empirical evidence. There have been two limitations to comparing analytical findings against empirical data in this study. The first is particular to this study and relates to the poor quality of the video evidence available, which made it difficult to be confident about some of the false positives (was it just that issues did not emerge because the interaction was too short or too undemanding?). The second issue is more general: it concerns the difficulty of relating behavioural observations to underlying causes. Hollnagel (1998) refers to this as the difference between genotypes (underlying causes) and phenotypes (surface manifestation); this is a difficulty that will continue to plague HCI, and remains a strong argument in favour of analytical techniques: observation of surface behaviour can highlight user difficulties, but does not directly point to the possible sources of those difficulties, and hence to design solutions that will remove them. Also, although

false positives are often considered undesirable (e.g. Cockton et al 2003), there may be usability difficulties that do not emerge in finite empirical data – whether because they are rare but critical difficulties (Connell *et al*, in press) or because they cause unnecessary mental workload but no physical manifestation.

In summary, the work reported here has made three important contributions to our understanding of analytical evaluation methods: methodological; on the nature of expertise in usability evaluation; and in the comparison and scoping of methods. By firmly rejecting numerical comparisons between techniques, this study has probed deeper issues about the application of such approaches.

Acknowledgements

Joanne Hyde was funded by a studentship from the School of Computing Science, Middlesex University. Work on CASSM is funded by EPSRC grant GR/R39018.

References

BAILEY, R.W., ALLAN, R.W. & RAIELLO, P. (1992). Usability testing vs. heuristic evaluation: a head-to-head comparison. *Proceedings of the Human Factors Society 36th Annual Meeting, 1992, **1**, 409-413*. Human Factors Society.

BARNARD, P. J. & MAY, J. (1999) Representing cognitive activity in complex tasks. *Human Computer Interaction*, 14, 93-158.

BASTIDE, R. & PALANQUE, P. (1990) Petri net objects for the design, validation and prototyping of user-driven interfaces. In DIAPER, D. GILMORE, D., COCKTON, G. & SHACKEL, B. (Eds.): Human-Computer Interaction - INTERACT'90. pp.625-631, Elsevier Science Publications, North Holland, Netherlands.

BLACKWELL, A. & GREEN, T. R. G. (2003) Notational Systems – The Cognitive Dimensions of Notations Framework. In J. Carroll (ed.), *HCI Models, Theories and Frameworks,* pp. 103-134. Morgan Kaufmann.

BLANDFORD, A. (2004) CASSM analysis of arm. Working paper available from http://www.uclic.ucl.ac.uk/annb/CASSMpapers.html

BLANDFORD, A. E., BUCKINGHAM SHUM, S. & YOUNG, R. M. (1998) Training software engineers in a novel usability evaluation technique. *International Journal of Human-Computer Studies*, 45(3), pp. 245-279.

BLANDFORD, A., CONNELL, I. & GREEN, T. (2003) CASSM Tutorial. Working paper available from http://www.uclic.ucl.ac.uk/annb/CASSMpapers.html

BLANDFORD, A. GOOD, J. & YOUNG, R. M. (1998) Programmable user modelling analysis for usability evaluation. Tutorial available as Working Paper WP11a from www.cs.mdx.ac.uk/puma/ (mirrored at http://www.uclic.ucl.ac.uk/annb/CASSMpapers.html)

BLANDFORD, A. & HYDE, J. (2004) Rational Reanalyses of a Robotic Arm. Working paper available from http://www.uclic.ucl.ac.uk/annb/CASSMpapers.html

BLANDFORD, A. E. & YOUNG, R. M. (1996) Specifying user knowledge for the design of interactive systems. *Software Engineering Journal*. 11.6, pp. 323-333.

CARD, S. K., MORAN, T. P. & NEWELL, A. (1983) *The Psychology of Human Computer Interaction*, Hillsdale NJ: Lawrence Erlbaum.

CHENG, P. C-H. (1999) Unlocking conceptual learning in mathematics and science with effective representational systems. *Computers and Education*, 33(2-3), 109-130.

COCKAYNE, A., WRIGHT, P.C. & FIELDS, B. (1999). Supporting Interaction Strategies Through the Externalization of Strategy Concepts. In: Sasse, M.A. and Johnson, C. (eds.) *Proceedings INTERACT'99*, pp. 582-588. IOP Press.

