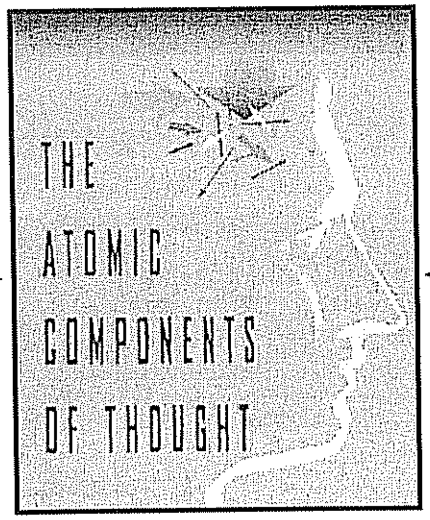


Twelfth Annual
ACT-R Workshop

Proceedings

July 15-17, 2005

Starhotel Savoia Excelsior
Trieste (Italy)

12th ACT-R Annual Workshop
Trieste, 15-17 July 2005

PROGRAM

Friday 15th

8:30 – 9:15	**Registration**
9:15 – 9:30	**Greetings and workshop opening**
	Talk session 1
9:30 – 9:50	Best, B. J. & Gunzelmann, G. Hierarchically-based perceptual grouping in ACT-R
9:50 – 10:10	Reifers, A., Schenck, I., Ritter, F. E., & Ignoscio, L. Using ACT-R to progress theories of pre-attentive visual search
10:10 – 10:30	Winkelholz, C. & Schlick, C A production system for the serial recall of object-locations in graphical layout structures
10:30 – 10:50	Rutledge-Taylor, M. F. & West, R. L. ACT-R versus neural networks in Rock=2 Paper, Rock, Scissors
10:50 – 11:15	**Coffee break**
	Talk session 2
11:15 – 11:35	Budiu, R. & Pirolli, P. Navigation in degree-of-interest trees
11:35 – 11:55	Gunzelmann, G. Spatial orientation in ACT-R: Architectural insights and extensions
11:55 – 12:15	Taatgen, N., van Rijn., H. & Anderson, J. R. Time interval estimation: Internal clock or attentional mechanism?
12:15 – 12:35	Mol, L., Taatgen, N., & Anderson, J. R. Individual differences in multitasking
12:35 – 12:55	Dzaack, J., Pape, N., Leuchter, S, & Urbas, L. How to integrate time-duration estimation in ACT-R/PM
12:55 – 14:30	**Lunch**
	Talk session 3
14:30 – 14:50	Brunstein, A. & Larrañaga, M. P. Is it a boy or a girl?
14:50 – 15:10	Douglass, S. A. Exploring the functional role semantics of instructions using a spatial module in ACT-R 6
15:10 – 15:30	Huss, D., Taatgen, N., & Anderson, J. R. Learning from instructions
15:30 – 15:50	van Rijn, H. Alphabetic retrieval & memory

15:50 – 16:10	Trafton, G., Altmann, E., & Brock, D. The long term disruption effect: A comparison of three memory models
16:10 – 16:30	**Tea break**
	Special session
16:30 – 17:30	Chater, N. General principles of cognition?
17:30 – 18:00	Anderson, J. R. Comments on "General principles of cognition?"
18:00 – 18:15	Open discussion
19:00 – 23:30	**Social event**

Saturday 16th

8:30 – 9:00	**Registration**
	Talk session 4
9:00 – 9:20	Anderson, J. R. Learning algebra in ACT-R
9:20 – 9:40	van Maanen, L. & van Rijn, H. RACE for retrieval: Competitive effects in memory retrieval
9:40 – 10:10	Elio, R. Modeling how delayed intentions impact current intentions in a prospective memory paradigm
10:10 – 10:30	Pavlik, P. I. An ACT-R based investigation of test and study temporal dynamics
10:30 – 10:50	Belavkin, R. V. Modelling the paradoxes of decision-making
10:50 – 11:15	**Coffee break**
	Talk session 5
11:15 – 11:35	Gamard, S., Schoelles, M.J., Kofila, C., Veksler, V. D. & Gray, W. D. CogWorks visualisation architecture: Cognitively engineering next generation workstations for decision makers.
11:35 – 11:55	Gray, W. D. & Schoelles, M. J. Profile before optimizing: A cognitive metrics approach to workload analysis
11:55 – 12:15	Gluck, K. & Gunzelmann, G. Informative failures on the path to a theory of degraded cognition
12:15 – 12:35	Ritter, F. E. & Reifers, A. The effects of pre-task appraisal and caffeine on cognition: Data and models
12:35 – 12:55	Dzaack, J., Kiefer, J., & Urbas, L. An approach towards multitasking in ACT-R/PM
12:55 – 14:30	**Lunch**

	Talk session 6
14.30 – 14:50	West, R. L., Emond, B., & Tacoma, J. Simple Object System (SOS) for creating ACT-R environments: A usability test, a test of the perceptual system, and an ACT-R 6 version
14:50 – 15:10	Royer, C., Farahat, A., & Pirolli, P. GLSA Server @PARC
15:10 – 15:30	Stewart, T. C. & West, R. L. Python ACT-R: A new implementation and a new syntax
15:30 – 15:50	Guhe, M., Gray, W. D., & Schoelles, M. J. New approaches for detecting workload and stress
15:50 – 16:10	Sims, C. & Gray, W. Interactive behavior at the sticking point: The curious persistence of apparently inefficient interactive routines
16:10 – 16:30	**Tea break**
	Free time: Exploring Trieste
20:00 – ??:??	**Dinner at the Löwenbräu Brewery (on your own)**

Sunday 17th

	Talk session 7
9:20 – 9:40	Qin, Y. & Anderson, J. R. ACT-R in the brain
9:40 – 10:10	Stocco, A. & Fum, D. From emotion to memory: An ACT-R view on the somatic marker hypothesis
10:10 – 10:30	Crescentini, C. & Stocco, A. Executive control in sentence comprehension: An ACT-R model of agrammatic aphasia
10:30 – 10:50	Kao, Y. Neural correlates of "expert" geometry problem solving
10:50 – 11:15	**Coffee break**
	Special Session
11:15 – 12:15	Bothell, D ACT-R 6: Official release
12:15 – 12:45	Open discussion: The future of ACT-R
12:45 – 12:55	**Workshop closure**

Hierarchically-Based Perceptual Grouping in ACT-R

Bradley J. Best (bbest@maad.com)
4949 Pearl East Circle, Suite 300
Boulder, CO 80301 USA

Glenn Gunzelmann (glenn.gunzelmann@mesa.afmc.af.mil)
Air Force Research Laboratory
6030 South Kent Street
Mesa, AZ 85212 USA

Introduction

When people encode visual information from the environment, they automatically organize the information to create a coherent image. This involves determining which areas go together to make objects, as well as determining what sets of objects go together to form groups. In ACT-R, the first of these processes is controlled by the creation of visual features to represent objects. The second process, however, has been essentially absent from the architecture.

In complex, irregular environments, such processes are not possible. We have developed applications of ACT-R to the Traveling Salesman Problem (TSP), as well as a 3D orientation task (Figures 1 and 2). In the TSP, the goal of the task is to produce the shortest possible route starting and ending with the same point that visits each point only once. In the 3D orientation task, the goal of the task is to identify ones location along the perimeter of the top-down view given a first person, egocentric view. There is evidence that people perform these tasks by organizing the objects into groups (Gunzelmann & Anderson, 2004; in press; Best, 2005).

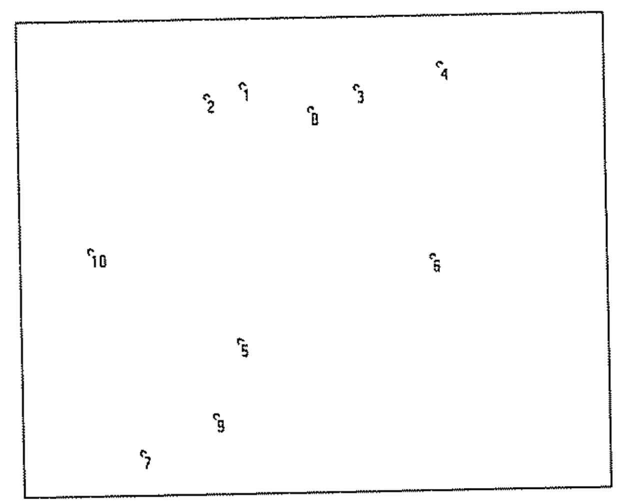


Figure 1: A Traveling Salesperson Problem Prior to Solution.

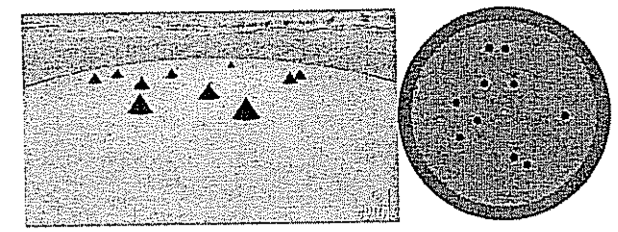


Figure 2: Sample orientation trial. Where is the viewer?

To faithfully model human performance in these tasks, we require the capability in ACT-R to recognize groups of objects in the display. To achieve this, we have implemented a hierarchical grouping algorithm that provides functionality and flexibility in this process. Many candidate algorithms for human grouping behavior exist. Compton and Logan (1993) developed a model that grouped points based on an exponential function of the distance between points. Graham, Joshi, and Pizlo (2000) developed a grouping model based on a pyramidal clustering algorithm while Pizlo (2005) described an evolution of this model that depended on a grouping algorithm based on the Minimax algorithm. Each of these algorithms require a threshold to be set for determining the groups that arise from the grouping process. However, human grouping behavior is likely to be much more interactive than these algorithms allow. We have chosen to implement an algorithm based on a hierarchical clustering method, specifically to produce groups consistent with existing approaches, but with the flexibility for central cognition to interact with the grouping process rather than through an all-or-none scheme.

Figures 3 illustrates the representation used by a hierarchical clustering method. This method produces a tree structure which allows groups to be determined at various levels of aggregation. A horizontal cut across the tree will produce a set of groups where the level of grouping is consistent for the all of the groups (i.e., it imposes a default spatial frequency grouping criteria). However, it is also possible for an interactive process to choose different levels of aggregation across different sections of the tree (e.g., to select groups of a certain size). Thus, this representation is capable of producing groups consistent with other methods, but also of supporting the interaction of cognition with the grouping process.

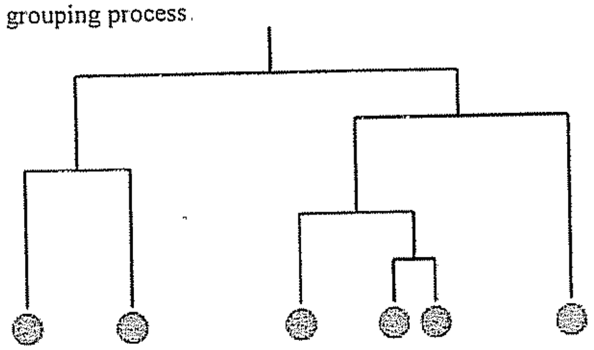


Figure 3: Hierarchical Clustering (in two dimensions).

Groups in ACT-R

To allow ACT-R to "see" the groups identified by the algorithm, they are instantiated in ACT-R as items in the icon, which can be attended and encoded. The location of the group is identified as the center of mass of the individual objects that comprise it. In the visual icon, the value slot is used to hold a group identifier. We use the same convention for the objects in the group to facilitate group-oriented visual search.

When a group is attended, a visual-object of type group is created. This chunk includes the group ID, and also has a slot for the number of objects in the group. There are sure to be limits to the ability to perceive this directly (i.e., subitizing), but so far we are dealing with relatively small groups (<5 objects), making this a reasonable assumption.

The appropriate level of the hierarchy on which to base group definitions is an open research question. We have chosen, as a first effort in this direction, to create an initial grouping by selecting the groups corresponding to a spatial frequency threshold determined dynamically from the density of points in the display. Spatial frequency is represented here as a parameter ranging from 0 to 1, where this frequency parameter represents the quotient of the dispersion (i.e., summed distances of these points from the mean of their group) of points within the group to the dispersion of points of the overall display. Thus, groups consisting of one point, which have no within-group dispersion, produce a spatial frequency of 0, while the group containing all of the points in the display produces a spatial frequency of 1. This relationship is given by the following formula:

Scaled spatial frequency =
within group dispersion / overall display dispersion

The result of this calculation can be used as a threshold for determining a consistent level of grouping across displays with different densities. This automatic adjustment of grouping to the density of display objects addresses the problem with alternative schemes which might produce very different size groups in displays with different densities (at the extreme producing one group for the whole display if points are tightly clustered relative to the chosen threshold).

There are a number of issues still to be resolved for this mechanism, including the selection of an appropriate level of the spatial frequency measure on which to base group definitions, and an empirical demonstration of the utility of grouping that automatically scales with density. There is also a need to resolve other issues, like whether objects of different types can be included in the same group and how to handle dynamic displays. These issues will be resolved as research issues demand and resources allow. However, the main contribution here, a grouping process that gives modelers a mechanism to leverage in their research, is already providing leverage in supporting modeling in spatial domains where grouping behavior is an essential aspect of human performance.

References

Best, B. J. (2004). Modeling Human Performance on the Traveling Salesperson Problem: Empirical Studies and Computational Simulations. Doctoral dissertation, Department of Psychology, Carnegie Mellon University, Pittsburgh, PA.

Best, B. J., and Simon, H. A. (2000). Simulating Performance on the Traveling Salesman Problem. Proceedings of the 2000 International Conference on Cognitive Modeling, pp. 42-49.

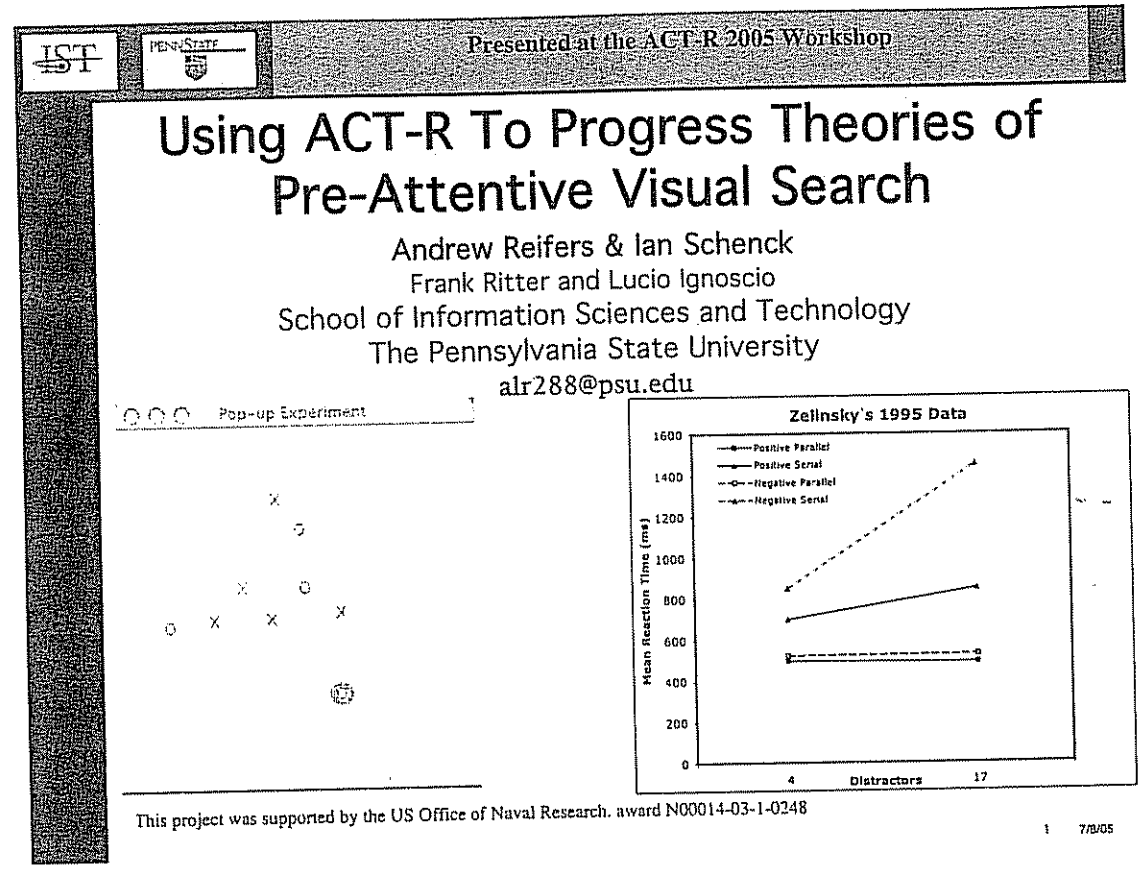
Compton, B. J., and Logan, G. D. (1993). Evaluating a computational model of pereptual grouping by proximity. Perception & Psychophysics, 53(4), pp. 403–421.

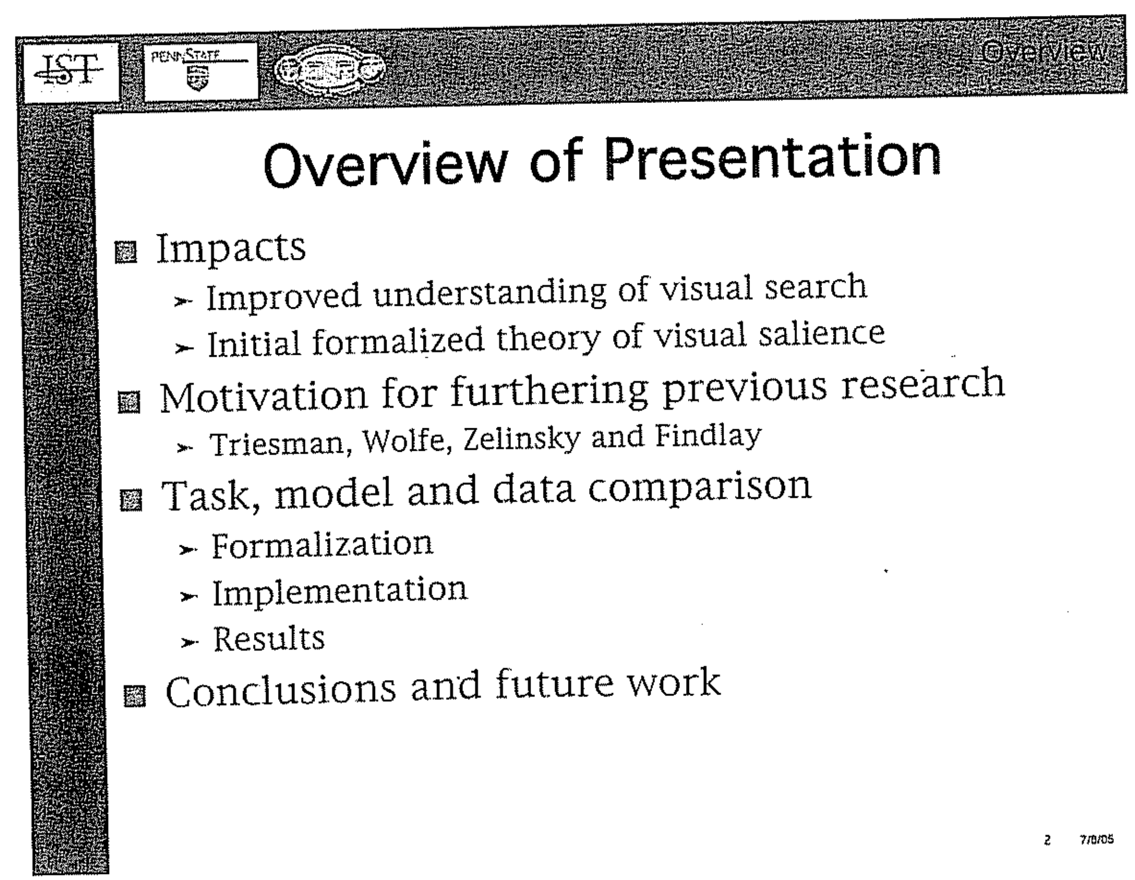
Graham, S.M., Joshi, A., and Pizlo, Z. (2000) The Traveling Salesman Problem: a hierarchical model. Memory & Cognition 28, pp. 1191-1204.

Gunzelmann, G., & Anderson, J. R. (in press). Location Matters: Why Target Location Impacts Performance in Orientation Tasks. *Memory & Cognition*.

Gunzelmann, G., & Anderson, J. R. (2004). Spatial orientation using map displays: A model of the influence of target location. In K. Forbus, D. Gentner, and T. Regier (Eds.), *Proceedings of the Twenty-Sixth Annual Conference of the Cognitive Science Society* (pp. 517-522). Mahwah, NJ: Lawrence Erlbaum Associates.

Pizlo, Z (2005). Towards a new computational model of human thinking and problem solving. Presented at the 2005 Air Force Office of Special Research (AFOSR) Cognition Program Review.





Motivation for Studying Visual Search and Pre-Attentive Distinctiveness

- Important for understanding aspects of human-computer, human-object interactions
- Large impact on safety critical scenarios
  - Air traffic control
  - Automotive and aircraft control
- Increasing dependence on graphical interfaces in e-commerce, as well as everyday life.
- Cognitive theories of visual search and salience still remain slightly stochastic and could benefit from developed formalizations.

Previous Approaches to Visual Search and Pre-Attentive Distinctiveness

- Cognitive psychological studies
  - Examples: Triesman, Wolfe, and Zelinsky
- Physiological studies
  - Examples: Mishkin, Livingston, Findlay and Gilchrist
- Human Factors
  - Examples: Wickens
- Cognitive Science
  - Examples: Kieras, Byrne, Hornof and Salvucci

The Task

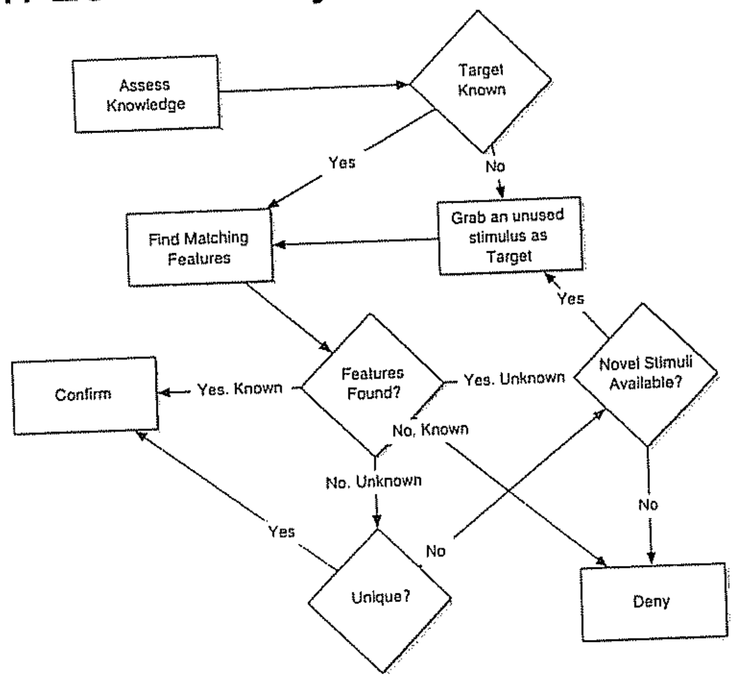
- Use basic visual search task paradigms
  - Top Down or Bottom Up
  - Conjunctive or Non-Conjunctive
  - Confirmation or Denial
- Create a model and formalization that can account for known trends (Zelinsky; Triesman; and Wolfe)

Our Approach ACT-R+PO

- Extend the base of ACT-R's currently instantiated perceptual motor buffers
- The following formula offers a very simplistic theory of pre-attentive visual salience

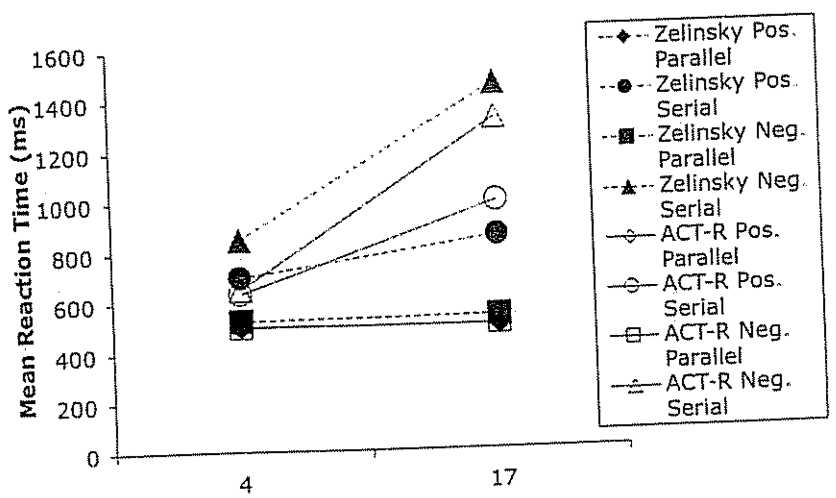
$$\mathrm{Pop(Stim)} = 1 - \left(\frac{f(\mathrm{Stim}_{\mathrm{color}}) \times f(\mathrm{Stim}_{\mathrm{feature}})}{f(\mathrm{Total\text{-}Stim})^2} \right)$$

A High Level Analysis of Our Model +PO



Testing the Formalization

- Initial results are encouraging.



Lessons from Initial Implementation of Pre-Attentive Distinctiveness

- Need inclusion of more basic features (motion, size, etc.)
- Need alternative formalizations
- Other models or theories of pre-attentive search will need the same testing paradigm with more complex visual field
  - Top Down or Bottom Up
  - Conjunctive or Non-Conjunctive
  - Confirmation or Denial

Summary of Results

- Our model and formalized theory succeeds in:
  - Simulating the pop out effect in visual search
  - Simulating differences between bottom up versus top down conjunctive search
  - Simulating differences between bottom up versus top down parallel search
  - Simulating differences between confirmatory versus denial searches
- Our theory suggests a graded "pop-out" effect. This is in line with Wolfe's theory of efficient searches.

Future Work

- Eye tracking studies
  - Use richer data sets to develop more complete theories
- Continue to develop a more complete formalization of pre-attentive visual search
  - Extending our formula to include more basic features (e.g. size, linear angle)
  - More data comparisons
  - Test and compare alternative formalizations

References

Anderson, J. R., Matessa, M., & Douglass, S. (1995). *The ACT-R theory and visual attention*. Paper presented at the Proceedings of the Seventeenth Annual Conference of the Cognitive Science Society, Hillsdale, NJ.

Byrne, M. D. (2001). ACT-R/PM and menu selection: Applying a cognitive architecture to HCI. *International Journal of Human-Computer Studies, 55*(1), 41-84.

Hornof, A. J., & Kieras, D. E. (1997). Cognitive modeling reveals menu search is both random and systematic. In *Proceedings of the CHI'97 Conference on Human Factors in Computer Systems* 107-114. New York, NY: ACM.

Livingstone, M., & Hubel, D. (1988). Segregation of form, color, movement and depth: Anatomy, physiology, and perception. *Science, 240*(4853), 740-749.

Salvucci, D. D. (2001). An integrated model of eye movements and visual encoding. *Cognitive Systems Research, 1*(4), 201-220.

Treisman, A., & Sato, S. (1990). Conjunction search revisited. *Journal of Experimental Psychology: Human Perception & Performance, 16*, 459-478.

Wolfe, J. M. (1998). Visual Search. In H. Pashler (Ed.), *Attention* (pp. 13-73). San Diego: Psychology Press.

Zelinsky, G., & Sheinberg, D. (1995). Why some search tasks take longer then others: Using eye movements to redefine reaction times. *Eye Movement Research*, 325 - 336.

A production system for the serial recall of object-locations in graphical layout structures

Carsten Winkelholz (winkelholz@fgan.de)
Research Establishment for Applied Science (FGAN),
Research Institute for Communication, Information Processing and Ergonomics
Neuenahrer Strasse 20, 53343 Wachtberg, GERMANY

Christopher Schlick (c.schlick@iaw.rwth-aachen.de)
Institute of Industrial Engineering and Ergonomics, RWTH Aachen University of Technology
Bergdriesch 27 52062 Aachen, GERMANY

Abstract

This paper presents a production system within the ACT-R theory of cognition for the serial recall of object-locations in a graphical layout structure. Concepts of noise and the encoding of object-locations in local allocentric reference systems have been integrated into the visual module for this purpose. The intrinsic reference axis of the local reference systems automatically result from the previously attended objects. The production system describes the process of encoding and rehearsal of object-locations at the stage of the presentation as well at the answer-stage. The model encodes environmental features of the object-locations by object-to-object spatial relations. The production system reproduces the main effects in an experiment which was carried out with 30 subjects.

Introduction

Ehret (2002) and Anderson et al. (2004) describe production systems that reproduce learning curves for the location of information on a display. In these examples the underlying mechanism for learning locations is the same as for the learning of facts. After some practice the location of specific objects like menu buttons can be retrieved without a time consuming random visual search and encoding of labels. In ACT-R the location of a visual object is represented in absolute screen coordinates. Furthermore there is no noise integrated into the visual module. Therefore the location of an object is learned independent of its position on the screen and its position within an object-configuration. But there is evidence that the kind of how objects are displayed has implications on object-location memory. One experiment of Travanti & Lind (2001) investigated object location memory in hierarchical information structures across different instances of 2D and 3D perspective displays. The results of their tests show, that the 3D display improves performance in the spatial memory task they designed. But beside the perspective view also the structure of the object-configuration was different in the 2D and the 3D display. Cockburn (2004) repeated the experiments where he displayed the object-configuration of the 3D display in 2D. He found, that if displayed in 2D the 3D object-configuration improved performance on object-location memory. In both studies the memory task was to associate alphanumerical letters to the object-locations. Therefore Cockburn suspected that the vertical orientation of Travanti & Lind's 2D display made the formation of effective letter mnemonics more difficult than the horizontal 3D layout, because words and word combinations normally run horizontally left to right. By analyzing these studies we came to the conclusion that one major factor had not been considered - the factor of the object-to-object spatial relations (the structure of the graphical layout respectively). Therefore we performed own experiments in which the structure of the object-configuration were varied. Furthermore, to avoid subjects to create letter mnemonics in our experiments the task was to memorize sequences of highlighted objects (Winkelholz et al. 2004). The object-configurations investigated are shown in figure 4a. In each encoding retrieval trial, the subject was presented one structure. After an acoustical signal the computer started to highlight objects of one randomly created sequence. Only one object of the sequence was highlighted at once. The sequences were five (A structures) and six (B and C structures) items long. The end of a sequence was indicated by a second acoustical signal. Each object of a sequence was highlighted for 2 seconds. Subjects were instructed to repeat the highlighted objects in correct order, by clicking them with the mouse. As a measure of performance the number of correct repeated sequences was chosen. The displayed dependencies of the overall performance on the object-configurations (figure 4b) show two things. First, that a horizontal orientation of a structure improves the performance in the memorizing task compared to a vertical orientation (A_1 compared to A_3). Second, performance increases the more distinct object-to-object relations are within a structure. E.g. in the matrix structures B_1 and B_2 the object-to-object relations covers the whole plane, whereas in the linear structure B_3 object-to-object relations are only in one dimension. Since there is no difference in the performance between structure B_1 and B_2 this effect can not result from spatial vicinity. As well suggests the effect in the performance between C_1 and C_2 that noisy object-to-object relations are needed to model this effect. While object-locations are represented in absolute screen-coordinates this effect can not be modeled on the level of production rules within ACT-R and some extensions to the visual module are needed. One promising approach in this direction was suggested by Wang et al. (2002) and Johnson et al. (2002) who extended ACT-R to automatically encode object-to-object relations between the previously and currently attended objects. Based on this approach we extended the visual module not only to encode the spatial relation of previously and currently attended object, but also to use the two previously attended objects to form a local reference axis according to which the location of the current attended object is encoded. Furthermore, we integrated a noise model into the visual module, extended the mechanism of visual indexing and integrated some kind of competitive chunking mechanism in the equation for the activation.

Visual Module Extensions/Restrictions

Reference systems

The location of an object can only be identified within a frame of reference. In experimental psychology it is well accepted to divide the frames of references into two categories: An egocentric reference system, which specifies the location of an object with respect to the observer and an environmental (allocentric) reference system, which specifies the location of an object with respect to elements and features of the environment. As mentioned above the visual module of ACT-R encodes object-locations in the reference-system of the screen, which is equivalent creating all spatial object relations to one edge of the screen. However, according to Mou & McNamara (2002) humans also use reference systems concerning the intrinsic axis of the object configuration. E.g. two salient objects create an axis that is used to specify the location of other objects. The most natural way to integrate this into the concept of attention of the visual module is to consider the last two attended objects as an axis of reference. This is an extension to the proposal of Johnson et al. (2003) considering only the previously attended object in creating object-to-object relations, which means that only the distance is represented in a pure environmental reference system and the angles in an egocentric reference system. However, creating object-location memory chunks in this "semi-allocentric" reference system is less effort to the visual module because it only needs to keep track of two objects, whereas in the case of the pure allocentric reference system three objects are needed. Therefore in some situations the production system might be forced to use spatial memory chunks in the semi-allocentric system. We considered in the visual module all three different reference systems, which are summarized in Figure 1.

The introduction of object-relations based on three objects is important for three reasons: First, it fits well with the concept of intrinsic axis in the object configuration as reported by Mou & McNamara (2002). Second the concept of angles is essential to most cognitive operations in geometric tasks. Third, it is the simplest percept for spatial memory chunks that allows reconstructing object locations, also if the whole configuration is rotated.

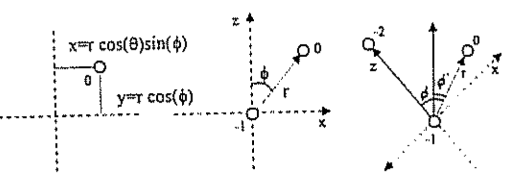


Figure 1: Three different reference systems. The objects are attended in the order (p_{-2}, p_{-1}, p_0)

Noise

The variances in recalled object-locations require the memory chunks to be noisy. To integrate noise into the memory chunks the first question is how object-locations in different reference systems are represented in memory. Huttenlocher et al. (1991) showed among other things, that the distribution of recalled locations supports the assumption that subjects imagine object-locations on a plane relative to a center in polar coordinates. We generalized this to use spherical coordinates in respect to an extension of the visual module in three dimensions. This assumption has also some interesting implications on the representation of locations on a screen. Spherical coordinates are a system of curvilinear coordinates that are natural for describing positions on a sphere or spheroid. Generally θ is defined to be the azimuthal angle in the xy-plane from the x-axis, ϕ to be the polar angle from the z-axis and r to be distance (radius) from a point to the origin. In the case of the allocentric reference system this means, that if the three points p_{-2}, p_{-1}, p_0 were attended and p_0 has to be represented in a local allocentric reference system, the point p_{-1} defines the origin, the polar axis is given by (p_{-1}, p_{-2}), and the local spherical y-axis points orthogonal into the screen. For the semi-allocentric reference system, again p_{-1} is the origin, but the polar axis is parallel to the vertical axis of the screen and the x-axis is parallel to its horizontal axis. In the case of the egocentric reference system the viewpoint of the subject is the origin. In the typical scenario of a user interacting with symbols on the screen the differences in the angles and distances between symbols represented in the egocentric system are very small compared to the differences if represented in an allocentric, or semi-allocentric reference system. Therefore, if the same magnitude of noise is assumed in all reference systems, memory chunks represented in the egocentric reference system would be extremely more inaccurate compared to object-locations represented in the other two reference systems and therefore can nearly be neglected. The next question is, if θ, ϕ, and r should be considered as single, independent memory chunks. Because it is impossible to imagine a distance without a direction and an angle without corresponding lines, it is reasonable to combine distance and angular as one percept in one memory chunk. Because of this argument, also in the case of the actual allocentric reference system the egocentric orientation of the reference system should be stored into the memory chunk. This does not imply that the angular or the different dimensions of one chunk can not be separated later. In spatial reasoning often two angles have to be compared. But this can be handled as commands to the visual module. Then also timing issues can be considered for example for the mental rotation of an actual allocentric reference system. In principle the spatial information of the semi-allocentric reference system is now also present in the chunk of an actual allocentric reference system. This might suggest discarding memory chunks of the semi-allocentric reference system. But as mentioned above, creating object-location memory chunks in this semi-allocentric reference system is less effort to the visual module and therefore in some situations needful. Finally a spatial location is represented by $D(r,\theta,\phi,\phi',e_{rs})$, where r,θ,ϕ are the spherical coordinates as described above, e_{rs} indicates in which reference system r,θ,ϕ have to be interpreted, and ϕ' is an additional attribute for the actual allocentric reference system and holds additionally the polar angle in the semi-allocentric reference system. The values of the spherical coordinates in the memory chunk are interpreted as random numbers distributed according to a truncated logistic distribution $f(x,x_0,\sigma_x)$, with to each dimension corresponding standard deviations $(\sigma_r(\phi',r),\sigma_\theta,\sigma_\phi)$. The scalar value in the slot of the memory chunk indicates the maximum x_0 of the distribution. The noise in the r-dimension is biased by a factor according to if the distance to be estimated is vertically or horizontally oriented. Furthermore, the noise σ_r is relative to r. As the final noise in the r-dimension we use:

$$\sigma_r(\phi',r) = (f_{\sigma_-} + (1-f_{\sigma_r})\cos^2(\phi'))\sigma_r r \qquad (1)$$

Every time a location is to be encoded, it is decided if the perceived values for the location correspond to an already existing memory chunk. The posterior probability $P_{Di}=P(D_i|F_x)$ that the location of a feature F_x belongs to a memory chunk M_i and the probability P_0 that no appropriate memory chunk already exists, are given by

$$P_{D_i} = \frac{P(F_x \mid D_i)}{V^{-1} + \sum_i P(F_x \mid D_i)}, \quad P_0 = \frac{V^{-1}}{V^{-1} + \sum_i P(F_x \mid D_i)} \qquad (2)$$

The parameter V^1 describes the weight of a noisy background and

$$P(F_i(r,\theta,\phi,\phi') \mid D(r_x,\theta_x,\phi_x,\phi'_x)) = \qquad (3)$$

$$f(r,r_x,\sigma_r(\phi',r_x)) f(\theta,\theta_x,\sigma_\theta) f(\phi,\phi_x,\sigma_\phi) f(\phi',\phi'_x,\sigma_\phi)$$

On the other hand, if an object-location is requested based on a memory chunk $D(r,\theta,\phi,\phi',e_{rs})$, the values are set to random values according to (3). After the noise has been added to the location request, it is decided if the values are latched on possible features in the display. Therefore, the object-locations of all features $F_i(r_i, \theta_i, \phi_i, \phi'_i)$ in question are calculated in the current local reference system corresponding to the reference system in the request. The probability P_{Fi}, that the location request is caught by feature F_i and the probability P_0 that it is not, are given similarly to (2) by

$$P_{F_i} = \frac{P(x \mid F_i)}{V^{-1} + \sum_i P(x \mid F_i)}, \quad P_0 = \frac{V^{-1}}{V^{-1} + \sum_i P(x \mid F_i)} \qquad (4)$$

These equations express the posterior probability $P_{Fi}=P(F_i|x)$ that if a noisy location x from the memory is given the location results from the feature F_i. The likelihood probability functions $P(x|F_i)$ are the truncated logistic distribution according to if the feature F_i would have been the stimulus and are similar to (3). The process of encoding and reconstruction of a location into a random number in memory is illustrated in figure 2.

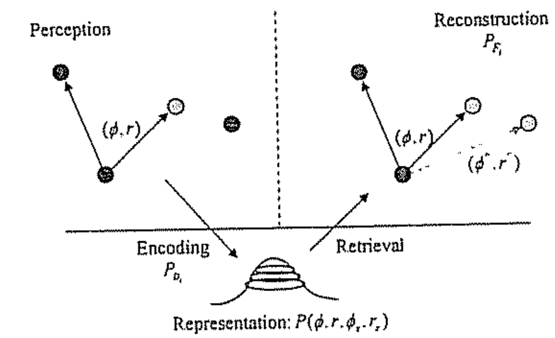


Figure 2: Perception, representation and reconstruction of a location

This noise model has two interesting properties. First, because the truncated logistic distribution is asymmetric, the expected report of an object-location is biased away from the reference axis. This is the same effect as reported at categorical boundaries. Second, for object-locations on a flat screen the values of θ are discrete $\theta=(\pi/2,0,-\pi/2)$ and encode whether the object-location in question is on the left side, on the right side, or aligned, when facing into the direction of the reference axis. This is consistence with the assumption to interpret the reference axis as a categorical boundary, where θ encodes the category.

Visual Indexing

It is evident that subjects browsing a graphical layout structure encode environmental characteristics of object-locations, e.g. if an object is located on the border of a matrix. To encode such environmental features the cognitive system needs to attend objects nearby. The crucial point is that after some objects in the environment have been attended, attention needs to return to the object in question. If this return would depend on noisy spatial memory chunks, the strategy to encode environmental features might be highly counterproductive. At this point the concept of visual indexing, or FINST - FINger INSTantiation, (Pylyshyn, 1989) is needed. According to this theory the cognitive system has "access to places in the visual field at which some visual features are located, without assuming an explicit encoding of the location within some coordinate system, nor an encoding of the feature type". Experiments suggest that the number of FINSTs in the visual system is limited to the number 4 to 5. In the visual module of ACT-R the concept of FINST is used to decide if an object has already been attended. Whenever an object is attended, a FINST is created. Because the number of simultaneously existing FINST is limited, any time a new visual object is attended the oldest FINST is removed to create a new FINST for the currently attended object. To implement environmental scan patterns, FINST need to provide additionally to the information that an object has already been attended also information for accessing its location without, or at least minimal noise. In the visual module interface described in the next section this has been accomplished by determining a visual index through the sequential position in the chain of attended locations. This index can be used in visual module commands to return (or avoid to return) attention to a particular location in the chain of attended locations.

