In Proceedings of the Sixth International Conference on Cognitive Modeling, 154-159.
Mahwah, NJ: Lawrence Earlbaum.

A Queuing Network Model for Eye Movement

Ji Hyoun Lim (limjh@umich.edu)
Department of Industrial and Operations Engineering, University of Michigan
1205 Beal Ave, Ann Arbor, MI 48109 USA

Yili Liu (yililiu@umich.edu)
Department of Industrial and Operations Engineering, University of Michigan
1205 Beal Ave, Ann Arbor, MI 48109 USA

Abstract

Eye movement is a basic human behavior that offers a valuable means to explore human cognitive processes. This article introduces two modeling studies of eye movement. First, random menu search was modeled using a queueing network approach and second, a reinforcement learning algorithm was used to generate various eye movement patterns. Random menu search is a task component involved in many human-machine interfaces and has been modeled with several cognitive models including ACT-R and EPIC. Based upon review of empirical data in menu search, and strengths and limitations of existing models, this article proposes a queueing network model, which has been successfully applied in some other task domains (e.g., response time, driver performance). The queueing network model for random menu search was implemented and evaluated through model simulation. In contrast to existing models that rely on four task-specific strategies to account for data, the queueing network model accounted for the same data using only one strategy already employed in cognitive modeling. To extend this parsimonious, "minimal task strategy" modeling approach, Q-Learning, one of the reinforcement learning methods, was adopted to generate different patterns in eye movement. The same strategy from random menu search was used to generate eye movement, and the simulated eye movements were qualitatively compared to the human eye movement.

I. Introduction

Computational modeling of menu search and eye movement has both practical and theoretical significance. Practically, menu search is almost universally required in computer interfaces. Most computer interfaces contain menus for users to select functions from. Selection of a menu is then followed by a visual search. Theoretically, eye movement is an integral component of cognitive performance. Many experimental studies (Nilsen, 1991; Byrne, Anderson, Douglass, & Matessa, 1999) and computational modeling efforts (Byren, 2001; Hornof, 1999; Anderson, Matessa, and Lebiere, 1997) have been conducted to investigate random menu search and eye movement.

Computational cognitive models are useful because they can integrate results of various experiments, provide understanding of perceptual and cognitive processes underlying observable behaviors, make quantitative predictions for scenarios not yet tested, and provide a precise common language for description of phenomena of interest. We examined existing computational cognitive models of menu search and eye movement and proposed a new computational cognitive model rooted in queueing network theory (Liu, 1996, 1997) that overcomes the limitations of the existing models.

Two studies are introduced here. The first focuses on the development of a computational cognitive model for menu search with a minimal number of eye movement strategies. Performance of the queueing network model is compared to existing models involving complex strategies. The second study investigated the potential performance of our model to generate eye movement patterns. Results show that our model generated various types of eye movement patterns based on the same simple strategy used in the first study. To explain the underlying rules for different eye movement patterns, reinforcement learning algorithm was adopted.

II. Random Menu Search Models

Random menu search is the simplest type of search task because it requires only one-dimensional eye movement (up or down). A menu search task consists of control of aimed movements and visual search of menus. Nilsen conducted a series of menu search experiments using sets of randomly ordered and vertically listed items as the stimuli for menu search (1991). The subject's task was to find a given target from a set of items. Nilsen collected only response time in this experiment. Nilsen's data have been used to validate existing models (Anderson, 1997; Hornof, 1999).

In addition to response time data, Byrne, Anderson, Douglass, and Matessa (1999) considered the use of eye movement data as validation of computational cognitive models for random menu search. According to their eye movement records of a random menu search task, there are two typical phenomena in eye movement for menu search: *varying lengths of saccade* and *overshooting*. A saccade is one type of eye movement characterized as a ballistic movement with a fixed destination. The records of eye movement showed that the length of saccade is not constant. Overshooting is a phenomenon in which the eye moves beyond the location of the target. Byrne's data revealed the existence of negative lengths on the last saccade in menu search, a finding that can be interpreted as overshooting.