COCKTON, G., WOOLRYCH, A., HALL, L. & HINDMARCH, M. (2003) Changing Analysts' Tunes: the Surprising Impact of a New Instrument for Usability Inspection Method Assessment. In: People and Computers XVII: Proceedings of HCI'03, Springer, pp 145-161.

CONNELL, I., GREEN, T. & BLANDFORD, A. (2003) Ontological Sketch Models: Highlighting User-System Misfits. In E. O'Neill, P. Palanque & P. Johnson (Eds.) *People and Computers XVII, Proc. HCI'03*. 163-178. Springer.

CONNELL, I., BLANDFORD, A. & GREEN, T. (in press) CASSM and Cognitive Walkthrough: usability issues with ticket vending machines. To appear in *Behaviour and Information Technology*. Draft available from http://www.uclic.ucl.ac.uk/annb/CASSMpapers.html.

CUOMO, D.L. & BOWEN, C.D. (1994). Understanding usability issues addressed by three user-system interface evaluation techniques. *Interacting with Computers, 1994, **6(1)**, 86-108*.

DESURVIRE, H.W. (1994). Faster, cheaper !! Are usability inspection methods as effective as empirical testing ? In J. Nielsen and R.L. Mack (eds.), *Usability Inspection Methods*, pp. 173-201. New York: John Wiley & Sons.

DESURVIRE, H.W., KONDZIELA, J.M. & ATWOOD, M.E. (1992). What is gained and lost when using evaluation methods other than empirical testing. In A. Monk, D. Diaper and M.D. Harrison (eds.), *People and Computers VII.* Proceedings of HCI '92, York, September 1992, pp. 173-201. British Computer Society Conference Series 5. Cambridge: Cambridge University Press.

DIX, A. J., FINLAY, J., ABOWD, G. & BEALE, R. (1993) *Human-Computer Interaction,* Hemel Hempstead: Prentice Hall International.

DREYFUS, H. L. (1992) *What computers still can't do: a critique of artificial reason*. Revised edition, MIT Press, USA

DREYFUS, H. L.& DREYFUS, S. E. (1985) *Mind over machine*. New York : Macmillan / The Free Press.

DUKE, D. J., BARNARD, P. J., DUCE, D. A. & MAY, J. (1998) Syndetic Modelling. *Human-Computer Interaction*. 13, 337-394

DUTT, A., JOHNSON, H. & JOHNSON, P. (1994). Evaluating evaluation methods. In G. Cockton, S.W. Draper and G.R.S. Weir (eds.), *People and Computers IX. Proceedings of HCI '94*, Glasgow, August 1994. Cambridge: Cambridge University Press.

GRAY, W. D. & SALZMAN, M. C. (1998a) Damaged merchandise? A review of experiments that compare usability evaluation methods. *HCI Journal*. pp. 203-261

GRAY, W. D. & SALZMAN, M. C. (1998b) Repairing Damaged merchandise: A Rejoinder. *HCI Journal*. pp. 325-335.

GREEN, T. R. G. & BENYON, D. (1996) The skull beneath the skin: entity-relationship models of information artifacts. *International Journal of Human-Computer Studies*, 44(6) pp. 801-828

HARTSON, H. R., SIOCHI, A. C. & HIX, D. (1992) The UAN: A user-oriented representation for direct manipulation interface designs. *ACM Transactions on Office Information Systems,* 8, 181-203.

HERTZUM, M., AND JACOBSEN, N.E. (2001). The Evaluator Effect: A Chilling Fact about Usability Evaluation Methods. *International Journal of Human-Computer Interaction*, 13(4), 421-443.

HOLLNAGEL, E. (1998) *Cognitive Reliability and Error Analysis Method (CREAM)* Oxford : Elsevier Science.

HYDE, J. K. (2002) *Multi-Modal Usability Evaluation*. PhD thesis. Middlesex University. Tutorial notes on applying EMU available from http://www.uclic.ucl.ac.uk/annb/CASSMpapers.html.

JACOBSEN, N., HERTZUM, M. & JOHN, B. (1998) The evaluator effect in usability studies: problem detection and severity judgements. *Proc. HFES 42nd Annual Meeting.* 1336-1340.