The visual module interface

Figure 3 shows the visual module interface, with the slots that have been added, and the slots whose meaning have been extended.

Perception:	Action:
=visual-location> vsl-e symbol ;egocentric vsl-a symbol ;allocentric vsl-sa symbol ;semi-allocentric kind [text,. .empty] index1 [nil. t] index2 [nil, t] index3 [nil. t] index4 [nil. t] index5 [nil. t]	+visual-location> vsl-r symbol vsl-phi symbol vsl-mphi symbol vsl-theta symbol vsl-mtheta symbol vsl-ix [back1,...back5] attended [not1. .not5.noti1. noti5]

Figure 3: Modified visual module interface

For each reference system one slot (*vsl-e*, *vsl-a*, *vsl-sa*) has been added containing a symbolic value of a memory chunk encoding the location in the respective reference system (egocentric, allocentric, semi-allocentric). These symbolic values can be used to request new locations in the visual field. For this purpose the command-slots *vsl-r*, *vsl-phi*, *vsl-mphi*, *vsl-theta*, and *vsl-mtheta* have been added. For each dimension (r,θ,ϕ) there is a slot extracting this dimension from the spatial memory chunk given to this slot. Thus, the dimensions of different spatial memory chunks can be combined to one request. The angular dimensions can be inverted through the slots *vsl-mphi* and *vsl-mtheta* $(\theta \rightarrow \theta > 0 ? \theta - \pi : \theta + \pi \text{ , } \phi \rightarrow \arccos(\cos(\phi + \pi)))$. This approach enables the visual module to compare the length of two distances or to scan an imagery path backwards. Only spatial memory chunks within the same reference system can be combined. Possible sub-symbolic parameters for timing and if any combination should be disabled, need to be investigated in future work. The request for a new location through these slots may prompt the visual module to attend an empty location. This case is indicated by the symbol EMPTY in the kind slot. First, we tried to implement the environmental scan patterns only by using these slots. But it turned out that as long as these requests are noisy operations, it was too risky to loose the actual object-location in question during an environmental scan. The possible gain of information for an object location was culled by this noise. Therefore we introduced the slots *index1,..,index5*, and *vsl-ix* to have precise access to indexed locations. The slots *indexN* indicate whether the currently attended location has already been attended at position *N* (counted backwards) in the chain of attended locations. By using the descriptive identifiers *backN* on the slot *vsl-ix* a particular location in the chain of already attended locations can be re-attended. The possible descriptive identifiers on the attended slot have been extended to *notN* and *notiN*. The *notN* identifier prevents the visual-module to attend a location that has already been attended within the last *N* attended locations. The *notiN* identifier prevents the visual module to attend a location that has been attended exact at position *N* in the chain of attended locations. Only with this access to indexed locations it is possible to "weave" a reliable network of object-to-object spatial relations.

Competitive Chunking

A subject learning object-locations in a graphical structure becomes familiar with the structure after some time. This means he recognizes environmental features faster and is therefore able to link environmental features more efficient to object-locations. The concept of familiarity within a symbolic architecture of cognition has already been discussed by Schreiber-Evert & Anderson (1990). They developed the theory of competitive chunking (CC), which assumes that memory chunks are supported by subchunks. For example subjects are able to learn sequences of letters more efficient, if the sequence contains well known words or syllables. This is because the memory chunk for the sequence can be compressed by replacing elements of the sequence by references to subchunks having a high activation and can therefore be retrieved reliably and fast from memory. The concept of CC as described by Schreiber & Anderson is not part of the curren: version of ACT-R. However, we suspect that such a concept is needed, to describe the effect of becoming familiar with a configuration of objects. One way to manage subchunks within ACT-R is to couple them tightly to their parent chunks by their symbolic values in specific slots. This method does not result in an effect considered as CC, because it doesn't allow accessing associated subchunks by free association. In many situations only one of possible several subchunks associated with the parent chunk needs to be retrieved, but by this approach the slots need to be retrieved consecutively. Therefore, a more promising approach is to couple chunks only by symbolic tags they share. This way e.g. an arbitrary number of environmental features can be associated with one object-location, and can be retrieved competitively. The problem is that subchunks that have been learned in context of different parent chunks carry the same information but differ in the tag shared with its parent chunks. In the sense of CC they should be supported because of their common patterns. To study this effect in the learning of environmental features we extended ACT-R's activation equation for memory chunks by the following term:

$$C_i = c_{cc} \sum_{m=1}^{n_s} \sum_{n=1}^{n_s} \sum_{k}^{N} I_{mni} K_{mnik} \left[B_k + \frac{\ln(1+e^{-c_d B_k})}{c_d} \right] \quad (5)$$

The index k runs through the chunks of the same kind, the index m and n through the slots of the chunk type. The parameter K_{mnik} compares the similarity of the slot values and can be expressed by the similarity parameters of the partial matching term:

$$K_{mnik} = \begin{cases} e^{M_{v_{ni}v_{nk}}} e^{M_{v_{mi}v_{mk}}} \text{, if } e^{M_{v_{ni}v_{nk}}} e^{M_{v_{mi}v_{mk}}} > c_\tau \\ 0 \text{ ,otherwise} \end{cases} \quad (6)$$

The partial matching parameter $M_{m_i m_k}$ we interpret as the log probability $ln(P(v_{mi}=v_{mk}))$ that the value in slot m of chunk D_i results from the same source as the value of slot m of chunk D_k. This is in accordance with the default choice of $M_{ij}=0$ if the slot values are equal.. Hence K_{mnik} is the probability that both values are equal. To limit the contributions, K_{mnik} is cut by a threshold c_τ. So roughly speaking the sum of the K_{mnik} over the slot pairs is a measure of how many equal slot values chunk i and k share. If only K_{mnik} is used as a factor for the competitive chunking, also slots contribute, which values are equal over all chunks, which means that they do not carry any information. Therefore we introduced the factor I_{mni}, that estimates how much normalized information the knowledge of the value $V_m=v_{mi}$ in slot m of memory chunk D_i contains about the values V_n in slot n of the other chunks.

$$I_{mni} = 1 - \frac{H(V_n \mid V_m = v_{mi})}{H(V_n)} \quad (7)$$

I_{mni} is zero if v_{mi} contains no information about V_n and 1 if V_m is fully determined by the knowledge of v_{mi}. If the slots only contained clearly distinguishable symbolic values, the entropies in (7) could be calculated by the frequencies. But in the case of spatial memory chunks the similarities have to be taken into account.

$$H(V_n) = -\frac{1}{N} \sum_{k}^{N} \ln \frac{\sum_{k'}^{N} e^{M_{v_{nk}v_{nk'}}}}{N} \quad (8)$$

$$H(V_n \mid V_{mi} = v_{mi}) = -\frac{1}{\sum_{k'}^{N} e^{M_{v_{mi}v_{mk'}}}} \sum_{k}^{N} e^{M_{v_{mi}v_{mk}}} \ln \frac{\sum_{k'}^{N} e^{M_{v_{nk}v_{nk'}}} e^{M_{v_{mi}v_{mk'}}}}{\sum_{k'}^{N} e^{M_{v_{mi}v_{mk'}}}} \quad (9)$$

In the limit of clearly distinguishable slot values the equations (8) and (9) are identical to a formula estimating the probabilities of the entropies for the information by the frequencies of the slot values. Further, the contribution of each chunk is weighted by a factor according to its basis activation B_k with a lower bound to zero for and approximating B_k for large activations.

Due to the additional term (5) in the activation equation virtual subchunks emerge through the clustering of attribute values, which support their container chunks.

Simulation

We used the extended visual module to model human performance in a task for the serial recall of object locations in graphical layout structures briefly reviewed in the introduction.

Production rules

The production system we developed describes the encoding and retrieval stage of the memorizing task. During the encoding of the sequentially presented object-locations the previously highlighted objects up to the current location are rehearsed. During the rehearsal, environmental features of the object locations are encoded or it is checked if one to the object-location retrieved environmental feature matches the environment of the current object. If the environment does not match, the reference system is restored through the visual indexes, and a new guess is made excluding the denied object-location. The environmental features are encoded in competing chunks with a symbolic tag to the corresponding object-location and spatial relations to objects in its neighborhood. To check an environmental feature is time consuming, because it has to be retrieved from memory. Therefore, the production rules for checking or encoding the environment compete. The answer stage is equal to the rehearsal stage, except that environmental features are not encoded anymore and are only checked. Overall the production system contains 142 rules. This unexpected high number of rules results from the time pressure set on the task. At any possible stage the model needs to check if a new card is highlighted, which leads to a lot of exceptions needed to be handled.

Most ACT-R parameters were left at their defaults, and subsymbolic computation was enabled. Further, retrieval threshold (:rt 0.0), latency factor (:lf 0.35) and maximum difference (:md -100). The variance $\sigma_{(\theta,\phi)}$ of the noise for the angular dimension was set to 0.06 radians and to 0.08 for the r-dimension. This is smaller than the standard deviation reported by Huttenlocher et al. (1990), but in their experiments no reference point was displayed, hence noise might be larger because of an uncertain reference location. The skewing factor f_r in eq. (1) of the noise in the r-dimension was chosen to be 0.8. The parameter for the background noise was set to V=2e3. The competitive chunking parameters were set to c_c=0.7, c_d=1.0 and c_τ=0.8. For all simulations and graphical structures the same parameters and production rules were used.

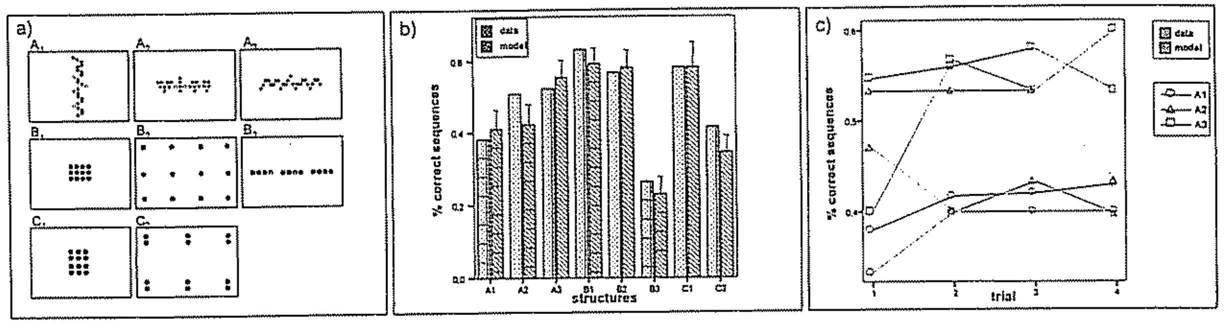


Figure 4: a) graphical layout structures used in the experiments b) overall performance, c) learning curves

Results

The results are shown in figure 4. The output of the simulation model adequately fits the data (R^2=0.83). However, the simulation exhibits no learning curve. The competitive chunking mechanism worked as intended. The traces of the model reveal that in the first trial only the environmental feature of the first object-location gets enough activation to be retrieved, at the last trial mostly the environmental features of the first four object-locations are retrieved. But to get some kind of saturation from existing chunks in the competitive chunking equation, we let the model first learn sequences in random object-configuration. After this saturation the other structures seems not to be distinct enough to change the effects in the competitive chunking equations. The learning curves in the experimental data are not significant, so they should not be over-interpreted. The model underestimates the performance of the subjects in the symmetrical tree structure A_2. This may indicate that the visual system takes advantage of symmetries in an object-configuration that are not captured by the model yet. This could be done by more sophisticated scan patterns or the saturation in the competitive chunking should have been done by training the model on more regular structures.

Conclusions and Future Work

This paper described extensions to the visual-module of the ACT-R/PM theory that allows developing very detailed models for the visual working memory. The concepts were derived from well known effects in experimental psychology. In conclusion the modeling gave us a deep insight into the mechanisms and bottlenecks of encoding object-locations. One challenge in modeling the memorizing task was the limited number of FINSTs. The number of FINST limits the complexity of environmental features that can be encoded. This is interesting with respect to visual working memory in three dimensions. In three dimensions encoding of an object-location in a real allocentric local reference system needs at least three object locations to define a reference plane. This reduces the number of free FINST in an encoding task. This might explain why spatial reasoning in three dimensions is for most people more difficult than spatial reasoning in two dimensions. In future work we will extend the concepts described in this paper to three dimensions. Furthermore, we currently investigate how the occurrence of noisy scalar values in attributes of memory chunks should be considered in the equation for learning of association strength and base level learning. Furthermore, spatial reasoning tasks might be modeled in future work.

References

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C., & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review 111*, (4). 1036-1060.

Ehret, B. D. (2002). Learning where to look: Location learning in graphical user interfaces. CHI Letters, 4(1), 211–218.

Cockburn A (2004). Revisiting 2D vs 3D Implications on Spatial Memory. *Proceedings of the Fifth Australasian User Interface Conference* (AUIC2004). Dunedin, New Zealand. January 2004, pages 25-32.

Huttenlocher J., Hedges L. V., Duncan S. (1991): Categories and Particulars: Prototype Effects in Estimating Spatial Location. Psychological Review, Vol. 98, No. 3, 352-376, 1991.

Johnoson, T.; Hongbin W.; Zhang J. (2003): Modeling the use of multiple frames of reference for object location memory. *Proceedings of 10th Annual ACT-R Workshop*

Mou, W., & McNamara, T. P. (2002). Intrinsic frames of reference in spatial memory. *Journal of Experimental Psychology. Learning, Memory, and Cognition*, 28, 162-170.

Pylyshyn, Z. W. (1989): The role of location indexes in spatial perception: A sketch of the FINST spatial-index model. *Cognition*, 1989, 32, 65-97.

Servan-Schreiber E., & Anderson J.R. (1990) Chunking as a mechanism of implicit learning. *Journal of Experimental Psychology: Learning, Memory, and Cognition*, 16, 592-608.

Travanti, M. & Lind M. (2001). 2D vs 3D, Implications on Spatial Memory. In proceedings of the IEEE Symposium on Information Visualization 2001

Winkelholz, C.; Schlick, S.; Brütting, M. (2004) The effect of structure on object-location memory. *Proceedings of the Twenty-Sixth Annual Conference of the Cognitive Science Society*. Mahweh, NJ: Lawrence Erlbaum Associates.

ACT-R versus Neural Networks in Rock=2 Paper, Rock, Scissors

Matthew F. Rutledge-Taylor (mrtaylo2@connect.carleton.ca)
Institute of Cognitive Science, Carleton University, 1125 Colonel By Drive
Ottawa, Ontario, K1S 5B6 Canada

Robert L. West (robert_west@carleton.ca)
Institute of Cognitive Science, Department of Psychology, Carleton University, 1125 Colonel By Drive
Ottawa, Ontario, K1S 5B6 Canada

Abstract

Recent research on cognitive modeling and game playing has focused on the game of Paper, Rock, Scissors (PRS). Models of PRS players have been created using both neural networks (West & Lebiere, 2001) and ACT-R (Lebiere & West, 1999; West, 1998). In all cases successful models of human play were created. This seems to be, in part, because of the simplicity of the game. In Rutledge-Taylor & West (2004) neural network models of players of a modified version of paper, rock, scissors were tested. The network model was able to fit the human data, but it was necessary to use a genetic algorithm to adjust the reward and punishment amount for the various game outcomes. The ACT-R model is much more constrained than a generic neural network in terms of how it can be adjusted. Since it uses the declarative memory system, learning is based on "harvesting" and "popping." In the present work the question of how to create an ACT-R model where rewards of various magnitudes need to be implemented is investigated. This is done by exploring some simple techniques such as the double retrieval and harvesting of chunks, and the manipulation of the default parameter for noise (ans). The results are compared to the human data from Rutledge-Taylor & West (2004) in which humans played the variant of PRS described above.

Keywords: ACT-R; cognitive architectures; paper, rock, scissors.

Introduction: The neural network models

Neural network models of human paper, rock, scissors game play have been described in Rutledge-Taylor & West (2004), and West & Lebiere (2001). The game of paper, rock, scissors was chosen for two reasons. It is a game familiar to most prospective experimental participants. Due to it's simplicity, an analysis of how it is played is tractable. In Rutledge-Taylor & West (2004), and West & Lebiere (2001) the same types of neural networks were used. The networks were perceptron-like in that they had no hidden layer. The output layer consisted in three nodes, one for each of the possible play options of paper, rock, and scissors. When presented with input, the play option associated with the output node with the greatest activation is chosen by the network. The input layer consisted in either one or two groups of three binary nodes, one for each of the possible play options. Each input group represented a move made by the model's opponent in the past history of a game underway. Models with three input nodes were called lag 1 models; they received as input the last move made by their opponent. Models with six input nodes were called lag 2 models; they received as input the last two moves made by their opponent. For each input group, the node corresponding to the move made would be have an activation of one, while the other two would have activations of zero. For example, if a lag 2 model's opponent had played paper last and scissors the time before that, the input pattern to the network would be ((1,0,0),(0,0,1)). The network weights were integer values, which in the case of West & Lebiere (2001) were initialized to 0, and in the case of Rutledge-Taylor & West (2004) were randomly initialized to –1, 0, or 1. For each model there is an associated three by three reward matrix which determines how the model's network weights should be adjusted after an iteration of play based on the outcome. Each cell in the matrix corresponds to the combination of an outcome from the model's perspective {win, tie, loss} and the move chosen by the model {paper, rock, scissors}. The value in the corresponding cell is added to the network weights that contributed to making the selection of the move played. The two most simple reward matrices were called aggressive and passive. In the passive reward matrix, cells associated with wins have values of 1, ties have values of 0, and losses have values of –1. The aggressive matrix is identical, with the exception that ties have values of –1. Thus, networks with passive reward matrices, hereafter, referred to simply as passive networks or passive models, treat ties as neutral events whereas the aggressive networks categorizes both ties and losses as non-win negative events.

West & Lebiere (2001) pitted human participants against several of the neural network models, and compared these results to model versus model games. They found that pairs of identical models, when pitted against one another, on average, tied; networks that processed two lags had a competitive advantage over networks that processed only one; and, aggressive networks had an advantage over passive networks. Interestingly, the extra lag and aggressive advantages were approximately equal in magnitude. Humans participants showed a performance profile similar to that of the aggressive lag 2 network model. On average, the human participants had won 9.99 more games than the aggressive lag 1 models after 300 games, and against the passive lag 2 models had won 11.14 more games after 287 games. However, against the aggressive lag 2 model, the network had, on average, a 8.89 win advantage after 20 minutes of play. The failure of the human participants to, at least, tie the aggressive lag 2 networks was attributed to a lack of motivation on the part of the human players; playing an opponent that is difficult to beat is less fun that playing one that can be taken advantage of.

Rutledge-Taylor & West (2004) investigated the performances of humans against computer neural network models in a slightly different version of paper, rock, scissors. In this version, called Rock=2, two points are awarded to the winner in the rock versus scissors combination of play, and one point is awarded to the winner in the paper versus rock, and scissors versus paper cases. Humans played against the aggressive lag 1 model, the aggressive lag 2 model, and new model called the rock=2 model. The rock=2 model was identical to the aggressive lag 1 model, with the exception that the rock=2 reward matrix cell corresponding to winning with rock had a value of 2. The standard aggressive models are not explicitly sensitive to the game condition that winning with rock is worth more that winning with either paper or scissors. The rock=2 model was designed to be, potentially, sensitive to this new aspect of the game.

The results of play were that, after 300 games, the human participants were, on average, able to earn 16.5 more points than the aggressive lag 1 models, 5.7 more points than the aggressive lag 2 models, and, 25.6 more points than the rock=2 model. Additionally, the expected points difference based on the ratios with which the players played each of the three possible moves was calculated. This expected points difference was subtracted from the actual points difference to produce a measure of the human players' abilities to orchestrate more wins than would be the case if the humans played randomly according to the ratio by which they made their choices of play. This measure was called the strategy index. Against the aggressive lag 1, the humans scored a strategy index of 11.6; against the aggressive lag 2, -5.7; and, against the rock=2 model, 19.8. This suggests that against the lag 1 models, the humans were able to predict the models' moves with greater than chance success. However, against the lag 2 model, it was the neural network that was better able to predict the humans' moves. That the humans enjoyed an average net points advantage was entirely due to their sensitivity to the fact that winning with rock was worth two points (i.e., they would play so as to maximize wins using rock).

In Rutledge-Taylor & West (2004) a neural network model of human play was proposed. It was produced using a genetic algorithm. The model was a lag 2 network with a unique reward matrix: rock wins = 3, paper wins = 2 scissors wins = 0; rock tie = -1, paper tie = -1, scissors tie = 0; and -2 for all losses. This model produced game results similar to that of the human participants. See table 1.

This GA model produced good results. However, there is, at best, a rather ad hoc story to be told about the values in the model's reward matrix. This concern motivated the production of an ACT-R model of human paper, rock, scissors play.

ACT-R models

There are many different ways to build a paper, rock, scissors game player in ACT-R. Not only are there many parameters, affecting the manner in which ACT-R models behave, there are also several different architecturally distinct ways that a model designed to play paper, rock, scissors could be built. For example, there is the distinction between a rule based ACT-R model and an exemplar based ACT-R model, as described in Anderson & Matessa (1998). To examine every architecturally distinct ACT-R model would be too grand a project, so we limited ourselves to an exemplar based model of game play proposed by Lebiere and West (1999). We examined both lag 1 and lag 2 ACT-R models of the normal rock=1 version of paper, rock, scissors, and the rock=2 version of the game.

The model is embedded in LISP code. The LISP code provides the ability to automatically reset and run the model multiple times, and to log data from those runs. It is also the means of sustaining an opponent for the ACT-R model. A LISP implemented neural network model is defined in the code.

How does the model work?

As mentioned above the ACT-R models are exemplar based. This means that the models make their moves based on predictions of what sequences of opponents' moves they've experienced in the past. There is a chunk in declarative memory corresponding to each distinct pattern of moves plus a prediction. For example, in the case of the lag 2 model, there are 27 such chunks; there are nine combinations of last and next to last move, and for each combination there are three possible predictions. For each iteration of play, the model recalls a chunk that matches the last two moves made by the LISP neural network opponent; the chunk with the greatest sum of activation and noise, is selected. The move that beats the predicted move is played, and the LISP network makes it's choice (as described above). The result of the play is recorded (using LISP). Were this all that the model did, it would not learn. This is because only the recalled prediction would be rewarded, and worse, regardless of whether the prediction was correct. Therefore, after the network's move is revealed, a series of productions retrieve the chunk in memory corresponding to this move (i.e., the chunk triplet of last move, second to last move, and the networks subsequent move). This way, the correct chunk is reinforced.

Table 1. Human and GA neural network model performances compared

	Human		GA model		Disagreement	
Opponent	Pts. Diff	S.I.	Pts. Diff	S.I.	Pts. Diff	S.I.
Agg. Lag 1	16.5	11.6	16.98	10.68	-0.48	0.92
Agg. Lag 2	5.7	-5.7	4.42	-1.09	1.28	-4.61
Rock=2	25.6	19.8	22.37	16.78	3.23	3.02

Testing the ACT-R models

In West & Lebiere (2001), humans were pitted against three different neural network models, as described above. In the case of the human versus the aggressive lag 2 model, the results were somewhat ambiguous. Whether the score difference in favour of the model was due to an inherent skill inferiority on the part of the human participants, or due to extraneous factors such as lack of motivation is unclear. Therefore, the ACT-R models of human play in the standard paper, rock, scissors game, were compared only to the results of humans versus the aggressive lag 1 model and the passive lag 2 model. For both the lag one and lag two ACT-R models, ans values of 0.35, 0.30, 0.28, 0.25, and 0.15, were tested. Also, the effect of optimized learning was tested. For each combination of parameters, 100 simulations of 300 games each were run. The models were reset between each simulation. The first obvious result was that optimized learning drastically reduced the performance of the ACT-R model. No combination of lag and noise value could produce an ACT-R model that came close to replicating the human data. However, when optimized learning was turned off (sgp :op NIL), several good models of human play were produced. See table 2 for a comparison of two models

Optimized learning drastically compromises the ACT-R models' abilities to compete against lag 2 neural network models (these models also performed poorly against the aggressive lag 2 model in pilot simulations). The best ACT-R model was the lag 2 model with a noise setting of 0.28, and optimized learning turned off. Although, the win differences against the two opponent neural networks was not a perfect match, this model is considered a success due to the fact that only a single parameter was manipulated; all of the other parameters were left at their default settings (with the exception of the optimized learning parameter, which is by default on).

Coming up with an ACT-R model of the Rock=2 paper, rock, scissors player proved to be a challenge. As a starting point, the model with a noise setting of 0.28 and optimized learning turned off was pitted against the aggressive lag 1, the aggressive lag 2, and the Rock=2 models. The result was a fairly good match of the human data. Next, a model with the default noise setting (sgp :ans 0.25) was tested. This model was a somewhat better match.

Table 3 presents a comparison of the points differences and strategy indices for human participants, and the two ACT-R models designed to play the Rock=1 game. The Sum SQ diff. row indicates the sum of the squares of the differences between the models' scores and the human data. The rating row is the sum of the points difference and double the strategy index difference, divided by 1000. The strategy indices were weighted more heavily in determining the rating of a model due to the fact that it's values tended to be smaller than the points difference values.

Despite the fairly good match of the Rock=2 models to the human data, there was one main concern. This was the fact that the naïve models lost to the aggressive lag 2 model. This, however, should not be unexpected. In West & Lebiere (2001) it was observed that human participants tended to lose to the aggressive lag 2 model. However, in Rutledge-Taylor & West (2004) it was reported that human participants were aware that winning with rock was more

Table 2. Naive ACT-R models of rock=1 paper, rock, scissors play

	Human	ANS=0.28, OL=NIL		ANS=0.35, OL=T	
Opponent	Win. Diff	Win. Diff	Error	Win. Diff	Error
Agg. Lag 1	9.990	12.297	2.307	4.337	-5.653
Pas. Lag 2	11.645	8.149	-3.496	-3.010	-14.655
Sum SQ. Err.			17.548		246.727

Table 3. Naive ACT-R models of rock=2 paper, rock, scissors play

	Human		ANS=0.28, OL=NIL		ANS=0.25, OL=NIL	
Opponent	Pts. Diff.	S.I.	Pts. Diff.	S.I.	Pts. Diff.	S.I.
Agg. Lag 1	16.500	11.600	16.921	16.781	11.554	14.748
Agg. Lag 2	5.700	-5.700	-7.762	-5.198	-4.267	-1.877
Rock=2	25.600	19.800	41.396	32.153	35.792	27.309
Sum SQ diff.			430.928	179.689	227.684	80.901
Rating			0.079		0.039	

Table 4. Savvy ACT-R models of rock=2 paper, rock, scissors play

	Human		Rock		Scissors		Rock+Scissors	
Opponent	Pts. Diff.	S.I.	Pts. Diff.	S.I.	Pts. Diff.	S.I.	Pts. Diff.	S.I.
Agg. Lag 1	16.50	11.60	37.70	40.64	4.09	19.14	26.24	21.63
Agg. Lag 2	5.70	-5.70	11.88	15.36	-16.40	-1.38	7.52	2.23
Rock=2	25.60	19.80	52.94	56.94	25.38	30.02	38.45	29.07
Sum SQ diff.			1235.22	2666.24	642.45	180.08	263.35	249.54
Rating			6.57		1.00		0.76	

valuable than winning with paper or scissors, and explicitly tried to maximize rock wins. This is apparent from the fact that human participants were able to achieve, on average, a positive points difference against the aggressive lag 2 model, despite the negative strategy index. Thus, an ACT-R model sensitive to the Rock=2 game parameters was designed.

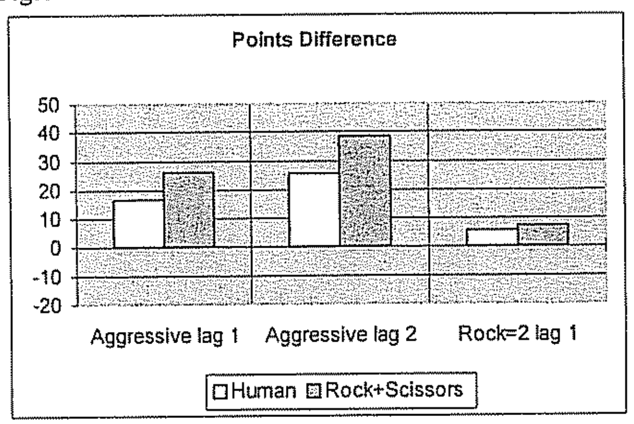
Just as there are many different ways to build a paper, rock, scissors player in ACT-R, there are various options for how to "build-in" knowledge that winning with rock is worth more than winning with either paper, or scissors. The option that was chosen for this experiment was to double harvest chunks associated with particular plays by the model's opponent. Three variations were tested. First, rock plays received extra attention. That is, when the neural network played something other than rock, the relevant chunk was retrieved and harvested once, just as is the case with the Rock=1 models. In the case of rock, this retrieval process occurs twice. The rational is that human participants might be merely paying more attention to when their opponents play rock. Technically, the effect is that the ACT-R model will be more likely to predict that the neural network opponent will play rock, and respond with paper. This will cause the ACT-R model to play paper more frequently, which will indirectly have the effect that the neural network will be more likely to play scissors, which is desirable for the ACT-R model. This is because by playing scissors more frequently, there is a greater likelihood that ACT-R will benefit by playing rock. Obviously, the dynamics of the interplay between the players is very complex. In fact, West & Lebiere (2001) argue that when two neural network players are pitted against one another, they form a chaotic dynamic system. Second, we tested models which gave extra attention to scissors. And last, we tried a model that gave extra attention to both rock and scissors plays by it's opponent.

In testing the ACT-R models in the Rock=2 game, optimized learning was turned off, and a noise setting of 0.25 was used.

The Results

Of the three ACT-R models, each paying extra attention to different subset of opponent moves, none matched the human data perfectly. However, the model paying extra attention to opponent plays of both rock and scissors produced a good qualitative fit to the human data. Figure 1 depicts a comparison of the points difference achieved by both the human participants and the Rock and Scissors ACT-R model versus each of the three neural network models. Table 4 summarizes the comparisons of the models' performances against the three neural network opponents and that of human experimental participants. Although the "scissors" model matched the strategy index values best, it's failure to beat the aggressive lag 2 neural network is the principle reason for disqualifying it as an adequate model of human play.

Figure 1. Rock and Scissors model of Rock=2 PRS



Conclusions

The results of human experimentation from West & Lebiere (2001) and Rutledge-Taylor & West (2004) were replicated with varying degrees of success. Good models of the human performance in the Rock=1 game were produced, without tinkering with the available parameters controlling the manner in which ACT-R behaves. However, in the case of Rock=2 game, the naïve ACT-R model was unable to take advantage of the fact that winning with rock was worth more than winning with either paper or scissors. Of three models designed to pay extra attention to certain opponent moves, the model that attended to both rock and scissors plays matched the human data well. Therefore, the process of retrieving chunks twice is a viable option for increasing the activation of chunks more than would be achieved by the normal "popping" and "harvesting" process. Intuitively there seems to be a need for this.

References

Anderson, J. R. & Matessa, M. (1998) The rational analysis of categorization and the ACT-R architecture. In M. Oaksford & N. Chater (Eds.), *Rational Models of Cognition*. Oxford University Press.

Lebiere, C., & West, R. L. (1999). A dynamic ACT-R model of simple games. *Proceedings of the Twenty First Annual Conference of the Cognitive Science Society*, 296-301. Mahwah, NJ: Erlbaum.

Rutledge-Taylor, M. F. & West, R. L. (2004) Cognitive modeling versus game theory: Why cognition matters. *Proceedings of the sixth International Conference on Cognitive Modeling*, 255-260. Pittsburgh, PA: Carnegie Mellon University/University of Pittsburgh.

West, R. L. (1998). Zero Sum Games as Distributed Cognitive Systems. *Proceedings of the Complex Games Workshop. Tsukuba*, Japan: Electrotechnical Laboratory Machine Inference Group.

West, R. L., & Lebiere, C. (2001). Simple games as dynamic, coupled systems: Randomness and other emergent properties. *Cognitive Systems Research*, 1(4), 221-239

Navigation in Degree-of-Interest Trees

Raluca Budiu (budiu@parc com)
Peter Pirolli (pirolli@parc com)
Palo Alto Research Center

ACT-R Workshop. 2005. Trieste. Italy

parc

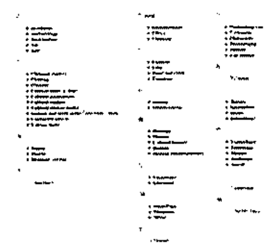
Hierarchical Displays

Many collections of information are organized hierarchically

2

Hierarchical Displays

Many collections of information are organized hierarchically

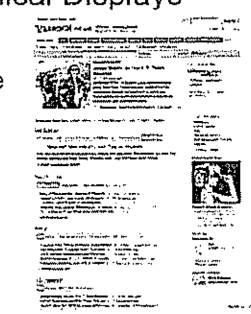


http://en.wikipedia.org/wiki/Category:Culture

3

Hierarchical Displays

Many collections of information are organized hierarchically



Good Visualization Requirements

1. Space in nodes to display information
2. Relationship between node and context
3. Quick search for a node
4. Fits into a bounded region

(Card & Nation 2002)

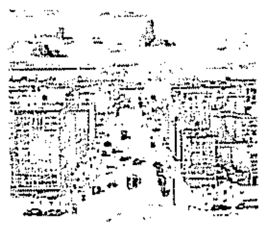
5

Overview

- Degree of Interest (DOI) trees
- DOI tree experiment and results
- Tentative ACT-R Model
- Problems
- Tentative solutions

6

DOI Trees



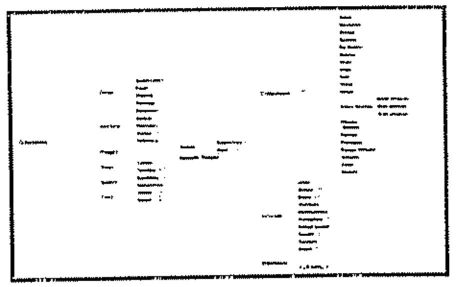
Saul Steinberg
"View of the world from Ninth Avenue"

Which portion of a tree should it be displayed?

A combined function of:
- importance of a node.
- distance to the user focus

7

Degree of Interest (DOI) Trees



Demo

8

DOI Experiment

Task:

Given a hierarchical information structure, find a given node in the structure.

Examples:

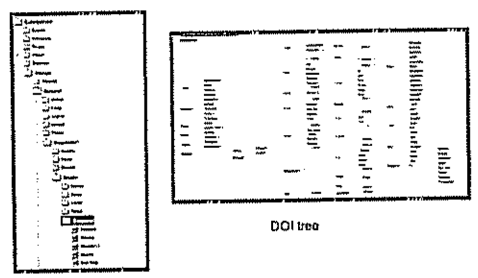
Find a banana.

Find the play "Romeo and Juliet".

(Run by J. Heiser and M. Fleetwood)

9

The Browsers



Windows Explorer - like

DOI tree

10

Task Difficulty

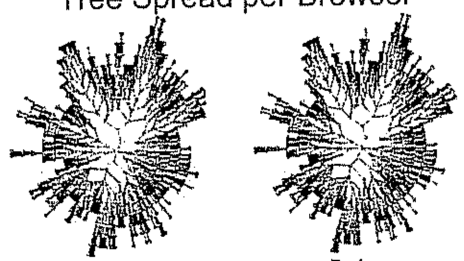
How likely a node is to be on its actual path

- High scent tasks (easy)
  Categories → Things → Natural → Vegetable → Fruits → Tropical → Banana
- Low scent tasks (hard)
  Categories → People → Specific People → Organizations → Governmental → United States → Legislative Branch → Library of Congress

A different rating experiment was conducted to assess the scent of the task

11

Tree Spread per Browser



DOI

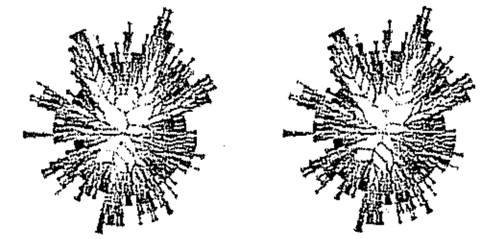
Explorer

Visited nodes

Find the banana

12

Tree Spread per Task



Find the Library of Congress

Categories → People → Specific People → Organizations → Governmental → United States → Legislative Branch → Library of Congress

Find the Chevy Corvette

Categories → Things → Artificial → Tangible → Mechanical → Machines → Vehicles → Cars → American → General Motors → Chevrolet → Corvette

13

Experimental Results

- No difference in RT for the two browsers
- Larger "spread" in the DOI browser
  - More nodes seen in the DOI browser
  - Fewer revisitations per node in the DOI
  - Wondered farther away from the solution path in the DOI browser
- Low scent task involved lower RTs, more nodes visited, farther away from the solution

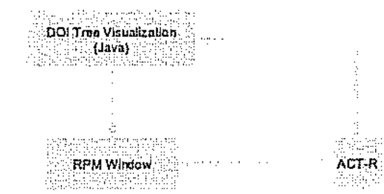
14

Overview

- ✓ DOI trees
- ✓ DOI tree experiment and results
- Tentative ACT-R Model
- Problems
- Tentative solutions

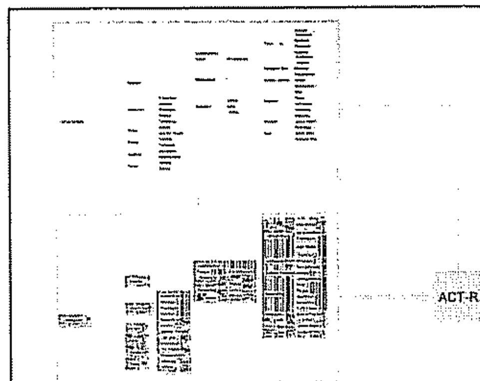
15

ACT-R Navigates DOI Trees



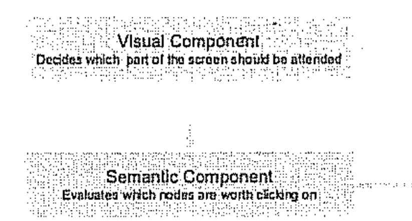
J. Heer M. Fleetwood

16



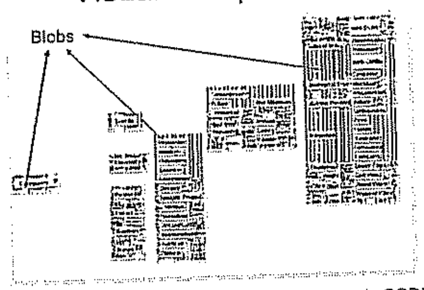
17

Model Components



18

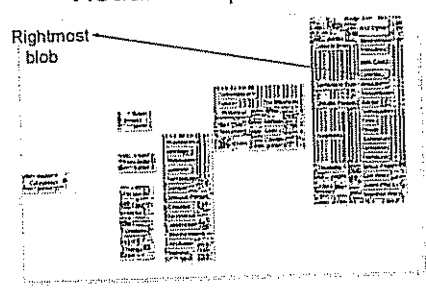
Visual Component



Blobs = groups of items close together (a la Logan's CODE theory)

19

Visual Component



The rightmost unattended blob is attended first

20

Semantic Component

Within a blob:

Attend to all nodes in the blob

For each node evaluate it for relevance to the goal

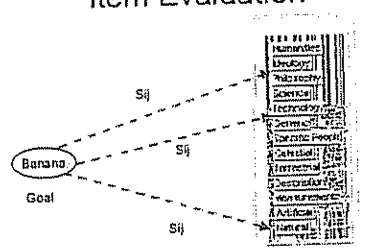
Select the node that has the best "evaluation" function

Click on that node

This algorithm is optimal within that blob (it should find the best item if the evaluation function is correct)

21

Item Evaluation



The item in the goal is associated to all the nodes
Sij = "semantic similarity" (GLSA value between the two words)

22

Information Scent and Activation Spreading

- Traditionally, information scent has been expressed as production utility (Pirolli 2005)
- We now attempt to go back to more natural ACT-R representations (strength of associations, partial matching)

23

Item Evaluation

- Attempt to retrieve each item in the blob
- If retrieval is successful, then mark the item as a possible candidate
  (This step will prune out all candidates with low similarity to the target item)
- When no more items within the blob:
  retrieve the candidate with the highest similarity (i.e., with highest activation value)
- Click on that candidate

24

Feedback to Visual Component

- All the nodes in a blob are attended one by one
- However, if the candidate with the highest similarity is not in the current blob, that candidate will be selected and attended next (if it is still on screen)

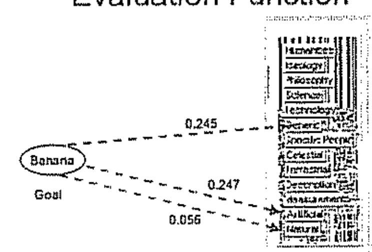
25

Problem

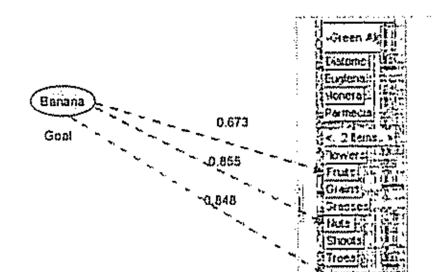
- This model cannot find the solution to some of the problems in a reasonable amount of time
- Reasons:
  - people do not necessarily attend the rightmost blob
  - similarity-based evaluation function may not be right

26

Evaluation Function



27



We compared GLSA/LSA selections for the first three levels in the tree with human ratings:
No rank correlation!