II. 1. ACT-R and EPIC Models

Anderson, Matessa, and Lebiere (1997) showed that the ACT-R model could reproduce menu search response time by incorporating a critical production called the *hunt-feature*. The hunt-feature production moves attention by hunting for objects that have a target feature. The hunt-feature production determines the next fixation position, which determines the length of the following saccade.

Hornof (1999) developed the *dual strategy varying distance hybrid model* (DSVDH model) based on EPIC. He addressed the difference between local and global control of eye movement in visual search. In the DSVDH model, he assumed four different search strategies and their fractions of use in their global control of eye movement.

Byrne (2001) compared the performance of ACT-R (Anderson et al., 1997) and DSVDH (Hornof, 1999) in random menu search using eye movement data (Byrne et al., 1999). ACT-R successfully generated the varying length of saccade with the hunt-feature production (Anderson et al., 1997). However, the fundamental serial cognitive processing assumption of the ACT-R architecture did not allow the eyes to overshoot. Since subsequent fixation location is determined before the eye moves in ACT-R, there is no reason for the eyes to move further than the target location.

Unlike ACT-R's serial cognitive processing assumption, the EPIC architecture assumes parallel cognitive processing. Thus, Hornof's (1999) DSVDH model allows multiple target items to be compared simultaneously. Hornof's assumption of multiple strategies allows DSVDH to reproduce eye overshooting and varying length of saccade, although the DSVDH model itself provides insufficient explanation of how and why they occur.

II. 2. Queueing Network Model for Menu Search

The *queueing network* modeling approach was proposed by Liu (1996, 1997) and has been applied successfully to integrate a large number of influential mathematical models in psychology (Liu, 1996), in modeling multitask performance (Liu, 1997), and in modeling driver performance (Tsihmoni and Liu, 2003).

The queueing network model consists of four components: customers, servers, arrivals and routes. The *stimulus*, either outside or inside of human body, is encoded as customers, which need to be processed by servers. Each *server* in a queueing network represents a fundamental functional unit in human information processing. The *arrival* of a customer at a particular server is an event that initiates information processing at that server, and the order of the servers though which a customer should go is the *route* traversed by that customer.

To model a random menu search task with the queueing network, three types of stimuli required in random menu search tasks were defined as customers: precue stimulus, reference stimulus (the target), and a list of menu items containing one target. Nine servers form the architecture of the queueing network model: four servers for the visual perceptual sub-network, three servers for the cognitive sub-network, and two servers for the motor sub-network. Figure 1 shows the layout of servers and routes.

