Modelling Interactive Behaviour with a Rational Cognitive Architecture

David Peebles,
University of Huddersfield, UK

Anna L. Cox,
University College London, UK

Abstract

In this chapter we discuss a number of recent studies that demonstrate the use of rational analysis (Anderson, 1990) and cognitive modelling methods to understand complex interactive behaviour involved in three tasks: (1) icon search, (2) graph reading, and (3) information retrieval on the World Wide Web (WWW). We describe the underlying theoretical assumptions of rational analysis and the adaptive control of thought-rational (ACT-R) cognitive architecture (Anderson & Lebiere, 1998), a theory of cognition that incorporates rational analysis in its mechanisms for learning and decision making. In presenting these studies we aim to show how such methods can be combined with eye movement data to provide detailed, highly constrained accounts of user performance that are grounded in psychological theory. We argue that the theoretical and technological developments that underpin these methods are now at a stage that the approach can be more broadly applied to other areas of Web use.

Introduction

With the rapid increase in Internet use over the past decade there is a growing need for those engaged in the design of Web technology to understand the human factors involved in Web-based interaction. Incorporating insights from cognitive science about the mechanisms, strengths, and limits of human perception and cognition can provide a number of benefits for Web practitioners. Knowledge about the various constraints on cognition, (e.g., limitations on working memory), patterns of strategy selection, or the effect of design decisions (e.g., icon style) on visual search, can inform the design and evaluation process and allow practitioners to develop technologies that are better suited to human abilities.

The application of cognitive psychology to human-computer interaction (HCI) issues has a long history going back to Card, Moran, and Newell's (1983) introduction of the goals, operators, methods, and selection (GOMS) task analysis technique and model human processor account of human information processing in the early 1980s. Since then, their cognitive engineering approach has developed into a family of methods (John & Kieras, 1994; Olson & Olson, 1990) which are widely used to produce quantitative models of user performance in interactive tasks.

Another, more recent approach to modelling human performance in interactive tasks has emerged in the last decade from theoretical and technological advances in research into cognitive architectures. Cognitive architectures are theories of the fundamental structures and processes that underlie all human cognition, of which there are several currently in existence including EPIC (executive process / interactive control; Kieras & Meyer, 1997), Soar (Laird, Newell, & Rosenbloom, 1987; Newell, 1990), and ACT-R (Anderson & Lebiere, 1998; Anderson et al., 2004). An important feature of these architectures is that they are all implemented as computer programming systems so that cognitive models may be specified, executed, and their outputs (e.g., error rates and response latencies) compared to human performance data.

Originally ACT-R and Soar were theories of central cognition only and did not explicitly specify mechanisms for perception or motor control. EPIC however, was unique in that from its inception it incorporated processors for cognition, perception, and motor control. Recent adaptations to ACT-R (Byrne & Anderson, 1998) and Soar (Chong & Laird, 1997) have now ensured that both architectures incorporate perceptual motor components that allow models to include visual attention processes and manual interactions with a keyboard and mouse. This is an important development for the study of HCI as cognitive models can now be *embodied* (Kieras & Meyer, 1997) in the sense that the architectures are now able to simulate perceptual-motor contact with computer interfaces and devices and so capture the complex interactions between the task environment, cognition, and perceptual-motor behaviour.

Modelling interactive behaviour with an embodied cognitive architecture has a number of advantages over the traditional cognitive engineering approach exemplified by GOMS and its relatives. Perhaps the most important of these is that computational models can actually execute the task, allowing a direct test of the sufficiency of the hypothesised processes. Second, although most cognitive architectures contain built-in timing parameters taken from the psychological literature, unlike cognitive engineering models, they do not require prior estimated times for all subcomponents of a task. In addition, some architectures—such as ACT-R and Soar—contain learning mechanisms which allow them to model various effects of practice on performance. This allows cognitive architectures to be used to model novel tasks, novice users, or tasks involving components without prior time estimates.

One of the promises of embodied cognitive architectures is that, once they are equipped with sufficient knowledge, they will begin to provide a priori predictions of user performance and eventually evolve into artificial users that can be employed to evaluate novel tasks and environments (Ritter, Baxter, Jones, & Young, 2000; Young, Green, & Simon, 1989). In this chapter we will describe one of these architectures, ACT-R, and show how it has been used to provide detailed and sophisticated process models of human performance in interactive tasks with complex interfaces. ACT-R is an appropriate choice for this discussion because, in contrast to other cognitive architectures, ACT-R also embodies the rational theory of cognition (Anderson, 1990) which analyses cognitive phenomena in terms of how they are adapted to the statistical structure of the environment. Rational analysis and ACT-R's mechanisms have been used recently to provide novel insights into Web-based interactions. The chapter proceeds as follows: First we describe the basic assumptions and mechanisms of rational analysis and the ACT-R cognitive architecture. We then show how these have been used to develop a model of information foraging on the Web and discuss the model in relation to a rational analysis model of the task and the data from eye-tracking studies of interactive search. In the final sections of this chapter we briefly outline ACT-R models of two interactive tasks; graph reading (Peebles & Cheng, 2003) and icon search (Fleetwood & Byrne, 2006). Although neither of these studies involves a specifically Web-based task, they both describe user interaction with items commonly found on Web pages. They are also illustrative of a methodology that combines task analysis, eye tracking, and formal modelling to provide a detailed account of the cognitive, perceptual, and motor processes involved in the performance of the task. These studies are also useful because in both cases the model is validated by comparing the simulated eye movements with those recorded from human subjects. Both studies, therefore, are clear demonstrations of a novel approach to understanding interactive behaviour that can be applied to Web-based tasks.

Rational analysis

Rational analysis (Anderson, 1990) is a method for understanding the task an agent attempts to complete. It assumes that humans have evolved cognitive mechanisms that are useful for completing tasks that we encounter in our environment, and that these mechanisms work in an efficient way to complete these tasks. Therefore, rather than concerning ourselves with firstly trying to define the cognitive mechanisms required by the agent to solve the task, rational analysis suggests that we should consider the structure of the task itself, the environment in which it is encountered, together with some minimal assumptions about the computational limitations of the system. From these initial statements the analysis proceeds by the specification of an optimal solution to the problem and the comparison of human behavioural data to see how close an approximation it is to the optimal solution.