28

Changes to the Evaluation Function

Perhaps evaluation is not similarity based, but categorization-based

Is banana more a fruit than a nut or a vegetable?

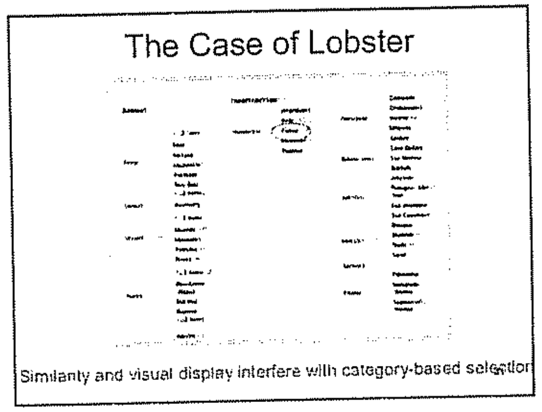
29

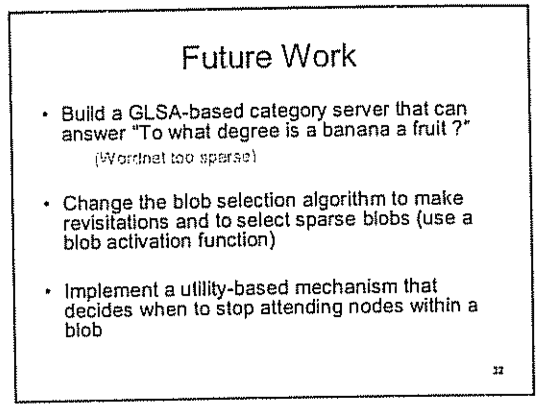
The Case of Lobster

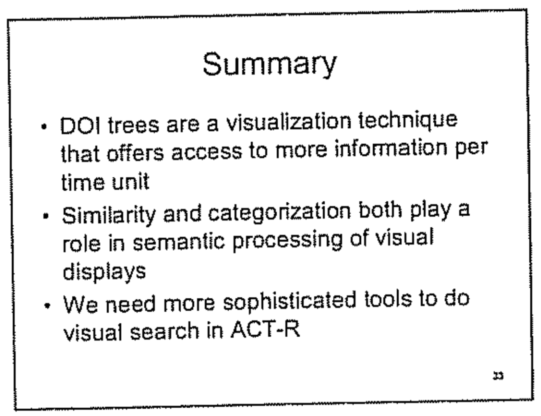
DOI Browser

Explorer Browser

30







Spatial orientation in ACT-R: Architectural insights and extensions

Glenn Gunzelmann (glenn.gunzelmann@mesa.afmc.af.mil)
Air Force Research Laboratory
6030 South Kent Street
Mesa, AZ 85212 USA

Spatial Orientation

This paper describes an ACT-R model that performs the spatial orientation task illustrated in Figure 1. The model has access to two views of a space, an egocentric visual scene and an allocentric map. It must identify the location of the viewer on the map based on the content of the visual scene. Responses are made by clicking on the dark green ring around the outside of the map. The model makes realistic predictions about the response times and errors of several pilot participants. We currently are collecting eye tracking data for eventual comparison with the model's predictions.

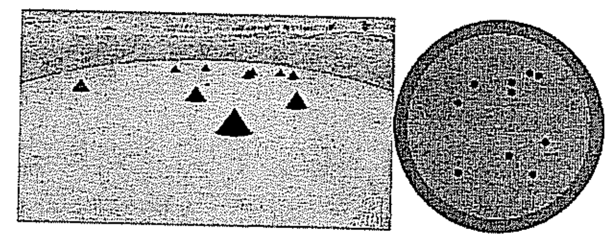


Figure 1: Sample Trial. Where is the viewer?

Instance-based Learning

The model's behavior is guided by instance-based learning (IBL). At various points in the solution process, the model must decide whether the current estimated response is "good enough" or whether further refinement/information-gathering should be done. To make this decision, the model retrieves a chunk from declarative memory that encodes the results from a past experience. If the chunk represents a previous error, more refinement is done to produce a better estimate. When a previous correct response is retrieved, the response is made. Noise and similarities between chunks impact these retrievals and make the model fallible.

Individual Differences In the model, instances stored in memory have a slot indicating whether it was a *correct* or an *incorrect* response. The retrieval requests can specify this value, and the similarity between the chunks representing the two slot values in memory can be varied as well. Together, these two features specify the response bias/criteria for the model. If correct and incorrect are highly dissimilar and the retrieval request is for a previous correct response, the model approaches the task with the philosophy: 'If this were ever a good enough approximation, then it is good enough now.' In contrast, if the retrieval request is biased toward retrieving an incorrect instance from memory, the model's philosophy is: 'If this approximation was ever not good enough, then it is not good enough now.' By varying the similarity between the two values intermediate degrees of bias can be obtained.

With this mechanism, overall accuracy can vary from around 35% in a model focused on retrieving a correct instance to 95% in a model focused on retrieving an incorrect instance. This covers the range of performance observed so far in the human data (58.1% to 85.1%). This suggests that individual differences in human performance on this task may be a function of response bias or tendencies, and illustrates how ACT-R can be used to capture these effects using IBL.

Perceptual Grouping and Visual Search

In addition to the instance-based learning mechanism, the model incorporates modifications to the vision module that enhance its functionality. Perceptual grouping has been added to allow the model to see clusters of objects on the screen. This allows ACT-R to use a strategy that is more in line with how participants report doing the task. The perceptual grouping algorithm is applied when *pm-proc-display* is called, producing a set of groups, which are then treated like any other item in the visual icon by the architecture. The value slot is tagged with an identifier both in the group and in the objects in the group. This allows for efficient search among group members. The algorithm, itself, is hierarchically based. Additional work is still needed on issues such as defining the appropriate level of the hierarchy for grouping.

In addition to the perceptual grouping mechanism, the model introduces enhancements to the visual search options that allow ACT-R to search along vectors, rather than just according to horizontal and vertical constraints. Additional slots have been added to the visual-location chunk for *bearing*, *range*, and *reference*. Visual search can be constrained by the bearing and distance from either the current location of fixation or another visual location, specified in the *reference* slot. These additional options add significant flexibility to visual search that is conducted relative to a given visual location on the screen.

Time Interval Estimation: Internal Clock or Attentional Mechanism?

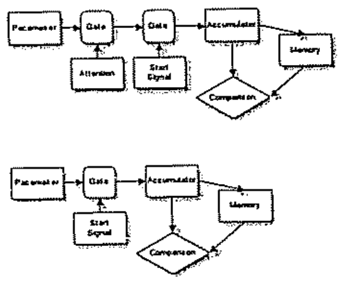
Niels Taatgen, Hedderik van Rijn and John Anderson
Carnegie Mellon University and University of Groningen

The human ability to accurately estimate time intervals in the order of 0 to 20 seconds can be explained by two seemingly incompatible theories: the internal clock and the attentional counter theory. The attention counter theory postulates that attention is needed to advance a counter during the estimation or reproduction of an interval. The internal clock model has no role for attention and can estimate intervals with need for intervention. Although empirical and neurophysiological data seem to favor the internal clock theory, there are nevertheless effects of attention that it cannot explain. Our ACT-R temporal module is modeled after the internal clock theory, but because it is a component of ACT-R the attentional expects of time estimation can be explained in quite a different way than the attentional counter theory proposes.
The support for this assertion consists of two experiments with respective models. The first experiment uses a dual-task paradigm in which time estimation has to be done together with an choice-reaction time task. This experiment was used to construct a model based on the temporal module. To further test this paradigm we designed a second experiment in which we combined estimating time with two other task, making it a triple-task paradigm. Before we actually did the experiment we constructed a model and made a prediction (announced on the ACT-R mailing list). The predictions turned out to be fairly accurate with a fit that is indisputably zero-parameter.

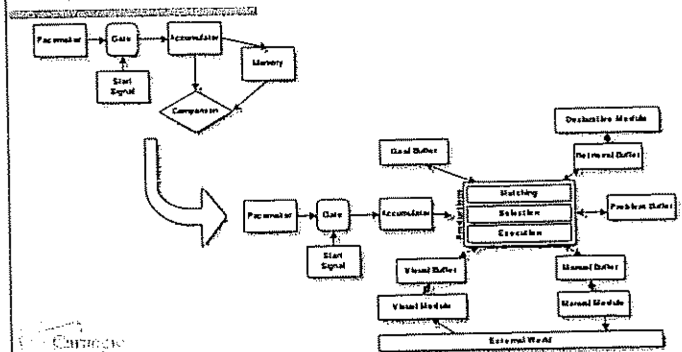
Two theories

Attentional counter (Zakay, Block):
Timing intervals demands attention

Internal clock (Matell & Meck):
No role for attention



ACT-R version of internal clock

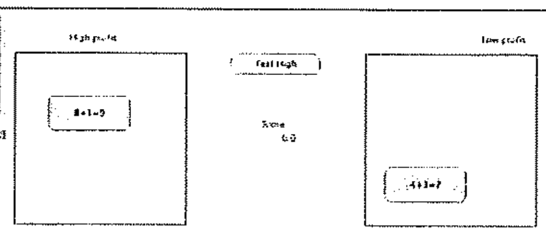


Idea of the experiment

- Subjects have to estimate an interval of unknown duration, but this is only one of three tasks they have to do
- The difficulty of the other tasks is manipulated to see the impact on timing accuracy

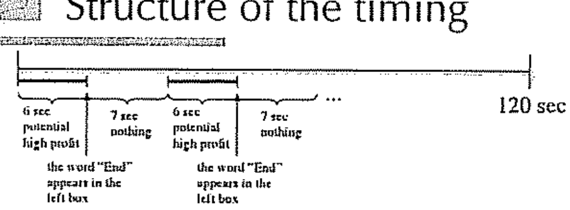
Task: Dual-task Timing

- Actually a Triple Task:
  - Two areas with visual stimuli
  - And a time estimation task



- Two boxes with stimuli
  - Stimuli are continuously presented in both boxes, correctly responding to a stimulus gives a score of 30 points. Respond in left box by pressing space, and in the right box by clicking the stimulus with the mouse
  - During certain intervals, stimuli in the left box can yield 100 points, but only if this interval is "activated" by pressing "Test High", making time interval estimation the third task
- Two types of stimuli (depending on condition):
  - Additions (respond to correct additions)
  - Letters (responds to "A"s but not to "B"s)

Structure of the timing



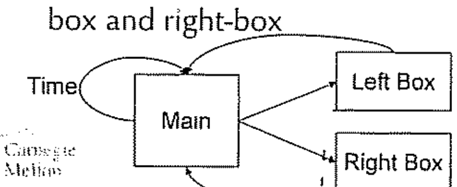
- If the participant clicks the "Test high" button during a 6 sec period, targets in the left box will yield 100 points for the remainder of the 6 sec period ("HIGH PROFIT" will appear to signify this). If the participant clicks during the 7 sec period, nothing happens.

Four conditions

- Blocks consist of 5 trials of 120 seconds
  - LL: 4 blocks of letters
  - AA: 4 blocks of additions
  - LA: 2 blocks of letters, then 2 blocks with additions
  - AL: 2 blocks of additions, then 2 blocks with letters
- 10 Subjects/condition

Model

- The model is based on an earlier experiment that was different in the sense that it was only a dual task.
- It uses three control states: main, left-box and right-box



Control State Main

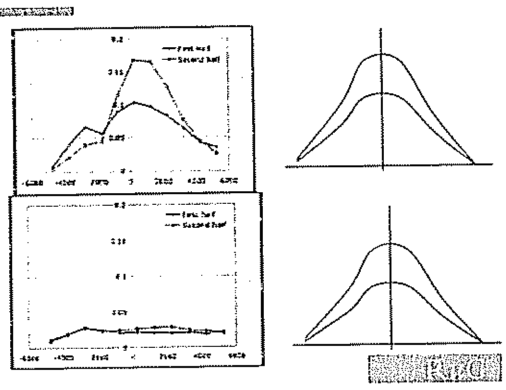
- If there is a stimulus on the left side, attend it and switch to the left-box control state
- If there is a stimulus on the right side, attend it and switch to the right-box control state
- If the profit is currently not "HIGH", try to retrieve an example of clicking on "Test" for the current time
- If a previous experience with the current time has been retrieved, then click "Test" if that experience was successful
- If retrieving an example yields a retrieval failure, then randomly decide whether or not to click "Test".

Effect of difficulty of the secondary tasks

- Attentional counter theory: when the secondary task is more difficult, counting is slower
- Internal clock/ACT-R: when the secondary task is more difficult, the probability that you completely forget the timing task is higher

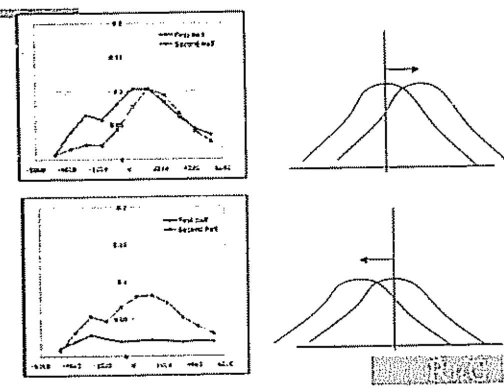
Predictions ACT-R vs. Attention

- Easy
- Hard



Predictions ACT-R vs. Attention

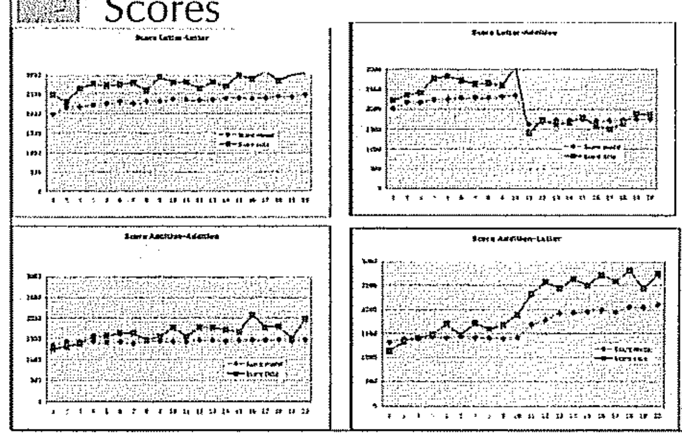
- Easy to Hard
- Hard to Easy



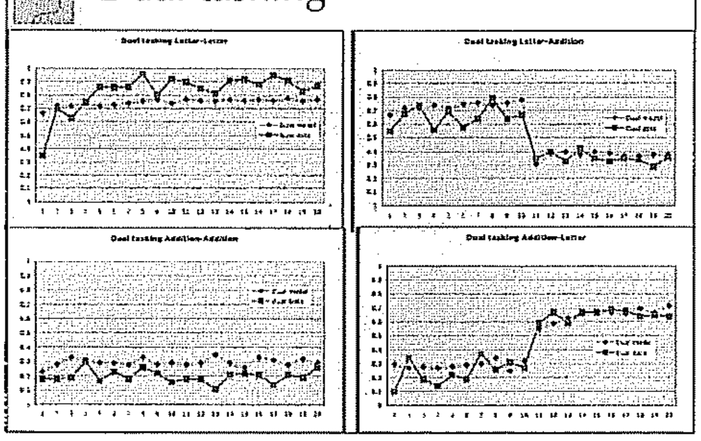
What else are we looking at?

- Score
- Amount of dual tasking:
  - During a high-profit period, how often do people also work on the low-profit stimuli?
- How often do subjects fail to attend to time at all?

Scores

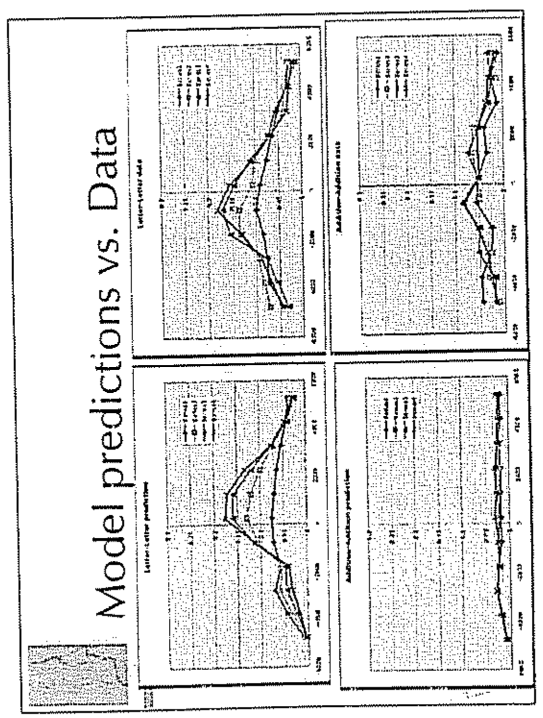


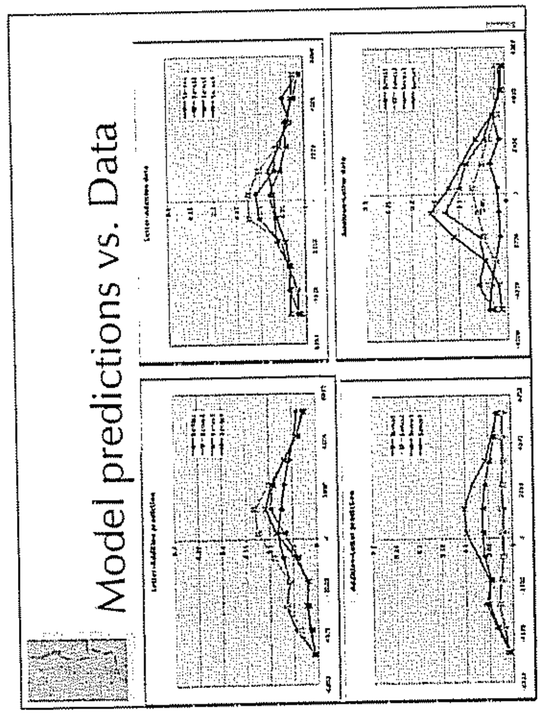
Dual tasking

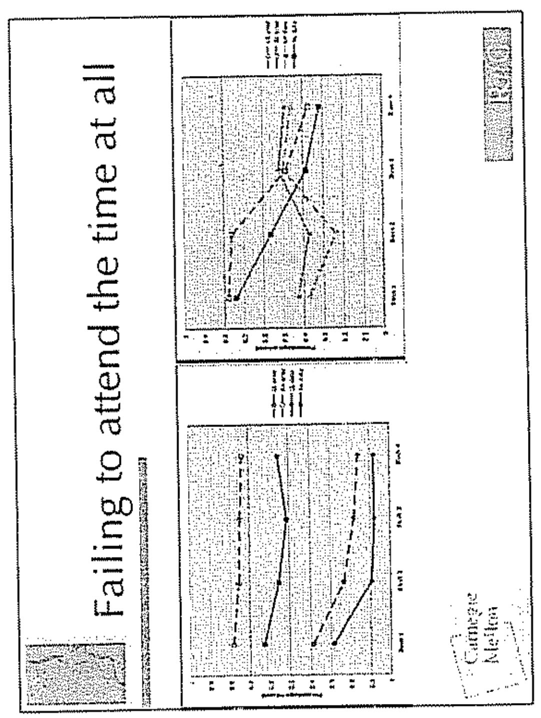


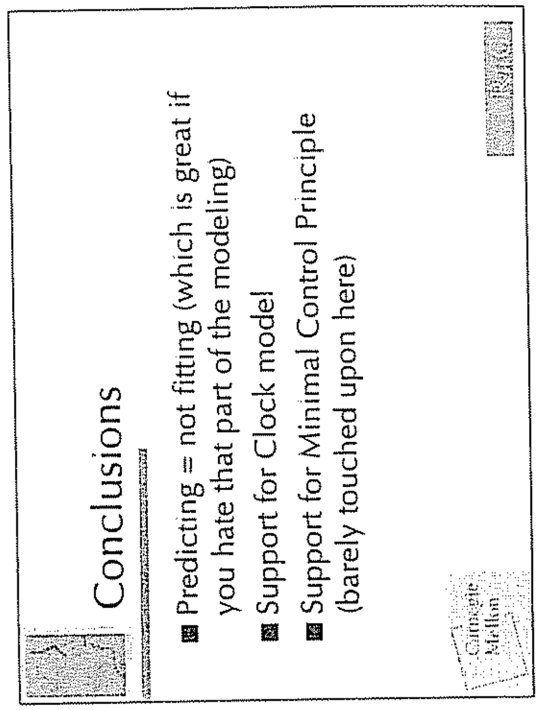
Time estimation

- We will look at the deviation from the ideal click-time
  - If subjects click too early, nothing happens
  - If they click up to 7 seconds too late, they still get a high profit period, but it will be shorter
  - If they don't click at all this is counted as a miss
  - Only the first click within a period is analyzed.
- Graphs will show the time distribution of the clicks for each of the four blocks of the experiment









Individual differences in multi tasking

Lisette Mol (lisettemol@cmu.edu), Niels Taatgen, John Anderson
Psychology Department, Carnegie Mellon University

An experiment consisting of two tasks was carried out in an attempt to measure structural individual differences. That is to say individual differences that vary over participants rather than over trials

The first task was a task in which targets could appear either in a window on the left or in a window on the right of the screen. Participants could earn points by responding to targets on the left by pressing the space bar and to targets on the right by clicking on them with the mouse. Targets could appear simultaneously on the left and on the right. In addition, participants could earn a higher profit for targets on the left, by estimating a time interval, of which they did not know the duration. They could click a button to see whether a new interval of high profit had begun. Periods of high and low profit were alternating and of even length. If the button was clicked during a high profit period, participants would receive the higher profit for responding to targets on the left until the end of that period.

The second task was the Abstract Decision Making task (ADM), developed by Joslyn and Hunt. In this task participants have to sort objects into boxes, which can only take objects with certain features. Participants cannot see the objects or the boxes. They can pose questions on the features of the objects using a text interface. The features of the boxes need to be remembered. During the task, objects become available once every 30 seconds in the practice trials and once every 15 seconds in the test trials. It is possible that a message that a new object has become available pops up while the participant is still working on the previous object. The return key has to be pressed in order to continue after such a message.

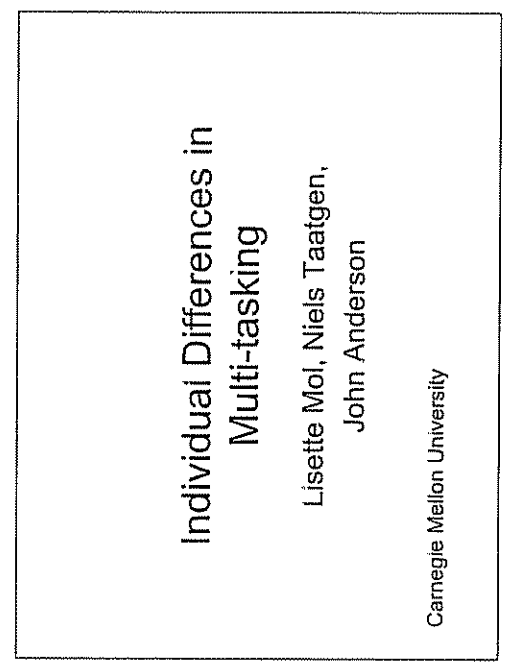
A correlation of 0.73 was found between the average time it took a participant to assign an object to a box in the ADM, and the percentage of high profit periods in which the participant did not click the button to try to estimate the time interval in the first task. This suggests that there are indeed structural individual differences underlying performance on these tasks. To investigate what these differences consist of, ACT-R models of both tasks were developed. Our hypothesis is that the coordination of top down deliberate reasoning and bottom up processing of visual stimuli can account for most of these differences.

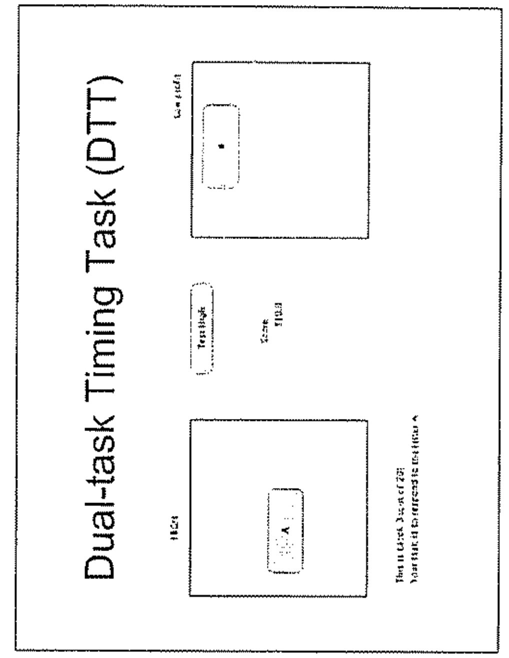
For the timing task a model was made before the experiment was conducted, to predict the data. This model was based on a simpler version of the task in a previous experiment. The predictions of this model matched the data quite well. To test our above hypothesis, a second version of the model was created, in which visual stimuli could not interrupt the process of testing for a high profit period, once it had been initialized. In this version top down processing had priority over bottom up processing whereas in the initial model this was the other way around. In the first model visual stimuli were given priority. Using the first model for the participants that did not learn to estimate the interval and the second model for participants who did, the models matched the data even better.

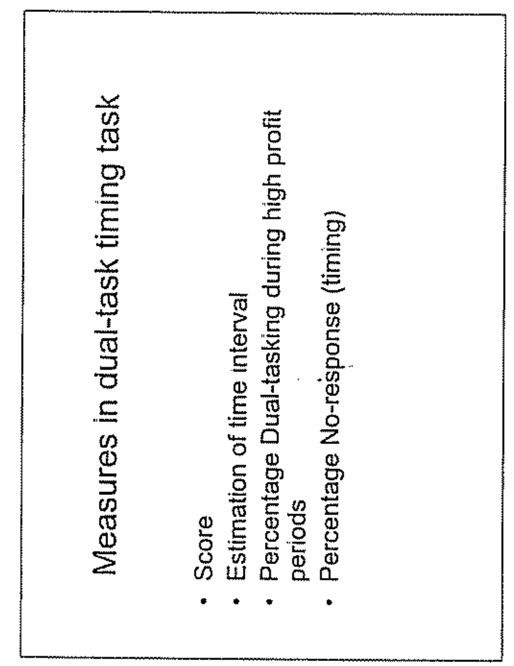
For the ADM task three different models were made. One in which the strategy was to think of a decision tree in advance. This allowed for very efficient questioning of the objects and corresponds to a top down strategy. In a second model, which corresponds to giving priority to bottom up processing, all features were asked before an object was assigned to a box. In a third model the strategy to ask for features until there was a single box left in which the object would fit was implemented. This corresponds to combining top down and bottom up processing.

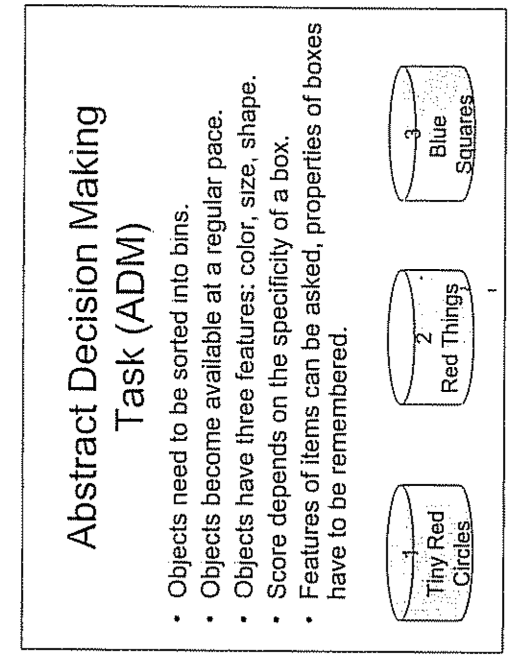
It was found that performance on the ADM of participants who were bad at estimating the time interval in the first task, was most often best described by the predictions of the model which always asked for all features. Performance on the ADM of participants who were good at estimating the timing interval, was most often matched best by the results of one of the other two models. This is in accordance with the hypothesis that some participants are better able to benefit from top-down processing than others.

So far, some support has been found for the hypothesis that the ability to coordinate top down and bottom up processing can account for some of the structural individual differences that we found in participants who performed the two tasks described above. In future work, we intend to do follow up experiments and to perform more statistical analyses on the experimental data and the outcomes of our models. For a follow up experiment on the ADM, we intend to make predictions on participants' strategy based on our ACT-R models and to monitor their behavior in a more detailed way.









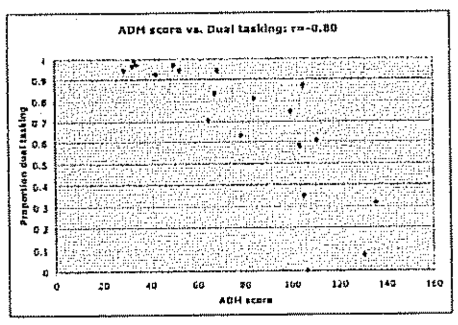
Measures in ADM

- Score, reflects how many objects are sorted correctly and wrongly and into how specific a box.
- Average 'lifetime' of an item: The time between the moment that an item becomes available and the correct classification of that item.

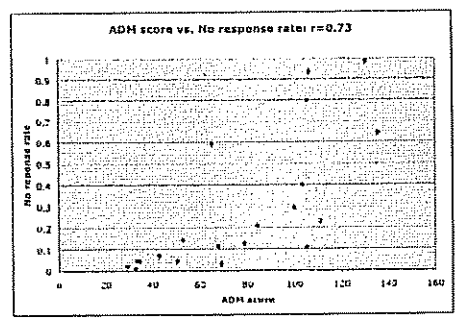
Correlation

- 20 participants
- Average over 2 blocks of Dual-task Timing task
- Average over 2 practice games & 2 test games of ADM
- Specific Dual-task measures of DTT correlate high with performance on the ADM task.

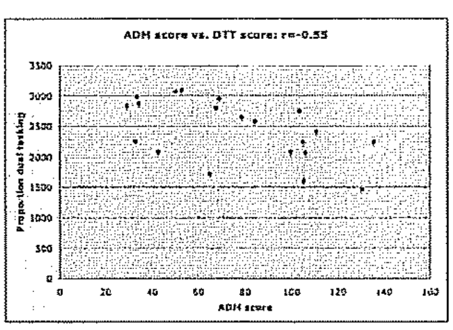
Average lifetime (high is bad) vs. Dual tasking rate (high is good)



Average lifetime (high is bad) vs. No-response rate (high is bad)



ADM score (high is bad) vs. DTT score (high is good)



Hypothesis

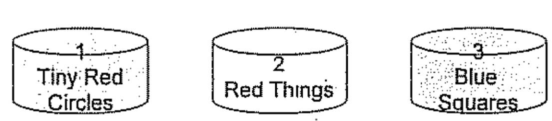
Individual differences are produced by how well people can balance top-down and bottom-up control, i.e., how much they let themselves be driven by the interface or maintain some measure of control themselves.

Modeling of dual-task timing task

- Model in which estimation of time interval is interrupted by the processing of visual stimuli.
  (bottom up processing gets priority)
- Model in which estimation of time interval takes precedence over the processing of visual stimuli.
  (top down processing gets priority)

Modeling of ADM

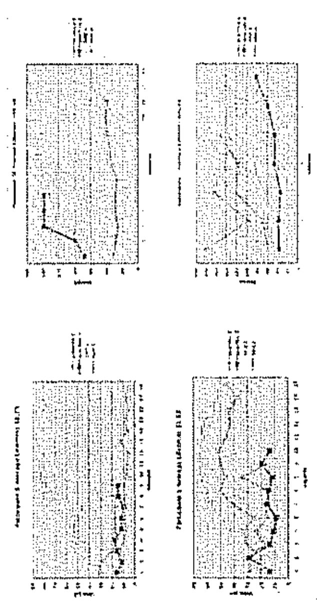
- Decision tree model (top down strategy)
- Model that always asks all features (bottom up strategy)
- Model that asks features until an item fits into a single bin (combined strategy)



Preliminary classification

	ADM Bottom-up	ADM Ask-until-one-bin	ADM Decision tree
Dual-task interruptible timing	16	5	1
Dual-task protected timing	3	9	6

Average lifetime does not capture details



Conclusion

In both the ADM task and the Dual-task timing task it seems that (slightly) more top-down control improves performance.

Future work

- Obtain more detailed data on performance on the ADM task, such that participants' strategy can be determined.

How to integrate time-duration estimation in ACT-R

Jeronimo Dzaack, Nele Pape, Sandro Leuchter, Leon Urbas
Research Group User Modeling in Dynamic Human-Machine-Systems
University of Technology of Berlin
Jebensstr. 1, J2-2
10623 Berlin, Germany
+49 30 314 7200

ABSTRACT
From literature a theory of retrospective time-duration estimation is derived. According to the theory a timer-module for the cognitive architecture is invented and described. In an empirical setting the implementation of the theory is proven. The results are discussed against the background of the assumptions.

INTRODUCTION

The estimation of time-duration in dynamic human-machine-systems is an essential requirement for system control (see Schulze-Kissing et al. 2004). Time-duration estimations help us to stay tuned to the sequential occurrence of events in a complex environment. Retrospective time-duration estimation is an important aspect in developing systems concerning human-machine interaction. In some situations only the processing of temporal information enables persons in complex human-machine systems for example to differentiate between a feedback delay caused by a system innate latency, and an expanded feedback delay that is caused by abnormal system performance. If the duration of a feedback delay exceeds the expected normal latency duration, an operator suspects a malfunction (Schulze- Kissing, et al., 2003). Time-duration can be estimated either prospectively from some event to some point in the future, or retrospectively form some point backward into the past.

Only a few studies are concerned with retrospective time-duration estimation and even less empirical studies can be found. Because of this lack of investigations of time-duration estimation in cognitive architectures, especially in ACT-R, our research group concentrate on retrospective time-duration estimation. We developed an ACT-R timer-module that implements a theory of human time-duration estimation distorted by information processing. In an experimental study the empirical validity of the new timer module of ACT-R and its mapping to real settings is tested.

THEORETICAL BACKGROUND

Regarding the literature several theoretical approaches of time-duration estimation can be found. It seems that the relative length of prospective and retrospective judgments may depend on several variables and that different processes are involved.

According to Ornstein retrospective duration judgments increases as a function of the amount of stored and retrieved information, or storage size allocated (Ornstein, 1969).

Hicks and co-workers (Hicks, Miller, & Kinsbourne, 1976) proposed a model where subjective time-duration estimation is assumed to be increased with subject's attention to time. This attention results in the storage of subjective temporal units. In the retrospective paradigm, subjective temporal units are presumably not stored.

In the contextual-change model proposed by Block and Zakay (1996) retrospective time-duration estimation depends on retrieval of contextual information which is encoded in association with event information. The estimated time-duration is dependent on the amount of contextual changes stored in memory until a point of request.

McClain (1983) conducted an experiment were subjects had to judge a time interval either prospective or retrospective. The subjects had to encode in a fixed time wordlists presented in several intervals. In 120 seconds the subjects had either to encode 15, 30 or 45 words in three different information-processing conditions. In contrast to the prospective condition subjects – under retrospective condition – the estimated time-duration did not differ so much depending on the information-processing condition (i.e. encoding the words in a deep or shallow level of cognition). In the retrospective task the time-duration estimation clearly increased with the amount of words encoded.

For our ACT-R timer-module we used the data presented by McClain to calculate the three factors used in the equation to compute the retrospective time-duration estimation.

In the data presented we added the average active time over the trials (i.e., the time subjects are concerned with encoding each word). The remaining time is seen as idle-time. We set up an equation concerning the estimated time-duration by the subjects (see equation 1). Based on this equation we extracted two independent factors describing the non-idle component as well the idle component of the estimated time-duration.

$$idleTime \times a + nonIdleTime \times b = estimatedTime$$

Equation 1: Equation to calculate factors for retrospective time-duration estimation (a, b: independent factors)

As gathered from the data a is weighted XXX and b is weighted XXX. These values are used in the timer-module described later.

THE ACT-R TIMER-MODULE

At the current state retrospective time-duration estimation is the focus of our work. We developed a retrospective timer-module for ACT-R, that can be integrated in the consisting architecture of the current specification (Anderson 1998). The timer-module attaches to the principal of buffers in ACT-R. We used the buffer-syntax to specify the interface for the application of the timer-module. Thus the timer-module can be added to and used like a new buffer.

As the timer-module is made available as a buffer in ACT-R the interface is obliged to be simple and supplies a comfortable way to be assessed. Three statements are important to work with the timer-module (see table 1).

RHS: +timer> isa timer-reference mode retro id =id	Set a new timer-reference
RHS: +timer> isa timer-reference id =id LHS: =timer> isa timer-reference id =id duration =duration =timer> isa timer-failure	Send a query to the timer-buffer Query the estimated time or a timer-failure
RHS: -timer> isa timer-reference mode retro id =id	Delete a timer-reference (only for technical reasons)

Table 1: Commands for the use of the timer-buffer in ACT-R

The timer-module allows to set reference-points according to reference-points in an episodic memory store (at least the first has to be set). The time between two different reference-point, as well the actual state of the model and a reference point can be enquired. The estimated time between these points is calculated by assessing an amount of time to every contextual change that is stored in this period (i.e., productions fired). The specification of the temporal weights is based on empirical evidence reported in literature as shown above. For technical reasons, as debugging and programming, it is possible to delete a timer-reference.

THE THEORY OF THE TIMER-MODULE

In the following we describe the theory of the retrospective time-duration estimation module. In an approximation it can be compared to an episodic memory store. New episodic reference points can be set to split up the passed time. In the retrospective approach this is made by the explicit setting of distinctive reference-points according to reality (i.e., distinctive actions will be hold in memory as landmarks and help to navigate through the passed time). The time between these reference-points is estimated by the timer-module algorithm (see equation 2).

$$DE = \begin{cases} (A + \frac{B}{\sqrt{AT/TT}}) \times AU \times TT & , AT/TT < 0{,}9 \\ C \times AU \times TT & , else \end{cases}$$

Equation 2: Duration estimation algorithm (DE: duration estimation, AU: action unit, AT: active time, TT: total time, A,B,C: variables based on empiric evidence)

In the timer-module contextual changes are seen as the smallest action units (AU) to calculate the time. In our current timer-module we distinguish between non-idle and idle productions that are weighted by different factors. Every production is regarded as non-idle as long there is no addition of "idle-" to the production-name. Idle productions are regarded as non/less cognitive productions. Thus every non-idle production is seen as a contextual change. In further development it could be possible to combine more productions to an action unit.

Every time a production fires (firing-hook-fn) the overall active time of the simulation is measured (the elapsed time between the last production and the current production is measured (active time – AT)) and divided by the total time from start of the simulation to the current time (TT). If the ratio is less than 0,9 (i.e., idle time is less than 10%) the product of both (AT x TT) is multiplied by a factor composed of the sum of a bias for idle-time (A) and the ratio of an idle-time factor (B) and the radical of the ratio of active time and total time. This allows to integrate the ratio of idle and non-idle time to the equation of the retrospective time-duration estimation. Otherwise the product of active

time and total time is multiplied by a fix factor (C), that means no specific influence of idle-time to the results. To separate these two cases is necessary because the idle-time slots do not give reliable time cues. Thus we integrated the specifics concerned with the different weights into the factors (i.e., independent factors of the equation).

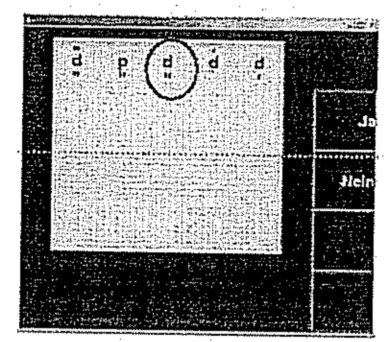
If the measured real-time of a production fired is 3/2 times greater than its default action-time (i.e., the time a production needs to be completed) the calculated time is balance with an idle-factor to adapt the according idle system waiting-time. Thus problems running a simulation in real-time environments can be adopted.

To take into consideration the theoretical background there are two assumptions concerning the retrospective time-duration estimation connected with the timer-module. First, the more productions are fired between two reference points (e.g., there is more cognitive work), the longer the estimated time. This can be explained by the high cognitive work and the connected amount of contextual changes that help to reproduce the passed time while doing a complex cognitive-demanding task. In contrary, the less productions are fired between two reference points, the shorter the time is estimated (e.g., less cognitive work).

Waiting time or idle-time is estimated in another way than cognitive-demanding time. It is estimated shorter than non-idle time, because the cognitive workload is low.