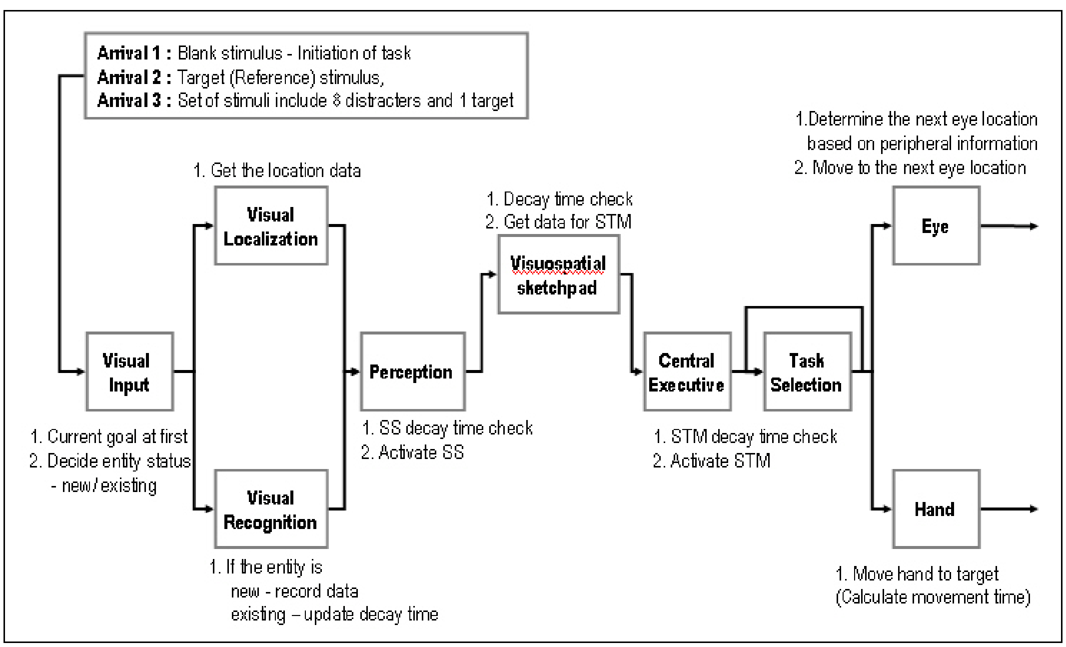

Figure 1. The layout of the servers and routes of the queueing network model for random menu search

The queueing network model for menu search has both serial and parallel routes. The servers located at the same horizontal location in Figure 1, for example *Visual Localization* server and *Visual Recognition* server, can be processed in parallel. The queueing network model relies on two distinct assumptions for menu search task. First, the model uses ACT-R's *hunt-feature* production (Anderson et al., 1997) as the only strategy for eye movement. Second, *separate routes* are suggested for attended and unattended visual information based on the preattentive location encoding theory (Wright and Word, 1998).

The queueing network model was implemented and simulated with Pro-Model—a commercially available simulation software widely used for manufacturing and operational applications. The stimuli consisted of nine randomly ordered numbers, one of which was the target. The location of the target was varied randomly from top to bottom.

The queueing network model was evaluated in two conditions—separate-routes versus same-route. The former had separate-routes for attended and unattended information, allowing partial parallel processing. The latter had the same-route for all information, which was expected to behave like ACT-R due to the shared assumption of serial information processing. The attended information was regarded as being achieved from the center zone of fovea (two degrees of visual angle) on the retina and the unattended information from outer area of fovea.

II. 3. Results and Discussion

Model simulation reproduced both the varying length of saccade and overshooting eye movement phenomena in random menu search task.

The relationship between the number of fixations and the location of target is presented in Figure 2. In the separate-routes condition, the queueing network model generated slightly more fixations because of the eye overshooting.

The major difference in the simulation results between the separate- routes condition and the same-route condition was the presence of eye overshooting. In the separate-routes condition, 30% of the menu selection trials generated overshooting eye movement. In contrast, there was no overshooting eye movement in the same-route condition. Byrne's eye tracking data showed that overshooting existed in visual search (about 32%).

Varying length of saccade was reproduced via use of the hunt-feature production. If the target stimulus was *out of* current unattended vision, the next destination of the saccade would be determined at the very end of the current unattended vision area. This condition generated a longer distance of eye movement than when the target stimulus was *within* an unattended vision area. Since the hunt-feature production used attended and unattended visual information to determine the next location of eye fixation and this was considered a unique rule of eye movement, multiple competing strategies were not required for each saccade.

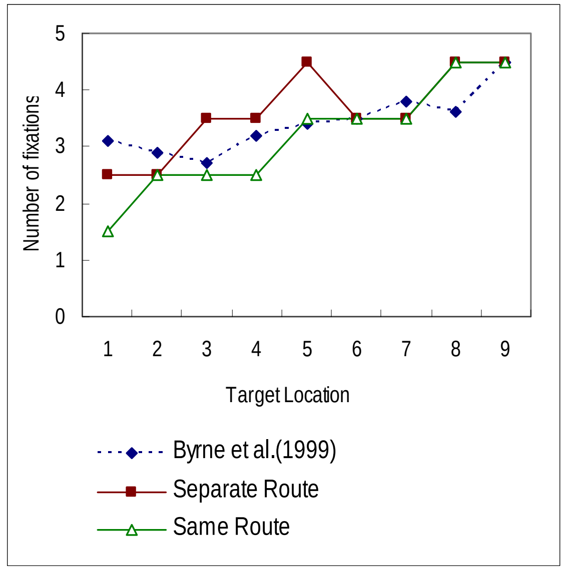

Figure 2. Comparison of the queueing network simulation result to Byrne et al.'s eye tracking data