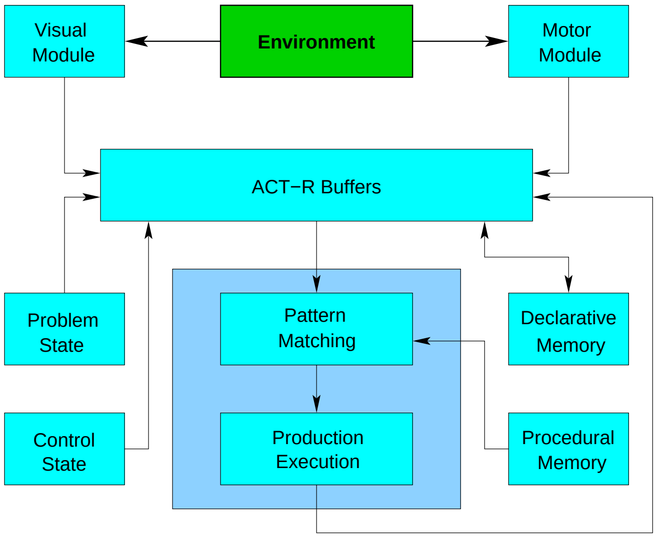

Figure 1: The modular structure of ACT-R 6.0

By identifying the best way to complete the task (the optimal strategy) we can often infer what the cognitive mechanisms of a rational agent must be as although humans do not always complete tasks in the most optimal way their behaviour is usually similar to the optimal strategy. That is, humans usually behave in such a way that they appear to be trying to complete their tasks in the most efficient manner by attempting to maximise their returns while minimising the cost of achieving their goals.

Rational analysis has been applied to several aspects of human cognition (see e.g., Oaksford & Chater, 1998), from the original analyses of memory, categorisation, causal inference, and decision making conducted by Anderson (1990), to more recent analyses of exploratory choice (Cox & Young 2004; Young, 1998) and the updating of memory during tasks in dynamic environments (Neth, Sims, Veksler, & Gray, 2004).

The ACT-R cognitive architecture

ACT-R is a theory of human cognition developed over a period of 30 years by John Anderson and his colleagues (Anderson & Lebiere, 1998; Anderson et al., 2004) that incorporates the theory of rational analysis. It is a principal effort in the attempt to develop a unified theory of cognition (Newell, 1990). As a cognitive architecture, ACT-R attempts to specify the basic cognitive structures and processes that underlie all human cognition.

Figure 1 illustrates the components of the architecture relevant to our discussion. ACT-R consists of a set of independent modules that acquire information from the environment, process information, and execute motor actions in the furtherance of particular goals. There are four modules that comprise the central cognitive components of ACT-R. Two of these are memory stores for two types of knowledge: a declarative memory module that stores factual knowledge about the domain, and a procedural memory module that stores the system's knowledge about how tasks are performed. The former consists of a network of knowledge chunks whereas the latter is a set of *productions*, rules of the form "IF <condition> THEN <action>": the condition specifying the state of the system that must exist for the rule to apply and the action specifying the actions to be taken should this occur. The other two cognitive modules represent information related to the execution of tasks. The first is a control state module that keeps track of the intentions of the system during problem solving, and the second is a problem state module that maintains the current state of the task.

In addition to these cognitive modules there are four perceptual-motor modules for speech, audition, visual, and motor processing (only the latter two are shown in Figure 1). The speech and audition modules are the least well-developed and, at present, simply provide ACT-R with the capacity to simulate basic audio perception and vocal output for the purpose of modelling typical psychology experiments. The visual and motor modules are more well-developed and provide ACT-R with the ability to simulate visual attention shifts to objects on a computer display and manual interactions with a computer keyboard and mouse.

Each of ACT-R's modules has an associated buffer that can hold only one chunk of information from its module at a time, and the contents of all of the buffers constitute the state of an ACT-R model at any one time. Cognition proceeds via a pattern matching process that attempts to find productions with conditions that match the current contents of the buffers. There then follows a process to select the "best" production from those that match the conditions, after which the most appropriate production "fires" and the actions (visual or manual movements, requests for the retrieval of a knowledge chunk from declarative memory, or modifications to buffers) are performed. Then the matching process continues on the updated contents of the buffers so that tasks are performed through a succession of production rule firings. As an example, two production rules (written in English rather than in ACT-R code) that instantiate part of a search task may look something like this:

IF	the goal is to find the meaning of "eudaimonia" (control state)
AND	there is nothing in declarative memory about "eudaimonia" (declarative)
THEN	set the goal to search the WWW for "eudaimonia" (control state)

IF	the goal is to search the WWW for "eudaimonia" (control state)
AND	the Web browser is open (problem state)
THEN	look for the menu labelled "Bookmarks" (visual)
AND	update the problem state to "looking for Google" (problem state)

The processing in ACT-R's modules is serial but the modules run in parallel with each other so that the system can move visual attention while also moving the mouse and attempting to retrieve knowledge from declarative memory. ACT-R processes also have associated latency parameters taken from the psychology literature. For example, it typically takes 50 ms for a production to fire and the time taken to move the mouse cursor to an object on the computer screen is calculated using Fitts' Law (Fitts, 1954).

ACT-R implements rational analysis in two ways. The first is its mechanism for retrieving knowledge chunks from declarative memory which is based on the notion of activation. Each chunk in declarative memory has a level of activation which determines its probability and latency of retrieval, and the level of activation for a chunk reflects the recency and frequency of its use. This enables us to understand how rehearsal of items in a short-term memory task can boost the activation levels of these chunks and consequently increase the chances of recall/retrieval from declarative memory. The level of activation of a chunk falls gradually over time, and without retrieval or activation spreading from chunks in the current goal, it may fall below a threshold level which then results in retrieval failure. This enables ACT-R models to forget knowledge without having to explicitly delete chunks from the declarative memory store.

The second way that ACT-R implements rational analysis is in its mechanism for choosing between alternative production rules. According to rational analysis, people choose between a number of options to maximise their expected utility. Each option (i.e., production rule) has an expected probability of achieving the goal and an expected cost. It is assumed that when carrying out computer-based tasks people interact with the task environment and choose actions that will optimise their efficiency (i.e., maximise the probability of achieving the goal while minimising the cost, usually measured in units of time). At each decision step in the cycle, therefore, all possible production rules that match against the current goal are proposed in a choice set, and the one with the highest level of efficiency is chosen and executed.

ACT-R has been used to model a wide range of cognitive phenomena (Anderson & Lebiere, 1998), and in recent years, with the inclusion of the perceptual-motor modules, it has been applied to a number of complex interactive tasks in the area of HCI and human factors research, for example, menu selection (Byrne, 2001), cell phone menu interaction (St. Amant, Horton, & Ritter, 2004), and driving (Salvucci & Macuga, 2002). Although individually these models do not yet offer us a virtual "user" which can be sat in front of a Web browser and asked to complete any goal, together they provide us with insights into how and why users behave in particular ways, for example, when searching for information on the Web. In this chapter we will concentrate on three particular areas of work that are relevant to understanding Web behaviour: icon search, graph reading, and information foraging on the WWW.

Modelling interactive behaviour

In the following section, we will summarise a number of recent studies which employ rational analysis, cognitive modelling, eye tracking, or a combination of all three, to understand human performance in Web-based or HCI tasks. We first discuss recent efforts to model information foraging and interactive search on the WWW. These studies show how ACT-R and rational analysis can be successfully applied to explain different aspects of people's behaviour when conducting interactive search tasks. This can include both high-level behaviours such as backtracking through Web-pages and low-level behaviours such as patterns of visual attention obtained from eye-tracking studies. We then describe two studies which combine experimental data collection, eye movement recording, and cognitive modelling methods using ACT-R to provide detailed accounts of the cognitive, perceptual, and motor processes involved in the tasks. These studies were chosen because both develop a detailed process model which not only captures the human response time data from the experiment, but also provides a close match to the patterns of visual attention revealed by the eye movement study. This level of detail in modelling is still relatively uncommon and the strong constraints added by seeking to match model and human eye movement scan paths during the course of the task provide a further validation of the models.