THE ACT-R MODEL

The next step is to verify the timer-module. Therefore we invented an experimental setting and integrated the timer-module in a running ACT-R model. Two analysis have to be made: the timer-module does not affect the performance of the ACT-R model, because retrospective time-duration estimation has no effect on the performance of humans. And the retrospective estimated time-durations have to correlate with estimations measured in reality.



Picture 1: D2-Drive version A

The used ACT-R model for both analysis is a model invented by the research group Modeling of User Behavior in Dynamic Systems (MoDyS) of the Berlin University of Technology, that displays the interaction of humans with a test. The engaged D2-Drive test refers to the D2 test of attention and concentration by Brinckenkamp (2001) The overall aim is to identify a pattern as correct according to Brinckenkamp's specification. Three versions of the D2-Drive were developed (Kiefer, Dzaack, Urbas 2005), whereas we used the version A. This version refers to (static) visual search of one specific pattern in a row (see picture 1). The aim is to identify the middle pattern and give the correct response (yes: is a D2-pattern, no: is not a D2-pattern). Pressing the response-button (yes or no) changes the represented pattern in the window and the process starts again.

The altered ACT-R model runs the D2-Drive test for a given time and changes to an idle mode to emulate non/less demanding cognitive work. The running-time was 60 seconds with different non-idle and idle times (non-idle/idle: 30/30, 45/15 and 55/5). The subjects of the experiment had to do the same as can be seen later.

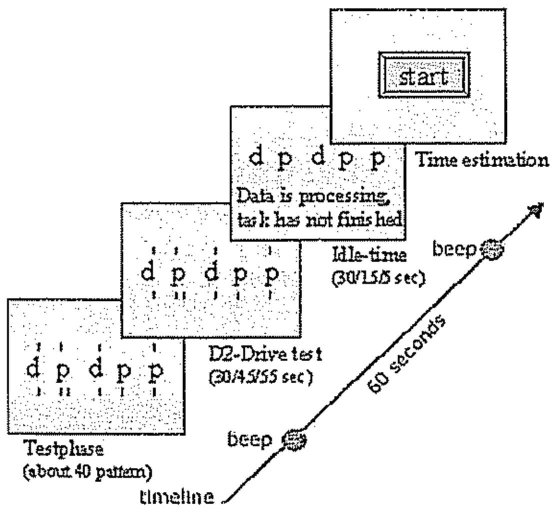
Results

The ACT-R model with and without the integrated timer-module does not show any differences in performing the task. This was measured by the given responses of identifying the pattern (i.e., while doing the non-idle task). Thus we conclude that the timer-module does not affect the cognitive workload of the model.

Running the ACT-R model with the timer-module shows the prospective estimation of time-duration as we anticipate based on literature: the longer the non-idle-time of the trial the longer should be the estimated time by the ACT-R model (i.e., the greater the amount of contextual changes the longer the estimated time-duration). The case with 55 seconds non-idle-time shows the longest estimated time of the model. The case with 30 seconds idle-time shows the shortest estimated time of the model. Thus we conclude that the timer-module is a suitable instrument for prospective time-duration estimation that shows the effects of prospective time-duration estimation as found and described in literature.

EMPIRICAL EVALUATION

To show the relevance of the models behavior to human performance we conducted an experiment. Subjects had to fulfill a simple task in a fixed time interval (60 seconds). The question was whether the time judgment would depend in the same way on the amount of produced contextual changes as the ACT-R simulation does. Thirty-one participants (twelve female, nineteen male) took part in the study in exchange for a donut. The sample covers students and graduates of the Berlin University of Technology. The participants were naïve to the purpose of the experiment and the relevance to time.



Picture 2: Experimental design

The subjects were instructed to complete version A of the D2-Drive test (as seen above). They were instructed to do the task as fast and correct as possible. After a training-phase the D2-Drive test started with a acoustic signal. After 30, 45 or 55 seconds depending on the condition of the observed paradigm the task stopped and the subjects were informed that data was processed and that the task had not finished. After the total of 60 seconds (non-idle time plus idle-time) the next acoustic signal finished this task period. Subsequently the subjects were asked to reproduce the passed time between the two acoustic signals. Therefore they should press a button and wait for the next button-press until they thought that the same amount of time had passed by (for all see picture 2). This way of reproducing the estimated time-duration was user because it had been suggested (McKay, 1977) that this measurement technique was more sensitive than verbal expression.

Results

Overall the time-duration estimations in the empirical setting are much shorter than the real time task duration of 60 seconds. The mean time-duration estimation for all subjects is approximately 42 seconds with a standard deviation (SD) of 21 seconds. The first condition with 30 seconds D2-Drive task and 30 seconds idle time was rated with a mean time-duration estimation of 45,6 seconds (SD 22,7 seconds). The condition with 45 seconds D2-Drive and 15 seconds idle time was estimated 49,5 seconds (SD 23,7 seconds). The results show that long idle time (30 or 15 seconds) seem to lead to considerable variances of time-duration estimation. The third condition with 55 seconds D2-Drive and 5 seconds idle time shows similar results in comparison to McClain (1983); estimated time-duration is 30,4 seconds (SD 9,8 seconds). The more pattern were judged in the given time the longer was the estimation of time-duration.

Observing the gained data from the case (55/5 seconds) we detected a correlation (R) between the results of the D2-Drive test (measured by the pressed buttons) and the estimated time-duration (R=0,72). That allows us to assume a connection between the processed information and the estimated time (i.e., contextual changes).

In the two other cases (30/30 and 45/15 seconds) no correlation could be found. In our opinion the long idle-time has implications on the estimation of time and is independent of the observed effects. Another explanation is that subject could change to an cognitive-demanding internal task (e.g., occupied by problem-solving) what we could not prevent with the experimental setting. Thus we claim to concentrate on the 55/5 seconds condition to investigate the correctness of the timer-module of ACT-R v.

DISCUSSION (ACT-R VS. REALITY)

OUTLOOK

To further examination of the timer-module some follow-up experiments are planed as well as the modification of the underlying ACT-R model. And if necessary the alteration of the timer-module. As discussed a long pooled idle-time period affects the estimation of time duration. To observe this issue of long idle times and the concerning estimation of time-duration we plan a first study with interleaved idle-time during the task. To investigate retrospective time-duration estimation and explicit setting of reference-points by humans we plan to integrate the version B of the D2-Drive in our experimental setting as a second study. In this study subjects have to fulfill five patterns arranged in a row and after completing this task the window changes to another view with new pattern – and the task starts again (Kiefer, Dzaack, Urbas 2005). We conclude this window-change as a distinctive event that allows us to assume to be a reference-point.

Although we think about changes concerning the introduced timer-module. The smallest entity for the estimation of time-duration in the timer-module are action units. At the current state of the timer-module these action units are formed through the productions fired. An interesting approach would be to substitute more than one production to an action unit. That allows to combine coherent productions to one action unit.

The next step in our research group is the integration of prospective time-duration estimation and its experimental prove. We just started investigating this issue.

The combination of retrospective and prospective time-duration estimation may be a promising approach. Both should be integrated in cognitive architectures. This opens a wide range of new applications in the field of designing dynamic human-machine-systems and in the field of research. We think that time-duration estimation in both characteristics – prospective and retrospective – gives new

answers to questions concerned with human behavior in the real world as well in the new virtual world.

ACKNOWLEDGEMENTS

The order of authorship was determined by alphabet.

REFERENCES

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

Block, R. A. & Zakay, D. (1996). Models of Psychological Time Revisited. In H. Helfrich (Eds.), Time and Mind. Proceedings of the International Symposium on Time and Mind held in Dec. 1994 at the University of Regensburg. (pp. 171-195). Seattle; Toronto; Göttingen; Bern: Hogrefe & Huber Publishers.

Bricnkenkamp, R. (2001). Test d2 Aufmerksamkeits-Belastungs-Test. 9., überarbeitetet und neu normierte Auflage. Hogrefe Verlage. Bern, Schweiz.

Hicks, R. E., Miller, G.W., & Kinsbourne, M. Prospective and retrospective judgments of time as a function of amount of information processed. *American Journal of Psychology,* 1976, 89, 719-730.

Kiefer, J. Dzaack, J. & Urbas, L. (2005) To interrupt and to assume : First Approach in ACT-R. This volume. ACT-R Workshop 2005, Trieste.

McClain, L. (1983) Interval estimation: Effect of processing demands on prospective and retrospective reports. Perception and Psychophysics, 34(2) 185-189.

McKay, T.D. Time estimation: Effects of attentional focus and a comparison of interval conditions. *Perceptual and Motor Skills,* 1977, 45, 584,586.

Ornstein, R.E. *On the experience of time. Middlesex,* England: Penguin, 1969.

Schulze-Kissing, D., van der Meer, E. & Urbas, L. (2004). A Psychological Analysis of Temporal Errors in Human-Machine-Systems. Proceedings of the IFAC Symposium: Analysis, Design and Evaluation of Human-Machine-Systems. Atlanta, USA, 07.-09. September 2004.

Is It a Boy or a Girl?

Angela Brunstein (angela.brunstein@phil.tu-chemnitz.de)
Department of Psychology, Chemnitz University of Technology
D-09107 Chemnitz, Germany

Maria Pilar Larrañaga (Maria.Larranaga@uwe.ac.uk)
School of Languages, Linguistics and Area Studies, Coldharbour Lane
BS16 1QY Bristol, GB

Abstract

Grammatical gender is early learned by children acquiring a language such as Spanish, but causes tremendous difficulties to L2 learners learning it with or without class-room instruction. The natural gender of referents can be a useful cue for choosing the correct articles given no other information as shown in previous studies with adult English speaking L2 learners of Spanish. The present study on natural gender ratings for name picture pairs as a nonlinguistic domain revealed comparable results. An ACT-R model was developed in order to investigate in detail how participants come up with their ratings identifying four strategies. This model was also used for deducing hypotheses for a further study which aims at collecting eye movement data. Altogether, results of both studies and heuristics implemented in the model demonstrate that at least adult L2 learners apply general, not language specific strategies for learning an unfamiliar phenomenon of a foreign language.

Natural versus Grammatical Gender

Children acquire the grammatical notion of gender at an early stage in development, and more importantly, well before they have a stable concept of natural gender (Karmiloff-Smith, 1979, for French; Pérez, 1990, for Spanish). Thus, Spanish and French children successfully use phonological cues at a very early stage in order to allocate the correct article irrespective of language external cues, such as the semantic category of the corresponding referent (Pérez, 1990). Results like these imply independence of language and other cognitive modules at least as far as gender for L1 learners is concerned. This is very much in line with the claim that gender is not a semantic phenomenon in languages like Spanish where rivers and mountains are masculine, whereas motorcycles are feminine.

As opposed to children who acquire their mother tongue, adult L2 learners often encounter difficulties when assigning gender to unknown nouns, especially when their L1 lacks grammatical gender (Franceschina, 2001). In that case, several strategies might be used by the L2 learner: skipping the article, guessing, transfer from L1 to L2, if the mother tongue has gender. In addition, natural gender of referents is probably a strong cue that enables the learner to choose the correct article. Larrañaga (2005) found that English L2 learners of Spanish more often chose the correct article for nonsense nouns when the natural gender of a pictured referent matched with the grammatical gender of the corresponding noun than without that correlation. In addition, their answers were biased to choose the male article more often and to prefer systematically nouns instead of pictures (see also Franceschina, 2001).

The present study investigated whether or not this is a specific strategy for language learning. Instead of having to allocate the articles, participants rated the natural gender of name picture pairs for unknown characters of the Harry Potter story. In addition, an ACT-R model was written to model results of both studies.

Method

Participants 179 participants (49% male, $M = 24$ years, $SE = 5$ 4 years) participated in this web experiment (www tu-chemnitz de/project/elearning/Potter_engl) All of them were familiar with main characters of the Harry Potter story but not with its secondary characters.

Material and Procedure Combinations of 24 non frequent male (*m_*) and female names (*f_*) and 72 pictures of male (*_m*), female (*_f*) and non identifiable secondary characters (*_n*) of the Harry Potter story were chosen for this study These characters are mentioned within the first five books of the story less than five times on the whole and are not mentioned by name in the films For non identifiable pictures, characters are either animals or they are shown from an unfavorable perspective. As fillers 18 female and male main characters were presented as converging name picture pairs (male - male and female - female combinations). In addition to the *names and pictures* condition (42 items per participant), 2 control conditions with *names only* (42 items) or *pictures only* (90 items) were presented. For the *names and pictures* condition altogether three lists were constructed for balancing male, non identifiable and female pictures per name between participants. Ratings and answering times were collected as dependent variables.

The study was conducted via internet. Participants filled in a questionnaire on general information first and rated afterwards the presented characters. They were instructed to demonstrate their familiarity with the magic world of Harry Potter in the prompt by rating characters presented by name and picture as male or female respectively. The complete study took about 10 minutes.

Results

As expected, participants rated converging name picture pairs more often as male (*mm*) or female (*ff*) respectively than not converging pairs (*mf* and *fm*; see Table 1), $F(2, 44) = 7.19$, $p < .001$, $\eta^2 = .75$. In addition, participants preferred systematically the "male" answer resulting in more "male" ratings for non identifiable pictures combined with male names (*mn*) than "female" ratings for non identifiable pictures combined with female names (*fn*) (see Table 1). For not converging pairs, pictorial information was systematically favored over verbal information. This effect was stronger for pictured male characters than for pictured female counterparts (see Table 1).

For identifying participants' strategies for rating the natural gender of characters, answers and answering times were compared to both control conditions. Answering times for rating *names only* ($M_N = 1827$ ms, $SE_N = 421$ ms) were shorter than answering times for rating *names and pictures* ($M_{NS} = 2523$ ms, $SE_{NS} = 726$ ms) (see Figure 1), $F(3, 172) = 24.29$, $p < .001$, $\eta^2 = .298$, indicating that participants integrated both sources of information into their rating. Answers did not differ for names in the *names only* condition ($M_{m\_} = .91$, $SE_{m\_} = .05$; $M_{f\_} = .22$, $SE_{f\_} = .27$) and in the *names and pictures* condition ($M_{m\_} = .84$, $SE_{m\_} = .29$; $M_{f\_} = .27$, $SE_{f\_} = .35$), $F(3, 172) = 0.16$, $\eta^2 = .00$.

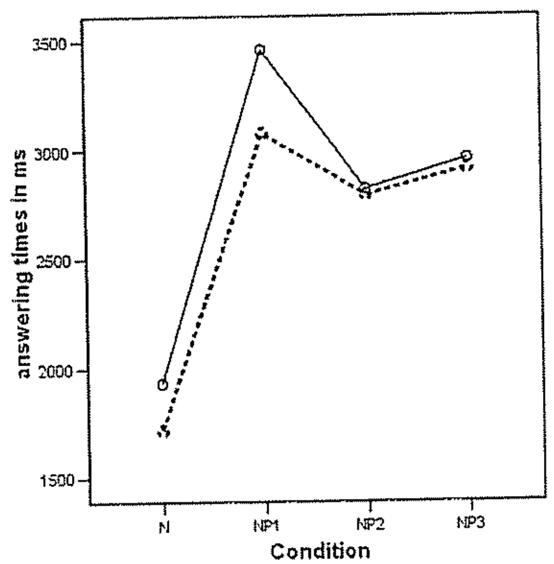


Figure 1. Answering times for male (line) and female names (dotted line) for *names only* (N) and *names and pictures* conditions (NP). Altogether three *names and pictures* lists (NP1, NP2, NP3) were presented for counterbalancing male, non identifiable and female pictures per name between participants.

In contrast to the comparison between *names only* and *names and pictures* conditions, answering times for rating *pictures only* ($M_S = 3451$ ms, $SE_S = 2111$ ms) were comparable to the *names and pictures* condition, $F(3, 221) = 1.86$, $p = .14$, $\eta^2 = .03$, with a very small trend toward shorter times for the name picture pairs. For that condition, names probably acted as a trigger for processing pictures.

Answers for male, non identifiable and female pictures did not differ for *pictures only* ($M_{\_m} = .98$, $SE_{\_m} = .06$; $M_{\_n\_} = .65$, $SE_{f\_} = .25$; $M_{f\_} = .08$, $SE_{f\_} = .16$) and *names and pictures* conditions ($M_{\_m} = .93$, $SE_{\_m} = .13$; $M_{\_n} = .59$, $SE_{\_n} = .35$; $M_{\_f} = .07$, $SE_{\_f} = .11$), $F(3, 221) = 2.05$, $p = .11$, $\eta^2 = .03$.

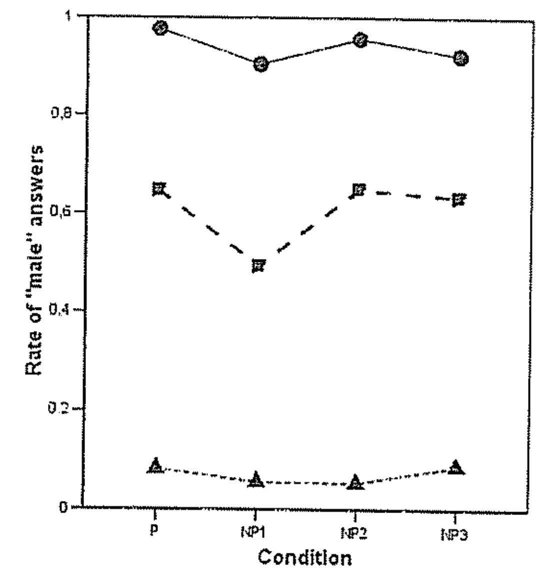


Figure 2. Rates of "male" answers for male (circle), non identifiable (square) and female (triangle) pictures for the *pictures only* (P) and for three lists of the *names and pictures* conditions (NP1, NP2, NP3).

The Model

For both studies, results indicate that participants integrate pieces of information from both names or nouns and pictures in order to allocate the correct gender. This follows from the different answering times for *names only* and *names and pictures* conditions and from the trend towards different answering times for *pictures only* and *names and pictures* conditions. In addition, performance was higher for converging pairs than for not converging pairs or pairs with non identifiable pictures.

In order to investigate in some detail how participants decided upon male of female ratings for name/noun picture pairs, an ACT-R model was developed.

Presented Stimuli As in the Potter study, one third of the stimuli were converging pairs, one third not converging pairs and the remaining third pairs had non identifiable pictures. In contrast to the Potter study, no fillers were presented. Moreover, letters were presented instead of names and numbers instead of pictures, because we were more interested in decision strategies than in encoding of pictures and names. Gender rating were taken from a pilot study and were implemented as declarative knowledge for names and pictures respectively. Male names and pictures were rated by 90 to 100 percent of participants as male in the pilot study. Female names and pictures were rated by zero to ten percent of the participants as male. Non identifiable pictures were rated by 40 to 60 percent of participants as male and were not implemented as declarative knowledge in the model. Correspondingly, non identifiable pictures resulted in retrieval errors during processing.

Processing name picture pairs For doing one trial, the model processed letters (instead of names) and /or numbers (instead of pictures) first. Secondly, gender information was retrieved if available. Thirdly, the model either chose one gender cue or realized conflicting information. Forth, it either answered corresponding to chosen gender, pressing the "m" or "f" button or resolved the conflict by preferring one of the given sources. Finally, it answered corresponding to chosen gender after conflict resolution.

Decision strategies By principle, participants could process either no, one or both sources of information. In addition, they could either know the gender of no, one or both stimuli. According to the above, four strategies are possible: *guessing* involving ignoring both sources of information, *preferring names* as seen for the Spanish study, *preferring pictures* as seen for the Potter study, or *integrating* both sources of information as seen for both studies. For *integrating* both sources of information, four cases are possible: pairs with two unknown stimuli resulting in guessing, pairs with one unknown stimulus resulting in preferring the known stimulus, converging pairs, and not converging pairs resulting in conflict resolution.

For *guessing*, all combinations would result in male or female ratings by chance. This strategy was not very likely to happen and was, therefore, restricted to cases of unknown names and non identifiable pictures in the model.

For *preferring pictures* and for *preferring names*, the model processed the preferred source of information first and the other source thereafter, only if the preferred one was unknown or not identifiable. In that case, the model preferred the previously not preferred source of information. If both stimuli were unknown to the model, it guessed the gender. According to this strategy, answering times and "rate of "male" answers should be equal for known converging and known not converging pairs. For unknown or non identifiable stimuli, answering times are predicted to increase and rates of "male" answer should be equal to the other preferring strategy.

For *integrating* both sources of information, performance and answering times should be highest compared to *guessing* and *preferring* either *names* or *pictures*. Within that strategy performance should be highest and answering times should be lowest for known converging pairs. Performance and answering times should fall slightly below in pairs with unknown or non identifiable stimuli. Lowest performance and highest answering times should be observed for not converging pairs because of conflict resolution.

For the first version of the model, we began with a strong preference of the *integrating* strategy as indicated by results of both studies. *Preferring names* or *pictures* were used only for pairs with one unknown or non identifiable stimulus. *Guessing* was used only for two unknown or non identifiable stimuli within one name picture pair. Because of presentation of letters and numbers instead of names and pictures, we focused on the performance instead of answering times. As to enable guessing, sub symbolic processing was enabled. No other parameters were adapted to the task.

Results Table 1 shows "male" rates for all six conditions for participants in the Potter study and for the model. For about 10 runs of the experiment, the model overestimates participants' performance for converging pairs slightly and underestimates their performance for the other conditions slightly. However, the correlation between models' and participants' performance is relatively high, $r = .987$.

Table 1: Natural gender ratings (percentage of "male" ratings) by participants and the corresponding ACT-R model for male (*m_*) and female names (*f_*) combined with male (*_m*), female (*_f*) and non identifiable (*_n*) pictures

	mm	mn	fm	mf	fn	ff
Participants	0.98	0.89	0.83	0.33	0.34	0.05
Model	1.00	0.75	0.70	0.28	0.23	0.00

In contrast to matching performance in respect to given answers, the model is far from producing answering times that are comparable to the participants at present, $r = -.23$.

Nevertheless, the actual version of the model has allowed us to draw conclusions and hypothesize on participants' answering behavior in a preliminary study that is been conducted at present with the aim of collecting eye movement data:

We expect that participants demonstrate either one of the *preferring* strategies or the *integrating* strategy for most of performed trials. Actual strategy should be identifiable using the converging pairs and fillers.

Participants identified as guessing should be extremely seldom within this study. They should demonstrate comparable low performance and short processing times for all trials.

Participants identified as using the *preferring pictures* strategy are expected to process pictures only ignoring names for all known and identifiable pictures. For these cases, answering times should be equal and comparably low. Answers should be given according to pictures exclusively. Processing of not converging pairs should not differ from converging pairs as long as pictures are known. It is predicted for non identifiable pictures these participants can either guess the actual gender of characters or, alternatively, process the name presented instead and estimate the gender according to the cues given by the names. According to the actual version of the model, preferring names for non identifiable pictures are more likely than guessing.

Participants identified as *preferring names* should demonstrate answering behavior corresponding to participants identified as *preferring pictures* with processing names only and processing pictures or guessing instead for unknown names only. This strategy is expected to occur more seldom than *preferring pictures* for the actually conducted Potter study.

Participants identified as *integrating* both sources of information should process both names and pictures for almost all pairs. For unknown or non identifiable stimuli, processing times should be enhanced compared to converging pairs. At the same time, performance is likely lower for these pairs. For not converging pairs, performance is expected to be poorer than with converging pairs. Both processing and answering times will probably improve for these pairs as compared to converging pairs. Eye movement pattern should shed light on how in detail these conflicts are resolved.

This study is being conducted with about 10 participants and one list of picture name pairs.

Discussion

Comparable to Larrañaga's (2005) results on grammatical gender, participants in the present study used both verbal and pictorial sources of information for judging the natural gender of persons. In addition, they demonstrated a strong preference for one of these sources when judging non-converging combinations, opting primarily for "male". These results support the claim that adult L2 learners resort to language-unspecific strategies for language production in the absence of relevant linguistic cues.
A corresponding ACT-R model fitted data out of the Potter study in respect to given answers, but not in respect to answering times. This missing fit can be explained primary by the fact that the task involved processing letters and numbers instead of names and pictures. In addition, the implemented conflict resolution process does not through satisfactory results yet. A plausible explanation is that participants check firstly whether conflicting gender is caused by encoding or retrieval errors. As a result, stimuli must be reread and processed once again, what would explain the higher answering times. Alternatively, they may rate the relevance of presented stimuli in order to decide which information to opt for. Finally, strategy management is very restrictive in the present version. For getting the model closer to participants' answering behavior, an increase of the probability for preferring either names or pictures should be implemented. Finally, for the actual version it is not possible to reduce processing times for pictures by names processed before. Nevertheless, implemented strategies allow hypotheses on answering behavior in a actually conducted study collecting eye-movement data.

To sum up, the present version of our model allowed us to discover more in detail which strategies participants use for rating natural gender of characters presented by name and picture and for rating grammatical gender presented by noun and picture. In addition to this, it worked as a "translator" between both domains of Spanish gender and natural gender ratings highlighting the communalities and differences of both domains. And finally, it functioned as a starting point for further research on the relationship between language and other cognition.

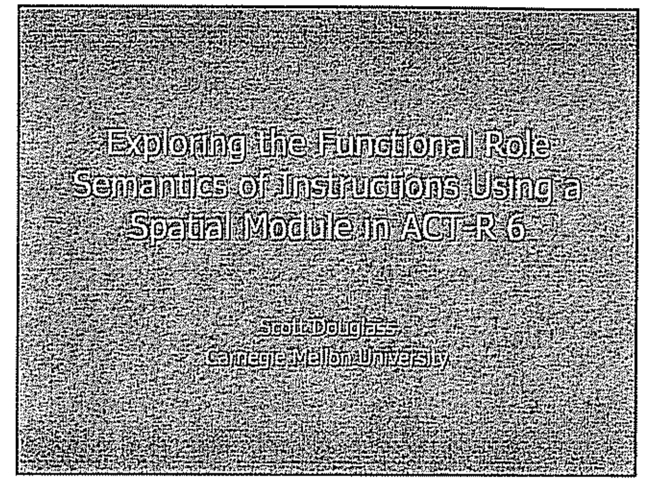
References

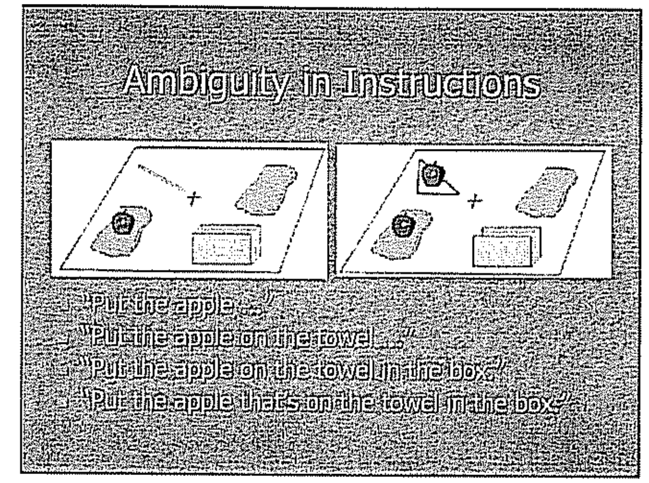
Larrañaga. M. P. (2005). *English native speakers acquiring Spanish and French gender* Manuscript in preparation.

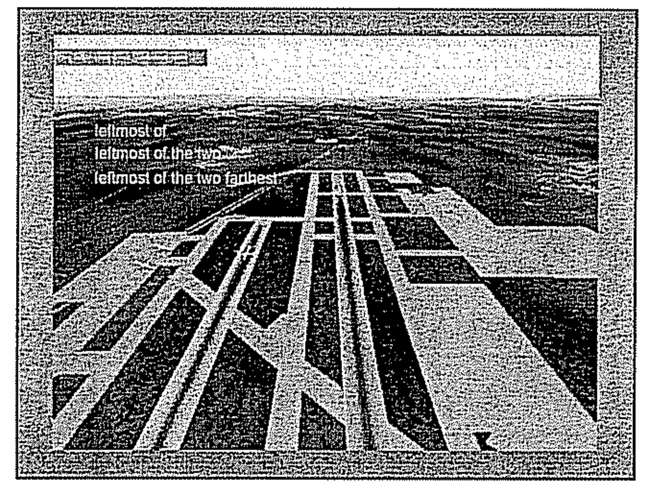
Francheschina, F. (2001). Morphological or syntactic deficits in near-native speakers? An assessment of some current proposals. *Second Language Research, 17*, 213-247.

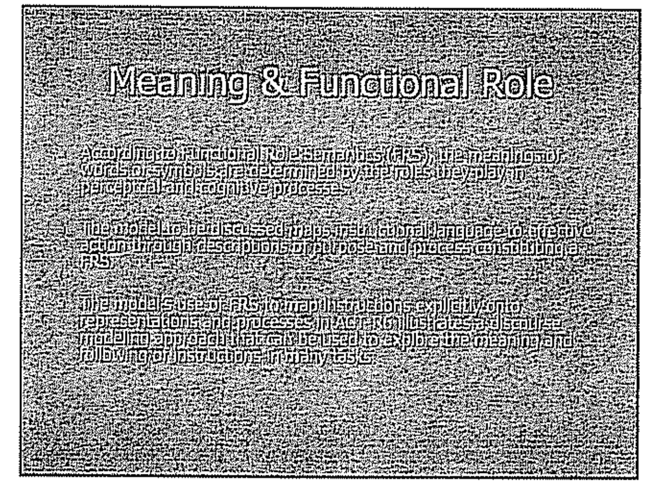
Karmiloff-Smith, A. (1979), *A functional approach to child language A study of determiners and reference* Cambridge: Cambridge University Press.

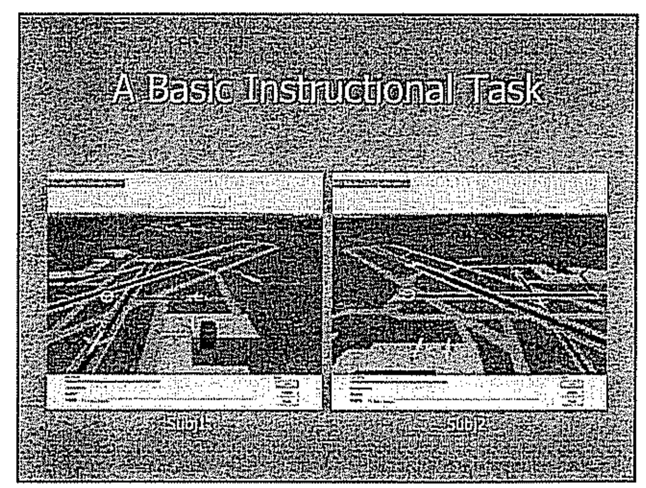
Pérez, M. (1990). ¿Cómo determinan los niños la concordancia de género? Refutación de la teoría de género natural *Infancia y Aprendizaje, 50*, 73-91.

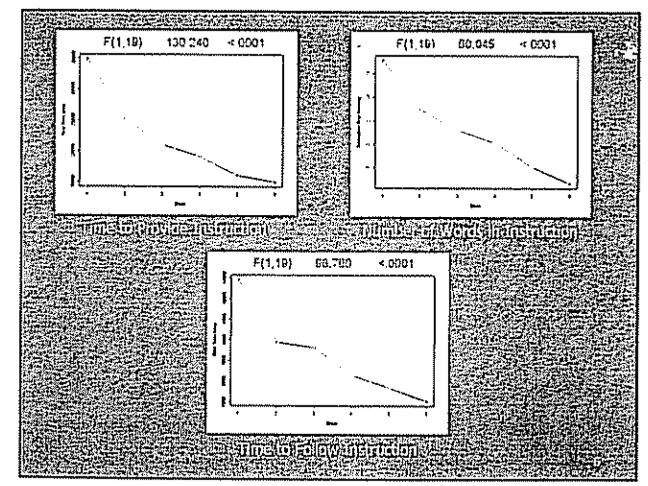




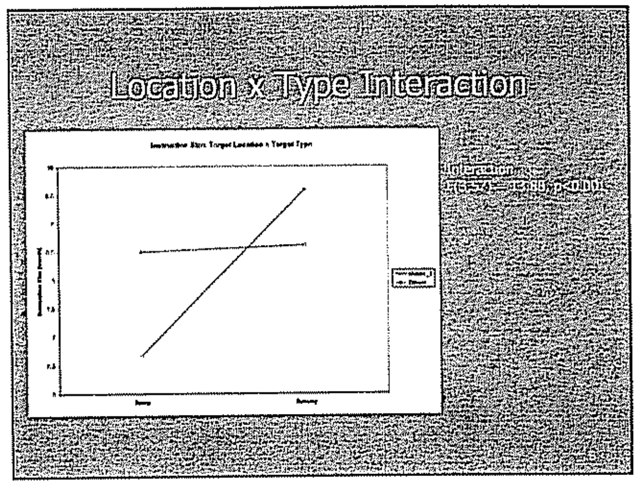


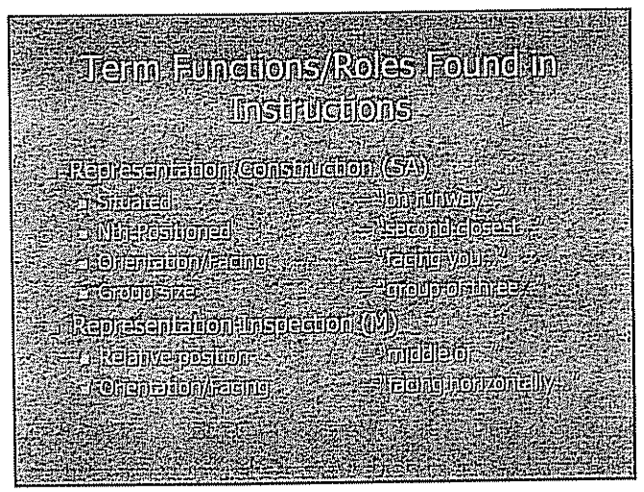


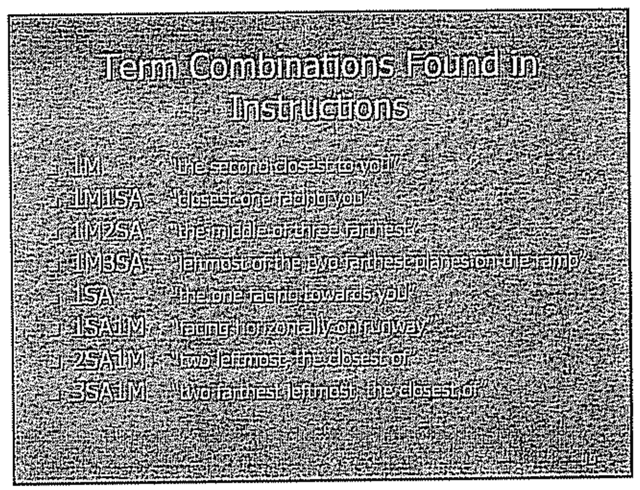


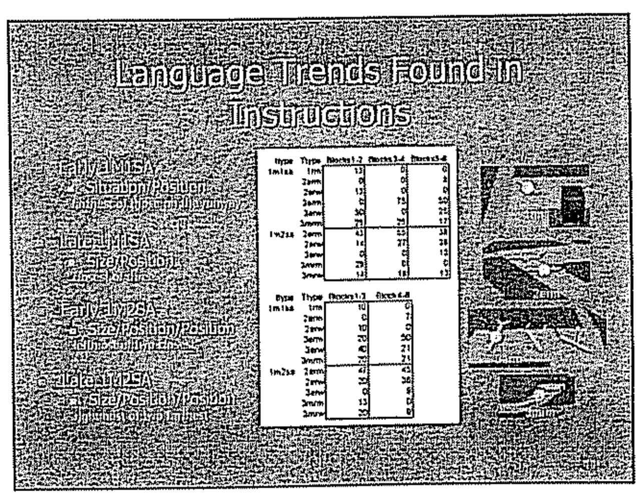


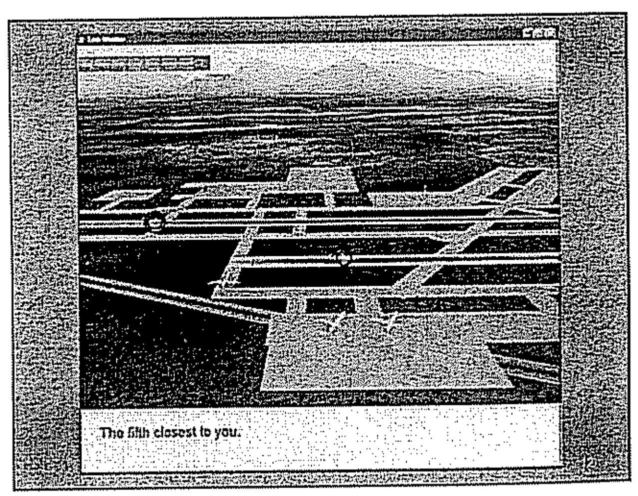

Target Airplane Locations
End

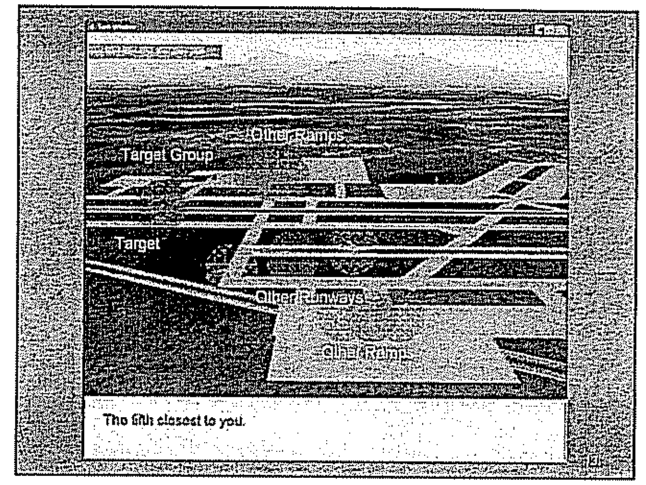

Location x Type Interaction

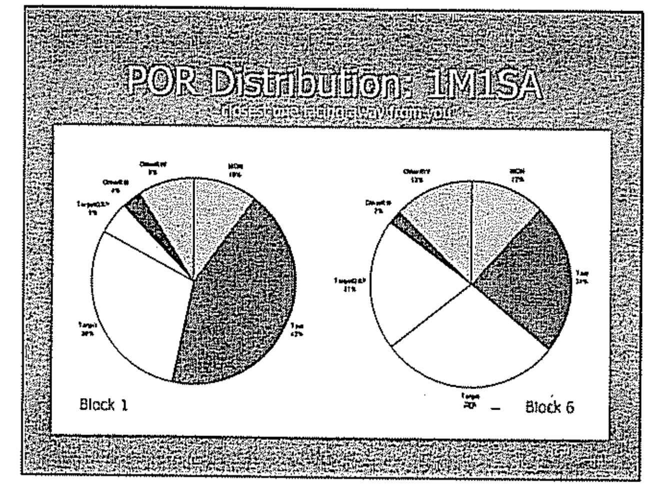

Term Functions/Roles Found in Instructions
Representation Construction (SA)
Situated — "on runway"
Nth Positioned — "second closest"
Orientation/Facing — "facing you"
Group Size — "group of three"
Representation Inspection (M)
Relative position — "middle of"
Orientation/Facing — "facing horizontally"

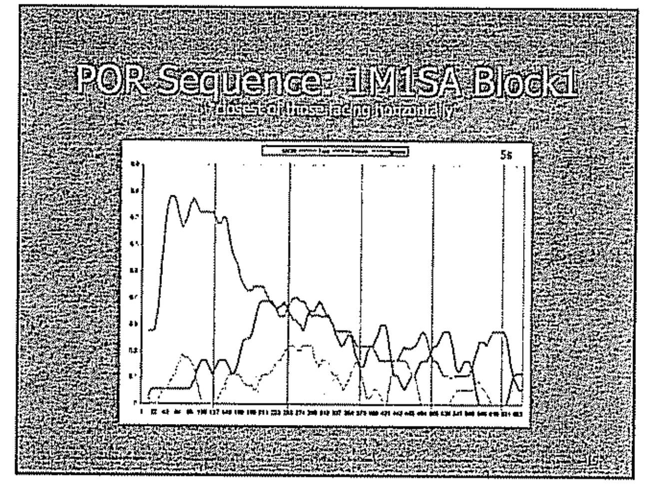

Term Combinations Found in Instructions
1M "the second closest to you"
1M1SA "closest one facing you"
1M2SA "the middle of three farthest"
1M3SA "leftmost of the two farthest planes on the ramp"
1SA "the one facing towards you"
1SA1M "facing horizontally on runway"
2SA1M "two leftmost, the closest of"
3SA1M "two farthest leftmost, the closest of"

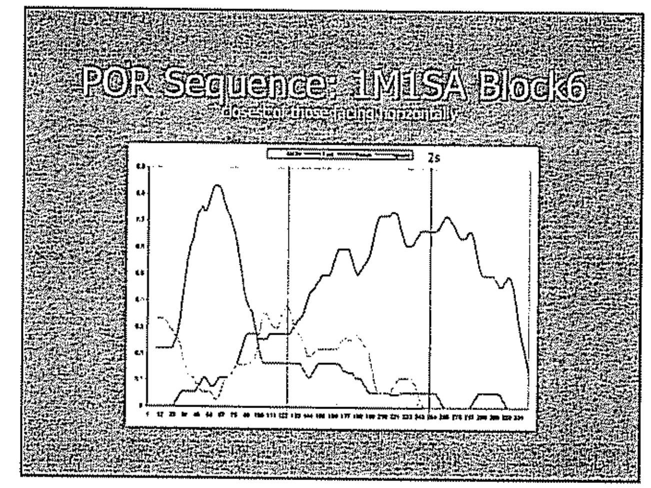

Language Trends Found in Instructions

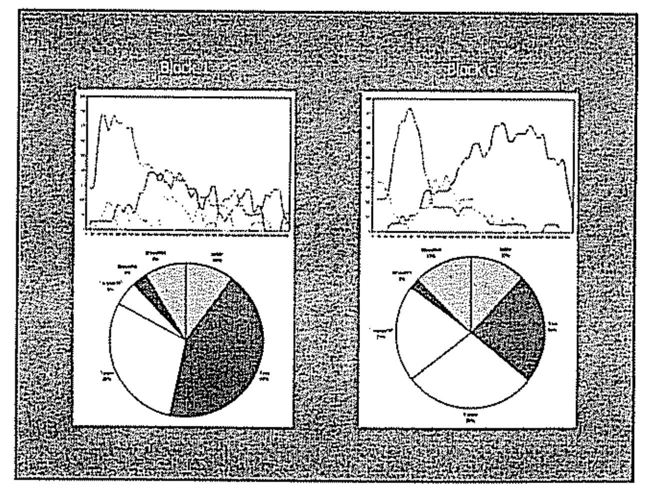

The fifth closest to you.