JACOBSEN, N. E., & JOHN, B. E. (2000). Two case studies in using cognitive walkthrough for interface evaluation. School of Computer Science Technical Report CMU-CS-00-132. Pittsburgh, PA: Carnegie Mellon University. PDF: http://reports-archive.adm.cs.cmu.edu/anon/2000/CMU-CS-00-132.pdf

JEFFRIES, R. & DESURVIRE, H. (1992). Usability testing vs. heuristic evaluation: was there a contest ? *SIGCHI Bulletin, October 1992, **24(4)**, 39-41*.

JEFFRIES, R., MILLER, J.R., WHARTON, C. & UYEDA, K.M. (1991). User interface evaluation in the real world: a comparison of four techniques. In S.P. Robertson, G.M. Olson and J.S. Olson (eds.), *Reaching Through Technology: CHI '91 conference proceedings*. ACM conference on human factors in computing systems, New Orleans, April-May 1991. New York: Addison-Wesley.

JOHN, B. & KIERAS, D. E. (1996a) Using GOMS for user interface design and evaluation: which technique? *ACM ToCHI* 3.4. 287-319.

JOHN, B. & KIERAS, D. (1996b) The GOMS family of user interface analysis techniques: comparison and contrast. *ACM Transactions on CHI*, 3, 320-351.

JOHN, B. & MARKS, S. (1997) Tracking the effectiveness of usability evaluation methods. *Behaviour and Information Technology* 16, No. 4/5, 188-202.

JOHN, B. E. & PACKER, H. (1995) Learning and using the Cognitive Walkthrough method: A case study approach. In *Proceedings of CHI'95.* pp.429-436. ACM Press: New York.

JOHNSON, H. & HYDE, J. (2003) Towards modeling individual and collaborative construction of jigsaws using task knowledge structures (TKS) *ACM Transactions on Computer-Human Interaction* 10.4, 339 - 387 .

KARAT, C-M., CAMPBELL, R. & FIEGEL, T. (1992). Comparison of empirical testing and walkthrough methods in user interface evaluation. In P. Bauersfeld, J. Bennett and G. Lynch (eds.), *CHI '92 Conference Proceedings: striking a balance*. ACM conference on human factors in computing systems, Monterey, California, May 1992. Reading, MA: Addison-Wesley.

KARAT, C. M. (1994) A comparison of user interface evaluation methods. In J. Nielsen & R. Mack (Eds.), *Usability Inspection Methods* (pp. 203-233) New York: John Wiley.

KARAT, J. (1997). User-centered software evaluation methodologies. In M. Helander, T.K. Landauer and P. Prabhu (eds.), *Handbook of Human-Computer Interaction*, 2nd edition, pp. 689-704. Amsterdam: Elsevier Science B.V.

KLEIN, G. A. (1998) *Sources of Power: How people make decisions*. Cambridge, MA: The MIT Press.

LAVERY, D., COCKTON, G. & ATKINSON, M. P. (1997) Comparison of evaluation methods using structured usability problem reports. *Behaviour and Information Technology*, 16 (4/5), 246-266.

LINDEGAARD, G. (2003) The Misapplication of Engineering Models to Business Decisions. In M. Rauterberg, M. Menozzi & J. Wesson (Eds.) *Proc. Interact 2003*. IOS Press. 367-374.

LONG, J. & DOWELL, J. (1990) Conceptions of the discipline of HCI: craft, applied science, and engineering. In A. Sutcliffe and L. Macaulay (Eds.) *People and Computers V*. 9-32 Cambridge: Cambridge University Press.

NIELSEN, J. (1994) Heuristic Evaluation. In J. Nielsen & R. Mack (Eds.), *Usability Inspection Methods* (pp. 25-62) New York: John Wiley.

NIELSEN, J. & PHILLIPS, V.L. (1993). Estimating the relative usability of two interfaces: heuristic, formal and empirical methods compared. In S. Ashlund, A. Henderson, E. Hollnagel, K. Mullet and T. White (eds.), *Human Factors in Computing Systems: INTERCHI '93*. Proceedings of INTERCHI '93. Amsterdam: IOS Press.

NORMAN, D. (2004) *Emotional Design: why we love (or hate) everyday things*. Basic Books.