Information foraging on the World Wide Web

Information foraging theory (IFT) (Pirolli & Card, 1999) describes an account of information gathering behaviour based on the ecological behaviours of animals when foraging for food. The account can be applied to situations in which people are searching for information in a number of different situations such as in a library or on the WWW. The theory rests on rational analysis in that it proposes that human behaviour is directed by the objective to maximise gain and minimise effort, and that this process is sensitive to changes in the environment. In contrast to animal studies, where the assumption is that animals seek to reduce the ratio of calorie intake to energy expenditure, the assumption in IFT is that people attempt to reduce the ratio of information gained to time spent.

The way in which the environment is structured determines the costs of search for information. For example, the structure of a Web site will determine how many pages the user has to navigate through in order to satisfy his/her goal. When searching for information on the WWW, many people make use of search engines. After entering some key words the user is presented with a list of search results which are usually ordered in terms of their relevance to the key words. Each of the results returned can be considered to be a "patch" of information. The user has to choose to either investigate one of the patches or to redefine their search criteria. Conducting another search using different key words will result in a change in the environment. This process is known as *enriching* the environment as it is hoped that the result is that the cost of obtaining the required information will be reduced compared to the perceived cost of obtaining it in the previous environment. Decisions about whether or not to pursue a particular information patch or to continue enriching the environment are based on a number of factors such as the perceived value of the information returned, the perceived costs of acquiring that information, interface constraints, and previous knowledge.

The decision to forage within a particular patch of information is based on an ongoing assessment of information *scent*. Information scent is the perception of the value of the distal information based on the proximal information available, that is, it is an estimate of the relevance of the information contained on a yet unseen page based on the cues from the icon

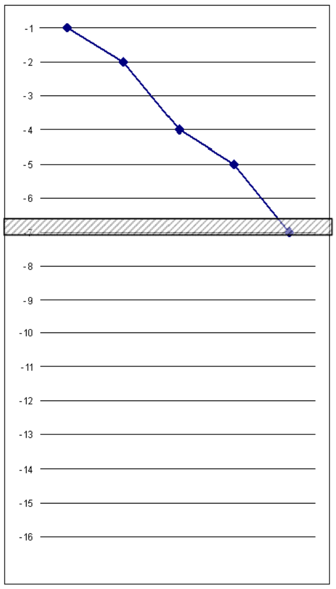

Figure 2: A simplified scan path of a participant performing an interactive search task

or wording of the link on the page currently viewed. The theory predicts that as more time is allocated to within-patch foraging, the rate of information return increases but only up to an optimal point, after which the rate starts to decrease. Therefore, after a particular amount of within-patch foraging (searching within a Web site) it becomes more profitable to move to the next patch (select another Web site from the list of search results) even though there are still pages within the previous patch that have not yet been visited.

SNIF-ACT

Scent-based Navigation and Information Foraging in the ACT architecture (SNIF-ACT) (Pirolli & Fu, 2003) is a model of human behaviour in an interactive search task. The model makes use of ACT-R's spreading activation mechanism so that the information scent of the currently viewed Web page activates chunks in declarative memory as does the spreading activation from the goal. Where these two sources of activation coincide there are higher levels of activation and this indicates a high degree of relevance between the goal and the page being attended to. This activation is what ultimately drives the behaviour of the model. The model includes the use of search engines to provide a set of search results and the processing of the page that is returned. The links on the page are attended to and eventually one of the links is selected.

The behaviour of the model is compared to user behaviour and successfully demonstrates that people tend to select the highest scent item in a list. SNIF-ACT does this by assessing the information scent of all the links on a page and then choosing the highest one. The model is also able to explain the point at which a user abandons a particular Web site and returns to the search results in order to select another item from the list or selects a link that takes them to another Web site. If the mean information scent of the currently viewed page is lower than the mean information scent of a page on another site the model selects that action that takes them to the other site.

Eye-tracking experiments in interactive search

When presented with a list of search results or items on a menu within a Web site (i.e., a patch of information), the user has to choose between selecting an item which will move him/her to another patch and doing some assessment on either the currently attended item or some other item in the list (i.e., consume the information presented within the current patch). As has been mentioned previously, IFT proposes that the user will make use of the information scent of the items to guide their behaviour. If the information scent of a particular item in the list is higher than the rest (i.e., that item appears to be relevant to the task and the user believes that clicking it will lead them to better information) then the item will be selected.

Eye-tracking experiments have been used to investigate what people attend to when conducting interactive search tasks (Brumby & Howes, 2004; Silva & Cox, 2005). Participants were given an information goal and a list of items and asked to select the label that they thought would lead to the information they required. Brumby and Howes demonstrated that people often examine only a subset of the list before selecting the target item, and that this behaviour is affected by the relevance of the other items in the list. When the other items in the list are more relevant to the goal (i.e., they have high levels of information scent), people tend to look at more items in the list and also tend to look at individual items on more occasions than when the items are irrelevant. When there are a number of items with high scent (i.e., two or more items look like they would lead to relevant information) people need to consider more items than when only one item looks sensible.

However, one limitation of this work is that the analysis of eye-tracking data is rarely sensitive enough to determine whether a lack of fixation of the eyes on an item really means that people have not assessed the relevance of the item. In order to address this, Silva and Cox (2005) additionally employed a recognition task in their study in order to assess the level of processing of each item in the list.

Figure 2 represents a simplified scan path of a participant completing one of these tasks. The items are represented on the y axis with time along the x axis. The highlighted item is the target item and was selected by the participant. The figure demonstrates how the user starts at the top of the list and scans down the list fixating items in the list. Some of the items (3 & 6) are skipped over. The results from Silva and Cox's (2005) recognition task suggest that in such cases the lack of fixations of particular items in the menu can be explained by parafoveal processing. However, parafoveal processing can only explain lack of fixations on up to two items below the last fixation (i.e., items 8 & 9) and cannot explain why the user does not attend to other items in the list (i.e., items 10 to 16).

SNIF-ACT would be able to produce a trace that would match the behaviour of users in these studies in terms of which items from the menus the user selected. However, the model does not account for the fact that some of the items in the menus were not assessed by the users as it assumes that users have knowledge about information scent of all the items in the list and then select the item with the highest level of scent. Consequently, SNIF-ACT is unable to provide us with any explanation for why users should choose to select an item when they have not even read the entire list presented to them.

Cox and Young (2004) propose an alternative model to that of SNIF-ACT that is able to capture this fine-grained level of detail of user behaviour. Their model is a rational analysis of an interactive search task that provides a rational explanation of why the user would select an item without first assessing all the items in the list.