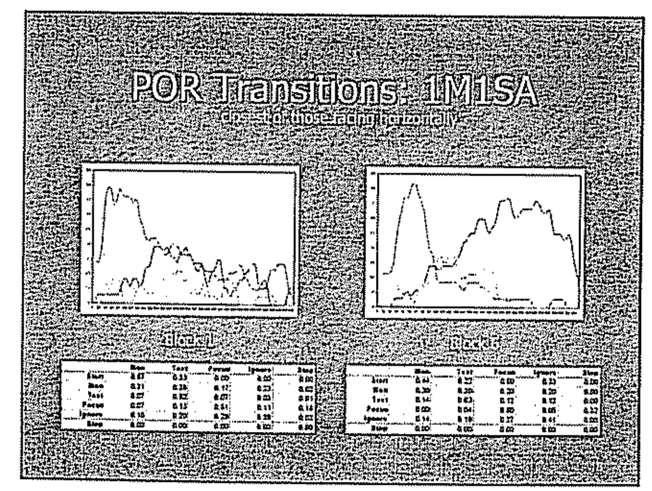

Target Group
Other Ramps
Target
Other Runways
Other Ramps
The fifth closest to you.


POR Distribution: 1M1SA
Block 1
Block 6


POR Sequence: 1M1SA Block1
"closest of those facing horizontally"
5s


POR Sequence: 1M1SA Block6
closest of those facing horizontally
2s




POR Transitions: 1M1SA
closest of those facing horizontally

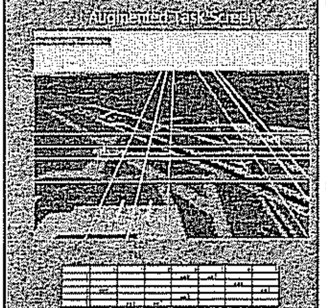
What the Model Does

- Reads the description (linguistic structure)
- Discerns the meaning of the utterance (intentional structure)
  - Uses references and spatial primitives
- Executes a scanning strategy (attentional structure)
  - Sequence of actions reflects the communicated intentional structure
  - Actions may depend on a spatial representation
- Identifies the described airplane

Discourse Module

- Purpose
  - Allow ACT-R models to recognize the intentions within discourse segments describing spatial references
- Basis
  - *Dialog grammars* & plan recognition ([illegible])
  - *Hierarchy* in discourse (Grosz & Sidner)
  - Multiple dialog threads ([illegible])
- Provides
  - Intention & propositional representational buffers
- Operations
  - Discourse segmentation: tracks utterance
  - Utterance interpretation: focus on action

Processing With Spatial Arrays



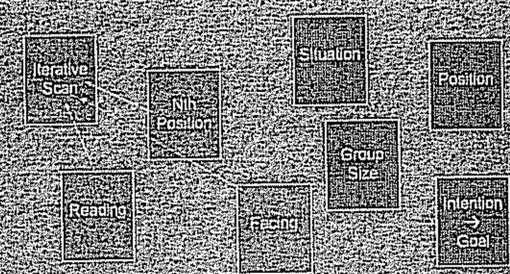
Combining Knowledge & Context

```
(f0) isa filter-fn relation facing
     property south fn SA-towardsPOV?)

(p manage-position-filter-SA
  =goal>
    ISA          position-goal
    state        execute-filter-fn
  =retrieval>
    ISA          filter-fn
    fn           =P
  ?spatial>
    state        free
==>
  +spatial>
    ISA          filter
    P            =P
  +intentional>
    ISA          focus-on
    info-type    next-action
)

(p manage-position-filter-SA
  =goal>
    ISA          position-goal
    state        execute-filter-fn
  =retrieval>
    ISA          filter-fn
    fn           SA-towardsPOV?
  ?spatial>
    state        free
==>
  +spatial>
    ISA          filter
    P            SA-towardsPOV?
  +intentional>
    ISA          focus-on
    info-type    next-action
)
```

Intention to Goal

```
(p first-action-to-intention
  =goal>
    ISA          goal
    state        visually-search
  ?intentional>
    buffer       empty
    enqueued     actions
==>
  +intentional>
    ISA          focus-on
    info-type    next-action
)

(p intention-to-goal
  =intentional>
    ISA          ipredicate
==>
  +goal>         =intentional
)

(p manage-size
  =goal>
    ISA          arity/1
    enacted      nil
    predicate    size
    arg1         =n
  ?visual>
    state        free
==>
  =goal>
    enacted      t
  +goal>
    ISA          size-goal
    state        request-group-VL
    group-size   =n
)
```

"Purpose-Based" Rule Groups



Combining Knowledge & Context

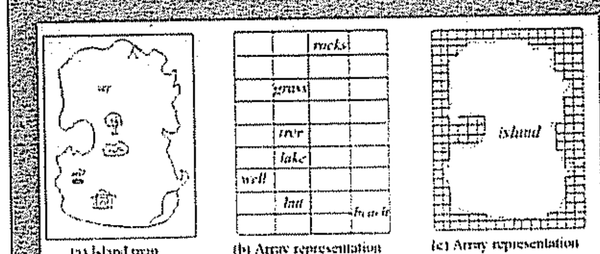
```
(s0) isa scan-fn relation closest
     axis screen-y progression highest)

(p* iterative-scan-nth-locate
  =retrieval>
    ISA          scan-fn
    axis         =axis
    progression  =constraint
  =goal>
    ISA          enumerate-goal
    state        request-VL
    cv           =n
  !safe-eval! (not (zerop =n))
  ?visual>
    state        free
==>
  +visual-location>
    ISA          visual-location
    kind         sp
    =axis        =constraint
    :attended    nil
  +retrieval>
    ISA          decrement-count
    current      =n
  =goal>
    state        counting-down
)

(p iterative-scan-nth-locate
  =retrieval>
    ISA          scan-fn
    axis         screen-y
    progression  highest
  =goal>
    ISA          enumerate-goal
    state        request-VL
    cv           3
  !safe-eval! (not (zerop 3))
  ?visual>
    state        free
==>
  +visual-location>
    ISA          visual-location
    kind         sp
    screen-y     highest
    :attended    nil
  +retrieval>
    ISA          decrement-count
    current      3
  =goal>
    state        counting-down
)
```

Acquired Context-Specific Procedural Skill

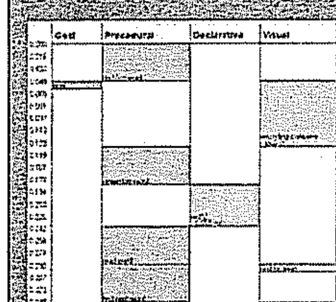
Spatial Module

- Purpose
  - Allow ACT-R models to create and maintain a qualitative geometrical spatial representation outside conventional declarative memory
- Basis
  - Representing spatial knowledge in *spatial arrays* (Glasgow & Papadias)
  - Model-based spatial reasoning (Johnson-Laird)
- Provides
  - Spatial (re)presentational buffer
- Operations
  - Construction/transformation: put, delete, filter, unfilter, focus, unfocus, orient, rotate
  - Inspection: user-specified inspection functions (leftmost, middle, left-of, etc.)

Representing With Spatial Arrays

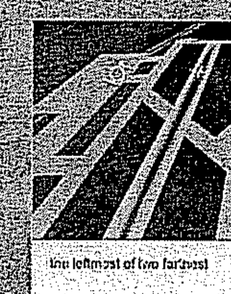


Reading the Instruction

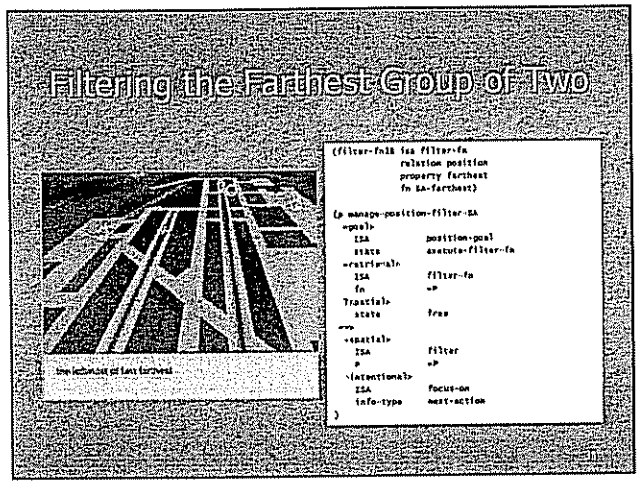


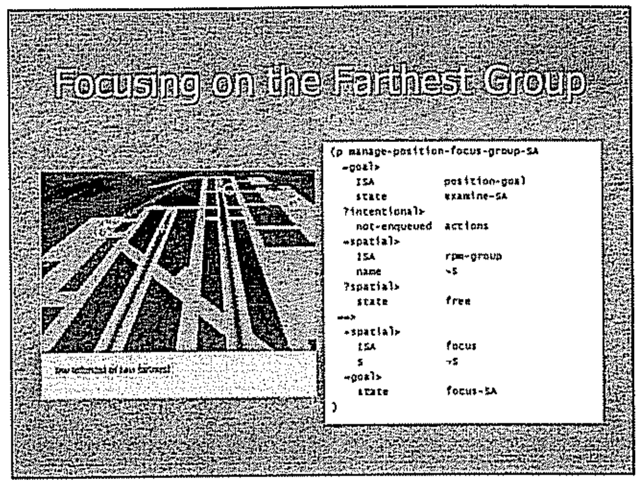

```
(p read-word
  =goal>
    ISA          goal
    state        remembering
  =retrieval>
    ISA          wd
    meaning      =word
==>
  =goal>
    state        find
  +visual-location>
    ISA          visual-location
    kind         text
    screen-x     greater-than-current
    nearest      current
  +intentional>
    ISA          track-utterance
    utterance    =word
)
```

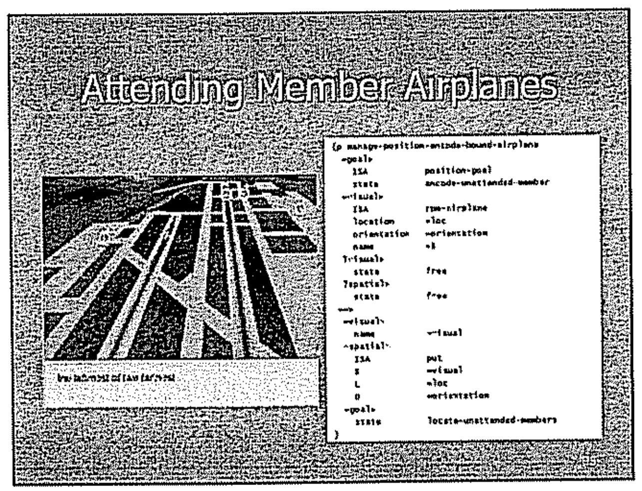
Attending to Groups of Two

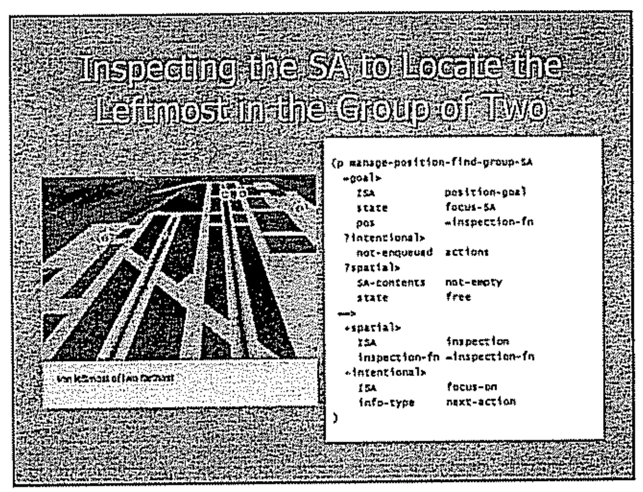


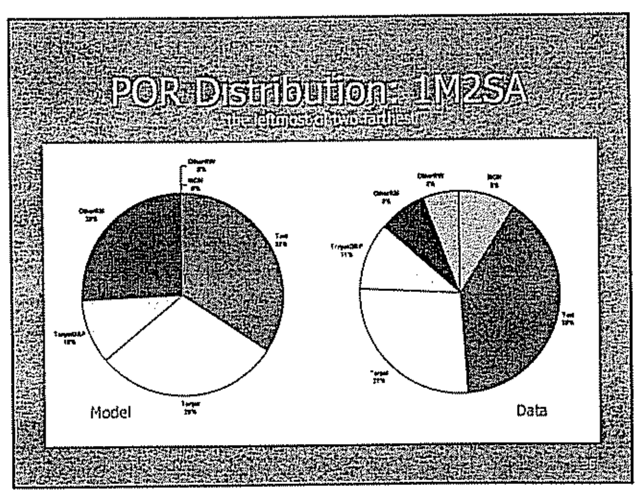

```
(p manage-size-encode-group
  =goal>
    ISA          size-goal
    state        encode-unattended-group
    group-size   =n
  =visual>
    ISA          rpm-group
    size         =n
    location     =loc
    orientation  =orientation
  ?spatial>
    state        free
==>
  =visual>
    name         =visual
  +spatial>
    ISA          put
    S            =visual
    L            =loc
    O            =orientation
  =goal>
    state        request-group-VL
)
```

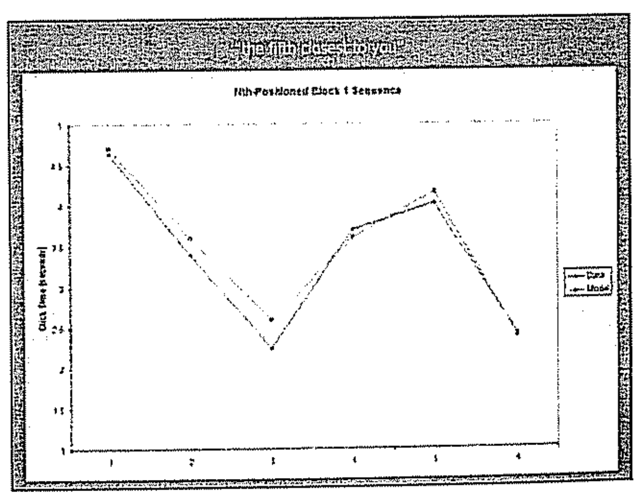












Learning From Instructions

David Huss, Niels Taatgen, and John Anderson

Understanding procedural learning is a continuing challenge. Previous work within ACT-R has produced the architecture's current methods of Production Rule Learning (PRL) and Production Rule Compilation (PRC). Through PRC and PRL ACT-R has been able to account for learning in tasks ranging from simple multitasking to air traffic control. Yet, both PRC and PRL rely on the modeler to provide a starting state in which sufficient task knowledge resides in the declarative and procedural systems. Currently, ACT-R provides little insight into how procedures are internalized from external sources such as instructions or display feedback.

In order to address this, our research examines learning on an aviation task. The task is autopilot navigation via the Flight Management System. We have created an accurate, computer-based simulation that is capable of interacting with both human and ACT-R subjects. In our empirical study involving CMU undergraduates, we manipulated the instructions provided to our participants. One group received theirs in a traditional list-based format while another was provided instructions inspired by recent work within the ACT-R architecture.

The traditional list-based instructions were taken directly from the United Airlines pilots' training manual. The opposing condition used environmental cues to reduce the need for internal control states and explained the results of a given action. Figure 1 provides examples of these instructions as well as a simplified view of how they may be represented.

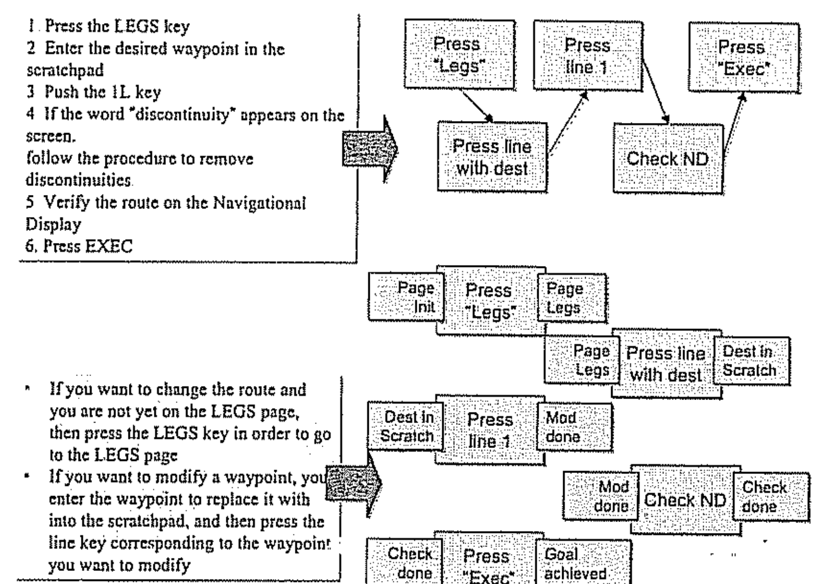


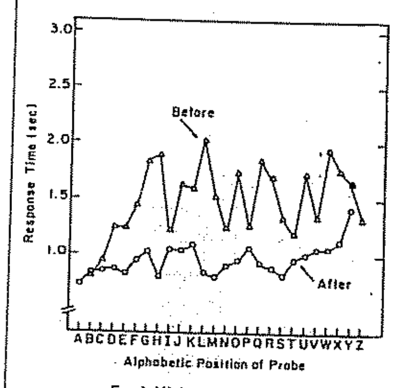
Figure 1: Examples of instructions

From our study we have revealed that, not only is performance worse in the list condition, but the ACT-R inspired condition demonstrates improved transference when tested on untrained procedures on the FMS. An ACT-R model is able to partially demonstrate this behavior. If steps are forgotten, the knowledge of productions' results allows for a process resembling a means-ends analysis. Yet this model fails to account for the significant within-trial learning exhibited by our participants. This has led us to develop a second model that will engage in the "trial and error" learning exhibited by our subjects.

Alphabetic Retrieval & Memory

Hedderik van Rijn
Artificial Intelligence, Groningen University

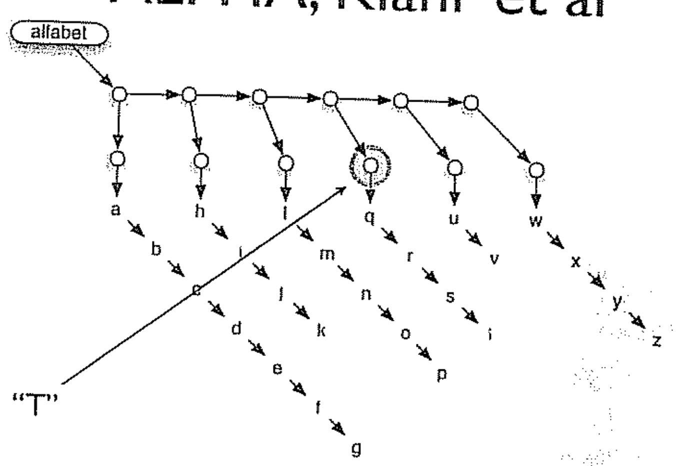
What are prototypical effects?



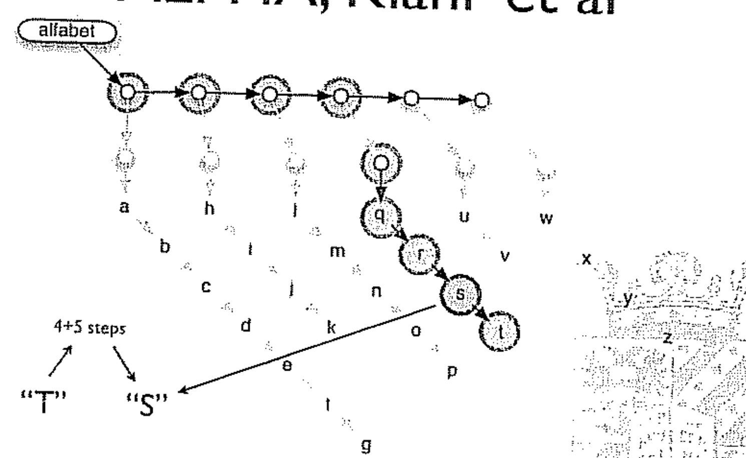
Exp1, Klahr, Chase, & Lovelace, 1983, JEP:LMC, 9(3)

- RTs:
  - Forward condition is faster than backward
  - Increased reaction times with alphabetic distance
  - Some letters are faster than others
- Interviews:
  - Sometimes parts of the alphabet have to be recited.

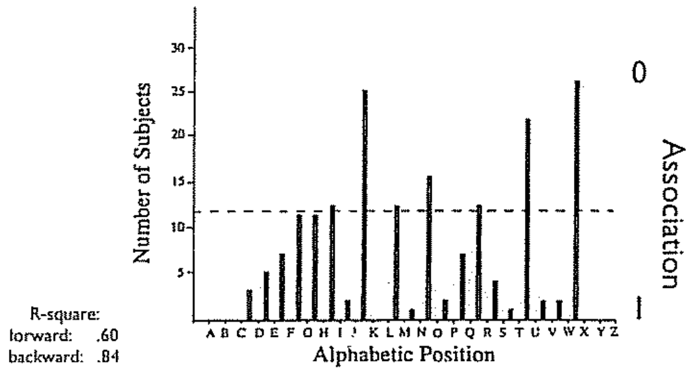
ALPHA, Klahr et al



ALPHA, Klahr et al



ASA Averaged Segmentations



ASA Conclusions

- Model *does* predict data better...
- ...but prediction is based on analysis of the to be predicted data.
- And, participants *do* consistently report using "alphabetical chunks".

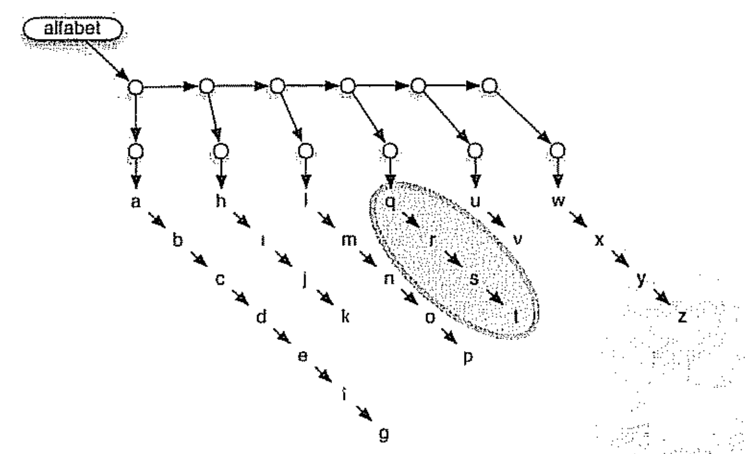
ALPHA + ASA

- Quite simple (from an ACT-R perspective) principle:

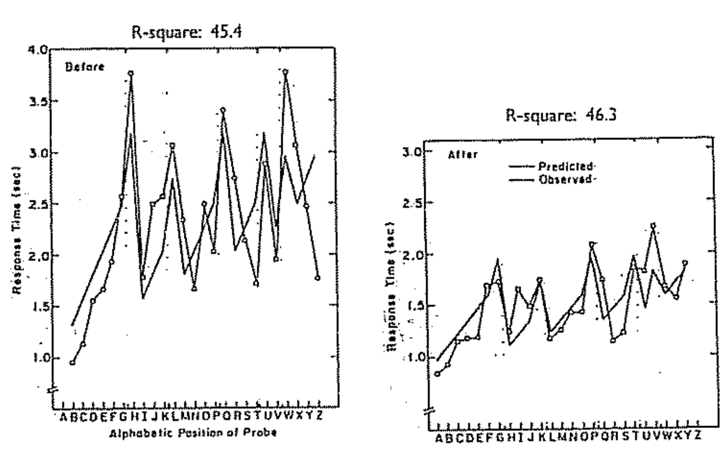
  Association based retrieval where possible,

  Strategy where necessary

Structure from ALPHA



ALPHA predictions



ALPHA Conclusions

- Reasonably good predictions (for 1982)
- Deterministic model
-

ASA

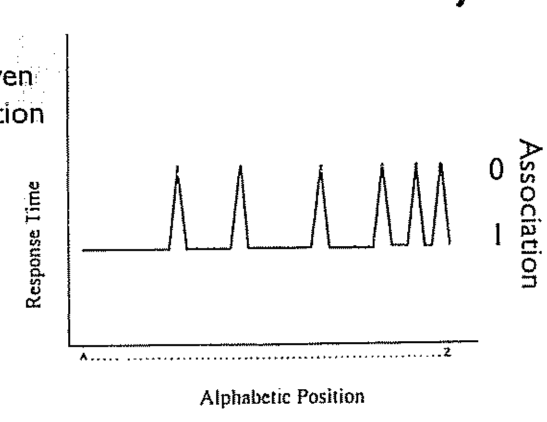
Alphabetic Search by Associations

Scharroo, Leeuwenberg, Stalmeier & Vos, 1994, JEP:LMC.

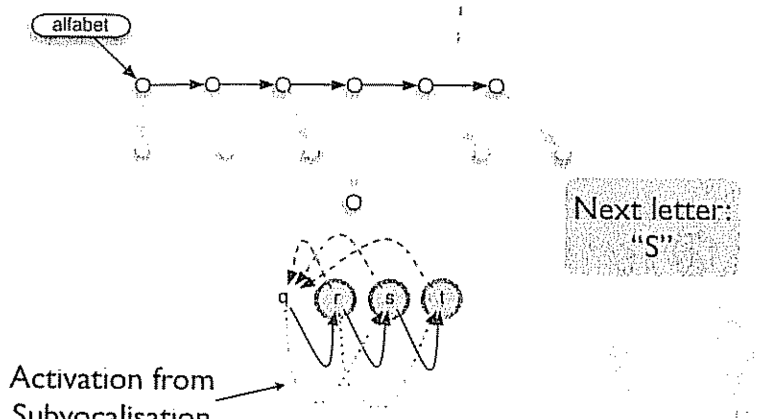
- Claim:
  - Model is implausible at different conceptual levels
  - Sawtooth-shaped pattern in Klahr et al's data is an artefact of averaging.
  - Pattern is dependent on "nursery rhyme", pattern is even weaker in people who learned alphabet in a different way
- Simple associations explain data better.

ASA Individual Subject

Data driven segmentation

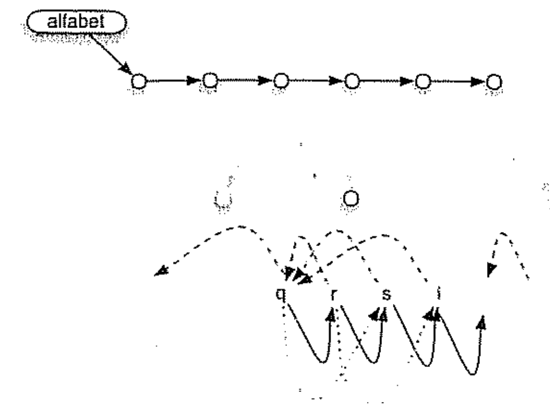


Associations from ASA



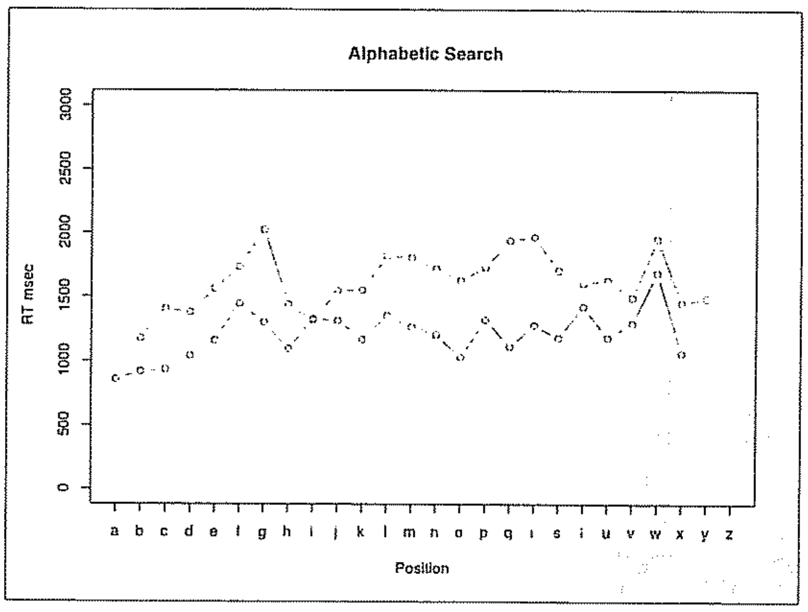
$(A_t = B_t + W_{ts}S + W_{rs}S) > \tau$

Structure + Association

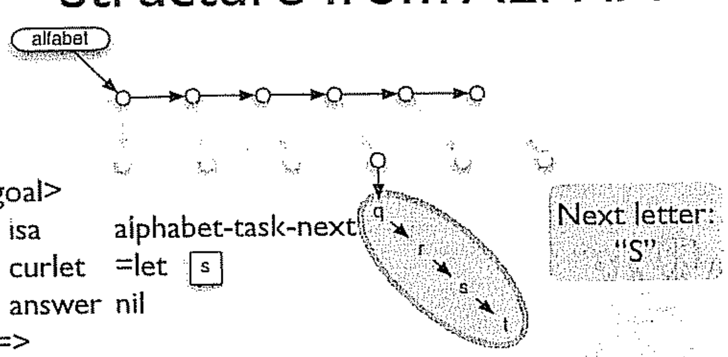


Structure + Association

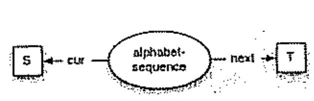
- Association strengths + activations determine position-related slope
- Association strengths determine "implicit" chunks
- Both "direct retrieval" and "walking-through-list" answers possible
- Individual RT are predicted to increase within chunks more than overall-slope predicts.



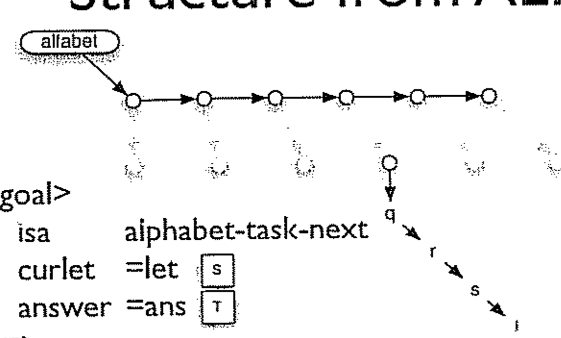
Structure from ALPHA



=goal>
isa alphabet-task-next
curlet =let $\boxed{S}$
answer nil
==>
+retrieval>
isa alphabet-sequence
cur =let $\boxed{S}$

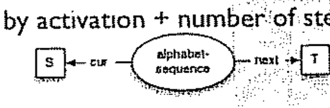


Structure from ALPHA

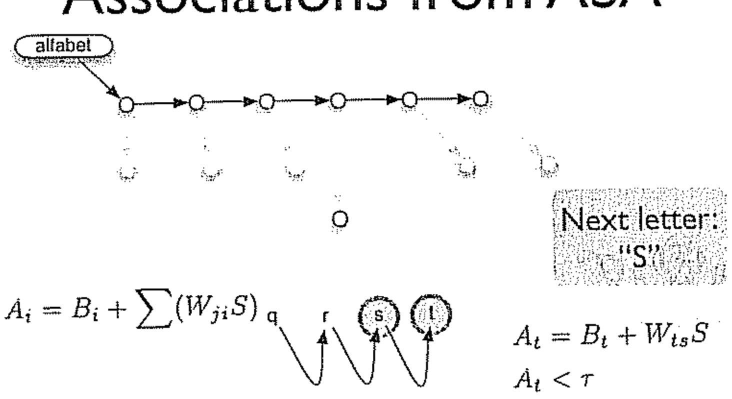


=goal>
isa alphabet-task-next
curlet =let $\boxed{S}$
answer =ans $\boxed{T}$
==>
"Say answer $\boxed{T}$"

Performance is driven by activation + number of steps



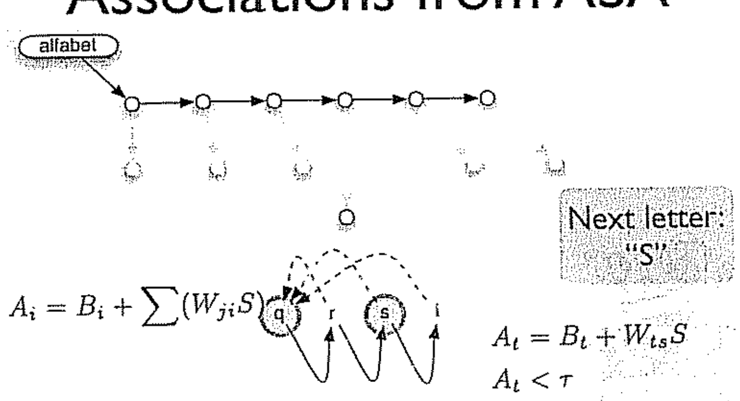
Associations from ASA



$A_i = B_i + \sum(W_{ji}S)$

$A_t = B_t + W_{ts}S$
$A_t < \tau$

Associations from ASA



$A_i = B_i + \sum(W_{ji}S)$

$A_t = B_t + W_{ts}S$
$A_t < \tau$
$(A_q = B_q + W_{qs}S) > \tau$

Conclusions

- The symbolic structure of ALPHA is reflected in the subsymbolic associations
- Neither simple letter-to-letter association account nor pure symbol manipulation can account for the data.

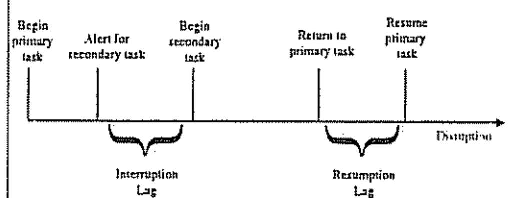
The long-term disruption effect: A comparison of three memory models

Greg Trafton (NRL)
Erik Altmann (MSU)
Derek Brock (NRL)

Collaborators (Reverse Alphabetical)

- Susan Trickett (GMU)
- Raj Ratwani (GMU)
- Chris Monk (GMU/Westat)
- Debbie Boehm-Davis (GMU)

Timeline of an Interruption



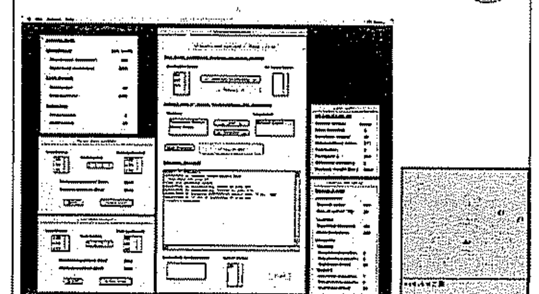
ACT-R model

- The interruption process is all about interrupted and resumed goals
- Altmann & Trafton (2002) present a computational cognitive model in ACT-R that deals with memory for goals

Theory of Interruptions

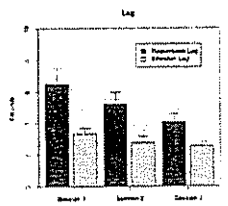
- Our model makes 3 key predictions about interruptions:
  - After being interrupted, the goal(s) of the primary task will decay (according to base-level equation)
  - Preparing to resume a task can slow down decay (Trafton et al. 2003)
    - Prospective encoding of goals (forward-looking)
    - Retrospective rehearsal of current state info
  - Priming of cues in the environment can facilitate resumption

Experimental task



What happens after an interruption?

- Immediately after an interruption, there is a large resumption lag compared to non-interrupted work



Trafton, Altmann, Brock, & Mintz, 2003

Calculating long term disruption

- I calculated RT between actions for the first 10 actions after each interruptions (first action is resumption lag)

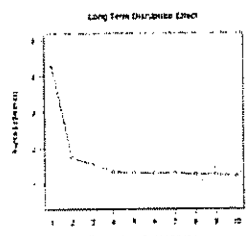
1	2	3	4	5	6	7	8	9	10
4.2	6.9	2.1	.2	3.3	10.	2.3	4.3	1.3	.2
3.7	1.7	.2	3.8	1.2	.7	1.5	.2	.2	.5
...	...	...	...	...	...	...	...	...	...

Average

4 6	1 8	1 6	1 9	1 4	1 4	1.1	1 0	9	8

What happens after the resumption lag?

- A couple of years ago I showed the long term disruption effect and a simple memory model for it. In general, the interruption is very disruptive immediately after an interruption, and then becomes less disruptive over time



Overall Goal: Zero Parameter memory model

- I took seriously what John has been saying for several years: Use zero-parameter models
- My overall goal is to use a good memory model to explain and predict the long-term disruptive effect
- (And then build systems that reduce the overall disruptiveness of interruptions, but that's a different project)

3 different memory models

- There are several different memory equations for base-level learning:
  - Simplified Base-level learning equation (Optimized learning, default in ACT-R)
    - Probably used the most in most ACT-R models
  - Base-Level Learning Equation 4.1 (full version)
    - Computationally intensive but good
  - Spacing effect equation (Pavlik & Anderson, in press)
    - Very computationally intensive; accounts for spacing effect data very well across multiple data sets
- Which one is the best model to explain long term disruption?

Base level learning equation

$$B_i = \ln(\sum_{j=1}^{n} t_j^{-d}) + \beta$$

Anderson and Lebiere. 1998

- B_i is the activation of the chunk
- n is the number of times that the chunk has been encountered at past lags t_j
- d is a free parameter set at 0.5
- β is typically absorbed in the estimates of other parameters

Simplified ACT-R equation

$$B_i = \ln[\frac{n}{(1-d)}] - d * \ln(T)$$

Anderson and Lebiere. 1998

- B_i is the activation of the chunk
- n is the number of times that the chunk has been encountered in the past
- T is the total time of the life of the chunk
- d is a free parameter set at 0 5

Spacing effect equation

$$B_i = \ln(\sum_{j=1}^{n} t_j^{-d_i})$$

$$d_i(B_{i-1}) = ce^{B_{i-1}} + a$$

Pavlik & Anderson. in press

- B_i is the activation of the chunk
- n is the number of times that the chunk has been encountered at past lags t_j
- d is the decay rate
- c is a free parameter (decay scale) = .217
- a is a free parameter (decay intercept) = .177

3 Models

- I calculated activation of each action in the entire task, then converted activation into predicted RT:
- $RT = F*e^{-activation}$
  - F is a free parameter (set to 1 as default)
- All models used zero free parameters:
  - d = 0 5 throughout
  - F = 1 throughout
  - c = 217 (From Pavlik & Anderson; Pavlik communication)
  - a = 177(From Pavlik & Anderson; Pavlik communication)

Method

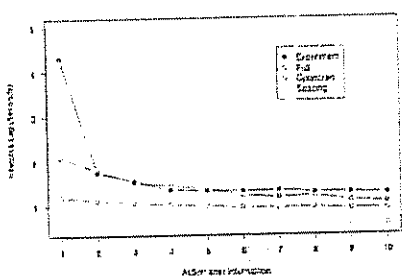
- Method:
  - 65 participants
  - 10 interruptions per session
  - Primary task: complex resource allocation task (Brock & Trafton, 1999)
  - Secondary task: Ballas task (ATC-like)
- All participants are switched to secondary task immediately: no alert and no environmental cues available at resumption
- 3 Sessions (within)

A priori concerns

- Zero parameter models may not be perfectly optimized
- Since I'm only using memory retrieval, any quantitative fit will (should?) be low (no /PM in my model)
- We know that the initial resumption lag depends on things other than just pure action/recall, so the resumption lag model point is likely to be a big underestimation

Guesses about which will be best?

Data and Model



Conclusions

- Surprisingly (to me, at least) the full model had the best fit
- The spacing model may have been at a disadvantage because we don't know about the specific parameters outside of spacing experiments
- The simplified model is in between the two quantitatively, but does not show the steep disruption at positions 2-4

Conclusions

- All three zero parameter models are quite good, picking up both qualitative and quantitative effects (sans resumption lag)
  - (not surprising, since they are all from the same family of models)
- Long term disruption effect can be captured by simple memory model, primarily use and re-use of action chunks (as opposed to a goal-based account we have been working on with the immediage (resumption lag) disruption)
- Not much room for production time, other memory retrievals, etc. Is the fit too good quantitatively, at least for the full model?