This study showed that both varying length of saccade and eye overshooting could be generated by a single strategy rather than multiple competing strategies. Furthermore, the unique strategy used in the queueing network model was not proposed here, but was originally incorporated by the ACT-R model (Anderson et al., 1997). This shows a potential strength of the queueing network structure compared to other computational cognitive models.

Based on these initial findings, an interesting question was raised. Would this simple, single strategy be sufficient to model more complex eye movement patterns? The following study shows our effort to answer this question.

III. Single Strategy Eye Movement

In 1967, Yarbus conducted an interesting experiment in which a subject was shown a picture and then asked seven different questions about the picture. The subject's eye movement was recorded for each question and the results showed that the subject moved his/her eyes over the picture very differently depending on the questions.

Modeling eye movement requires considering various patterns in eye movement. If we need multiple strategies to move our eyes along only one-dimension (up and down), as EPIC (Hornof, 1999) and ACT-R/PM (Byrne, 2001) do, it is difficult to estimate the number of strategies required to move the eyes in a two-dimensional space. Furthermore, explaining the different patterns in eye movements would very likely require even more strategies.

This speculation was confirmed in Hornof's recent study on visual search of hierarchical menus, in which additional strategies were needed to produce the varying length of saccade (Hornof, 2003). One of the questions addressed in our modeling work is whether there exists a modeling approach that relies on intrinsic behaviors of a cognitive architecture and is more parsimonious than depending on a large and growing number of task-specific strategies.

Our study shows that a single strategy can explain various patterns in human eye movements by adopting a Q-learning algorithm, which is task independent and is one type of method in reinforcement learning.

III.1. Reinforcement Learning and Eye Movement

This study is not the first attempt to adopt learning methods to visual search or eye movement study. Reward maximization has been used to model human eye movement (Sprague and Ballard, 2003). The ACT-R/PM model (Byrne, 2001) adopted learning method of conflict resolution for multiple competing strategies. Their study focused on 'where' an eye looked at (fixating location) rather than 'what' guided the movements of eye because each strategy is associated with the next eye location. The present study focuses not simply on the location of eye fixations, but on what guides the eye movement

The difference in the research approach between the queueing network model and other models is not only the number of strategies, but also the method used to guide the eyes. The hunt-feature production, which is used in queueing network model for menu search as a unique strategy for eye movement, guides eyes to the next location depending on the information in the unattended visual zone. In other words, the strategy is not directly related to the next location of eye fixation, while each strategy used in the other models explicitly tells the eyes where they should go next.

In the hunt-feature production rule originally used in ACT-R (Anderson, 1997), feature selection is assumed to be random. Here, we suspect that the randomness may be associated with the type of eye movement patterns. Therefore, instead of assuming numerous different strategies to explain different patterns of eye movements, a *reinforcement learning algorithm* is proposed to explain different kinds of patterns by adjusting the probability of selecting a feature used in hunt-feature production rule.

If we sketch eye movement as a problem of location, it becomes a complex problem in a continuous two-dimensional space. When we formulate the various patterns of eye movement into the feature selection problem, we can reduce the problem space into a finite number of alternative features rather than a continuous two-dimensional space problem. The picture used in Yarbus (1967)'s experiment was also used for this study.

III. 2. Q-Learning Model for Eye Movement

Our approach to modeling eye movement was to consider each eye movement as a Markov decision process and then to find an underlying policy using reinforcement learning method. A Markov decision process is described by four attributes: $\boldsymbol{S}$ is the state space, $\boldsymbol{A}$ is the action space, $\boldsymbol{T}(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{s'})$ is the transition function that indicates the probability of arriving in state $\boldsymbol{s'}$ when action $\boldsymbol{a}$ is taken in state $\boldsymbol{s}$, and the reward function $\boldsymbol{R}(\boldsymbol{s}, \boldsymbol{a}, \boldsymbol{x})$.