PARSONS, B., PRIOR, S. D. & WARNER, P. R. (1995) An Approach to Designing Manipulator Controller Software Employing Context Dependent Command Interpretation. In: *European Conference on Assistive and Rehabilitation Technology*, Lisbon, October 1995

PARSONS, B., WARNER, P., WHITE, A.. & GILL, R. (1997) An Approach to the Development of Adaptable Manipulator Controller Software. In: *Proc. International Conference on Rehabilitation Robotics (ICORR97).*

PAYNE, S. J. & GREEN, T. R. G. (1986) Task-Action Grammars: The Model of the Mental Representation of Task Languages *Human-Computer Interaction*, 2, 2, 93-133

RAUTERBERG, M. (2003). Human computer interaction research--a paradigm clash?. In: *Proceedings 1st International Conference Tales Of The Disappearing Computer*; Santorini (Greece). Available from http://www.ipo.tue.nl/homepages/mrauterb/publications/DCtales2003paper.pdf (accessed

2/8/4)

SANDERSON, P. M. & FISHER, C. (1994) Exploratory sequential data analysis: foundations. Human-computer Interaction 9, 215-317.

SEARS, A. (1997). Heuristic walkthroughs: finding the problems without the noise. *International Journal of Human-Computer Interaction, 1997, **9(3)**, 213-234*.

SPIVEY, J. M. (1989) *The Z Notation: A Reference Manual.,* Prentice-Hall International.

VIRZI, R.A., SORCE, J.F. AND HERBERT, L.B. (1993). A comparison of three usability evaluation methods: heuristic, think-aloud and performance testing. *Proceedings of the Human Factors and Ergonomics Society 37th Annual Meeting, 1993, **1**, 309-313.* Human Factors and Ergonomics Society.

WHARTON, C., RIEMAN, J., LEWIS, C. & POLSON, P. (1994) The cognitive walkthrough method: A practitioner's guide. In J. Nielsen & R. Mack (Eds.), *Usability inspection methods* (pp. 105-140) New York: John Wiley.

WINOGRAD, T. & FLORES, F. (1986) *Understanding Computers and Cognition,* Reading MA: Addison-Wesley.

YOUNG, R. M., GREEN, T. R. G. & SIMON, T. (1989) Programmable User Models for Predictive Evaluation of Interface Designs, in K. Bice. & C. Lewis (eds.), *Wings for the Mind: CHI '89 Conference Proceedings*, pp.15-19. ACM conference on human factors in computing systems, Austin, Texas, April-May 1989. Reading, MA: Addison-Wesley.

Appendix 1: definitions of the usability issues identified

1. Long sequence of (mental) operators to move arm

The number of decision and action steps needed by the user to get the arm going is greater than necessary (does not apply to pre-taught positions). This is particularly so if the user wishes to move an individual joint, or to change the speed of arm movement.

2. Inability to backtrack.

In the first version of the system analysed, there was no 'undo' option. As shown in the 'second STN' (Figure N), this omission was soon corrected.

3. Difficulty of choosing between Move Arm or Move

The user's first decision is between MoveArm (which moves the whole arm) and Move, which then allows the user to select an individual joint to move. The semantics of this choice may be difficult for novice users to grasp.

4. Lack of short cuts

There is no quick way to return to the direction menu, which might be required if the arm overshoots or if the whole arm is being moved and needs a change of direction.

5. Continue serves same function as Go, and is redundant

There was originally an option called 'continue', which served exactly the same function as 'go' and was therefore eliminated in an early redesign of the interface.

6. Confusion over joint called Arm

The term 'Arm' is used to refer to both the whole arm and an individual joint called 'arm'.

7. Gesture input with twice as many operations as voice because dependent on cursor movement

This refers to mental operations, not physical ones. The number of physical operations is the same in both cases, but it takes more mental effort to spot the gesture option and maintain attention on it until the cursor is in the correct place to select that option.

8. Problem if head moved to look at arm while gesture system operational may be interpreted as a command

Since gestures may be part of user's normal repertoire of head movements, it is possible that the user might move their head in a way that is interpreted by the system as a gesture when it was not intended as such.

9. If user pauses in middle of saying "Move arm"...

Because "move arm" is made up of "move" and "arm, and "move" and "arm" are also valid commands, a pause in the middle could cause misinterpretation by the system.

10. If user engaged in conversation...

If the user of the speech controlled system is also engaged in another conversation, it is possible that some conversational words might be interpreted as commands by the system.

11. Lack of feedback about selection

This arose in the CW analysis specifically in relation to MoveArm. This reflects a broad concern that the system as analysed did not give feedback on selections at the time of analysis, although the gestural and voice input mechanisms did request user confirmation of choice.