ACT-R conclusions

- Turn optimized learning off as a default within ACT-R
- How do we decide which memory model / memory equation to use for different tasks or situations? The full model is best here, but clearly not for traditional spacing effect experiments

fin!

General principles of cognition?

Nick Chater
Institute for Applied Cognitive Science
Department of Psychology
University of Warwick

OVERVIEW

I. Fragmentation and integration
II. Case Studies
  A: Scale invariance
  B: Probabilistic modelling
  C: The simplicity principle
III. Where next?

I. Fragmentation and integration

Fragmention in the cognitive and brain sciences

- ...of theory
  - Language acquisition
  - Perception
  - Memory
  - Reasoning

  are independent
- ...of experiments
  - Focus on increasingly detailed behavioral and/or imaging studies of specific phenomena
  - Extrapolation typically secondary

Computational architectures are an integrative step

- Candidate architectures
  - Physical symbol system hypothesis
  - ACT-R
  - Connectionism
  - Exemplar models
- But architectures are only one type of constraint
  - E.g., the myriad connectionist architectures for reading single words

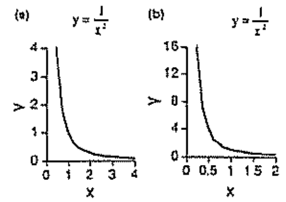
Some candidate general principles

Principle	Maths	Domains	Examples	References
Scale-invariance	•Statistical self-similarity •Fractals	•Psychophysics •motor control •memory •learning	•Weber's Law •Stevens' Law •power law of forgetting •power law of practice •Fitts Law	Chater & Brown (1999). Cognition; Brown, Neath & Chater (in press). Psych Review; Stewart, Brown & Chater (in press). Psych Review
Probabilistic models	•Bayesian statistics •Bayesian networks	•Perception •Language processing •Reasoning	•Bayesian vision •Statistical linguistics •Conditionals •Wason's selection task •Syllogisms •Imagery?	•Chater, Tenenbaum, Wolpert, Yuille (Eds) (2006). Special issue of TICS; Oaksford & Chater (1994). Psych Review; Oaksford & Chater (2006). The probabilistic mind, OUP.
Simplicity	•Kolmogorov complexity	•Perception •Language acquisition •Causal reasoning	•Perceptual organization •Learning from "positive" evidence •Baker's paradox	•Chater (1996). Psych Review. Chater (2004) J. Child Language. Chater & Vitányi (under revision) Cognitive Science.

II. Case Studies

A. Scale-invariance

- In a nutshell:
  - Throw away "units"
  - Can you reconstruct them from your data?
- If *not*, phenomenon is scale-invariant



Only power laws $y \propto x^{a}$ are scale invariant

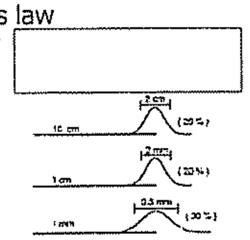
The ubiquity of scale-invariance

- City sizes
- Size of firms
- River sizes
- Earthquakes
- Distribution of digits (Benford's Law)
- Word frequencies (Zipf's Law)
- Stock fluctuations
- Mandelbrot: Scale-invariance as a primitive
- Scale-invariance as a "null hypothesis" for the cognitive and brain sciences
- This null hypothesis implies many well-known psychological laws...

From scale-invariance to psychological "laws"

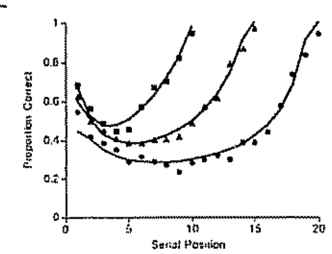
Regularity	*Form*	*Explanation*
Weber's Law	$\Delta I \propto I$	$\Delta I/I$=constant, if independent of units
Stevens' Law	$I^{n} \propto S$ (power law)	$\Delta I/I \propto \Delta S/S$ Ratio preserving: input-output
Power law of forgetting	$m(t) = m(0)t^{-a}$	Ratio preserving: memory-time
Power law of practice	$RT(N) = RT(N)t^{-a}$	Ratio preserving: trials-speed
Fitts' Law	$T = a + b\log(\Delta D/D)$	(*nb* Non-invariant *a* for initiating movement)

Weber's law



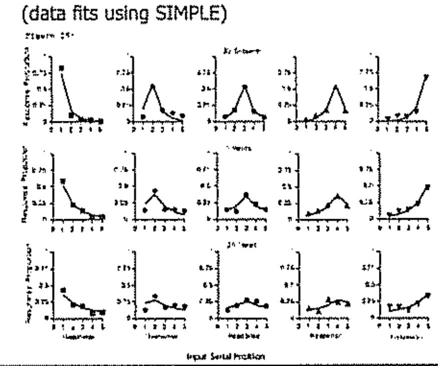
Endless cases of invariance, in perception, motor control, learning and *memory*

Serial position in immediate free recall

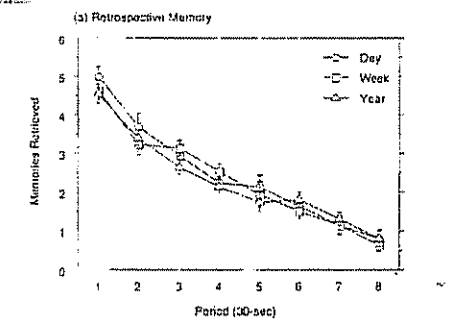


Data from Murdock, 1962; model fits using SIMPLE (Brown, Neath & Chater)

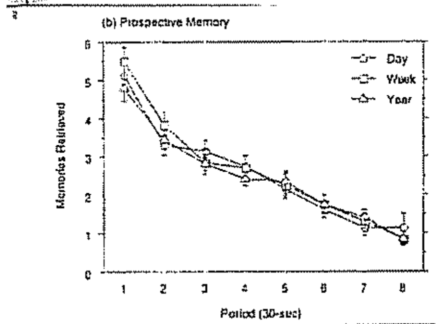
Confusion in memory for serial order
(data fits using SIMPLE)



Memory retrieval over different time periods in retrospective memory
(Maylor, Chater & Brown, 2001, *PB&R*)



And prospective memory



B: Probabilistic modelling

Subjective probability

- Uncertainty represented as subjective degree of "belief"
- Mild normative assumptions (e.g., Dutch book)

∴ Probability calculus

Bayesian updating

- Given evidence, E, models, M_i
- $\Pr(M_i|E) \propto \Pr(E|M_i)\Pr(M_i)$
- Iterate, as new evidence arrives

Domains for probabilistic models of cognition

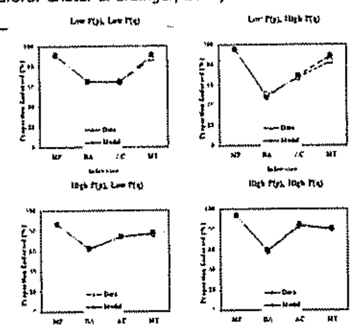
Domain	*Topic*	*Principle*	*References*
Reasoning		Human uncertain reasoning is modelled by probability not logic	
	Conditionals	If *P* then *Q*; Pr(*Q*|*P*)	Oaksford, Chater & Grainger, 2000
	Selection task	Optimal data selection	Oaksford & Chater, 1994
	Syllogisms	Some *A* are *B*; Pr(*A*, *B*) > 0	Chater & Oaksford, 1999
Causality	Representing and learning causal knowledge	Bayesian belief nets; Intervention as "do"-operator	Pearl, 2000; Tenenbaum, Griffiths, Sloman, Lagnado; Ali, Chater & Oaksford
Vision		Sensory processing as Bayesian updating	Knill & Richards; Weiss; Yuille, computer vision
Language	Parsing	Parsing using stochastic grammars	Manning & Schütze; computational linguistics
Motor control		Bayesian optimal control	Wolpert

Conditionals: Contrasting probability and logic

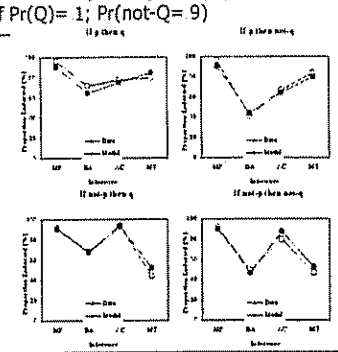
Inference	*Additional premise*	*Candidate conclusion*	*Logical validity*	*Probabilistic comparison*
MP: Modus Ponens	*P*	*Q*	Y	Pr(*Q*|*P*) ≥ Pr(*Q*)
DA: Denial of the Antecedent	Not-*P*	Not-*Q*	N	Pr(not-*Q*|not-*P*) ≥ Pr(not-*Q*)
AC: Affirming the Consequent	*Q*	*P*	N	Pr(*P*|*Q*) ≥ Pr(*P*)
MT: Modus Tollens	Not-*Q*	Not-*P*	Y	Pr(not-*P*|not-*Q*) ≥ Pr(not-*P*)

- Probabilistic predictions are graded
- Depend on Pr(*P*) and Pr(*Q*)
- Fit with data on argument endorsements

Varying probabilities in conditional inference
(Oaksford, Chater & Grainger, 2000)



Negations implicitly vary probabilities
(e.g., if Pr(Q)= 1; Pr(not-Q= 9)



Extensions of the approach to:

- Wason's selection task
- Syllogisms
- Causal reasoning: Ali, Chater & Oaksford (multiple, inter-related conditionals; causal structure matters)

- A unified for understanding reasoning
- Integration with probabilistic perspective across cognition

C: Simplicity

- Find explanation of "data" that is as simple as possible
  - An 'explanation' *reconstructs* the input
  - Simplicity measured in code length
  - Mimicry theorem with Bayesian inference (e.g., Chater, 1996, *Psych Review*, "deep" analysis by Li & Vitányi, 1997)

Simplicity as "ideal" inductive method, when no probabilistic model available

Mathamatics

- Deep mathematical theory: Kolmogorov complexity theory (Li & Vitányi, 1997)
- Predicting using simplicity converges on correct predictions (Solomonoff, 1978)

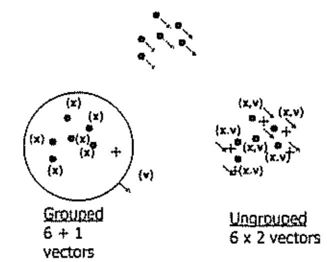
Statistics

- An ultra-general/neutral probabilistic model over all computable hypotheses (Solomonoff, 1964)
- Practical statistical /machine learning method: Minimum description length (Grünwald et al, 2005)

Simplicity has broad applications

Domain	*Principle*	*References*
Perceptual organization	Find grouping that minimizes code	Koffka, 1935; Leeuwenberg, 1971; Attneave & Frost, 1969;
Early vision	Efficient coding & transmission	Blakemore, 1990; Barlow, 1974; Srinivisan, Laughlin
Causal reasoning	Find minimal belief network	Wedelind
Similarity	Similarity between representations measured by code length between them	Chater & Vitányi, 2003; Hahn, Chater & Richardson, 2003
Categorization	Categorize items to find shortest code (high level perceptual organization)	Pothos & Chater, 2002; Feldman, 2000
Memory storage	Shorter codes easier to store	Chater, 1999
Memory retrieval	Explain interference by size-trace complexity; a rational foundation for distinctiveness models	Rational foundation for distinctiveness models: SIMPLE (Brown, Neath & Chater, 2003)
Language acquisition	Find grammar that best explains child's input	Chomsky, 1955; J. D. Fodor & Crain; Chater, 2004; Chater & Vitányi, 2003

Long tradition of simplicity in perception
(Mach, Koffka, Leeuwenberg); e.g., Gestalt laws



But we focus instead on simplicity as a model of language acquisition

- Undergeneral grammars predict that good sentences are not allowed
- just wait til one turns up

- Overgeneral grammars predict that bad sentences are actually ok
- Need negative evidence—say a bad sentence, and get corrected

The logical problem of language acquisition
(e.g., Hornstein & Lightfoot, 1981; Pinker 1979)

- Without negative evidence can never eliminate overgeneral grammars
- "Mere" non-occurrence of sentences is not enough.
- ...because almost all acceptable sentences also never occur
- Backed-up by formal results (Gold, 1967)

→ Argument for innateness?

But simplicity offers an alternative

Specifying an "ideal" learning set-up

- Linguistic environment
  - *Positive evidence only; computability*
- Measures of learning performance
  - *Statistical*
- Learning method
  - *Simplicity*

Prediction by simplicity

- Find shortest 'program/explanation' for current 'corpus'
- Predict using that program
  - Strictly, use 'weighted sum' of explanations, weighted by brevity

Prediction is possible (Solomonoff, 1978)
Summed error has finite bound

$$\sum_{j=1}^{\infty} s_j \le \frac{\log_e 2}{2} K(\mu)$$

So prediction converges [faster than 1/nlog(n)], for corpus size n

This is an amazing, and fundamental, result about the possibility of inductive inference

Overgeneralization Theorem
(Chater & Vitányi)

- Suppose learner has probability Δ_j of erroneously guessing an ungrammatical jth word

$$\sum_{j=1}^{\infty} \langle \Delta_j \rangle \leq \frac{K(\mu)}{\log_e 2}$$

- Intuitive explanation:
  - overgeneralization underloads probabilities of grammatical sentences;
  - Small probabilities implies longer code lengths

Absence as implicit negative evidence

- Overgeneral grammars predict missing sentences
- And their absence is a clue that the grammar is wrong

This overgeneralization theorem makes this intuition rigorous

Extensions and implications 1

- An ideal language learning can learn. from positive data, to
  - Predict
  - Make grammaticality judgements
  - Produce language
  - Relate form and meaning
- all to a high level of accuracy
- This does not imply that the language learner converges precisely on the "true" grammar but arbitrarily close seems good enough

Extensions and implications 2

- So (enough) positive evidence can support language acquisition
- Also "scaled-down" information-investment methodology. to assess which aspects of linguistic structure are learnable (Onnis, Roberts & Chater. 2005)
- Future question:
  - How far does simplicity predict empirical data
  - Relate to other theories of acquisition. e g . Tomasello. Culicover

III. Where next?

Towards the re-integration of cognitive science?

- Further integration of general principles + resolving clashes between them
  - Perceptuo-motor control vs. perceptual judgement
  - Models of decision making (DbS)
  - Memory retrieval—distrinctiveness meets simplicity?
- Other candidate principles?
  - Reversibility of cognition (but irreversibility of the production rule?)
  - Seriality constraints (e.g., memory retrieval)
  - Wide range of principles from ACT tradition; ideas from connectionism etc

And integrating with computational frameworks

- Connectionism
- Bayes nets
- Exemplar models
- ACT-R

Comments on General Principles of Cognition

John Anderson
Department of Psychology
Carnegie Mellon

1. Discuss the issues of general principles versus cognitive architectures.
2. Discuss two of Chater's specific General Principles
3. Discuss how to integrate the insights of general principles and cognitive architectures

I. Fragmentation and integration

A single system (mind) produces all aspects of behavior. It is one mind that minds them all. Even if the mind has parts, modules, components, or whatever, they all mesh together to produce behavior. Any bit of behavior has causal tendrils that extend back through large parts of the total cognitive system before grounding in the environmental situation of some earlier times. If a theory covers only one part or component, it flirts with trouble from the start. It goes without saying that there are dissociations, independencies, impenetrabilities, and modularities. These all help to break the web of each bit of behavior being shaped by an unlimited set of antecedents. So they are important to understand and help to make that theory simple enough to use. But they don't remove the necessity of a theory that provides the total picture and explains the role of the parts and why they exist (Newell, 1990; pp. 17-18).

What's wrong with Computational Architectures?

- Architectures are no protection against specific assumptions as the modules movement in ACT-R proves -- but the brain/mind seems fundamentally specialized
- Nonetheless the brain/mind is integrated
- Chater: Architectures do not seem to lead to unique explanations -- e.g., multiple ACT-R models for Broadbent sugar factory, list learning, task switching
- Points to need to have principles of model acquisition
- But inevitably this leads us down the road to dealing with specifics -- try understanding how children learn to solve algebra and you wind up having to reconstruct all of their past experiences.

What's wrong with General Principles?

- My experience with rational analysis: simply predicted behavior under the assumption that is optimized to the environment "minimal" assumptions about computational limitations
- Fall short of achieving integration and accounting for the real details of human thought; better suited to characterizing abstractions about human cognition than cognition itself.
- "The question for me is how can the human mind occur in the physical universe. We now know that the world is governed by physics. We now understand the way biology nestles comfortably within that. The issue is how will the mind do that as well. The answer must have the details. I got to know how the gears clank and how the pistons go and all the rest of that detail. My question leads me down to worry about the architecture." (Newell -- Dec 4, 1991)

Comparing General Principles (GP) and Cognitive Architectures (CA)

Chater and Vitányi:

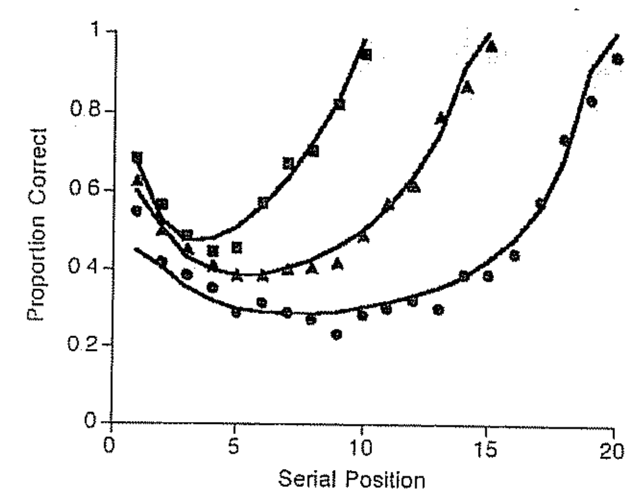
K(Complete explanation)
$\leq$ *K(Theory)* K(GP)< K(CA) at least weakly *(a.l.w)*
+ *K(Parameters|Theory)* K(Params|GP) < K(Params|CA) *a.l.w.*
+ *K(Data|Parameters, Theory)* *Not Clear*

It is not clear because the GP approach tends to deal in abstractions that ignore how the gears clank and how the pistons go.

Chater and Vitányi: "every piece of data must be 'accounted for'---even if only be reproducing it verbatim"

Of course it should not be a competition of approaches -- the question is how to combine GP and CA to get an optimal account of the data.

Serial position in immediate free recall



Data from Murdock, 1962; model fits using SIMPLE (Brown, Neath & Chater)

ACT-R: Anderson, Bothell, Lebiere, & Matessa, 1998

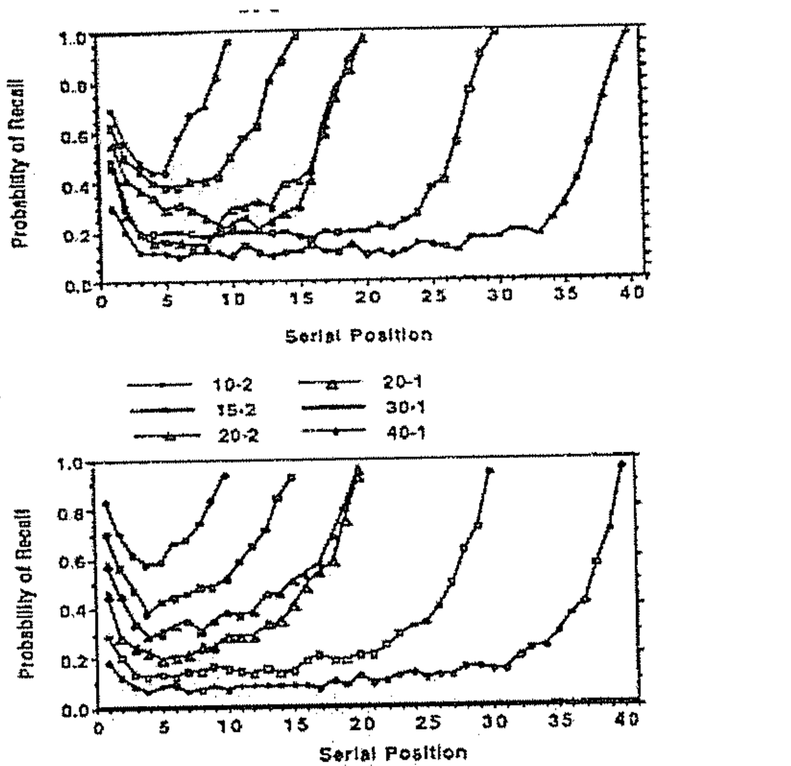


FIG. 13. Probability of recall of lists of various lengths (10, 15, 20, 30, 40) and amount of study time (1 or 2 seconds) as a function of serial position. (a) Data from Murdock (1962). (b) Predictions of the ACT-R theory.

Comparison of Models

1. Complexity of theories favors Simple
2. But ACT-R predicts
   a. Rehearsal patterns during study
   b. Recall order and dependency on rehearsal pattern
   c. Recall latency
   d. And every other detail

But what about

Study:

1. garrison—GARRISON, LIEUTENANT, DIGNITARY.
3. vulture—VULTURE...bird, there was a bird PRESENT...VULTURE, bird...GARRISON.
13. lieutenant—LIEUTENANT is in the GARRISON...and he is being attacked by a VULTURE that came through the window.
32. destroyer—the LIEUTENANT is an OFFICER, DESTROYER, MERCENARY...the LIEUTENANT is too much...he's a DESTROYER.

Recall:

The LIEUTENANT...lieu-ten-ant...is a MERCENARY with SIDEBURNS...DESTROYER...OFFICER...who's in the GARRISON...and is being attacked by VULTURES

The logical problem of language acquisition meets the past tense debate

- Note the past tense debate reflects a major abstraction and simplification from real language acquisition
- But at least it deals with real data (well, sort of real data)
- Strategies in the ACT-R model (Taatgen & Anderson, 2002):
  - Do nothing but pay communication cost
  - Retrieve an example and use analogy -- very costly but after production compilation leads to regular rule
  - Use regular rule(s)
  - Retrieve answer (but must be frequent enough to be active)
- Unlike most connectionist models it learns from a representative stream of input (both in distribution of items and numbers of items)
- Produces the U-shaped learning function --gradual onset of overgeneralization and even more gradual disappearance

The logical problem of language acquisition meets the ACT-R model of the past tense

- ACT-R model requires no feedback
- Hearing allows it to absorb the statistics of past tense in the language which are embedded in the base level activations of declarative memory
- Irregular past tense generations are preferred over regular because they are more regular in the phonology
- Explains why exceptions are both high frequency and phonologically regular
- Indeed it explains why there are exceptions
- Explains why the learning curves are gradual -- why one encounter with "break" has virtually no impact on behavior
- It is not clear that the ACT-R model contradicts anything that would follow from the simplicity principle but it addresses detail that is not obvious from that principle

If it really is cognitive architectures why do general principles work as well as they do?

- I think this follows from the simplicity of science itself -- because the whole universe is compressed by Kolmogorov complexity metric the human mind is forced into operating according to general principles (could try to develop this argument from the mere 30,000 genes we have).
- One could take the view that general principles just provide good "approximate" characterizations of human cognition and from that perspective maybe they should stay at the level of abstraction that they usually address data.
- However, an alternative is that they are actually deeply embedded in the architecture because they reflect broad regularities in the universe -- "We may look into that window of the mind as through a glass darkly, but what we are beginning to discern there looks very much like a reflection of the world" (Shepard, 1990, p. 213)
- We have embedded some of rational analysis into the subsymbolic level of ACT-R
- Perhaps we should be looking at how to embed some of Chater's principles.

Learning Algebra in ACT-R
John R. Anderson

1. Give the system the abilities that a prepared student entering Algebra 1 should have. These include the abilities to perform basic arithmetic and to parse arithmetic expressions. These are clearly not challenging abilities for an AI system.
2. Give the system a representation of the instructions that appear in a standard algebra textbook. It should be stressed that these instructions are only sometimes precise and never complete specifications of how to do the operations.
3. Have the system learn by feedback on its solution efforts how to solve the class of problems that appear in the textbook.
4. Match on learning time, performance statistics, and brain imaging.

2 Chapter 1 Expressions and Equations

1-1 OPERATIONS WITH NUMBERS

An *operation* in mathematics is something you do to numbers, such as adding, subtracting, multiplying, or dividing. For instance, in

$$3 + 11,$$

the operation of addition is performed on the numbers 3 and 11. Difficulties may arise if there are several different operations. For example, what number does

$$3 + 2 \times 7$$

represent? If you add 3 and 2 first and then multiply by 7, you get

$$5 \times 7 = \underline{35}.$$

But if you multiply 2 by 7 first, then add the result to 3, you get a different number:

$$3 + 14 = \underline{17}$$

To avoid this difficulty, symbols of inclusion—parentheses, ( ), or brackets, [ ]—are used to tell which operation to do first.

$(3 + 2) \times 7$ means 5×7, or 35

$3 + (2 \times 7)$ means $3 + 14$, or 17

$12 \div [2 \times 3]$ means $12 \div 6$, or 2.

Another symbol of inclusion is the bar used in fractions, called a vinculum. For example, in

$$\frac{6 + 7}{2 \times 7}$$

the operations $6 + 7$ and 2×7 are done *first*, giving

$$\frac{13}{14}$$

Then 13 is divided by 14, giving about 0.929.

A collection of numbers, operation signs, and symbols of inclusion (parentheses, brackets, vinculums) is called an expression. Finding the value of an expression is called evaluating the expression. An expression such as

$$36 \div 3 \times 4 + 2$$

can have different values, depending on how the operations are grouped.

Examples

EXAMPLE 1

Evaluate $[(36 \div 3) \times 4] + 2$.

$$\begin{aligned} &[(36 \div 3) \times 4] + 2 \\ &= [12 \times 4] + 2 \\ &= 48 + 2 \\ &= \underline{50} \end{aligned}$$

Note that you do what is inside the *innermost* symbols of inclusion *first*

EXAMPLE 2

Evaluate $[36 \div (3 \times 4)] + 2$.

$$\begin{aligned} &[36 \div (3 \times 4)] + 2 \\ &= [36 \div 12] + 2 \\ &= 3 + 2 \\ &= \underline{5} \end{aligned}$$

Objective

Given an expression, be able to *evaluate* it

Instruction?

As shown in the examples, you should do the following to evaluate an expression:

1. Write the given expression.
2. Do the innermost operation and write the result. Use an = sign to connect the new expression to the original one
3. Keep doing operations until you reduce the expression to a *single* number. Use = signs to connect each expression to the one before, as shown in the examples.
4. Clearly indicate the answer by underlining or boxing it.

Now *you* work the examples on the next page. Put a piece of paper along the dotted lines. This will cover the answer, leaving only the original expression showing. (If the writing shows through, use more sheets of paper or an index card.) Then evaluate the expression. Last, uncover the answer in the book to make sure your *work* and your *answer* are right.

Issues and Progress

1. Can students learn with as little instruction as this?
2. What are the relative contributions of instructions versus examples?
3. Problems with addressing this with the original Forester material -- prior knowledge and population. We developed a data-flow isomorph of the Foerster material for college students.
4. Shawn Betts developed a computer system that extends to most of equation solving and tested with 10+ subjects. They could indeed learn with this minimal instruction.
5. We have also developed an parallel system for teaching children regular algebra but not tested
6. Model extends to first 4 sections including preliminary equation solving.
7. The model illustrates many features in ACT-R 6.0 -- some of which I will try to illustrate.

Instruction Comprehension?

```
0.000   GOAL          SET-BUFFER-CHUNK GOAL GOAL REQUESTED NIL
0.000   GOAL          SET-BUFFER-CHUNK GOAL GOAL REQUESTED NIL
0.000   SCREEN        SET-BUFFER-CHUNK SCREEN SCREEN0
0.050   PROCEDURAL    PRODUCTION-FIRED FOLLOW-INSTRUCTIONS
5.050   VERBAL        SET-BUFFER-CHUNK VERBAL VERBAGE0

(p follow-instructions
   =goal>
      isa task
      state read-instructions
      step read
==>
   +verbal>
      isa verbage
      command read-instructions
      arg1 start
   =goal>
      step ready
      state start)
```

"To evaluate a diagram find an operation with all numbers, evaluate the operation, copy the results according to the arrow, and repeat until it becomes a number."

```
-->
(op1 isa operator pre start action find-with-all-numbers arg1 box post all-numbers)
(op2 isa operator pre all-numbers action copy arg1 evaluated-results arg2 arrow post
start subgoal t)
(op3 isa operator pre start action test-lone-number post lone-number)
(op4 isa operator pre lone-number action go-on post start))
```

Evaluate a Diagram

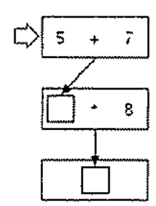
To evaluate a diagram find an operation with all numbers, evaluate the operation, copy the results according to the arrow, and repeat until it becomes a number

Follow the instructions above the problem to solve it.

Click the box pointed to by the arrow.

Last Diagram State

Current Diagram State



"To evaluate a diagram find an operation with all numbers, evaluate the operation, copy the results according to the arrow, and repeat until it becomes a number."

```
OP1                                   OP3
 ISA OPERATOR                          ISA OPERATOR
  PRE  START                            PRE  START
  ACTION  FIND-WITH-ALL-NUMBERS         ACTION  TEST-LONE-NUMBER
  ARG1  BOX                             ARG1  NIL
  ARG2  NIL                             ARG2  NIL
  POST  ALL-NUMBERS                     POST  LONE-NUMBER
  SUBGOAL  NIL                          SUBGOAL  NIL

OP2                                   OP4
 ISA OPERATOR                          ISA OPERATOR
  PRE  ALL-NUMBERS                      PRE  LONE-NUMBER
  ACTION  COPY                          ACTION  GO-ON
  ARG1  EVALUATED-RESULTS               ARG1  NIL
  ARG2  ARROW                           ARG2  NIL
  POST  START                           POST  START
  SUBGOAL  T                            SUBGOAL  NIL
```

Representation

```
op
   isa OPERATOR
   pre state -- both an index to operator and description of a
           potentially recognizable external situation
   action executable or action
   arg1 argument -- referent either bound or to be bound
   arg2 argument -- referent either bound or to be bound
   post -- both an index to operator and description of a
      potentially recognizable external situation
   subgoal t or nil -- flag on action
```

Similarities to list representation
- positional rather than associational
- list representation did not have pre and post
- hierarchical
- as list representation, allows one to begin in arbitrary position
- suggests positional confusions

Retrieving Operators

5.100	PROCEDURAL	PRODUCTION-FIRED FIND-OPERATOR-EXTERNAL
5.142	DECLARATIVE	SET-BUFFER-CHUNK RETRIEVAL OP1
5.192	PROCEDURAL	PRODUCTION-FIRED FIND-WITH-ALL-NUMBERS-BOX
5.692	GRAPHICAL	SET-BUFFER-CHUNK GRAPHICAL BOX0
5.742	PROCEDURAL	PRODUCTION-FIRED COLLECT-GRAPHICAL-RESULT
5.792	PROCEDURAL	PRODUCTION-FIRED FIND-OPERATOR-INTERNAL
6.329	DECLARATIVE	SET-BUFFER-CHUNK RETRIEVAL OP2

```
(p find-operator-external
   =goal>
      isa task
      state =state
      step ready
   =screen>
      isa screen
      state =state
   ?verbal>
         state free
==>
   =goal>
      step retrieving-operator
   +retrieval>
     isa operator
     pre =state)
```

```
(p find-operator-internal
   =goal>
      isa task
      state =state
      step ready
   ?verbal>
         state free
!safe-eval! (internal-state =state)
==>
   =goal>
      step retrieving-operator
   +retrieval>
     isa operator
     pre =state)
```

Using Retrieval Finsts

5.100	PROCEDURAL	PRODUCTION-FIRED FIND-OPERATOR-EXTERNAL
5.172	DECLARATIVE	SET-BUFFER-CHUNK RETRIEVAL OP3
5.222	PROCEDURAL	PRODUCTION-FIRED TEST-LONE-NUMBER
5.772	PROCEDURAL	PRODUCTION-FIRED FAILED-GRAPHICAL
5.962	DECLARATIVE	SET-BUFFER-CHUNK RETRIEVAL OP1
6.012	PROCEDURAL	PRODUCTION-FIRED FIND-WITH-ALL-NUMBERS-BOX

```
(p test-lone-number
   =goal>
      isa task
      step retrieving-operator
    =retrieval>
         isa operator
         action test-lone-number
         post =post
  ?digital>
      state free
==>
  +graphical>
    isa box
    command test-lone-number
  =goal>
    step looking
    arg1 result
    post =post)
```

```
(p failed-graphical
   =goal>
      isa task
      state =state
      step looking
    ?graphical>
      state   error
!eval! t
==>
   =goal>
      step retrieving-operator
   +retrieval>
     isa operator
     pre =state
     :recently-retrieved nil)
```

Processing Definite References: "the box"

5.692	GRAPHICAL	SET-BUFFER-CHUNK GRAPHICAL BOX0
5.742	PROCEDURAL	PRODUCTION-FIRED COLLECT-GRAPHICAL-RESULT
5.792	PROCEDURAL	PRODUCTION-FIRED FIND-OPERATOR-INTERNAL
...		
10.449	DECLARATIVE	SET-BUFFER-CHUNK RETRIEVAL OP2A
10.499	PROCEDURAL	PRODUCTION-FIRED CLICK-ARG
10.799	SCREEN	SET-BUFFER-CHUNK SCREEN SCREEN2
10.849	PROCEDURAL	PRODUCTION-FIRED FIND-OPERATOR-EXTERNAL

```
(p collect-graphical-result
   =goal>
      isa task
      post =post
      step looking
      arg1 =var
   =graphical>
      isa box
      val =val
==>
  +label>
    isa assoc
    variable =var
    value =val
  =goal>
     step ready
     state =post
     post nil
     arg1 nil
     arg2 nil)
```

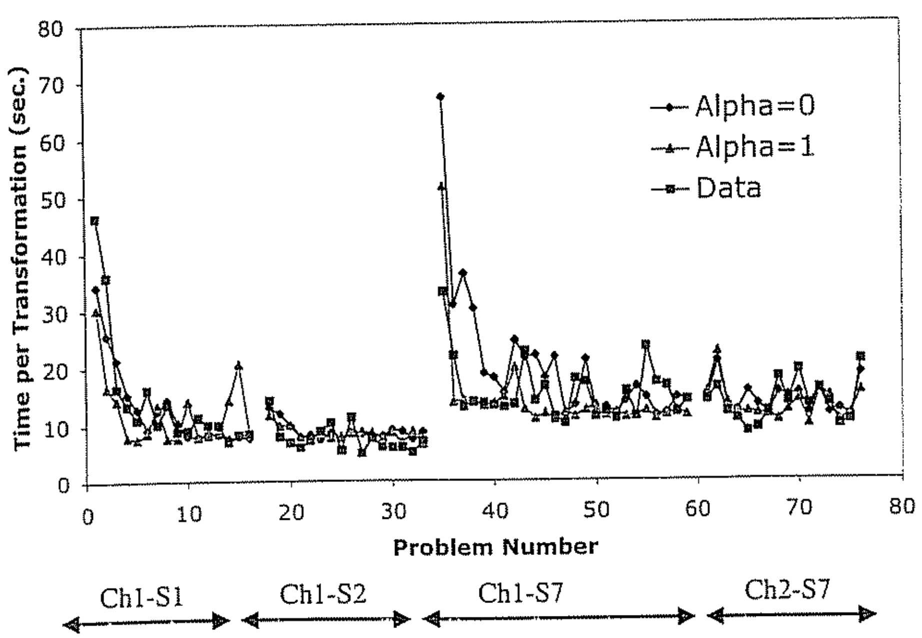
```
(p click-arg
   =goal>
      isa task
      step retrieving-operator
   =retrieval>
         isa operator
         action click
         arg1 =var
         post =post
   =label>
      isa assoc
      variable =var
      value =arg
   ?digital>
      state free
==>
   +digital>
      isa digital
      command click
      item =arg
   ...
```

Confusion in Instruction Interpretation

1. One sort of confusion is semantic -- incorrectly interpreting the verbal instructions. It is not clear than there is much of this in the first 4 sections.
2. Errors are substantially clicking extra boxes, failing to click boxes, and hitting the wrong operation.
3. The one semantic error would be converting 3 - x = 1 into x = 1+3 -- not clear how frequent it is.
4. The other errors can be produced by allowing confusions among adjacent operators through the positional confusion mechanism
5. This then creates the need to respond when things do not turn out the way they were expected to.
6. This can be achieved by looking at the external state and retrieving an operator for it.
7. This also implies that students should be able to pick it up in the midst of a problem.

Task is Action-limited

Minimal Effect of Production Compilation



RACE for retrieval: Competitive effects in memory retrieval

Leendert van Maanen & Hedderik van Rijn
Artificial Intelligence, Groningen University

When a question is stated such as "What is the capital of Australia?", various answers start competing for retrieval from declarative memory. If a hint is given during this retrieval process ("The name of the capital begins with the letter C."), retrieval may be facilitated, but if a distractor is presented ("Amsterdam"), retrieval may be inhibited. A well-known example of these kinds of effects is Picture-Word Interference, a task similar to the Stroop-task (Glaser & Düngelhoff, 1984; Glaser & Glaser, 1989; Schriefers *et al.*, 1990). In these tasks, it has been shown that SOAs have differential effects. For example, presenting a drawing that depicts a concept 50 ms after a word-form of that concept has appeared, speeds up processing of that word in comparison with a neutral condition, whereas in other conditions SOAs might have a negative effect.

The ACT-R Latency Equation as it is defined now, $RT_i = Fe^{-A_i}$ (Anderson *et al.*, 2004), cannot account for these phenomena, as it suggests that retrieval latency *only* depends on the state of the buffers and declarative memory at the exact time of retrieval *onset*, both reflected in A_i. This is easiest demonstrated in the condition where a facilitating word is presented at a short SOA after the picture is shown. After a retrieval request, the chunks that match the request are identified and the one with the highest activation is selected. The Latency Equation determines how long it will take to complete that retrieval, and takes the current level of activation at retrieval onset into account. Thus, at present in ACT-R, the presentation of an interfering stimulus after retrieval onset simply does not influence the calculated latency. Likewise, when another stimulus is presented before retrieval onset, retrieval latency depends (in part) on the spreading activation from the stimulus in a sensory buffer to the to be retrieved chunk: higher levels of activation result in shorter latencies. However, as this can only explain a speed-up, this does not comply with the observation that a condition in which a distractor from the *same category* as the target stimulus is presented has a *larger retrieval latency* than a condition in which an *unrelated distractor* is presented (Glaser & Düngelhoff, 1984). The intuition at least is that concepts of the same category have higher inter-associations than unrelated concepts, which in ACT-R would lead to higher activation levels and shorter latency.

A solution to this issue might be to regard the retrieval process as an instance of a sequential sampling mechanism (Ratcliff & Smith, 2004). Sequential sampling models follow the hypothesis that a neural representation of a stimulus is inherently variable or noisy, and in order to retrieve the required representation, enough samples of the stimulus representation have to be accumulated. Thus, sequential sampling models offer a mechanism that allows for a specification of the time course of retrieval. At retrieval onset, sequential sampling of evidence will allow for the activation of chunks to increase, until at least one chunk's activation has crossed a threshold. Using an adapted version of the leaky competitive accumulator model for perceptual choice (an example of a sequential sampling mechanism, (Usher & McClelland, 2001)), we show that the ACT-R activation function can be extended to account for the *time course of activation* without changing current mechanisms. We will refer to this new set of mechanisms as Retrieval by ACcumulating Evidence (RACE).

In RACE, the time of retrieval is defined as the time at which the activation of a chunk crosses a threshold. We assume that after initiating a retrieval request, the activation of matching chunks is updated per time step. The function underlying this updating has two components. A long-term activation component that is identical to the current ACT-R activation formula (henceforth base-level activation), and a short-term, more volatile activation component that represents the current context (context activation). The total activation is calculated by summing both activation components, and retrieval is finished when this summed quantity reaches a fixed threshold (the context activation threshold $\theta^{context}$). As in the leaky accumulator models, the context activation is based on "evidence ticks". At each time step, if positive context evidence outweighs negative context evidence, the amount of evidence increases. This evidence, gathered during the retrieval phase. is subject to decay Each piece of evidence can be seen as a contribution to the context activation of that chunk. similar to how the prior occurrences of a chunk contribute to its base-level activation. But different from occurrences of a chunk, pieces of evidence cannot be considered having an infinitely high activation. since the chunk is not (yet) retrieved Therefore we modeled the volatile context activation using the ACT-R Optimized Equation:

$$C_i(t) = \ln(n/(1 - d^{evidence})) - d^{evidence} \ln(T)$$

This equation represents the Power Law of Learning (Anderson & Lebiere, 1998) As this equation represents the built-up of evidence, instead of T representing the age of prior occurrences, T is a representation of the age of the sequential sampling process.

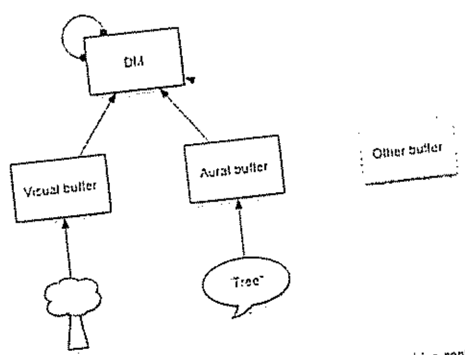


Figure 1 The RACE model. Chunks in the buffers spread positive context activation to chunks in declarative memory. Chunks in declarative spread negative association to each other (only during a retrieval process).