The objective of the reinforcement learning algorithm is to discover an optimal policy $\pi^*(\boldsymbol{s})$ (Sprague and Ballard, 2003) that includes which action should be taken in a certain state to maximize discounted long term reward.

One way to find the optimal policy $\pi^*(\boldsymbol{s})$ is to obtain the optimal value function $\boldsymbol{Q}(\boldsymbol{s}, \boldsymbol{a})$. This function represents the expected discounted return if action $\boldsymbol{a}$ is taken in state $\boldsymbol{s}$ and the optimal policy is followed thereafter. There are many algorithms (Sutton and Barto, 1998) for learning $\boldsymbol{Q}(\boldsymbol{s}, \boldsymbol{a})$. For the feature selection in eye movement, the on-line $\boldsymbol{Q}$-learning update rule is used.

$$\boldsymbol{Q}(\boldsymbol{s}, \boldsymbol{a}) \leftarrow \boldsymbol{Q}(\boldsymbol{s}, \boldsymbol{a}) + \alpha \,[\, \boldsymbol{r} + \gamma \max_{a'} \boldsymbol{Q}(\boldsymbol{s'}, \boldsymbol{a'}) - \boldsymbol{Q}(\boldsymbol{s}, \boldsymbol{a}) \,]$$

Here, α is a learning rate parameter, and γ is a discounting factor of future reward (Sutton and Barto 1998). In this study, the initial value of learning rate parameter α_0 was 0.3 and it decreased by following rule: $\alpha_n = 0.999\ \alpha_{n+1.}$ The discounting factor of future reward γ is set as 0.5 and does not change.

To apply a reinforcement learning method to our eye movement study, the four attributes such as agent, environment, states, action and reward, were defined in the context of Yarbus' experiment as follows.

The *Agent* was defined as an eye that has two kinds of visual zones: attentional and unattentional visual zones. In Figure 3, the small square indicates attended visual zone and the large square indicates unattended visual zone. The Environment ($\boldsymbol{E}$) was the picture shown to the Agent. The *State* ($\boldsymbol{s} \in \boldsymbol{S}$) was a feature selected by the agent and in this case we assumed there were three alternative features which determined the next eye location. Depending on the state (selected feature), the *Location* ($\boldsymbol{x} \in \boldsymbol{E}$ and $\boldsymbol{x}$ is two-dimensional vector) of eye fixation was determined as the location of attended visual zone.

The *Action* ($\boldsymbol{a} \in \boldsymbol{A}(\boldsymbol{s})$) was a selection of the next hunting feature. There were three alternative criteria of hunting feature: the most similar contrast, the most different contrast, or the most edge information. Though the next eye location was not directly selected by the action, a selected feature determined the next location $\boldsymbol{x}$ of eye as described in Figure 3. Once the next location of eye has been determined, the eye moves to the location in the following saccade.

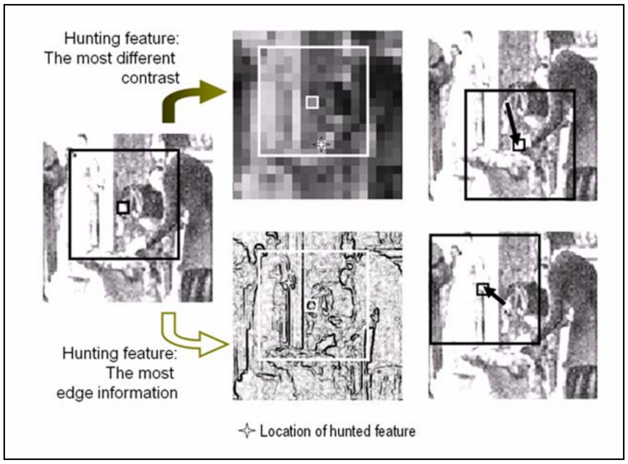

Figure 3. Different location is selected for next eye fixation depend on hunting feature used

Figure 4 presents the nine transition functions. The three states are the three alternative features currently used, and each action can keep the current state or move to other state.