In interactive search, the agent has the goal of selecting the item that will lead to goal completion. However, as the menu presented is novel, the first thing that the model has to do is to gain some information about the menu. The model therefore includes two types of exploratory acts (EAs) (these are the different types of things the model can do): assess information SCENT and ANTICIPATE the result of selecting this item. The SCENT EA should be thought of as being an amalgamation of perceiving the label, reading the label (at a lexical level), and considering the semantic similarity between the label and the current task. The ANTICIPATE EA should be thought of as some additional cognitive effort that considers whether the label is likely to lead to the goal. For example, given the goal of finding an armchair for your living room on a furniture shop Web site, imagine the model considering the first item in the menu "home". The SCENT EA would return a moderately high rating as the label has a moderately high level of information scent given the goal ("home" and "armchair"). The ANTICIPATE EA models the agent's consideration of whether the label home is likely to lead to the home page of the site, or to a list of home furnishings. Each of these EA types has a cost associated with it with the ANTICIPATE EA type being more expensive in mental effort than the first type. There is also a fixed cost of moving attention from one item in the menu to the next.

Before assessing any items, the model "knows" the number of items in the menu and considers each of these items to be equally (ir)relevant to completing the task. The scent ratings of the items in the menu are used as the basis for determining the new relevance (R) value of an item following an assessment. On each page, the set of relevancies R_i are mapped into a set of probabilities P_i by the transformation $P_i = odds(R_i)/\Sigma odds(R_j)$, where odds(R) is defined in the standard way as odds(R) = $R/(1-R)$. Note that $\Sigma P_i = 1$, reflecting the fact that exactly one option on the page leads to the goal.

When the model is run on a set of menus it demonstrates how different patterns of information scent result in different behaviours. As Brumby and Howes (2004) demonstrated, the levels of information scent of both the goal item and the distractors affect behaviour. However, it is also interesting to note that the model predicts that just the change in position of the goal item relevant to the distractors results in different patterns of behaviour: Sometimes the model predicts that users will scan to the bottom of the menu before selecting the target item, and other times they will select the item immediately after assessing the item leaving other items in the menu unassessed. To explain how this occurs we will compare the behaviour of the model when the high scent item is in position two (as an example of occurring early in the menu) and in position 12 (as an example of occurring late in the menu) in more detail. In both examples, initially, all 16 menu items are rated equally and all have an R value of 0.06. The relevance values are translated into efficiencies (E) which are then used to determine which of the EAs is most likely to lead to the goal and therefore which EA is executed in each cycle. In the first cycle, the EA that proposes assessing the scent of the first item in the menu is rated as having the highest E value due to it having the lowest cost. Consequently, the model assesses the first item which gets rated as very low scent. As a result, the new R value of this item is set at 0. On the next cycle, the EA that proposes SCENT assessment on the second item in the list is the most efficient (due to the lower cost) so this item gets assessed. This behaviour continues until the model assesses the high scent item.

In menus where the high scent item occurs early on in the menu, the second item in the menu gets an R value of 0.5097 which raises the probability that this item will lead to the goal to 0.6220. On the following cycle the R value of the high scent item leads to an E value of 0.008 while the second best item (an item yet to be assessed) has an R value of 0.06 which results in an E value of 0.006. Although the E values of the two EAs are very similar, one is larger than the other, and this is what determines which EA is chosen.

In our example of a menu where the high scent item occurs later on in the menu, the relevance of each of the low scent items that have already been assessed falls to 0. When the model assesses the twelfth item its R value is 0.5097, which raises the probability that this item will lead to the goal to 0.6220. On the following cycle the R value of the high scent item only has an E value of 0.005 while the item with the best efficiency (an item yet to be assessed) has an R value of 0.05 which results in an E value of 0.006. The result is that the model continues to assess each item in the menu until it reaches the bottom because the efficiency of conducting a SCENT assessment of a new item is greater than the efficiency of conducting the ANTICIPATE assessment on the high scent item in position 12. This has the effect of slowly increasing the probability of the item in position 12 leading to the goal.

The detail of the model explains that the reason the behaviour is different for the two types of menus is because the detail of the mathematics of the rational analysis. Comparisons of the traces of the model with the empirical data suggest that the model provides a good explanation of the cognitive processes involved in this task. This suggests that participants make an assessment of the relevance of a label to the current goal and then, together with the estimated relevance of previous items, choose to either (1) select that item as the one that will lead to the goal, (2) conduct some further assessment of the current item, or (3) move on to another item and assess that. Which of these EAs is chosen is driven by the pattern of information scent that has been experienced so far.

The model provides us with an explanation of how and why the position of the goal and the quality of the distractor items affect the behaviour of the participants on the task. Regardless of the pattern of scent of the menu, the model predicts that the agent will tend to stop exploring the menu as soon as it comes across a menu item that has high information scent (self-terminates) if this is encountered early in the menu. On menus where there is one high scent item among a set of low scent items and the high scent item occurs later in the menu, the agent continues to assess the other items in the menu before conducting further assessment of the high scent item and finally selecting it. The model enables us to explain why we see these different patterns of behaviour on menus which have such similar patterns of information scent. This is due to the effect of the interdependence of the probability that each of the items will lead to the goal. The actual point on the menu at which the model swaps from one behaviour to the other is sensitive to a number of factors such as the length of the menu and the costs of the EAs. It would appear therefore that it is in the nature of interactive search that there are close calls which suggest that people can rationally do either behaviour and that a number of factors have an effect on the behaviour of participants exploring real menus.

Together the two models described previously provide us with a good understanding of how people perform search tasks on the WWW. SNIF-ACT and the rational model explain different aspects of the interaction: SNIF-ACT demonstrates the higher level, page by page, link following behaviour seen in such tasks, whereas the rational model explains the lower level interactions with just one page. Given information about the information scent of the items on a new Web site both models are able to make predictions about user behaviour on the site.

Modelling Graph Reading

Peebles and Cheng (2003) conducted an experiment, eye movement study and cognitive modelling analysis to investigate the cognitive, perceptual, and motor processes involved in a common graph-reading task using two different types of Cartesian graph. The purpose of the study was to determine how graph users' ability to retrieve information can be affected by presenting the same information in slightly different types of the same class of diagram. The two types of graph, shown in Figure 3, represent amounts of UK oil and gas production over two decades. The only difference between the two graph types is in which variables are represented on the axes and which are plotted. In the *Function* graphs, the argument variable (AV: time in years) is represented on the x axis and the quantity variables (QV: oil and gas) on the y axis whereas in the *Parametric* graphs, the quantity variables are represented on the x and y axes and time is plotted on the curve.

In the experiment, participants were presented with the value of a "given" variable and required to use the graph to find the corresponding value of a "target" variable, for example, "when the value of oil is 2, what is the value of gas?" This type of task has typically been analysed in terms of the minimum sequence of saccades and fixations required to reach the location of the given variable's value and then from there to the location of the corresponding value of the target variable (Lohse, 1993; Peebles & Cheng, 2001, 2002; Peebles, Cheng, & Shadbolt, 1999). Experiment participants (some of whom had their eye movements recorded) completed 120 trials, each participant using only one graph type. The 120 questions were coded into three classes (QV-QV, QV-AV, and AV-QV) according to which variable's value was given and which was required (QV denotes a *quantity* variable, oil or gas, and AV denotes the *argument* variable, time). On each trial, a question (e.g., "GAS = 6, OIL = ?") was presented above the graph and participants were required to read the question, find the answer using the graph on the screen and then enter their answer by clicking on a button labelled *Answer* in the top right corner of the window which revealed a circle of buttons containing the digits 0 to 9. RTs were recorded from the onset of a question to the mouse click on the Answer button.