Whether a certain point in time is associated with evidence depends on the amount of context activation coming from the current buffers, context activation (currently only inhibition) coming from other chunks in declarative memory, and base-level activation (Figure 1): If at a certain time step excitatory context activation minus inhibitory context activation crosses a threshold, that time step is associated with positive evidence, and context activation increases A lack of positive evidence can be considered inhibition as the context activation decays strongly The name RACE reflects how different chunks compete on this basis of their accumulated evidence for retrieval

Using the RACE approach. the ACT-R latency equation can be rewritten as:

$$L_i = t \mid \left[(B_i(t) + C_i(t)) \geq \theta^{context}\right]$$

Note that $\sum WS$. the term reflecting context activation in the default ACT-R equations, is discounted for in the C_i component. as the slots of the goal buffer spread context activation to the associated chunks In this approach. the source of positive activation is still the chunk available in the buffers, and filled slots in the sensory buffers still inhibit retrievals as in the fan experiments

We conducted two experiments with RACE In experiment 1 we compared the predicted latencies of RACE with the predicted latencies from the ACT-R Latency Equation. In experiment 2 we fitted data from a Picture Word Interference experiment.

Experiment 1
Numerous models in ACT-R have shown that default ACT-R provides accurate predictions of retrieval latency (Anderson *et al*. 2004) To

ensure that our approach does not invalidate these results. we show that our model predicts the same latencies as the ACT-R Latency Equation in a non-competitive condition. In this experiment we fitted the ACT-R retrieval latency for different times after the chunk to be retrieved is presented In other words, we will predict the ACT-R latency for different base-level activation levels

We first chose reasonable parameters for the ACT-R Latency Equation These values were not updated in the optimization process. to ensure a fair comparison between the models. The crucial parameters in our model were the context activation threshold $\theta^{context}$, and the evidence threshold $\theta^{evidence}$ These indicate whether a chunk is retrieved ($\theta^{context}$) and whether evidence may be sampled ($\theta^{evidence}$). The evidence threshold was noisy with a standard deviation of σ=0.3 $d^{evidence}$ turned out to be less important in this condition because due to the absence of other chunks no inhibition was present, and evidence was sampled at almost every time step. However, because the age variable in the context activation function is in ms. the decay parameter and the time step frequency are related: If the frequency is high, so is the chance of sampling evidence (more opportunities), and decay may be higher All parameters are presented in Table 1

Table 1. Parameters experiment 1

RACE Parameters:	0.1
$d^{evidence}$	5
$\theta^{context}$	1000 Hz
time step frequency (f)	1.7
$\theta^{evidence}$	1
σ	
ACT-R Parameters:	0.5
$d^{base-level}$	0.35
F	

For a fixed set of retrieval onsets we calculated both the latency predicted by ACT-R and the prediction of our model. The retrieval onsets were chosen to ensure that different base-level activation levels were tested (0.5, 1.0, 3.0, 6.0. 9.0, 12.0, 15.0 seconds after chunk presentation) The experiment was performed 30 times. and the results were averaged The results are shown in Figure 2.

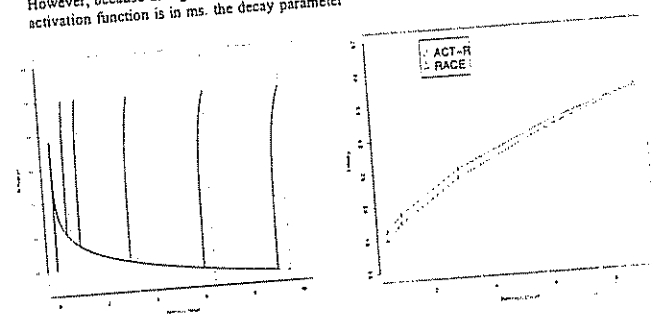


Figure 2. The retrieval process at different time steps (left) and associated latencies (right). Left: the grey dotted lines depict retrieval onsets. the black dotted lines depict predicted ACT-R latency, and the black solid lines depict activation. The simulation was terminated after reaching the retrieval threshold.

This simulation shows that in a single chunk retrieval task where similarity-based interference does not play a prominent role, the model predicts similar latencies to the classical ACT-R retrieval latency function Therefore. we assume that this extension does not limit ACT-R in its ability to capture the already modeled phenomena.

Experiment 2
Picture Word Interference experiments are characterized by a double stimulus paradigm, in which a distractor stimulus is presented at

different SOAs from the target stimulus, and subjects are asked to name the target stimulus (Glaser & Düngelhoff, 1984; Glaser & Glaser, 1989; Schriefers *et al.*, 1990) Distractors will interfere with the naming process, unless it is indicative of the same concept as the target stimulus, in which case its presentation will facilitate naming (see Figure 3, left adapted from (Glaser & Düngelhoff, 1984)). We will refer to this condition as *concept congruent* The conditions in which distractors interfere with the naming process will be referred to as *category congruent*, indicating that distractor and target are concepts from the same category, ensuring a high association, and *incongruent*, indicating that distractor and target are unrelated, ensuring that no association exists.

The addition of a short-term component makes it possible to predict priming effects at short SOAs As soon as a secondary stimulus is present in one of the buffers, this stimulus will start influencing the primary stimulus and thus cause the interference patterns typically observed in Picture Word-like experiments When the evidence threshold of a buffer chunk is crossed, evidence is sampled and context activation increases. This increase leads to a higher probability that evidence of positively associated chunks will be sampled. A higher context activation level of a chunk in declarative memory decreases in turn the likelihood that the evidence thresholds of competing chunks are crossed. Less evidence leads to an increased retrieval latency This process accounts for the latencies observed in the Picture-Word paradigm. Moreover, a highly associated chunk will decrease the likelihood that the evidence thresholds of competing chunks are crossed further than an unassociated chunk, with as a result a higher retrieval latency To keep the model simple and focus on the behavior of the concept component of the activation, base-level activation was set at a fixed constant of -2.5 The evidence threshold was set high enough that without external stimulation, evidence sampling was unlikely, and decay was set high enough that in the event of spontaneous evidence, context activation would quickly decay back to base-level

Table 2. Parameters experiment 2

RACE Parameters:	
$d^{evidence}$	0.5
$\theta^{context}$	0
f	1000 Hz
$\theta^{evidence}$	1.7
σ	1
B_i	-2.5

We ran our model through a series of retrieval experiments The model was set to retrieve concepts from memory that were highly associated with the presented items. At different SOAs (-400ms, -200ms, 0ms, 200ms, 400ms) distractor items were presented, and the latency was recorded. The results are presented in figure 3 (right).

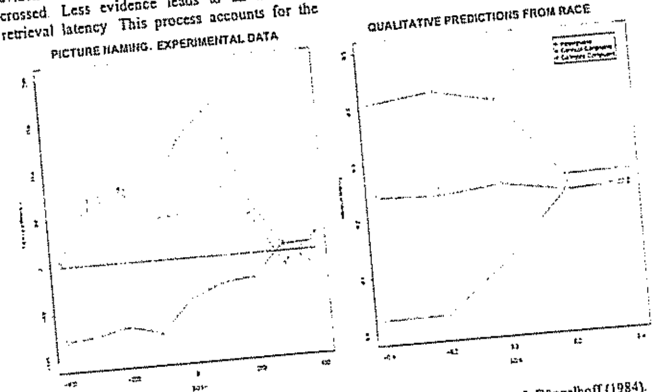


Figure 3. The left panel shows data from a Picture Naming Experiment by Glaser & Düngelhoff (1984). Distractors were visually presented. The right panel shows results from the RACE model.

Figure 3 (right) shows that both the facilitating and interfering effects are strongest at negative SOAs In those trials the distractor has more time to accumulate evidence, thereby influencing the target stronger when it is presented. At 0 or positive SOAs, the influence of the distractor is less as, on retrieval onset of the target, it does not have any already gathered evidence. For the concept congruent (facilitating) condition, a similar pattern is observed in the data, but for the interfering condition the data shows a peak at SOA=100ms, and smaller values at negative SOAs The current version of our model does not capture this, but could easily be extended to explain this effect: An explanation is that when a chunk is retrieved, accumulation of evidence stops, and the context activation quickly decays. However, the chunk itself is retrieved, and therefore has a higher base-level activation. If the prior stimulus was category congruent, this higher base-level slightly inhibits the target chunk, but not as much as when the distractor was still being processed. Therefore, the category congruent conditions at long negative SOAs will still show longer latencies than the neutral condition, but not as long as on positive SOAs If the prior stimulus was concept congruent the retrieval has increased the base-level of the concept that also needs to be retrieved for the target. Therefore, these conditions will still show significantly shorter latencies than the neutral condition In our model, however, base-level activation effects are not taken into account, but instead the context activation remains active, prolonging the interfering effect.

Another difference between the two panels of Figure 3 is the effect of the incongruent distractor condition In the left panel, this condition shows slower reaction times than the neutral condition (which is the condition in which no distractors are present) In our model, the neutral and incongruent distractor conditions have a similar effect as we did not aim at explaining these differences However, the model can easily be expanded to explain this effect. The expected increased retrieval latency in the incongruent distractor condition might be explained by assuming weak associations between the target and the incongruent distractor. The same effect as in the category congruent condition occurs, but to a lesser extent. Another possibility is that the activation of the incongruent concept is influenced by the number of competing chunks (i.e., the fan-effect). In the neutral condition (dotted line in figure 3), only one chunk is in the buffers In the other conditions, more chunks are present in the buffers. An extra competitor would mean that the "total amount" of activation is divided by one more, having a negative effect on the activation of the chunks and thus on the latency A third, equivalent option would be to calculate the *Luce Ratio* (Luce, 1959), indicating that the probability that a chunk will be retrieved depends on its part of the total activation. An extra competitor would mean higher total activation, and would in that way affect the latency of the target chunk, a mechanism which is implemented in the ACT-R competitive latency equation. We are planning to test these hypotheses in a next implementation of the model

Without much emphasis on optimization, RACE shows the most important phenomena observed in Picture-Word like tasks: a facilitating effect for the concept congruent stimulus, and a convergence of the effects to the neutral condition at positive SOAs. Besides the explanatory power for these previously unexplained effects, RACE is still compatible with current ACT-R latency predictions

Anderson, J. R., Bothell, D., Byrne, M. D., Douglass, S., Lebiere, C. & Qin, Y. (2004). An integrated theory of the mind. *Psychological Review, 111*(4), 1036-1060.

Anderson, J. R., Lebiere, C. (1998). *The Atomic Components of Thought.* Mahwah, NJ: Lawrence Erlbaum.

Glaser, W. R., & Düngelhoff, F. J. (1984). The time course of picture word interference. *Journal of Experimental Psychology-Human Perception and Performance, 10*(5), 640-654.

Glaser, W. R., & Glaser, M. O. (1989). Context effects in Stroop-like word and picture-processing. *Journal of Experimental Psychology-General, 118*(1), 13-42.

Luce, R. D. (1959). *Individual choice behavior.* New York, NY: Wiley.

Ratcliff, R., & Smith, P. L. (2004). A comparison of sequential sampling models for two-choice reaction time. *Psychological Review, 111*(2), 333-367.

Schriefers, H., Meyer, A. S., & Levelt, W. J. M. (1990). Exploring the time course of lexical access in language production: picture-word interference studies. *Journal of Memory and Language, 29*(1), 86-102.

Usher, M. & McClelland, J. L. (2001). The time course of perceptual choice: the leaky, competing accumulator model. *Psychological Review, 108*(3) 550-592

Modeling how delayed intentions impact current intentions in a prospective memory paradigm

Renée Elio
University of Alberta

Prospective memory is a term used to denote the process of remembering to do a particular action at some future time, either after some time period has elapsed or when some event has occurred (event-based prospective memory) The prospective memory literature also describes this process as setting and executing 'delayed intentions ' The typical laboratory paradigm for studying prospective memory requires people to remember to perform an infrequently occurring task (the delayed intention) whenever some event occurs in the environment; otherwise, they are preoccupied with executing some on-going cover task When the event occurs, the ongoing task must be interrupted, and the action associated with delayed intention must be performed There are two basic types of explanation offered for event-based prospective memory The automatic-retrieval explanation assumes that intentions have a special representation that includes its cue, and the appearance of the cue triggers the retrieval of the intention In contrast, monitoring explanation argues that performance of delayed intentions is never automatic, and requires ongoing preparatory and capacity-consuming processes, characterized as non-automatic monitoring of the environment for the target event Smith (2003) provides empirical evidence for this monitoring-explanation, which she argues cannot be accounted for by the automatic-retrieval explanation In brief, Smith found a reaction time cost to performing the on-going task, even when the prospective memory task itself was not being performed In my presentation, I will describe a relatively simple ACT-R model of Smith's main results This model uses an additional "control buffer" to effect a deliberation decision, which is the retrieval of some currently unsatisfied intention, and its associated preconditions for execution, from declarative memory The general trends of the Smith reaction time data emerge from a kind of competition among these intentions for the monitoring process Whether this is a plausible perspective from which to view these particular results is open to discussion As a computational account of the Smith data, the model provides some clarity about the descriptive characterizations of prospective memory as remembering to perform delayed intentions My main interest in modeling this data, however, was to better understand what it means to 'set' and process intentions within the current ACT-R modeling framework and its representational constraints My general goal is to relate the notions of intention cueing and intention monitoring, as used in descriptive prospective memory accounts, to theoretical distinctions in intention theory and also to computational mechanisms within a modeling framework such as ACT-R

Smith, R (2003). The cost of remembering to remember in event-based prospective memory: investigating the capacity demands of delayed intention performance *JEP LMC*, 29, 347-361

An ACT-R Based Investigation of Test and Study Temporal Dynamics

Philip I. Pavlik (ppavlik@andrew.cmu.edu)
Department of Psychology, Carnegie Mellon University
Pittsburgh, PA 15213 USA

1. Introduction

The questions to be answered here are somewhat long standing in the field of memory and cognition. The experiment grows out of questions raised in Pavlik and Anderson (2005). In this experiment, participants were trained in Japanese-English word pairs over the course of several hundred learning trials with a variety of spacing, repetition, and retention conditions. This experiment used a drill procedure for training. In this drill procedure, each item was introduced with a presentation of both members of the pair followed by spaced testing practice, which included corrective feedback immediately after any incorrect responses. This procedure was chosen because, according to ACT-R's assumption that test practice and study practice are equivalent, it should result in equal practice for each presentation regardless of the correctness of any particular response.

Of course, this assumption of equal effect for study and test practice is merely an approximation. A large variety of research has shown differences between test and study practice (Thompson, Wenger & Bartling, 1978; Runquist, 1983; Slamecka & Katsaiti, 1988; Carrier & Pashler, 1992; Cull, 2000). Though this work is interesting, much of it is incomplete or suffers from methodological flaws that make it difficult to come to clear conclusions about the differences between tests and studies. The first issue is that many of these prior studies have not clearly determined whether the advantage to testing is due to a benefit to encoding or a reduction in forgetting. A second question to be addressed is to what extant varying the duration of study opportunities affects learning.

2. Experiment Design

To examine the issues above requires a complex design. This design will use 2 spacing conditions (a spaced practice will follow either 2 or 30 trials after an initial study), 2 retention intervals (a performance test either 2 or 60 trials after the spaced practice trial), 5 practice types for the spaced practice (recall-or-study, test-and-study, pure-studies, pure-tests, or no practice), 2 study duration conditions for the spaced practice types (either 3 or 7 seconds for studies in the practice conditions that include study). Initial studies in all cases were fixed at 5 seconds.

Since, of the 5 spaced practice types, 2 do not include study presentations, there are 2 + 2 + 2 + 1 + 1 = 8 different study duration by practice type cells. Therefore, there are 8 x 2 = 16 study duration by practice type by spacing interval cells. Given the 2 retention intervals, there are 16 * 2 = 32 total cells within-subjects. This design will be repeated using two items per cell for each subject. These conditions were essentially delivered in 3 parts. The experiment began with 20 buffer trials to reduce primacy effects. Buffer items were always introduced with a study trial of 5 seconds and given recall-or-study trials for subsequent practice, with a 3-second study feedback in the case of failures to recall. Following these trials there were 160 trials during which the first replication of the design occurred, and then 160 more trials in which the second replication of the design occurred. Because the design itself required only 92 trials per replication, this meant that 68 trials were used for buffers in each case. A computer algorithm randomly interleaved the conditions with the buffers individually for each subject.

The stimuli and buffers were 100 Japanese-English word pairs. English words were chosen from the MRC Psycholinguistic database such that the words had familiarity ratings between 406 and 621, with a mean of 547, and had imagability ratings between 343 and 566, with a mean of 464.

All studies (whether they occurred as feedback or alone) and tests were cued with the prompts "Study" or "Test" for 5 seconds Tests involved presentation of the Japanese word on the left side of the screen Participants typed the English translation on the right If no response was made, the program timed-out in 10 seconds In the recall-or-study condition if correct the response was followed by a 0.5 second presentation of the word "Correct" and the next trial began If incorrect in the recall-or-restudy condition a study presentation for the word (which was introduced by the word "Study") was given The and test-and-study condition was identical to the recall-or-restudy condition if the response was incorrect, however, if correct the response was followed by a 0 5 second presentation of the word "Correct" followed by a study presentation for the word (which was introduced by the word "Study") In the pure-test condition, no feedback occurred following the test The pure-study and no practice conditions were self-explanatory

The experiment used 160 subjects recruited from the Pittsburgh, Pennsylvania community They were mostly college students responding to an online advertisement All participants completed the experiment. Eighty participants each were randomly assigned to 2 strategy conditions (free strategy or mnemonic training), data for which is not reported here for space reasons. Sessions lasted slightly less than one hour Only participants who professed no knowledge of Japanese were recruited

3. ACT-R Declarative Memory Model

The model used and developed in this report, an extension of the ACT-R theory, currently captures three major effects in declarative memory The ACT-R model captures the recency and frequency effects, i e that performance is better the more recently or frequently a memory item is practiced (Anderson & Lebiere, 1998) Anderson and Schooler (1991) originally developed this model by showing that memory strength for an item matches what would be optimal in the environment given the frequency and recency of usage of an item A recent extension of ACT-R (Pavlik & Anderson, 2005) captures the spacing effect

These effects are captured by an activation equation that represents the strength of an item in memory as the sum of these differences and the benefits from a number of individual memory strengthenings each of which corresponds to a past practice event (either a memory retrieval or study event) Eq 1 proposes that each time an item is practiced the activation of the item, m_i, receives an increment in strength that decays away as a power function of time

To deal with the spacing effect Pavlik and Anderson (2005) developed an equation in which decay for the i^{th} trial, d_i, is a function of the activation at the time it occurs The implication of this is that higher activation at the time of a practice will result in the benefit of *that* practice decaying more quickly On the other hand, if activation is low, decay will proceed more slowly It is important to note that *every practice has its own* d_i that controls the forgetting of that practice Specifically, I propose Eq 2 to specify how the decay rate d_i is calculated for the i^{th} presentation of an item as a function of the activation m_{i-1} at the time the presentation occurred. Eq 1 shows how the activation m_n after n presentations depends on these decay rates, d_i's, for the past trials

$$m_n(t_{1..n}) = \ln\left(\sum_{i=1}^{n} t_i^{-d_i}\right) \qquad \text{Eq 1}$$

$$d_i(m_{i-1}) = ce^{m_{i-1}} + a \qquad \text{Eq 2}$$

In Eq 2, c is the decay scale parameter, and a is the intercept of the decay function For the first practice of any sequence, $d_1 = a$ since m_0 is equal to negative infinity These equations are recursive because to calculate any particular m_n one must have previously calculated all prior m_n's to calculate the d_i's needed These equations result in a steady decrease in the long-run retention benefit for more presentations in a sequence of closely spaced presentations As spacing gets wider in such a sequence, activation has time to decrease between presentations, decay is then lower for new presentations, and long-run effects do not decrease as much

4. Results and Discussion

The performance data were aggregated by condition and several repeated measures ANOVAs were completed to examine main-effects and interactions in the data The first ANOVA (retention x spacing x study duration x trial type x strategy condition) compared the test-and-study, recall-or-restudy, and pure-spacing trial type performance after the retention interval. Main effects were as expected, with retention, spacing, study duration, and trial type all having significant effects [$F(1, 158) = 1080, p < 0.001$, $F(1, 158) = 5.81, p < .001$, $F(1, 158) = 15.2, p < 0.001$, and $F(1, 158) = 430.9, p < 0.001$]

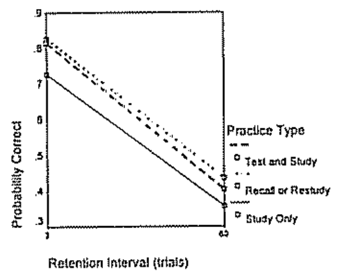


Figure 1 Effect of retention interval depending on trial type

Of primary importance, this analysis showed no indication of a retention by trial type interaction If study practice leads to a less permanent memory encoding, such an interaction should occur since forgetting in the study only condition would be faster than in those conditions that include testing. See Figure 1 As can be seen, there is no suggestion of quicker forgetting in the study only condition

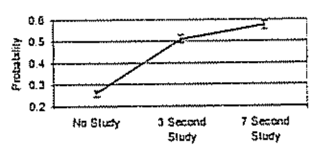


Figure 2 Effect of study duration on recall for study only condition

A second repeated measures ANOVA was identical in design to the first, but only included the pure-study condition. Main effects were as expected, with retention, spacing, and study duration (3 or 7), all having significant effects [$F(1, 158) = 398, p < 0.001$, $F(1, 158) = 7.44, p < .01$, and $F(1, 158) = 14.4, p < 0.001$] Again the retention by spacing interval interaction was significant, $F(1, 158) = 10.8$, $p < 0.01$ Figure 2 shows the 3-point study duration function (the no study point comes from the control condition which included no practice of any sort after the spacing interval)

5. Modeling the Results

The fits detailed here involved capturing the aggregate results of the experiment While this is a fit of aggregate data points, the model produces these aggregate data points by averaging a model of each trial of each sequence of practice of each word for each subject Of course, since this model does not consider individual differences in ability between subjects, difference in what the model predicts between subjects will come only from different histories of tests and studies in those cases where scheduling is contingent on recall, and from the different time values that occur for faster or slower subjects

5.1 *Tested Study Models*

Twenty declarative memory equation models of all the conditions of the experiment (except the test only condition which results in selection effects which can only be handled by a more elaborate version of the model) were tested and compared These twenty models were configured in a 5 x 4 design with 5 different study models and 4 different combinations of c and/or s being optimized In addition to the free parameters optimized for each of the 20 cells of this design, all of the models also fit a model of retrieval latency (not reported here) For all of these models a and τ were fixed at Pavlik and Anderson (2005) values, and in models where c and s were fixed, they were also fixed at these values

Using c and s as the main free parameters allows them to capture more accurately the slopes of the practice and forgetting functions, while fitting the study functions captures initial practice better. While the model used the spacing effect mechanisms discussed in Pavlik and Anderson (2005), they were not integral to the models of study duration Since study practice had no significant effects on the decay rate, the models worked by assuming that the strength of each $t_i^{-d_i}$ in the activation equation (Eq. 1) should be weighted to capture the effect of study trials (Eq. 3) In all of these models, retrieval practice was fixed at a weighting of 1. Equation 3 shows how the more complex models use a value (b) to scale strengths in the activation equation.

$$m_n(t_{1..n}) = \ln\left(\sum_{i=1}^{n} b_i t_i^{-d_i}\right) \qquad \text{Eq. 3}$$

Model 1 tested the standard ACT-R assumption that study practice also has a weight equal to 1 This is the comparison condition with no new parameters from which performance of the additional processes and parameters in the follow models can be judged.

Model 2 tested the hypothesis that study weight is constant, but not equal to 1 This model corresponds to the idea that studies may be simply weaker than tests and that study duration does not matter much. (In this case the $b=m$ in Eq 3)

Model 3 tested the idea that study weight is a linear function of study duration This is closely equivalent to one interpretation of how to count the benefit of each study practice in ACT-R Unlike Model 1 in which a single study opportunity occurs whenever study practice of a stimulus is offered to a participant, in this conceptualization a study occurs every 370ms The 370ms figure comes from ACT-R's perceptual motor assumptions (not discussed here) and corresponds to an estimate of the minimal time necessary to form an association While this model is suggested by ACT-R, it seems to be in direct conflict with the spacing effect since this model results in no penalty for 2 back-to-back study trials (e.g one 740ms study trial) in comparison to two 370ms studies spaced apart. (In this case $b = m/10000*(time-370ms)$ in Eq. 3.)

Model 4 tested the model where $b = m(1-e^{(time-370ms)\,v})$, where m is the maximum benefit of study and v describes the rate of approach to the maximum. This simple model is intended to capture the fact that study practice has diminishing marginal returns and appears to reach an asymptote (Metcalfe & Kornell, 2003). It is used to compute activation according to Eq. 3.

Model 5 was formulated and added to the list of hypothetical models because of misfit of Model 4 Model 5 was designed to capture the fact that study following a failed retrieval had a strong tendency in the data to proceed more effectively than study practice alone (study practice after a success is not so relevant since the high decay in this case wipes out gain). This model tests the assumption that after a failed test feedback-study proceeds more quickly due to prior cue encoding by using 2 values for v in the Eq 3

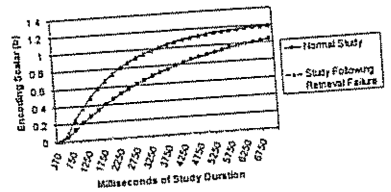


Figure 3 Effect of study on the encoding strength scalar parameter for best fitting model

Rather than fitting 2 values for v, which works only slightly better, the model assumed a simple process explanation to explain the value of v in terms of the stimulus In this conceptualization, the v is divided by the number of terms in the stimulus. This component of the model says that during study trials subjects deploy an attentional resource (typically in a strategic fashion, but also through rote processes) to encode the stimulus being studied Because this resource is limited, it must be divided among the components of the stimulus (in the model this is done by dividing the encoding rate by the stimulus size.) This mechanism explains the advantage of a study after a failed test comes from the opportunity to pre-encode the cue Because of this pre-encoding, during the following study opportunity the encoding of the single response term proceeds twice as quickly. The exact function mapping number of stimulus terms to the value of v remains uncertain based on the following research, more research will certainly be needed to determine its true form

5.2 *Fit of Models*

To fit these models I simultaneously found the best latency model parameters, study model parameters, and c and/or s to minimize the sum of an overall fit statistic for the 20 models tested. Since fitting latency had very little effect on the fit for correctness, here I will only report values for the 44 *df* correctness model Figure 4 shows the best model ($\chi^2 = 66.1$).

Table 1. Parameters and Model Statistics

	Parameters and Model Statistics				
Study Model	c	s	m	v	χ^2 Recall
1	0.401	0.340	1.000	n/a	245.77
2	0.312	0.340	0.901	n/a	116.93
3	0.323	0.342	1.960	n/a	134.38
4	0.305	0.333	1.100	0.373	84.81
5	0.344	0.341	1.217	0.582	66.11

Table 1 summarizes the fit of the models which varied both c and s. models (varying only one or less of these 2 parameters resulted in poor fits χ^2s > 160) As Table 1 shows, model 5 is clearly superior, with model 4 performing fairly well

In contrast, the other two variable study effect models (2 and 3) fit somewhat adequately because of the fact that they were able to count study practice as being less than test practice. This is particularly true in the case of model 2 in which study practice has about 90% of the effect of test practice. Study model 1, the current ACT-R assumption performed particularly poorly.

For the best fitting model of the 20, the study parameters m and v were 1.217 and .582 These values describe the effect of study duration on encoding strength shown in Figure 3

The fit to the main conditions for this best fitting model is shown in Figure 4, which includes all 44 data points modeled The fit is clearly rather good and captures all the main effects and significant interactions

Modelling the Paradoxes of Decision–Making

Roman V. Belavkin (r.belavkin@mdx.ac.uk)
School of Computing Science
Middlesex University, London NW4 4BT, UK

8 July 2005

OVERVIEW

1. Expected utility and ACT–R
2. The Rational donkey paradox
3. Noise and dynamic variance
4. The Allais paradox
5. The random utility solution
6. The Ellsberg paradox
7. Future work

Abstract

Decision–making paradoxes, such as the Allais and Ellsberg paradoxes, have shaken the foundations of the rational decision–making theories (e.g. the von Neumann-Morgestern approach and Savage axioms). These paradoxes deserve special attention because they point at our lack of understanding of how the brain makes decisions. Moreover, a theory of mind should be able to explain and predict these effects. I will attempt to explain these paradoxes using the random utility theory of choice. Then I will discuss how ACT–R can use this theory through a slight modification of the conflict resolution mechanism and will demonstrate the results on simple models.

DECISION MAKING

- Classical decision–making theory is due to von Neumann and Morgenstern (1944), Savage (1954) and Anscombe and Aumann (1963).
- Despite the differences in treating the *uncertainty*, the main idea is that of *utility* and its *expected value* (the EU), and the choice made by maximising EU

$$\text{Decision}(i) = \arg_i \max \sum_{i=1}^{n} P_i U_i$$

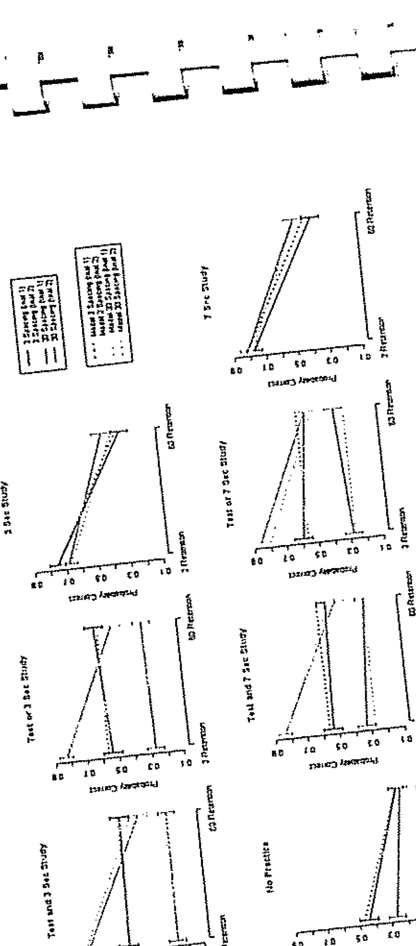


Figure 4 Correctness data and model for the experiment 2 SE bars

6 Discussion

The results here show that the differences between tests and studies can have important implications for recall Fortunately, these differences seem to be captured fairly well by scaling the contribution of each practice This analysis does not discuss the origin of these functions Certainly subjects can engage in a variety of study process during any particular study event, and the results here average over these process and differences by subject in order to capture an aggregate result This tells us very little about the underlying strategies subjects use Fortunately the methods of modeling applied here can be extended to capture results for individual strategies by condition This sort of analysis (omitted for space reasons) shows that mnemonic training results in a lower ν parameter (indicating slower encoding) and a higher m parameter (indicating a higher level of learning) for study model 5 This more detailed fitting does provide interesting reflections of subject processes; however, the large amount of seemingly arbitrary trial to trial variability in the strategies subjects exhibit may make it impossible to do more than fit a different aggregate study function for each subject

7. References

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

Anderson, J R. & Schooler. L. J (1991) Reflections of the environment in memory *Psychological Science* 2 396-408

Carrier, M., & Pashler, H. (1992) The influence of retrieval on reention *Memory & Cognition*, 20, 633-642.

Cull, W. (2000) Untangling the benefits of multiple tstudy opportunities and repeated testing for cued recall *Applied Cognitive Psychology*, *14*, 215-235

Pavlik Jr , P I., & Anderson, J R. (in press) Practice and Forgetting Effects on Vocabulary Memory: An Activation-Based Model of the Spacing Effect. *Cognitive Science*.

Runquist, W. (1983) Some effects of remembering on forgetting. *Memory & Cognition*, *11*, 641-650

Slamecka. N J & Katsaiti. L T (1988) Journal of Experimental Psychology: Learning. Memory. and Cognition. 14. 716-727

Thompson, C P., Wenger. S. K., & Bartling, C A (1978) How recall facilitates subsequent recall: A reappraisal *Journal of Experimental Psychology: Human Learning and Memory*. *4*. 210-221

GAMMA NOISE (OPTIMIST)

- The uncertainty about the utility can be used directly control the variance from probability distributions.
- The time component of the cost can be estimated using Poisson distribution $p = 1 - e^{-1/\theta}$ (Belavkin, 2003)

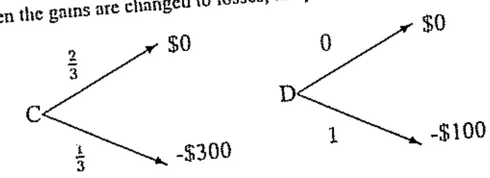
$$U_i = P_i G - \mathrm{Gamma}(\theta_i)\,, \quad \text{where } \theta = \frac{\text{Efforts}}{\text{Successes}}$$

- The OPTIMIST overlay (Belavkin & Ritter, 2004) for ACT–R is available at

http://www.cs.mdx.ac.uk/staffpages/rvb/

THE ALLAIS PARADOX (LOSSES)

When the gains are changed to losses, the preferences reverse

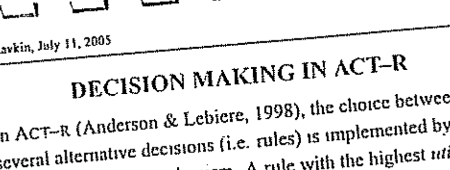


$\frac{2}{3} \cdot 0 - \frac{1}{3} \cdot \$300 = -\$100 \qquad 0 \cdot \$ - 1 \cdot \$100 = -\$100$

About 80% of subjects express preference $C \succ D$

THE ALLAIS PARADOX (GAINS)

Due to Allais (1953). Also studied by Tversky and Kahneman (1974) in many interpretations. Consider two lotteries A and B



$\frac{1}{3} \cdot \$300 + \frac{2}{3} \cdot \$0 = \$100 \qquad 1 \cdot \$100 + 0 \cdot \$0 = \$100$

About 80% of subjects express preference $A \prec B$

FRAMING OF DECISIONS

- Tversky and Kahneman (1974) suggested *decision framing* theory of using a function $\pi(P)$ of the probability.
- In ACT–R, one suggests to use G as the 'framing' global parameter

Lottery A and B $\frac{1}{3} \cdot G - \$0 \prec 1 \cdot G - \$0$

Lottery C and D $\frac{2}{3} \cdot G - \$0 \succ 1 \cdot G - \$100$

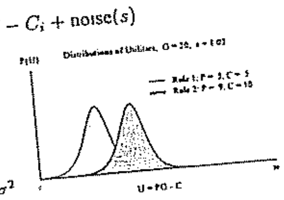
- However, the above formulae are incorrect as C should also be relative to goal value G. The correct formula is $P(U - G)$
- Note also that not 100% of subjects preferred as above.

DECISION MAKING IN ACT–R

In ACT–R (Anderson & Lebiere, 1998), the choice between several alternative decisions (i.e. rules) is implemented by the conflict resolution mechanism. A rule with the highest *utility* is selected: $i = \arg\max U_i$, where

$$U_i = P_i G - C_i + \mathrm{noise}(s)$$

rule's properties
P_i – probability of success
C_i – cost (e.g. time)
global parameters (constants)
G – goal value
s – controls noise variance σ^2



THE RATIONAL DONKEY PARADOX

?

↙ ↘

$\boxed{\text{Haystack A}}$ $\boxed{\text{Haystack B}}$

- max EU theory fails when there is no unique max.
- ACT–R uses noise (:egs) which ensures this does not happen
- How large should be noise variance?
- There are other paradoxes related to max EU.

ACT–R AND EXPECTED UTILITY

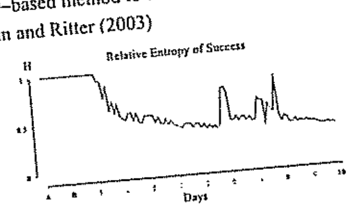
- For each decision, two outcomes: Success ∨ Failure
- Let $U^s = U(\text{Success})$ and $U^f = U(\text{Failure})$. Then

$$\begin{aligned} E\{U\} &= P^s U^s + P^f U^f \\ &= P^s U^s + (1 - P^s) U^f \\ &= P^s (U^s - U^f) + U^f \end{aligned}$$

- If $G = U^s - U^f$ and $U^f = -C$, then $E\{U\} = PG - C$
- ACT–R uses the expected utility and therefore is prone to all the paradoxes.

DYNAMIC EXPECTED GAIN NOISE

- Dynamic noise variance has been discussed recently (e.g. Belavkin, 2001; Taatgen, 2001)
- Entropy–based method to control :egs was proposed in Belavkin and Ritter (2003)



RANDOM UTILITY IN ACT–R

Each rule i has history of successes and failures $P(\text{Outcome} \mid i)$. For a set of conflicting rules, the following scheme is used to generate random utilities RU_i

$$\begin{aligned} P(\text{Outcome} \mid i) &\to \text{Success} \vee \text{Failure} \\ RU_i &= U_i^s \vee U_i^f \\ &= G + U_i^f \vee U_i^f \\ &= G - C_i \vee -C_i \end{aligned}$$

where C_i is the cost. We can also use Gamma noise

$$RU_i = G - \text{Gamma}(\theta_i) \vee -\text{Gamma}(\theta_i)$$

PROPERTIES OF RANDOM UTILITY

- The expected value of random utility

$$\begin{aligned} E\{RU_i\} &= P_i(G - C_i) - (1 - P_i)C_i \\ &= P_i G - C_i \end{aligned}$$

- Allows to model the Allais paradox
- The use of Gamma noise implements the features of the OPTIMIST conflict resolution: Rule specific and dynamic noise variance $\sigma^2 = \theta^2$

RANDOM UTILITY

For each decision i, the outcome is sampled from its distribution $P(\text{Outcome} \mid i)$ conditional to rule i. The utility of this outcome is called *random utility* RU_i

Decision $i = \arg\max RU_i$, where $RU_i \leftarrow P(\text{Outcome} \mid i)$

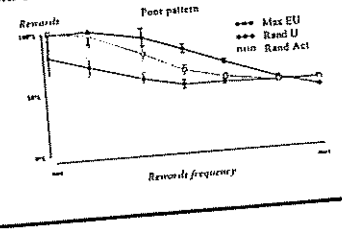
Here $P(\cdot \mid i)$ is probability distribution of successes and failures for a given rule, and RU_i is the utility of each outcome.

Sampling can be implemented using the inverse PDF method

$$\text{Outcome} = F^{-1}(P), \quad \text{where } P \in (0,1)$$

RANDOM UTILITY vs max EU

- Tested on agents with Bayesian learning of Markov Decision models (i.e. transitional probability tables P_{ij}^k).
- The random utility agents are as good as the max EU agent, and often outperformed them 2:1 (Belavkin, 2005)



THE ELLSBERG PARADOX

Due to Ellsberg (1961). Consider two lotteries A and B, and probabilities of outcomes for A are given

A: $\frac{1}{2}$ → \$100, $\frac{1}{2}$ → \$0 $\qquad$ B: $(1,0)$ → \$100, $(0,1)$ → \$0

$EU(A) = \$50 \qquad EU(B) = \$50$

$$A \succ B$$

UNCERTAINTY OF INFORMATION

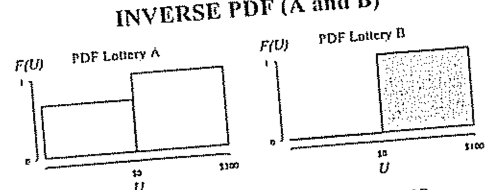
$$PU^s + (1 - P)U^f$$

Although the expected utilities are the same, the procedures involved in choosing are clearly different

$$\frac{1}{2} \cdot \$100 + \frac{1}{2} \cdot \$0 \quad \neq \quad \begin{cases} \frac{1}{100} \cdot \$100 + \frac{99}{100} \cdot \$0 \\ \frac{99}{100} \cdot \$100 + \frac{1}{100} \cdot \$0 \end{cases}$$

Using random utility would involve drawing two samples in lottery B (one for P and one for U) while only one sample is needed for lottery A, and may be perceived as less risky.

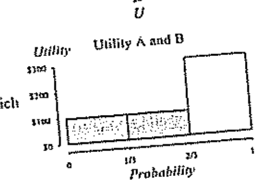
INVERSE PDF (A and B)



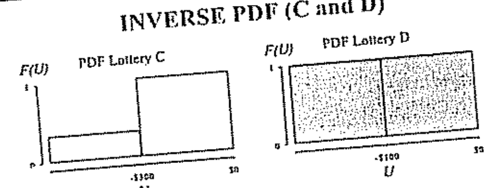
$$\text{Utility} = F^{-1}(P)$$

$RU_A < RU_B$ 2 out of 3 times, which supports experimental evidence

$$A \prec B$$



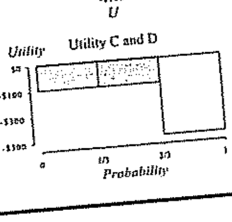
INVERSE PDF (C and D)



$$\text{Utility} = F^{-1}(P)$$

$RU_C > RU_D$ 2 out of 3 times. Again, corresponds to experimental results

$$C \succ D$$



CogWorks HCIL

CogWorks Visualization Architecture

Cognitively Engineering Next Generation Workstations for Decision Makers

Stephane Gamard, Michael J. Schoelles, Christopher Kofila, V. Daniel Veksler, & Wayne D. Gray

ACT-R Workshop
Trieste, Italy
15-17 July 2005

Rensselaer | Cognitive Science

CogWorks

- Wayne D. Gray
  - Professor & Director CogWorks Laboratory
- Michael Schoelles
  - Research Professor & co-Director CogWorks Laboratory
- Stephane Gamard
  - Visiting Scholar, Institut National Polytechnique de Grenoble, France
- Chris Kofila
  - Lab Assistant
- V. Daniel Veksler
  - Research Assistant

Rensselaer | Cognitive Science

Interface Design

- Which interface works best with what new technology?
- A bad interface to a good tool can negate the utility of the tool
- How can we ensure that the new technologies will bridge the human-computer-information gap?