12. Problems of determining left and right, especially when arm contorted

If the arm is contorted then "its" right and left may be different from right and left (or indeed up and down or in and out) as perceived by the user.

13. User cannot check direction choice until arm starts to move

This is really a combination of 11 and 12: that the user neither gets feedback on what they have selected nor can anticipate which actual direction corresponds to the command for a contorted arm until the arm starts to move.

14. Time taken to interact with system to stop arm

The user has to anticipate how long it will take the system to respond to 'stop' and issue the command at the right time.

15. Similarity between moving joint and moving whole arm

Both moving the joint and moving the arm follow a similar pattern of states and transitions. The interaction could be made more efficient and maybe clearer by combining these options into a single menu.

16. Illegal options

When the arm has reached its limit of movement, it is possible to issue command that would, in principle, send it beyond its limit. The only feedback to the user is that the arm does not move.

17. Mismatch between way that arm works and way that user would move arm

The way the user conceptualises what they are doing 'in the world' does not map readily on to the way the user has to program the arm to work.

18. Not clear that End returns user to main menu

This is about labelling: firstly, 'end' is semantically confusable with 'stop'; secondly, 'end' does not mean 'return to initial menu', although that is the effect of this action.

19. End having two meanings

Under all circumstances, 'end' returns the user to the initial menu. Other than at the end of the overall interaction, the user has a motivation to complete this step; right at the end of the interaction the user has no reason to restore the interface to its initial state, and may therefore omit the 'end'. This is unlikely to cause substantive user difficulties in the circumstances.

20. Lighting conditions

If lighting is poor, the user may have difficulty seeing options or seeing the arm's current position.

21. Difficulty for user to move field of vision

Disabled users may have difficulty shifting their visual attention from the display to the arm and vice versa.

22. User looking one way, menu options in other direction

The user has to divide their visual attention between the arm position or movements and the display that controls the arm.

23. Difficulty of judging arm movements

For novice users, it is likely to be difficult to judge exactly how the arm is moving and where it currently is. This issue is expanded below as more detailed issues.

24. Difficulty in judging speed and direction as getting close to target

As the gripper gets close to the target, it needs to reach it without overshooting or colliding. Depending on the direction of approach, the user may find this very difficult to judge.

25. Difficulty in judging position, orientation and aperture of gripper as approaching target

Similarly, the position of the gripper may be difficult to ascertain.

26. Position and movement of most joints is of limited interest to the user

Since the user's main concern is with the position of objects in the world, which can only be manipulated by the gripper, the main concern is about getting the gripper in the right place, i.e. by moving the whole arm. Exceptions might be when fine-tuning the angle of the gripper on approach, and if avoiding other obstacles in the room.

27. Possible difficulty of timing gesture accurately as cursor moves between options

The user of the gestural interface has to time their gesture to select the correct option. This timing may be difficult for novices.

28. Voice recognition problems

If the user does not speak clearly, their words may not be interpreted correctly by the voice recognition system.

29. Speaking with mouth full…

If the user of a voice recognition system tries speaking while eating, there are likely to be voice recognition problems.

30. No display of speed

There is no feedback (other than the perceived speed of the arm while actually moving) of the current speed setting.

31. Arm obscuring user's view

The arm itself may get in the way of the user's view of the target object in the world.

32. No arm reversing.

It is not possible to reverse direction of the arm without going all the way through the set-up procedure again. This matters in cases where the user overshoots.

33. Difficult to match names to joints

For the novice user, it may take a while to learn the names of all the joints.

Appendix 2: extracts from the reanalyses

The full reanalyses are available from Blandford and Hyde (2004). Here, the most interesting issues are presented – typically those that illustrate points made in the discussion section.

STN re-analysed

1. Long sequence of operators to move arm

Since the STN shows the number of states that the user has to navigate through before the robotic arm can be moved, this issue should have been identified in the original analysis. However, STN deals with only physical state changes, and does not consider mental operations, so the effect is less marked for STN than it was for GOMS. That it was not identified shows the extent to which the analysis was dependent on the craft skill (or lack thereof) of the analyst.

2. Inability to backtrack [STN]

This issue is apparent from the STN, and was identified as a problem. However, the identification of this issue was possibly influenced by the explicit mention of this kind of problem in a discussion on "undo" in the source materials (Dix et al, 1993, p.291). This shows the effect that the source materials of a technique has on the application of a technique.