The RT data from the experiment, displayed in Figure 4, showed that the graph used and the type of question asked both had a significant effect on the time it took for participants to retrieve the answer. This was all the more surprising because, for two of the three question types, participants were faster using the less familiar parametric graphs by nearly a second.

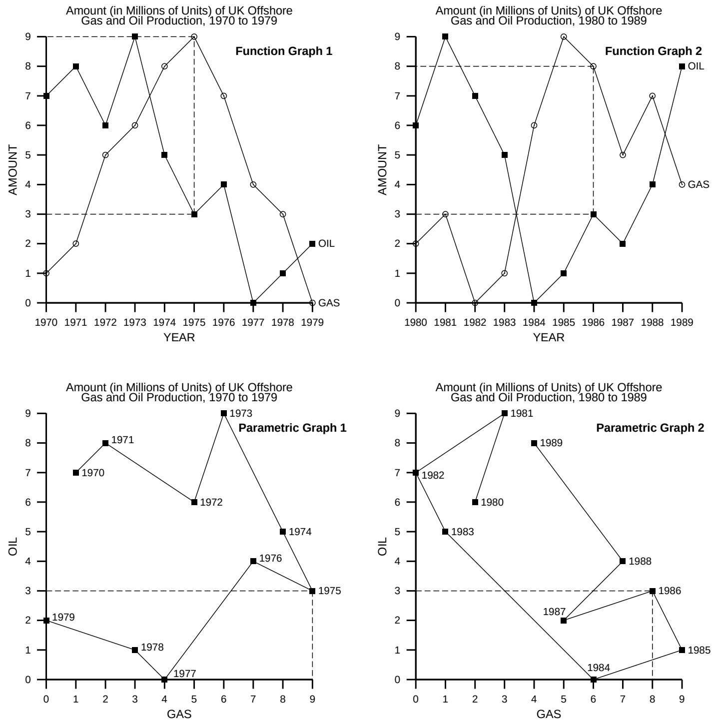

Figure 3: Function and parametric graphs used in Peebles and Cheng (2003) depicting values of oil and gas production for each year. NB. The graphs on the left (labelled 1) show years 1970 to 1979 while those on the right (labelled 2) show years 1980 to 1989. Dashed lines indicate the optimal scan path required to answer the question, "when the value of oil is 3, what is the value of gas?"

The results of the eye movement study were also surprising. It was found that in 63% of trials (irrespective of the graph used or question type being attempted), after having read the question at the start of a trial, participants redirected their visual attention to elements of the question at least once during the process of problem solving with the graph. This was not predicted by the simple minimal fixation sequence account outlined previously but two possible explanations may be provided: (1) participants initially encode the three question elements but are unable to retain all of them in working memory and retrieve them by the time they are required to do so, or (2) to reduce the probability of retrieval failure, participants break the problem into two sections, the first allowing them to reach the given location and the second to then proceed to the target location corresponding to the solution.

Peebles and Cheng constructed two ACT-R models of the experiment (one for each graph type) that were able to interact with an exact replica of the experiment software. The models consisted of a set of productions to carry out the six basic subgoals in the task; (1) read the question; (2) identify the start location determined by the given variable; (3) identify the given location on the graph representing the given value of given variable; (4) from the given location, identify the target location representing the required variable; (5) identify the target value at the target location; and (6) enter the answer. Many of the productions were shared by the two models, the main difference between them being the control structure that sequences the execution of the productions. Figure 4 shows that the mean RTs from the parametric and function graph models are a good fit to the observed data (R^2 = .868, RMSE = 0.123, and R^2 = .664, RMSE = 0.199 respectively). Perhaps more importantly however, were the insights into the observed eye movement data that came from the modelling process itself. When ACT-R focuses attention on an object on the screen, representations of the object and its location are created in the system's visual buffers which can be accessed by productions.

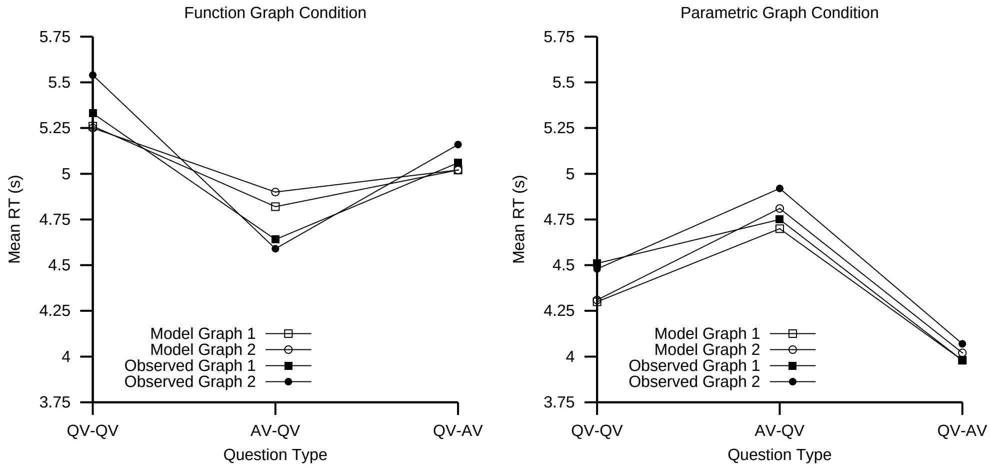

Figure 4: Mean response times for experimental participants and ACT-R models for each question type (Peebles & Cheng, 2003)

Eventually these representations go into declarative memory with initial activation values and, as long as these values are above a certain threshold, they can be retrieved by the cognitive system and replaced in a buffer. However, ACT-R includes a mechanism by which the activation of representations in declarative memory decreases over time which allows it to simulate processes involved in forgetting. These mechanisms played a crucial role in the ACT-R models' ability to capture the eye movement data observed in the experiment.

At the start of each trial, the models read the three question elements and during the problem solving these elements are placed in declarative memory. As a consequence, at least one question element must be retrieved from memory at each stage of the problem in order to continue. However, as soon as a question element is placed in declarative memory its activation starts to decay and, as a consequence, the probability that it cannot be retrieved increases. Typically, if a retrieval failure occurs, an ACT-R model will halt as it does not have the appropriate information to solve the problem. During the process of model development it was found that on a significant proportion of trials the model was not able to retrieve question elements at the later stages of the trial because their activation had fallen below the retrieval threshold. As a consequence new productions had to be added to allow the model to redirect attention to the question in order to re-encode the element and then return to solving the problem. This was precisely the behaviour observed in the eye movement study. This is illustrated in Figure 5 which compares screen shots of the model scan path and eye movements recorded from one participant for the same question using the 1980's parametric graph. The numbered circles on the model screen shot indicate the sequence of fixations produced by the model. The pattern of fixations in both screenshots is remarkably similar.