The *Reward* (***R***(***s***, ***a***, ***x***)) was information given in a certain location ***x*** which was determined by an action ***a*** taken in the state ***s***. In this case, the amount of information was encoded as High (10), Low (5), and None (0). The location of reward was assumed to be different according to the goal. For example, under 'give the ages of the people', the rewards were assumed to be distributed mainly at people's face in the picture whereas under 'what family had been doing', the rewards were assumed to be distributed at people's posture. For different reward conditions, a different optimal policy could be obtained by Q-Learning method.

Following the different optimal policy, the simulated eye movement data were collected and they were qualitatively compared to the record of actual human eye movement patterns from Yarbus' experiment.

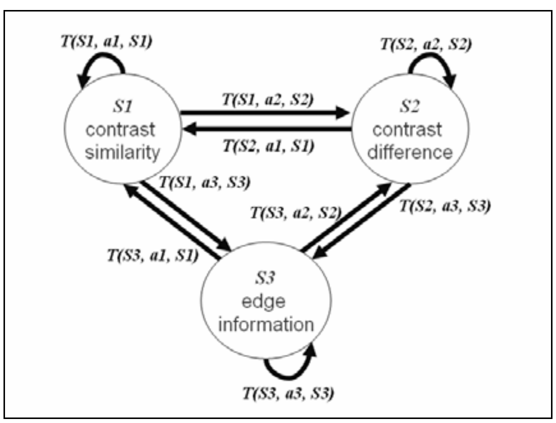

Figure 4. The nine transitions with the three states and the three actions

III. 3. Results of Q-Learning and Discussion

Using 800 randomly selected actions, the value of each ***Q*(*s*, *a*)** was recorded. Two different reward arrays were used for the 'Give the ages of the people' condition and 'What the family had been doing' condition. The results are summarized in Table 1. The proposed policy would be: *In the state **s**, the action **a** is selected with probability **F(s, a)**.* Two different conditions showed the most different policy in state 1.

Table 1. The summary of different optimal policies for two different conditions

a. Give the ages of the people	
T(s, a, s')	***F*(*s*, *a*)**
T(S1, a1, S1)	0.43
T(S1, a2, S2)	0.25
T(S1, a3, S3)	0.32
T(S2, a1, S1)	0.31
T(S2, a2, S2)	0.43
T(S2, a3, S3)	0.26
T(S3, a1, S1)	0.34
T(S3, a2, S2)	0.32
T(S3, a3, S3)	0.34

b. What family had been doing?	
T(s, a, s')	***F*(*s*, *a*)**
T(S1, a1, S1)	0.24
T(S1, a2, S2)	0.29
T(S1, a3, S3)	0.47
T(S2, a1, S1)	0.33
T(S2, a2, S2)	0.35
T(S2, a3, S3)	0.32
T(S3, a1, S1)	0.33
T(S3, a2, S2)	0.34
T(S3, a3, S3)	0.33

S1: the most similar contrast
S2: the most different contrast, and *S3*: the most edge information

Using the two policies above, the viewing tasks were simulated. Figure 5 shows the simulated eye movement of 100 saccades and the human eye movement patterns from Yarbus' experiment.

Due to the lack of actual data from Yarbus' experiment, it was impossible to compare the simulated and the human eye movements quantitatively. However, the simulated eye movement patterns showed striking difference between the different conditions, and also showed similarity with the human eye movement patterns of the same conditions (the questions asked).

The only difference in modeling the two conditions—the 'Give the ages' and 'What the family had been doing'—was the *location of rewards*. All other parameters, input data, and strategy for eye movement were exactly same for both cases.