3. Difficulty of choosing between Move Arm or Move

The STN concentrates on the actual choice between the system states, rather than on the difficulties the user has in choosing between them. It is therefore not an issue that the STN on its own would be expected to identify, but might have been identified through craft skill – looking at the problem with particular questions in mind.

4. Lack of short cuts

Since the STN explicitly shows the possible path of the interaction through the various states, the lack of short-cuts was an issue that might have become apparent if the analyst had been looking for it. This is therefore an issue that is a combination of craft skill and representation. That it was not noticed was possibly because the analyst's attention was more on obtaining the correct representation of the system states.

6. Confusion over joint called Arm

The STN did not go into the detail of the individual options, so this issue did not arise. If the STN had been done at a different level of abstraction, this issue might have been identified through the craft skill of the analyst. It is not something that the STN would identify directly however, since it is concerned more with the user understanding of what a particular option choice means rather than with the option choice itself. Thus this issue highlights questions associated with both craft skill and appropriate levels of abstraction.

7. Gesture input with twice as many operations as voice

The STN was not written at the level of abstraction which would identify this issue. If it had been, this issue would probably have been identified, since it would be concerned with the number of states and transitions. For the gesture input, there are a series of states and transitions between them as opposed to the voice input which has one state with multiple transitions coming from it. This raises questions concerning the appropriate level of abstraction of an analysis.

16. Illegal options

This issue was not represented on the STN. There was no state showing that the arm had reached its limit of movement, nor was there an end option leading from the travel until stop state which might also represent it. This shows how difficult it is to draw STNs correctly, and relates to the level of skill of the analyst in determining how the system states should be represented. However, even if the STN diagram had been correctly drawn, it is still unlikely that this issue would have been identified without explicitly checking for illegal options.

27. Possible difficulty of timing gesture accurately as cursor moves between options

STN does not explicitly consider timing. With a more detailed STN (level of abstraction), this issue might have been spotted through craft skill. In the event, it was not.

32. No arm reversing.

Because the STN focuses on the device states, and the direction of motion is simply a parameter on that state, the domain requirement to make it easy to reverse does not appear through the STN. It would have required a very different kind of STN to allow this issue to emerge.

CW Re-Analysis

2. Inability to backtrack

CW does not deal with error in terms of its implications, therefore would not find this issue, although it might come out from the craft skill of the analyst through thinking about rectifying errors.

11. Lack of feedback about selection

For the purposes of the original analysis, this was not relevant, since the feedback had not been implemented, but it was an important issue raised by the method that would have to be addressed once feedback had been implemented.

17. Mismatch between way that arm works and way that user would move arm

One of the aims of the method is to uncover this kind of issue, however there is not much support within the questions for this to be identified at a high level, because of the method's concentration on the step-by-step nature of the task. This is more likely to be uncovered by craft skill therefore.

30. No display of speed

Because the display of speed (or the lack of it) is outside the essential task definition (unless the task were to be to move the arm at a particular speed, which would involve craft skill in perceiving the need for such a task), this would not naturally emerge from a CW analysis.

CMN and CPM GOMS Re-Analyses

The CPM GOMS analysis was unable to identify many issues over and above those identified by CMN GOMS, other than the difference between the use of voice and gesture operators. Thus only issue seven was able to be identified, and this was the only issue that can be considered to be within the bounds of the method. This re-analysis consequently focused on the use of CMN GOMS. A different CPM GOMS analysis that indicated where the user would want to look at the arm to check its position or movement would have raised more issues.

4. Lack of short cuts

By writing out the methods, the long sequence showed that this would take a long time and that there were no short cuts. Whether this emerges from the analysis or is derived through craft skill is a moot point.

10. if user engaged in conversation...

This issue is outside the scope of CMN GOMS and was not identified.

If the task description included reference to another conversation, this issue should be identified through CPM GOMS; however, this depends on analyst insight in specifying such a task.

26. Position and movement of most joints is of limited interest to the user

This issue did not emerge. Indeed, a task definition would include a specification of which joints to move, so this issue is more strongly excluded from the set of possible issues than most.

27. Possible difficulty of timing gesture accurately as cursor moves between options

CMN GOMS does not consider timing issues such as this.