Modelling icon search

Fleetwood and Byrne's study of icon search (2002, 2006) is a another demonstration of how an ACT-R cognitive model can provide a detailed account of the cognitive and perceptual processes involved in a common HCI task that closely matches people's response times (RTs) and patterns of eye movements. Fleetwood and Byrne's model differs from that of Peebles and Cheng in that it incorporates eye movements and movement of attention (EMMA; Salvucci, 2001), a computational model of the relationship between eye movements and visual attention. EMMA can be easily integrated into the ACT-R architecture, allowing models to make more detailed predictions of actual eye movements, rather than simple shifts of visual attention.

One of the main aims of Fleetwood and Byrne's study was to investigate the notion of icon "quality" (defined in terms of an icon's distinctiveness and visual complexity) and to examine

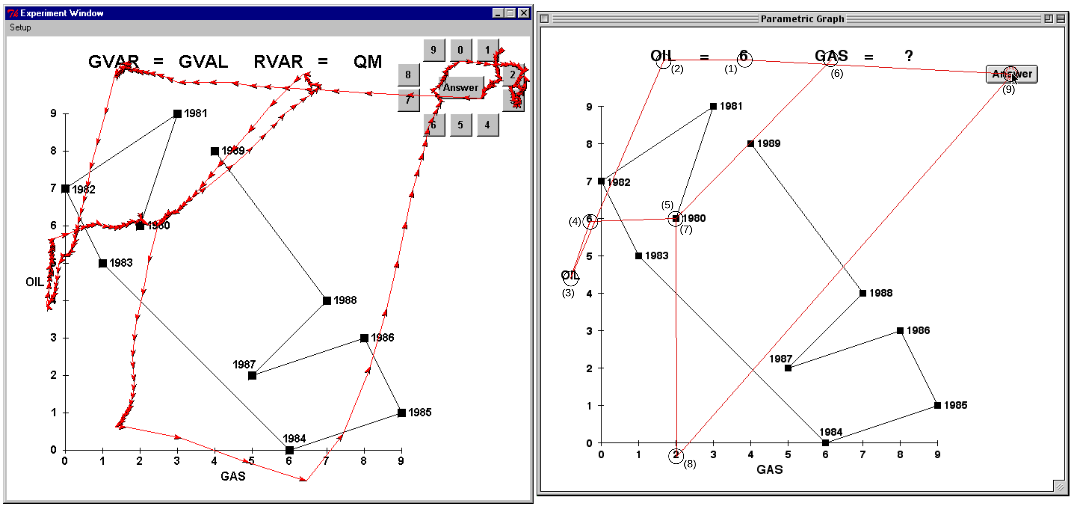

Figure 5: Screen shots showing an experimental participant's eye movement data (left) and the ACT-R model's visual attention scan path (right) for the QV-QV question "oil = 6, gas = ?" using the 1980's parametric graph. NB. In the model screen shot, numbered circles on the scan path indicate the location and sequence of fixations.

the effect that differences in quality may have on identification performance. They created three classes of icon (examples of which are shown in Figure 6). "Good" quality icons were designed to be easily distinguishable from others based on the primitive features of colour and shape. All icons in this set were a combination of one colour (from six) and one shape (from two).

In contrast, "poor" quality icons were designed to be distinguishable only by a relatively careful inspection but to be relatively indistinguishable in a large distractor set. These poor quality icons were all of the same basic shape and colour (a combination of black, white, and shades of grey). An intermediate class of "fair" quality icons was also designed with shapes more distinctive than the poor quality icons but more complex than the good quality icons, and with the same range of greyscale colours as the poor quality icons. The main effect of the manipulation was to produce a different similarity structure for each class of icons. Good quality icons could be identified as a single combination of features, for example, "yellow triangle". In contrast, fair quality icons were defined by more than one combination of features (typically three, for example: "grey rectangle; black square; black diagonal-right"), some of which were shared with other icons. In the poor quality group, icons were defined by an average of four feature combinations and many more of these were shared by several other icons in the group. From the visual search literature, it can be predicted that search time will increase as icon distinctiveness decreases. An additional factor in Fleetwood and Byrne's study also known to affect search time (at least for certain stimuli) is the number of distractors in the display, with search time increasing with the number of distractors in the search set. In their experiment, Fleetwood and Byrne had search sets of 6, 12, 18 and 24 icons.

In the experiment, participants were required to find, as rapidly as possible, different quality target icons in search sets of differing sizes. On each trial, a target icon and file name were presented followed 1500 ms later by a button labelled *Ready* for the participant to click when he/she felt ready to continue. When this button was clicked, the target icon was replaced by the search set and the participant had simply to look for the target icon and click on it as quickly as possible; when an icon was clicked upon, the next trial started. Participants completed a total of 144 trials, involving all levels of the search set and icon quality variables, and on each trial the participant's RT (the duration between clicks on the Ready button and an icon in the search set) was recorded. The results of the experiment (shown in Figure 7)

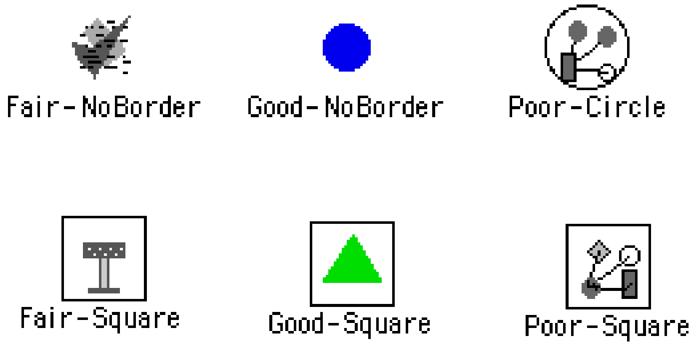

Figure 6: Examples of icons of good, fair, and poor quality used in the experiment of Fleetwood and Byrne (2006)

revealed that, as predicted, both icon quality and search set size had a significant effect on search time.

To provide an explanation of their data, Fleetwood and Byrne produced an ACT-R model of the task that was able to interact with the same experiment software as the participants. As described previously, each experiment trial is comprised of two stages, the first where the target icon and its file name are encoded and the second in which it is sought. The model has a set of seven productions to carry out the first stage: locate the target icon and encode an attribute pair (e.g., “grey rectangle”), look below the icon and encode the associated file name, and finally locate and click on the Ready button. In the second stage, the model locates and attends to an icon with the previously encoded target feature and then shifts visual attention to the file name below it. If the file name matches the target file name, visual attention is returned to the icon and the mouse clicks on it. If the file name is not the target, however, the model continues the search by locating another icon at random with the same target features. This sequence of events requires four productions and takes 285 ms to complete.