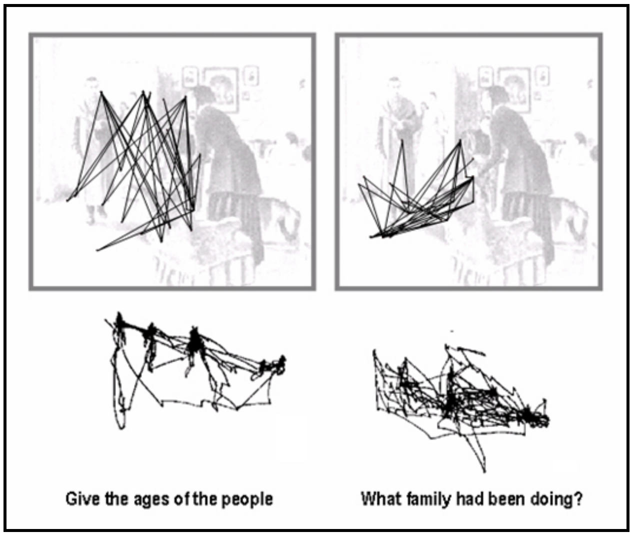

Figure 5. The simulated eye movements (top) and human eye movement (bottom)

In the two studies reported in this article, patterns of eye movement emerged mainly as the behavior of the underlying queueing network with far less reliance on task specific performance strategies than other models. The two studies demonstrated some of the unique aspects of the queueing network approach for modeling eye movement in specific and human performance in general.

ACKNOWLEDGMENTS

This article is based upon work supported by the National Science Foundation under Grant No. NSF 0308000. However, any opinions, findings and conclusions or recommendations expressed in this article are those of the authors and do not necessarily reflect the views of the National Science Foundation (NSF).

REFERENCES

Anderson, J. A., Matessa, M., & Lebiere, C. (1997). ACT-R: A theory of higher level cognition and its relation to visual attention. *Human-Computer Interaction*, 12, 439-462.

Byrne, M. D. (2001). ACT-R/PM and menu selection: applying a cognitive architecture to HCI. *Int. J. Human-Computer Studies*, 55.

Byrne, M. D., Anderson J. R., Douglass, S., & Matessa, M. (1999). Eye tracking the visual search of click-down menu. *Human Factors in Computer Systems: Proceedings of CHI*, 402-209.

Hornof, A. J. (1999). *Computational Models of the Perceptual, Cognitive, and Motor Processes Involved in the Visual Search of Pull-down Menus and Computer Screens*. Ph. D. dissertation in computer science engineering, The University of Michigan, Ann Arbor, Michigan.

Hornof, A. J., & Halverson, T. (2003). Cognitive strategies and eye movements for searching hierarchical computer displays. *ACM CHI 2003: Conference on Human Factors in Computing Systems*, New York: ACM, 249-256.

Liu, Y. (1996) Queuing network modeling of elementary mental processes. *Psychological Review*, 103, 116-136.

Liu, Y. (1997). Queuing Network Modeling of Human Performance of Concurrent Spatial and Verbal Tasks. *IEEE Transactions on system, Man, and Cybernetics – Part A: Systems and Humans.* 27(2), 195-207.

Nilsen, E. L. (1991). *Perceptual-Motor Control in Human-Computer Integration*. Ph.D. dissertation in Psychology, The University of Michigan, Ann Arbor, Michigan..

Sprague, N. and Ballard, D (2003), Eye Movements for Reward Maximization, *Advances in Neural Information Processing Systems*, 15.

Sutton, R. S. and Barto, A. G. (1998), *Reinforcement* ambridge, Massachusetts, London, England

Tsihmoni, O., and Liu, Y. (2003). Modeling steering using the Queueing Network-Model Human Processor (QN-MHP). *In Proceedings of the 2003 Annual Conf. of the Human Factors and Ergonomics Society*.

Wright, R. D., & Ward, L. M. (1998) The control of visual attention. Wright, R. D. (ed.), *Visual Attention*, Oxford; Oxford University Press.

Yarbus, A. L. (1967). Eye movements during perception of complex objects, in L. A. Riggs, ed., *Eye Movements and Vision*, Plenum Press, New York, chapter VII, pp. 171-196.