CPM GOMS should have spotted this issue, had the interface been described at the appropriate level of abstraction.

29. Speaking with mouth full…

This issue is outside the scope of CMN GOMS and was not identified.

It would only be identified by CPM GOMS with a very inspired choice of tasks.

PUM Re-Analysis

1. long sequence of operators to move arm

This issue was mentioned in the original analysis, but not in a strong enough way for it to be apparent as an issue of consequence. It was identified from looking at the heavy ordering identified by the analysis, and was therefore dependent upon the craft skill of the analyst.

2. inability to backtrack

The original analysis found a heavy ordering, which is within the bounds of the PUM technique. However, from this was derived the lack of backtracking provision, which is therefore identified by the craft skill of the analyst, based on the representation provided by the method.

7. Gesture input with twice as many operations as voice because dependent on cursor movement

This did not come out in the original PUM analysis, because the analysis was not written at a low enough level of abstraction for this to be apparent.

11. Lack of feedback about selection

The output was not included in the original analysis. If it had been then the PUM analysis might have picked up on this issue, in the modelling of the user knowledge, because the user would not know that the option had been selected.

15. similarity between moving joint and moving whole arm

The way that the PUM analysis was conducted meant that this issue was not identified, although it would probably have been recognised if the analyst was looking for it. Therefore, although the PUM analysis represented the operations, it would take the craft skill of the analyst to identify their similarity.

24. Difficulty in judging speed and direction as getting close to target
If a much more detailed PUM model had been constructed, it is possible that this issue might have been identified, through the process of describing a 'monitoring' activity more detailed than the 'wait and then stop' implemented in the current model. This is therefore both a level of abstraction and a craft skill issue.
26. Position and movement of most joints is of limited interest to the user
Because PUM doesn't encourage the analyst to 'step back' in this way, it is unlikely that this issue would fall inside the scope of a PUM analysis.
27. Possible difficulty of timing gesture accurately as cursor moves between options
It would be necessary to construct a PUM model at a much finer grain of detail for this issue to emerge. This is not a level at which PUM naturally works, so it is unlikely that this issue would be spotted.

Z re-analysis

1. long sequence of operators to move arm
This issue was not apparent because of the way that the specification was constructed, although the specification did represent it. This issue therefore highlights the important difference between an issue being represented and identified. It would take a certain amount of craft skill on the part of the analyst to identify this issue.
4. Lack of short cuts
The Z specification represented the lack of backtracking opportunities, due to its concentration on the ordering of the interaction. The lack of short-cuts was therefore also represented. However, the issue, although represented, was not identified, which again illustrates the difference between an issue being represented and identified, and the importance of the craft skill of the analyst in identifying significant issues.
7. Gesture input with twice as many operations as voice because dependent on cursor movement
This issue was not identified by the Z specification because the specification was not written at a low enough level of detail to represent the cursor movement. This illustrates the need for the appropriate level of abstraction of the representation.

EMU re-analysis

10. if user engaged in conversation...
This was not an issue identified by the method since it was not indicated in the initial scenario that the user would be engaged in conversation. If that information had been included in the environment profile, then this issue would have been identified by EMU.
19. End having two meanings
This issue was not identified in the original analysis and should have been, since it is a potential mismatch. This demonstrates how the identification of any issue is dependent upon the analyst, and that mistakes and omissions can occur.
23. difficulty of judging arm movements
This is a clash unless expert issue, and the method instructs the analyst to look for these clashes.
29. Speaking with mouth full…
This issue should have been identified by EMU had a different task been considered – i.e. one that included feeding.
31. Arm obscuring user's view
Paradoxically, this issue is outside the scope of EMU, unless it were identified through craft skill, because the bulk of the rest of the arm (other than the gripper) is not represented.

CASSM re-analysis

6. Confusion over joint called Arm
With a slightly expanded CASSM description that includes the concept of the whole arm as being made up of joints, this issue should have emerged. This issue *should* have been identified.
12. problems of determining left and right, especially when arm contorted
The issue of judging directions when the arm is contorted emerged (with some craft skill) from looking at joints and what the user knows about the directions in which joints can move. It does not emerge directly from the CASSM representation.
30. No display of speed
This probably should have emerged through the consideration that there is a difference (misfit?) between the perceived speed of the arm as moving and the speed setting as determined (but not displayed) through the interface. This one's a bit marginal…