Figure 7 reveals a close correspondence between the mean RTs produced by the model and those of the experiment participants (R^2 = .98, RMSE = 126ms) and shows that an ACT-R model based on the similarity structure of the search set and the strategy of identifying a single combination of features and random search can provide a reasonable account of the data. However, Byrne, Anderson, Douglass, and Matessa (1999) had shown in an earlier study of visual search in a menu selection task that alternative strategies can produce similar aggregate RTs, necessitating the incorporation of eye movement data to add further constraints on the proposed theory. As a result, Fleetwood and Byrne carried out an eye movement study to test their model further and found two major discrepancies between the observed eye movements and the patterns of visual attention produced by their model. First, they found that, although the model successfully reproduced the patterns of visual attention across the icon quality and set size conditions, for all conditions the number of saccades per trial produced by the model was significantly greater than those recorded in the experiment. Second, when analysing the eye movement data, Fleetwood and Byrne found that patterns of icon search were not random as their model predicted, but were systematic, in the sense that participants sought to minimise the distance between successive fixations, typically looking at target icons closest to their current fixation point. This produced a search pattern that revealed a systematic scanning of areas of the display.

Both of the discrepancies between the model and human data are explained by Salvucci’s (2001) EMMA model. It is been demonstrated previously that the relationship between eye movements and visual attention is not direct, and that people often do not move their eyes to their focus of attention (e.g., Henderson, 1992; Rayner, 1995). EMMA attempts to capture this relationship by providing an account of if and when eye movements occur, and if they do occur, the location of their landing relative to their targets. Integrating EMMA into ACT-R allows models to simulate actual eye movements rather than just visual attention shifts and

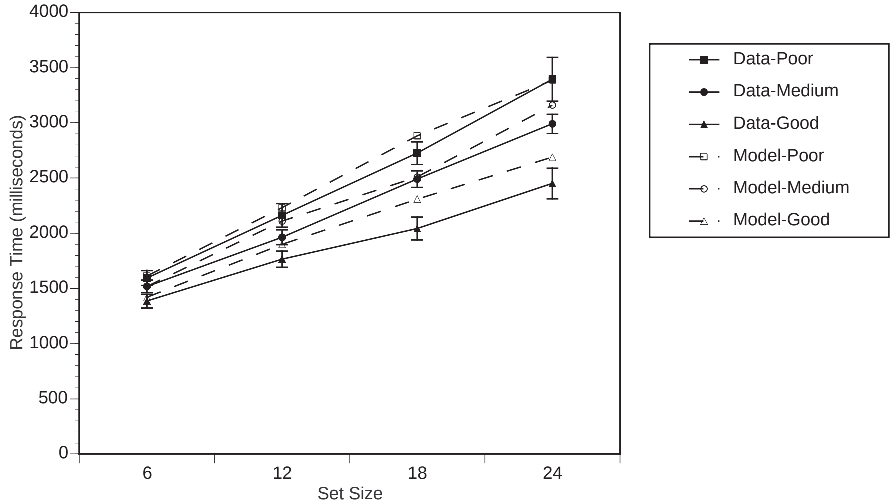

Figure 7: Response time by set size and icon quality for Fleetwood and Byrne's (2006) revised model and the experiment data.

provides a more realistic output to be compared with human eye movement data. In addition, EMMA predicts that efficient search strategies minimise average saccade distance, resulting in search patterns in which objects nearest to the current fixation point are examined soonest. Fleetwood and Byrne modified their model's search strategy according to the EMMA account and incorporated EMMA's eye movement computations into their model, resulting in a greatly improved fit (shown in Figure 8) to the human eye movement data (R^2 = .99, RMSE = 0.58).

Conclusion

In this chapter we have presented a number of recent examples of research that we believe clearly demonstrate the value of rational analysis and cognitive modelling in the study of complex interactive behaviour. Such tasks typically involve the complex interaction of three elements: (1) the perceptual and cognitive abilities of the user; (2) the visual and statistical properties of the task environment; and (3) the specific requirements of the task being carried out. The use of rational analysis and an embodied cognitive architecture such as ACT-R allows all three of these elements to be brought together in an integrated theoretical account of user behaviour. Rational analysis provides a set of assumptions and methods that allow researchers to understand user behaviour in terms of the statistical structure of the task environment and the user's goal of optimising (i.e., reducing the cost/benefit ratio of) the interaction. Developing cognitive models of interactive behaviour in a cognitive architecture such as ACT-R allows researchers to specify precisely the cognitive factors (e.g., domain knowledge, problem-solving strategies, and working memory capacity) involved. In addition, the recent incorporation of perceptual-motor modules to cognitive architectures allows them to make predictions about users' eye movements during the entire performance of the task, which can be compared to observed eye movement data a highly stringent test of the sufficiency and efficacy of a model. The use of these methods has increased rapidly over the last 5 years, as has the range of task interfaces being studied. Although we are still a long way from achieving the goal of an artificial user that can be applied "off the shelf" to novel tasks and environments, the models of interactive behaviour described here demonstrate a level of sophistication and rigour still relatively rare in HCI research. As these examples illustrate, developing more detailed accounts of interactive behaviour can provide genuine insights into the complex interplay of factors that affect the use of computer and Web technologies, which may inform the design of systems more adapted to their users.

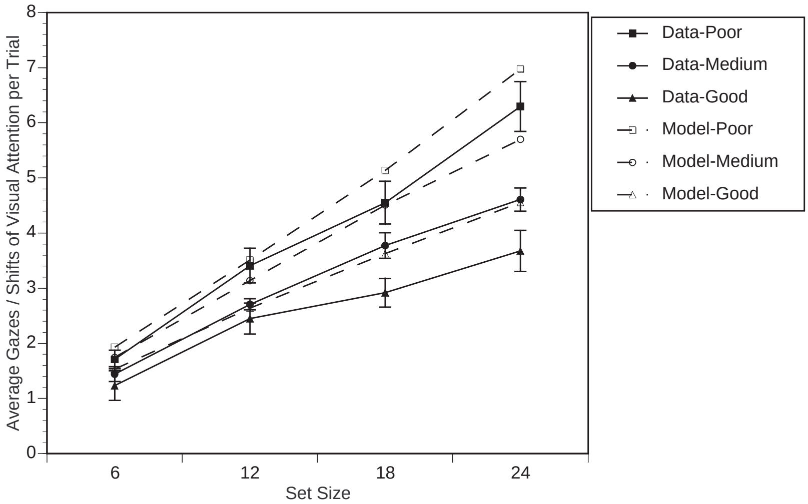

Figure 8: Mean number of shifts of visual attention per trial made by Fleetwood and Byrne's (2006) revised model relative to the mean number of gazes per trial made by participants.

References

Anderson, J. R. (1990). *The adaptive character of thought.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review*, *111*, 1036–1060.

Anderson, J. R., & Lebiere, C. (1998). *The atomic components of thought.* Mahwah, NJ: Lawrence Erlbaum Associates.

Brumby, D. P., & Howes, A. (2004). Good enough but I'll just check: Web-page search as attentional refocusing. In *Proceedings of the 6th International Conference on Cognitive Modelling.* Mahwah, NJ: Lawrence Erlbaum Associates.

Byrne, M. D. (2001). ACT-R/PM and menu selection: Applying a cognitive architecture to HCI. *International Journal of Human-Computer Studies*, *55*, 41–84.

Byrne, M. D., & Anderson, J. R. (1998). Perception and action. In J. R. Anderson & C. Lebiere (Eds.), *The atomic components of thought.* Mahwah, NJ: Lawrence Erlbaum Associates.

Byrne, M. D., Anderson, J. R., Douglass, S., & Matessa, M. (1999). Eye tracking the visual search of click-down menus. In *Proceedings of the ACM CHI '99 Conference on Human Factors in Computing Systems*, (pp. 402–409). New York: ACM.

Card, S. K., Moran, T. P., & Newell, A. (1983). *The psychology of human-computer interaction.* Hillsdale, NJ: Erlbaum.

Chen, M. C., Anderson, J. R., & Sohn, M. H. (2003). Eye-hand coordination during Web browsing. In J. Ratner (Ed.), *Human factors and web development.* Mahwah, NJ: Lawrence Erlbaum Associates.

Chong, R. S., & Laird, J. E. (1997). Identifying dual-task executive process knowledge using EPIC-Soar. In *Proceedings of the 19th Annual Conference of the Cognitive Science Society*, (pp. 107–112). Mahwah, NJ: Lawrence Erlbaum Associates.

Cox, A. L., & Young, R. M. (2004). A rational model of the effect of information scent on the exploration of menus. *In Proceedings of 6th International Conference on Cognitive Modeling.* Mahwah, NJ: Lawrence Erlbaum Associates.

Fitts, P. M. (1954). The information capacity of the human motor system in controlling the amplitude of movement. *Journal of Experimental Psychology*, *47*, 381–391.

Fleetwood, M. D., & Byrne, M. D. (2002). Modeling icon search in ACT-R/PM. *Cognitive Systems Research*, *3*, 25–33.

Fleetwood, M. D., & Byrne, M. D. (2006). Modeling the visual search of displays: A revised ACT-R/PM model of icon search based on eye tracking data. *Human Computer Interaction*, *21*, 153–197.

Henderson, J. M. (1992). Visual attention and eye movement control during reading and picture viewing. In K. Rayner (Ed.), *Eye movements and visual cognition: Scene perception and reading*. New York: Springer-Verlag.

John, B. E., & Kieras, D. E. (1994). *The GOMS family of analysis techniques: Tools for design and evaluation* (Tech. Rep. No. CMUHCII94106). Pittsburgh, PA: Carnegie Mellon University, Human-Computer Interaction Institute.

Kieras, D. E., & Meyer, D. E. (1997). An overview of the EPIC architecture for cognition and performance with application to human-computer interaction. *Human-Computer Interaction*, *12*, 391–438.

Laird, J. E., Newell, A., & Rosenbloom, P. S. (1987). SOAR: An architecture for general intelligence. *Artificial Intelligence*, *33*, 1–64.

Lohse, G. L. (1993). A cognitive model for understanding graphical perception. *Human Computer Interaction*, *8*, 353–388.

Neth, H., Sims, C. R., Veksler, V. D., & Gray, W. D. (2004). You can't play straight TRACS and win: Memory updates in a dynamic task environment. In K. D. Forbus, D. Gentner, & T. Regier (Eds.), *Proceedings of the 26th Annual Meeting of the Cognitive Science Society* (pp. 1017–1022). Hillsdale, NJ: Lawrence Erlbaum Associates.

Newell, A. (1990). *Unified theories of cognition*. Cambridge, MA: Harvard University Press.

Oaksford, M., & Chater, N. (Eds.). (1998). *Rational models of cognition*. UK: Oxford University Press.

Olson, J. S., & Olson, G. M. (1990). The growth of cognitive modeling in human-computer interaction since GOMS. *Human Computer Interaction*, *5*, 221–265.

Peebles, D., & Cheng, P. C.-H. (2001). Graph-based reasoning: From task analysis to cognitive explanation. In *Proceedings of the 23rd Annual Conference of the Cognitive Science Society* (pp. 762–767). Mahwah, NJ: Lawrence Erlbaum Associates.

Peebles, D., & Cheng, P. C.-H. (2002). Extending task analytic models of graph-based reasoning: A cognitive model of problem solving with Cartesian graphs in ACT-R/PM. *Cognitive Systems Research*, *3*, 77–86.

Peebles, D., & Cheng, P. C.-H. (2003). Modeling the effect of task and graphical representation on response latency in a graph reading task. *Human Factors*, *45*, 28–46.

Peebles, D., Cheng, P. C.-H., & Shadbolt, N. (1999). Multiple processes in graph-based reasoning. In *Proceedings of the 21st Annual Conference of the Cognitive Science Society* (pp. 531–536). Mahwah, NJ: Lawrence Erlbaum Associates.

Pirolli, P., (2005) Rational analyses of information foraging on the Web. *Cognitive Science*, *29*, 343–374.

Pirolli, P., & Card, S. K. (1999). Information foraging. *Psychological Review*, *106*, 643–675.

Pirolli, P., & Fu, W.-T. (2003). *SNIF-ACT: A model of information foraging on the World Wide Web*. Paper presented at the Ninth International Conference on User Modeling, Johnstown, PA.

Rayner, K. (1995). Eye movements and cognitive processes in reading, visual search, and scene perception. In J. M. Findlay, R. Walker, & R. W. Kentridge (Eds.), *Eye movement research: Mechanisms, processes, and applications*. New York: Elsevier.

Ritter, F. E., Baxter, G. D., Jones, G., & Young, R. M. (2000). Supporting cognitive models as users. *ACM Transactions on Computer-Human Interaction*, *7*, 141–173.

Salvucci, D. D. (2001). An integrated model of eye movements and visual encoding. *Cognitive Systems Research*, *1*, 201–220.

Salvucci, D. D., & Macuga, K. L. (2002). Predicting the effects of cellular-phone dialling on driver performance. *Cognitive Systems Research*, *3*, 95–102.

Silva, M., & Cox, A. L. (2005). *Eye-movement behaviour in interactive menu search: Evidence for rational analysis.* Paper presented at the BPS Cognitive Section Conference 2005, University of Leeds.

St. Amant, R., Horton, T. E., & Ritter F. E. (2004). Model-based evaluation of cell phone menu interaction. In *Proceedings of the CHI'04 Conference on Human Factors in Computer Systems* (pp. 343–350). New York: ACM.

Young, R. M. (1998). Rational analysis of exploratory choice. In M. Oaksford & N. Chater (Eds.), *Rational models of cognition* (pp. 469–500). UK: Oxford University Press.

Young, R. M., Green, T. R. G., & Simon, T. (1989). Programmable user models for predictive evaluation of interface designs. In *Proceedings of CHI '89: Human Factors in Computing Systems* (pp. 15–19).

Published in Zaphiris, P. & Kurniawan, S. (Eds). (2006). *Human Computer Interaction Research in Web Design and Evaluation* (pp. 290–309). London. Idea Group Inc.