HCIL

Rensselaer | Cognitive Science

The Focus Today

- The CogWorks Visualizations Architecture (VizArch)
- An enabling technology
- First will enable research, and
- Then will enable the development of theory-based tools and techniques for the rapid evaluation and redesign of complex visualizations

Rensselaer | Cognitive Science

CONCLUSIONS

- The Expected utility theory is probably not a good model of the decision-making in the brain.
- Cognitive architectures and ACT-R need to consider the paradoxes arising from the max *EU* principle.
- The random utility method has been suggested as a cost-effective solution to the problem.
- The role of uncertainty in decision-making is not well understood (e.g. Ellsberg, 1961).

References

Allais, M. (1953). Le comportement de l'homme rationnel devant le risque: Critique des postulats et axiomes de l'Ecole americaine. *Econometrica, 21*, 503–546.

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

Anscombe, F. J., & Aumann, R. J. (1963). A definition of subjective probability. *Annals of Mathematical Statistics, 34*, 199–205.

Belavkin, R. V. (2001, March). The role of emotion in problem solving. In C. Johnson (Ed.), *Proceedings of the AISB'01 Symposium on Emotion, Cognition and Affective Computing* (pp. 49–57). Heslington, York, England: AISB. (ISBN 1-902956-19-7)

Belavkin, R. V. (2003). *On emotion, learning and uncertainty: A cognitive modelling approach*. PhD Thesis, The University of

Nottingham, Nottingham, UK.

Belavkin, R. V. (2005). *Acting irrationally to improve performance in stochastic worlds*. (Submitted to *The Twenty-fifth SGAI International Conference on Innovative Techniques and Applications of Artificial Intelligence*)

Belavkin, R. V., & Ritter, F. E. (2003, April). The use of entropy for analysis and control of cognitive models. In F. Detje, D. Dörner, & H. Schaub (Eds.), *Proceedings of the Fifth International Conference on Cognitive Modelling* (pp. 21–26). Bamberg, Germany: Universitäts-Verlag Bamberg. (ISBN 3-933463-15-7)

Belavkin, R. V., & Ritter, F. E. (2004). Optimist: A new conflict resolution algorithm for ACT-R. In *Proceedings of the Sixth International Conference on Cognitive Modelling* (pp. 40–45). Mahwah, NJ: Lawrence Erlbaum. (ISBN 0-8058-5426-6)

Ellsberg, D. (1961, November). Risk, ambiguity, and the Savage ax-

ioms. *The Quarterly Journal of Economics, 75*(4), 643–669.

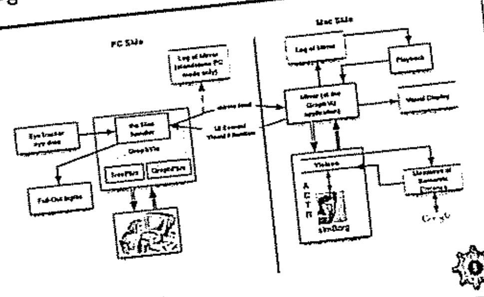
Neumann, J. von, & Morgenstern, O. (1944). *Theory of games and economic behavior* (first ed.). Princeton, NJ: Princeton University Press.

Savage, L. (1954). *The foundations of statistics*. New York: John Wiley & Sons.

Taatgen, N. A. (2001, July). Production compilation. In *Eights annual post-graduate summer school*. Retrieved from act-r.psy.cmu.edu.

Tversky, A., & Kahneman, D. (1974). Judgment under uncertainty: Heuristics and biases. *Science, 185*, 1124–1131.

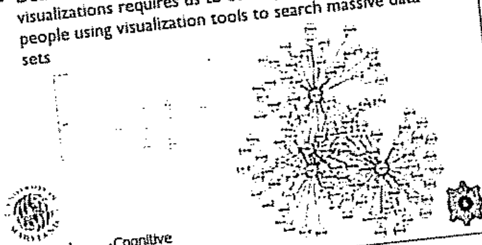
CogWorks VizArch



Rensselaer | Cognitive Science

CogWorks VizArch

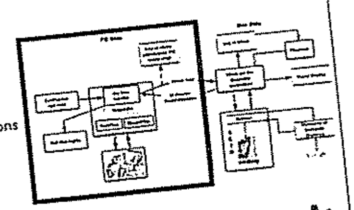
- Detailed study of humans use of UMd's innovative visualizations requires us to do empirical studies of people using visualization tools to search massive data sets



Rensselaer | Cognitive Science

1st Mode: Human Data Collection

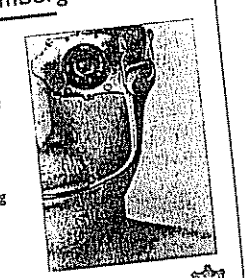
- *Human Mode*
  - One machine
  - All user interactions
  - All system states
  - Eye data



Rensselaer | Cognitive Science

2nd Mode: Collecting Data from simBorgs

- To be used to evaluate alternative designs, simBorgs need to
  - interact with the same software interfaces as people, and
  - make human-like judgments during their searches



Rensselaer | Cognitive Science

Human Search Thru Large Data Sets

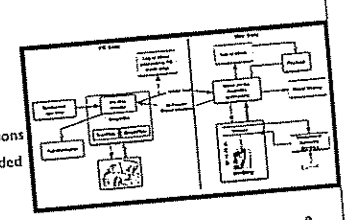
- When searching for information people follow the *information scent*
- The weaker the scent, the more likely we are to abandon the trail we have been following and either backup or start over
- The stronger the scent the more likely we are to click on the link, open the file, etc

parc

Rensselaer | Cognitive Science

2nd Mode: Collecting Data from simBorgs

- *simBorg Mode*
  - Two machines
  - All system states
  - All simBorg interactions
  - simBorgs search guided by *information scent*



Rensselaer | Cognitive Science

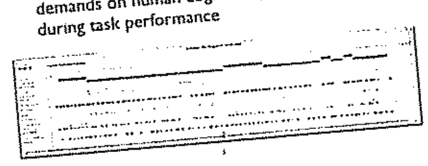
simBorgs as Information Foragers

- To the extent that a visual interface is not sufficient to guide the human or misleads the human, it should also fail to guide or mislead the simBorg

Rensselaer | Cognitive Science

Forecasting Dynamic Changes in Cognitive Workload

- Log file that simBorg produces will be used to generate a Cognitive Metrics Profile of the changing demands on human cognitive, perception, and action during task performance



Rensselaer | Cognitive Science

The Result

- Cognitive Metrics Profiles that
  - Reveal the *dynamic changes in interface demands on the users cognitive, perceptual, and motor resources*

Rensselaer | Cognitive Science

Examples

- Temple of the Sun
- Blocks World
  - Focus on memory demands

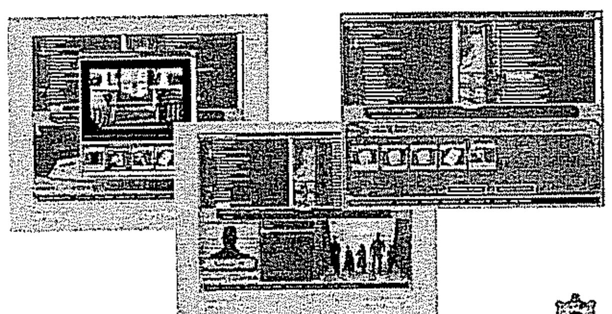
Rensselaer | Cognitive Science

Example: *Temple of the Sun*

- Temple of the Sun (ToS) is a synthetic task environment designed to study intelligence analysts in a non-classified environment

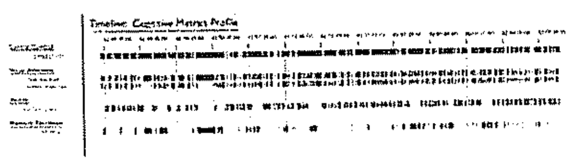
Rensselaer | Cognitive Science

Example: *Temple of the Sun*



Rensselaer | Cognitive Science

CMP of ToS



Rensselaer | Cognitive Science

Blocks World

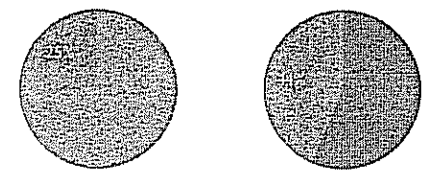
- Used to illustrate some of the features of cognitive metrics profiling
- Easy to explain Blocks World task in a short talk

Rensselaer | Cognitive Science

Cognitive Metric Profiling Reveals Hidden Workload

Encode Blocks

Retrieve & Place



Rensselaer | Cognitive Science

When to Profile?

- Not during initial design
  - Designers should not try to second guess how a design stresses visual attention, memory, or whatever
- Focus on designing consistent systems that are
  - Pleasurable to use
  - Easy to learn
  - Easy to recover from errors
  - & that meet the general performance requirements for which the system is being designed

Rensselaer | Cognitive Science

When to Profile?

- As in software development: *Profile before Optimizing*
- Before changing the interface to reduce workload
  - Profile the model to see where it's actually spending its time
  - Focus on the few high-payoff areas and leave the rest alone

Rensselaer | Cognitive Science

Results of Profiling

- Profiling focuses on the *dynamic changes in workload that the interface imposes* on the users
- *In a multitasking world any one tool cannot be permitted to greedily hoard the user's cognitive resources*
- *CMP is the fMRI for applied modeling*

Rensselaer | Cognitive Science

Advances Required to Extract CMP from Computation Cognitive Models Data

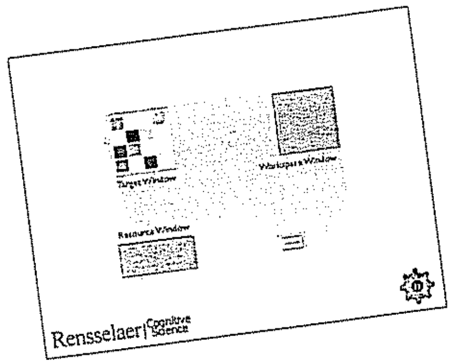
- Extend 6.0 by adding CMP module
- Write cognitive metric logfile into format that is directly importable into MacSHAPA™

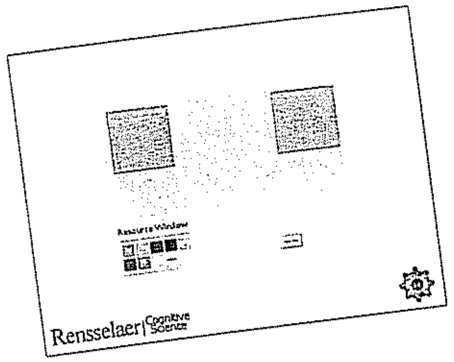
Rensselaer | Cognitive Science

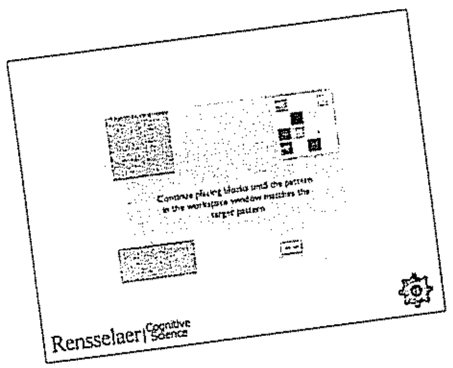
CogWorks

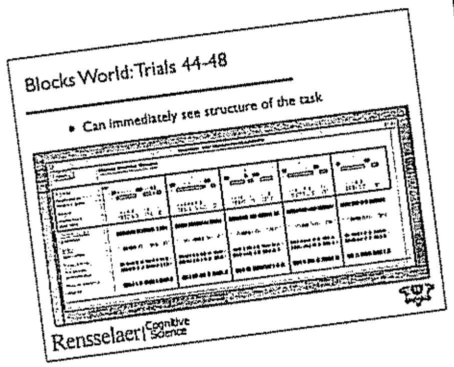
Questions?

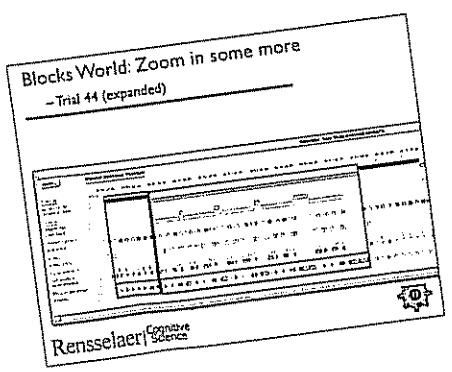
Rensselaer | Cognitive Science

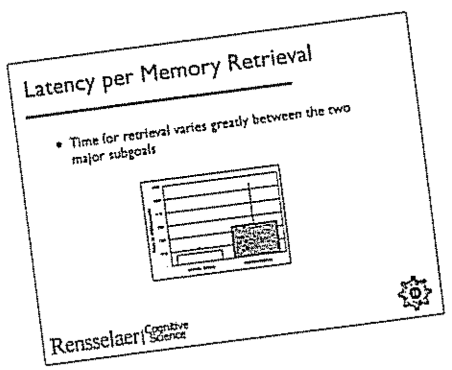












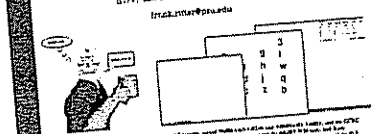
Kevin Gluck
Glenn Gunzelmann
Air Force Research Laboratory
Mesa, AZ USA 85212

Informative Failures on the Path to a Theory of Degraded Cognition

Factors such as extended wakefulness, chronic sleep restriction, and circadian desynchrony reduce cognitive effectiveness and have dramatic effects on human performance We have a research effort underway to identify or create mechanisms in ACT-R that account for (and eventually predict) changes in performance that result from these degraded cognitive states We have made some progress, and the current set of mechanisms is described in a paper to appear in this year's Cognitive Science Society proceedings The current mechanisms were not obvious to us initially and were attempted only after other approaches failed They also include varying parameters that traditionally are considered fixed (e g , utility threshold) This begs the question: what mechanisms did we try that proved to be insufficient for modeling degraded cognition? This presentation will focus on these informative failures on the path to a theory of degraded cognition

The effects of pre-task appraisals and caffeine on cognition: Data and models

Frank E. Ritter & Andrew Reifers
Laura C. Klein, Courtney Whetzel
Mike Schoelles, Karen Quigley
IST/Psy, BBH @ Penn State, RPI, DVA/UMDNJ
frank.ritter@psu.edu



Overview of Presentation

- Impacts
  - The effects of stress and caffeine on cognition
  - Stress, caffeine, & cortisol w/implications for health
  - Lessons on testing large scale theories
- Overview of our research line
  - Blascovich, Lazarus, and other work, short review
- Tasks, models, and data. CafeNav Project
  - 1. Task and model suite
    - Approach/initial fits/analyses
  - 2. Large study. CafeNav (N=45 x 3 tasks)
    - CafeNav-Σ study (update, 45/45, preview of results)
    - CafeNav-Argus study (update: 8/45 + 17)
    - Implication: Caffeine moderate dosages
  - 3. Overlays, theories of stress on cognition
    - a. Appraisal & Stress Overlays
    - b. Caffeine review
    - Implication: Caffeine low dosages
- Conclusions and future work

Motivation for Studying Moderators

- Behavioral moderators appear to influence cognition
  (maybe they don't, we just remember things differently!)
  - heat
  - affect
  - stress (multiple causes)
- Important for understanding aspects of human-computer, human-object interactions
- Language can be muddled: affect, emotions, moods, arousal
- Work in this area has not combined physiology and cognition that often (e.g., performance on cognitive-stressor not recorded)

Motivation for *Modeling* Moderators

- Modeling behavioral moderators that influence architecture processing
  - Development
  - Affect
  - Stress (multiple causes)
- Important for modeling aspects of human-computer interactions
- Extending applied models from Quake to ModSAF
- Example validated model near affect

Previous Approaches to Stress/emotions and Cognition

- Physiology studies
  - Examples: Blascovich, Klein, Lazarus, Lieberman
- AI & Cognitive Science
  - Examples: Sloman, Picard, Self-El Nessar, Norling & Ritter
- Human Factors
  - Examples: Woods, Hancock, and in overlays
- Cognitive Science
  - Belavkin, Gunzelmann, Chong, Jongman
- Perhaps need for several approaches

Our Approach

(Ritter, Avraamides, & Councill, 2002)

- Cognitive architecture
  (e.g., ACT-R, COJACK) (Ritter, 2004)
- Biopsychology models and data
- Validation of model's behavior
- Specifically
  - Task appraisal
    ("Challenging" or "Threatening")
  - Caffeine
- Displays to explain model to
  - analysts
  - readers

1. CafeNav Measures and Tasks (N=45+8/135)

- Heart rate, BP/3 min., Cortisol, αAm. DHEA TimeE [Taatgen], mood, appraisal
- Visual signal detection task [Reifers, task, model 5, d', ß]
- Simple reaction time task [Reifers, task, model 5, simple RT]
- Working memory task (MODS) [Reder-Lovett-Lebiere, task, model 4, W]

(a) Serial subtraction task [Reifers, voice task & keyboard task, model in 5, RT errors]
(b) Argus Prime [Schoelles, task, model in 5 about 6 measures]
(c) Argus Prime - Dual task

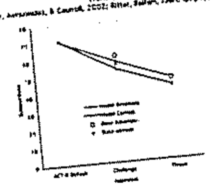
1. Task & Model Suite

WM task (MODS, vers A & B) (MCL)	Act-R 4 (headed to 5)
VSDT task (vers A & B) (MCL)	Act-R 5 + EMMA
RT (MCL)	Act-R 5 + EMMA
Time estimation (paper)	Act-R 5 (Taatgen)
Serial subtraction (paper, keypad, Allegro)	Act-R 5
Argus (ATC-like task) (MCL)	(Act-R 5)
Argus Dual-task (MCL)	(Act-R 5)

AC T-R (4) Model of Subtraction

- Create goal to serial subtract
  - Subgoal to do current column
    - Two strategies: count-down and subtract
    - Get column answer
  - Repeat across columns
  - Report result
- 28 rules
- 15 state chunks + 230 math facts (~250 total chunks)

Predicted and Actual on Serial Subtractions



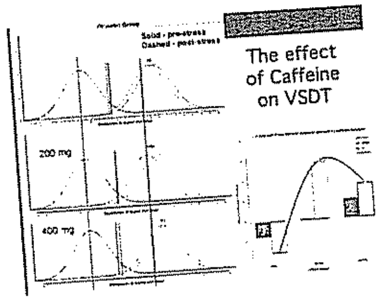
ACT-R (5) Model of Subtraction

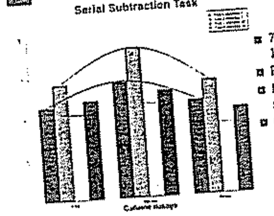
- Create goal to serial subtract
  - Utilizes a Borrowing Sub-goal
  - Repeating sub-goal across columns
  - Report result
- 34 rules
- 10 chunk types
- 318 original chunks of DM

Lessons from 1st Model

- Need complete data
- Need more overlay theories
- Other lessons not reported here (see *HFES* paper)
- Other models will need the same testing
- Recording User Input (RUI) software (Kukreja & Ritter accepted BRMIC)

The effect of Caffeine on VSDT





- 7's easier than 13's
- Practice effect
- Inverted U-shaped curve
- 0 < 200 > 400

Lessons from Café Nav I

- More control and care of subjects
- Tasks work, cognition, stress, caffeine effects
- Reuse, because we have to
  Reuse: BP, HR, cortisol, mood, time-task, time-data, time-model?, working memory task, model?, Argus task and model, ACT-R, /PM
  New: vigilance-task & model. serial-sub model, overlays
- Moderate caffeine may be more helpful
- Caffeine and stress effect on cortisol needs to be kept in mind

Overlays

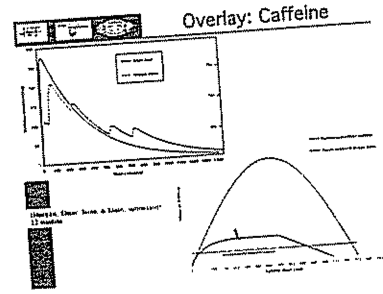
Summary of Stress Theories

	Type 1 - Central	Type 2 - Functional	Type 3 - Physio.
Wickens-CT	Central		
Wickens-PT		Vision	
Wickens-WM	Central		
Wickens-SS	Central		
Hancock-Szalma-FN		Vision	
Avraamides-IV	Central		
Belavkin-IAV	Central		
Processing speed	Central		
Learning rate	Central		
Associations	Central		
Worry, on-, off-task	Central		
Cannon, Selye, Mason			Physiol

Summary of Stress theories

- Stress theories are incomplete—do not touch enough mechanisms (or does tunneling arise through WM?)
  - Many affect the central processor
  - Few affect periphery processors and processes E.g., No motor
  - The trick will be making this dynamic
  - And then analysing dynamic data
- These theories are unlikely to be complete e.g., how will mental arithmetic be influenced by perceptual narrowing theory? Where is tremor?
- Might be combinable
- To test them, will need
  - Multiple tasks
  - Physiological data (HR, BP, cortisol.....)
  - Experimental psych data
- Pointer to overlay chapter

Overlay: Caffeine



Caffeine: Summary

.. (Morgan, Ritter, Stine, & Klein, submitted)

- Caffeine influences
  - RT -7% (not WM, not DM)
  - Self-reports on attention and alertness 20-100%
  - Vigilance stays good up to 3-6 hours (flat or +15%-30%)
  - 50-80 mg looks good for most effects
- Have a reusable overlay as a review
  - Reusable by CoJACK (DMSO, MoD project)
  - Suggests several caffeine studies
- Suggests adding appraisal and fatigue to ACT-R
  (cf. Gratch & Marsella, 2004; Gunzelmann et al. in press)

Summary

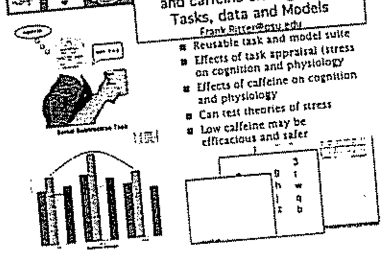
- CafeNav Suit: Set of tasks used by subjects and models, and models
- Headed towards detailed data set
  - Biopsychology + cognitive
  - Ready for model comparisons
- Overlays for pre-task appraisal & caffeine
- Suggestions for all cognitive architectures
  - Physio effects, Appraisal effects
    Vigilence effects, Strategies
- May be a problem fitting the data
- Caffeine, low doses may be as good cognitively and high doses bad physiologically

Future Work

- Studies
  - Self-report study on caffeine use (why how much)
  - Caffeine dosage-response curves
  - Study without caffeine users on caffeine
- Models and overlays
  - Finish packaging
  - More data comparisons (Cycling, dentist Quigley)
  - Move MODS to ACT-R 5
  - Consider different ways to compute best W (MSE vs. correlation computations of W)
  - Finish overlays for pre-task appraisal
- Fitting the data/develop the models
- Implication for caffeine users:
  - High caffeine and stress leads to cortisol
  - CafeNav data suggests 200 mg is more than enough
  - Review suggests lower (50 mg) is enough

The Effects of task appraisals and caffeine on cognition: Tasks, data and Models

Frank.Ritter@psu.edu



- Reusable task and model suite
- Effects of task appraisal (stress on cognition and physiology
- Effects of caffeine on cognition and physiology
- Can test theories of stress
- Low caffeine may be efficacious and safer

References

- At acs.ist.psu.edu/papers
- [illegible] U., & Ritter, F. E. (accepted pending revisions, March, 2005). [illegible] Recording user input from interfaces under Windows. Behavior Research Methods, Instruments, and Computers.
- Morgan, C. P., Ritter, F. E., Stine, M., Klein, L. C. (submitted). The effects of caffeine on cognition.
- Ritter, F. E. (2004). Choosing and getting started with a cognitive architecture to test and use human-machine interfaces. [illegible] Journal, 7, 17-37.
- Ritter, F. E., [illegible], Reifers, A. L., & Klein, L. C. (in prep.). [illegible] the effect of dental work as a stressor on cognition.
- Ritter, F. E., Quigley, K. S., & Klein, L. C. (submitted). Determining the number of model runs: Treating user models as theories by not sampling their behavior.
- Ritter, F. E., Reifers, A., Klein, L. C., Quigley, K., & Schoelles, M. (2004). Using cognitive modeling to study behavior moderators: Pre-task appraisal and anxiety. In Proceedings of the Human Factors and Ergonomics Society. 2121-2125. Santa Monica, CA: Human Factors and Ergonomics Society.
- Tomaka, J., Blascovich, J., Kelsey, R. M., & Leitten, C. L. (1993). Subjective, physiological, and behavioral effects of threat and challenge appraisal. Journal of Personality and Social Psychology, 65(2), 248-260.

An approach towards multitasking in ACT-R/PM

Jeronimo Dzaack, Juergen Kiefer, Leon Urbas
Research Group User Modeling in Dynamic Human-Machine-Systems
Technische Universität Berlin
Jebensstr 1, J2-2
10623 Berlin, Germany
{jdz, jki,lur}@zmms.tu-berlin.de
+49 30 314 7200

ABSTRACT

In human-machine-interaction, interruptability and resumption of different tasks are common aspects to influence human performance. To understand human behavior in such situations, we conducted a multitasking experiment where subjects had to perform a test of attention when driving. In this paper, we present an ACT-R/PM model on how people perform the test of attention (secondary task of multitasking scenario). This first model serves as basis for our long-term-objective: to model multitasking, i.e. to simulate how people interrupt a task and recover, considering and integrating individual differences in human performance models. The reported study is a first attempt and smoothes the way for ongoing studies

INTRODUCTION

In early stages of the design of technological systems, aspects of users more and more call for attention. Computer simulations, in our eyes, seem to be an appropriate method to consider these aspects The resulting simulations are based on empirical, psychological results. One main feature we are interested in is multitasking.

Multitasking is often seen as natural ability Interruptability and resumption of tasks are implicit components of multitasking, constituting a need which, so far, has not been fully considered in high-level cognitive architectures. Some, for instance, call for a single model to capture performance on multiple tasks This reflects a part of the aim of our research group: we strive for a cognitive architecture to simulate multitasking, i.e performance during interruption and resumption. Most models focus on single tasks like the Sternberg task or the Sperling task. In this case, interruption is forced by a self-defined break. We therefore investigated how people handle interruptability and resumption by using a multitasking scenario The overall aim is to integrate these aspects in cognitive architectures. This paper is intended to illustrate our approach: we present an ACT-R/PM-model of the secondary task performed in the presented multitasking experiment conducted in a driving simulator at the Technische Universität Berlin

EXPERIMENTAL SETTING

For the experimental setting, the situation while driving was used as multitasking scenario: driving was taken as main task. For the secondary task we referred to the D2 test of attention by Brickenkamp (2001). investigating individual differences on attention and concentration The aim is to identify a pattern as correct according to Brickenkamp's specification As in real-life-scenarios of human-machine-interaction, e g. when managing a navigation system or switching the radio. the used test acquires visual attention and. on an abstract level, can be seen as model for a driver infotainment system. Hence. it is referred to as D2-Drive test (Urbas et al . 2005)

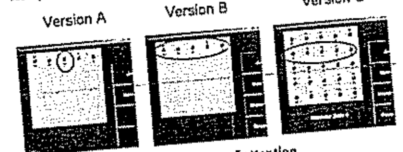


Figure 1: Test of attention

Three versions of the D2-Drive test were developed (see figure 1): the first relies on (static) visual search of the pattern in the middle (version *A*). The second maintains sequential, horizontal visual search of multiple patterns in a row (equivalent to reading from left to right, version *B*). The third is a combination of horizontal and vertical visual search (i.e., version *A* and *B*) with a recall task (called version *C*): within a field of information, one specific pattern (determined by row and column) must be found We assume our electronic implementation to be cultural independent requiring no previous knowledge or special expertise The configuration of our realization of the original paper and pencil test is appropriate to investigate interruptability and resumption. in a multitasking in-car-scenario Consequently. each subject was asked to perform the test of attention as pre-test (single task. baseline), real-test (while driving) and post-test (single task) after the experiment.

After a training phase in driving and a pre-test of the D2-Drive test. at three task-switching points a D2-Drive test had to be performed. The duration for each test was one minute. Subjects were instructed to attend driving as main task (high priority); they were asked to pay attention and to drive safely

While the task scenario (single task vs multitasking) was treated as within-subjects-design (i.e., each subject performed single and multitasking condition), we used a between-subjects-design to investigate the performance of subjects in the three different test versions (version *A* vs version *B* vs version *C*)

Twenty-four subjects joined the study Regarding to the D2-Drive test. they were told to work carefully but concurrently to complete as many patterns as possible.

HYPOTHESIS: D2-DRIVE (SINGLE TASK)

In this paper, we focus on performance in the test of attention. Please note that performance was measured by number of items per minute and not by number of correct items because the error rate approximates 0.

Assuming different levels of complexity, we expected the number of resulting patterns to be different in the three versions for the single task condition. Using performance rate r. we hypothesize

$$H1: r(A) > r(B) > r(C)$$

More concrete, we expected an improved performance of version *A* to *B* to *C* (a higher number of processed items)

RESULTS: D2-DRIVE (SINGLE TASK)

We observe a significant better performance in version *A* compared to *B* as well as compared to *C* ($p < .05$ for both), but there is no difference between *B* and *C* But a huge range in *C* can be observed: *C* evokes individual performance contrary to *A* where most of the subjects seem to perform approximately the same number (see figure 2)

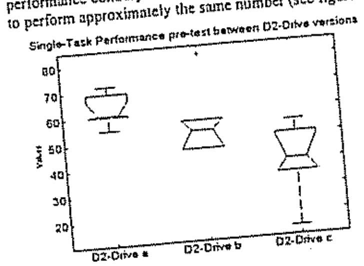


Figure 2: Task Difficulty

In version *A*, most people reach between 60 and 70 items per minute We therefore assume that one second for each pattern is a valid time prediction for subjects' performance in version *A* Observing the data more precisely we found different strategies of performing the given task of identifying a pattern as correct or not. Over all subjects we see a significant difference ($x^2 = 26.7$, $p < .001$) in the response time for *d*-patterns or *p*-patterns Observing the data individually per subject we find that the response times of more than 70% of the subjects show at least a small advantage for *p*-patterns in respect to response time, but only about 30% of the subjects show a significant difference ($p < .05$) in the time for judging a *p*- or *d*-pattern. 50 % of their response times are between 562 and 688 ms for a p-pattern (median = 610), 625-782 for a d-pattern (median = 687). We further identified other strategies: rhythmic key-presses with a quite impressive low deviation between response times, response-bursts for two or three elements due to separation of encoding and answering (variant B and C only) Finally 10% of the subjects show a clear advantage of the d-pattern in respect to response time

A FIRST ACT–R/PM – MODEL OF D2-DRIVE

The goal of our research project is to predict users performance in human machine interaction. To do this, we use the ACT-R architecture (Anderson, 2004). Based on the described experimental study an ACT-R/PM model of human performance in the test of attention (D2-Drive) was derived

In what follows we introduce the ACT-R/PM model of D2-Drive used in the experimental setting to collect empirical data. We start with the given assumptions which form the basis of our model Subsequently the structure of the model is explained. We prove the correctness of this approach in comparing the model with data of the experiments The results are discussed at the end

Assumptions

To implement a model that predicts user behavior means reducing human beings to specific elements (e.g., goal-oriented. emotionless, perfect) because of the interference and the complexity Not every aspect of human behavior can be integrated, and at the current state there is no need to do so To predict user behavior a wide understanding of psychological theories to build and empirical data to verify the model is needed. The underlying concepts have to be outlined and, consequently. integrated within the implementation of the model

The implementation of the ACT-R model of the secondary task – the interaction of humans with D2-Drive – implies assumptions that need to be integrated

Declarative Memory

The main task of the D2-Drive test is to identify a given pattern as a correct (D2-) pattern: If the letter in the center position is a *d*. only nine arrangements of the determined alphabet (i.e., nothing: 0. one stroke: 1. two strokes: 2) are possible to derive the conclusion. To enable the model to identify a pattern with a *d* as correct or not correct, we put nine chunks in the declarative memory concerning all possible statements (e.g.. 00 → No. 01→No, 11→Yes, 02→Yes) Retrieving the chunk connected with the given signs allows the model to derive a conclusion (yes or no)

Strategies

The results of the structured interviews at the end of the experiments and the supervision of subjects show that there are several strategies to solve the problem of identifying the pattern as correct or not as described before. For the implementation of this model, we used one strategy for all three cases of the D2-Drive test based on own experience and empirical evidence (as can be read in the results section of this paper)

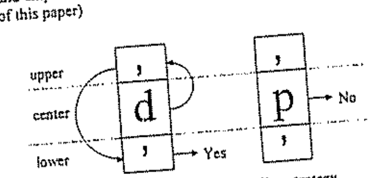


Figure 3: D2-pattern and encoding strategy

Each pattern is cut into three segments (upper: sign. center: letter, lower: sign) that are treated separately (see figure 3). A pattern is identified as correct D2-pattern if there is a *d* and two strokes

1. SET VISUAL ATTENTION to center segment
2. ENCODE letter
    a. p → D2-pattern. no → END
    b. d → SET VISUAL ATTENTION to upper segment
3. ENCODE sign
4. SET VISUAL ATTENTION to lower segment
5. ENCODE sign
6. RETRIEVAL IN DM (both signs)
    a. is a D2-pattern → D2-pattern: yes → END
    b. no D2-pattern → D2-pattern: no → END

Figure 4: Algorithm to identify a specific pattern

The algorithm to identify a pattern as correct or not correct works as described (see figure 4):

The visual attention is set to the center segment of the pattern and the letter is encoded If the letter is a *p*, the pattern cannot be a D2-pattern per definition. The procedure stops and the pattern is encoded as no D2-pattern Otherwise the letter is a *d* and the visual attention is set to the upper segment to encode the upper sign and afterwards to the lower segment to encode the lower sign In the next step a retrieval with both encoded signs is set to the declarative memory to identify the pattern. Both signs have to be encoded, because in all nine possible cases it is necessary to control both signs to identify a pattern as correct or not correct.

Visual Search

The first visual search in the model has to be treated separately because the empirical data shows a gap of orientation between the first and the subsequent tasks

The visual focus is always on the considered pattern and changes to the next one when the (keyboard-) key is pressed to enter the conclusion of the model.

In versions *B* and *C*, the model returns after 5 (9, 13) patterns to the beginning of a row If the "Eye" of ACT-R cannot see anything (i.e., visual-location throws an error) the "Eye" has to be redirected to the (next) starting point because the end of the row is reached and the visual-location points an empty space.

In version *C* the number of the next row has to be stored We assume this to happen before pressing the button the last time in a row

General

The trial of performing the D2-Drive test is determined by the version of D2-Drive and a given time Both parameters are set with the call of the start-function *s* of the model (i.e. s "a" 10).

Because of the re-use of specific structural elements all three versions are integrated in one single model.

Structure

The structure of the implementation of the ACT-R/PM model of D2-Drive is quite simple (see figure 5). It is a loop of separated tasks starting with a first visual search for orientation reasons. The loop consists of reading the pattern. interpretation of the pattern (both steps are described as identifying a pattern above), resetting the visual component (visual-location) and pressing the key

1. SET STARTing point
2. READING pattern
3. INTERPRET pattern
4. MOVE VISUAL-LOCATION
5. PRESS-KEY

Figure 5: Structure of the ACT-R model

To cope with different settings of the three versions, the part *resetting the visual component* has to be altered for each version of D2-Drive because every version requires different coordinates for the observed pattern (see figure 6)

In version *A*, only the middle pattern is observed. Thus, the visual component has to be reset to the center segment of the middle pattern after identifying one pattern. In version B, one row is observed successively and is then started again with a changed pattern. Hence the X-coordinate of the visual component has to be changed to step from one pattern to the next pattern If the visual-location is empty (i.e., throws an error) the end of a row is reached and the visual component has to be reset to the starting coordinates of the row In version C, additionally the number of the next row to be observed has to be stored. Thus the number has to be stored if the end of a row is reached. Changing the coordinates means to change the X- as well as the Y-coordinates of the visual component after reaching the end of a row

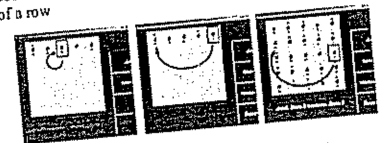


Figure 6: What to do at the end of a row?

Compare: Model vs Reality
The output of the implemented ACT-R/PM model shows that a *d*-pattern requires 950 milliseconds and a *p*-pattern requires 750 milliseconds In comparison with the data of the experiment, we conclude that the chosen strategy can be managed for this purpose. The amount of processed pattern in 60 seconds in all three versions (A: 75, B: 73. C: 56) is comparable with the results from the experiment regarding the upper third of the derived data. This can be explained by the assumptions of cognitive models to be perfect (i.e.. no interfering variables) The hypothetical assumptions on performance (A > B > C) are not approved by the model In version A and version B, nearly the same amount of patterns are processed. The model predicts the same performance for version A and version B, further it predicts for version C to be the less effective one (i.e.. A = B. A > C)

All together the model shows a slightly over-estimation of performance But the relative tendency of the tree different versions can be predicted by the model (A > B > C)

Using the Stand-Alone-Version of ACT-R 5.0, the performance times predicted by the models were close to subjects real performance. Integrating visual attention and motor action, ACT-R/PM turned out to be appropriate for our attempt to model user behavior in each of the three versions of the intended test-version

DISCUSSION

The model presented in this paper is an attempt in modeling the secondary task and a first step of our vision to simulate interruptability and resumption of tasks in the driving simulator scenario we used

In our model we implemented one strategy observed by most subjects. A next step in understanding individual behavior therefore must be the extension to a model including individual differences in strategic behavior

The final aim is modeling multitasking in cognitive architectures. Thus we have to combine the developed cognitive model with another model to observe the model-behavior in multitasking and compare the results with the data from the experiment.

Individual differences
Based on subjects statements in the feedback questionnaire as well as on observations measured by eye movements, this section attends the importance of individual differences of subjects. People differ in their performance, behavior and (working memory) capacity (see Jongman et. al , 1999) Work by Daily et al. (2001), for instance, suggests an individual component of working memory capacity Rehling et al (2004) refer to individual difference factors in a complex task environment. All this recommends various derivatives of the starting ACT-R/PM model we derived. Please keep in mind that the focus of this paper is only on performing the test of attention in a single task condition Ongoing research will investigate how to approach a multitasking ACT-R/PM model by questioning how. or if at all. to handle this complexity

Another observation on how subjects process is the dimension of steps each one uses: in version *B*, it seems to be of advantage not to compare and to press a key (y/n) after each pattern but to keep in mind the answer and then insert as many answers as can be kept in memory This strategy has not been considered so far

A general executive for multitasking
The next step in our research group is to combine our model with the car-driving scenario to analyze the effects of multitasking. To do so, we will use the *General Executive* described in Salvucci (2004). He proposes a general executive for multitasking suggesting to allow concurrent goals stored in a goal set. Because of the serial processing of ACT-R. only one single goal can be executed at the same time. Two heuristics define when to switch between goals. To determine which goal is chosen next the urgency of the concurrent goals is calculated and the most urgent is chosen

In this case, we want to use the scenario of Salvucci to represent the primary task of our experiment and the ACT-R model of D2-Drive as secondary task.

Conclusion
We are aware of the limits of our model, although it is a starting point in our research A next step concentrates on the question whether existing multitasking models in ACT-R are appropriate for our purpose (for instance, see Salvucci, 2001) Models of driving as well as of D2-Drive, taken together. will enlighten our way of modeling

interruptability and resumption in human machine interaction.

ACKNOWLEDGEMENT
This work was funded by grants of VolkswagenStiftung (research group user modeling). DFG (Research training group GRK 1013 prometei) and IBB PROFit (HMI Engineering for networked driving).

REFERENCES

Anderson, J., Bothel, D., Byrne, M.D., Douglass, S., Lebiere, C., and Qin, Y (2004) An Integrated Theory of the Mind. Retrieved from http://act-r.psy.cmu.edu/papers/403/IntegratedTheory.pdf

Brickenkamp, R. (2001) Test d2 Aufmerksamkeits-Belastungs-Test. 9., überarbeitetet und neu normierte Auflage Hogrefe Verlage Bern. Schweiz.

Daily, L. Z., Lovett, M. C., & Reder, L. M. (2001) Modeling individual differences in working memory performance: A source activation account in ACT-R. Cognitive Science 25. 315-353

Heinath, M., Dzaack, J., Kiefer, J. Urbas, L. (unpublished) Handbook of D2-Drive. Technical University of Berlin. Berlin 2005

Jongman, L. & Taatgen, N. A. (1999). An ACT-R model of individual differences in changes in adaptivity due to mental fatigue (pp 246-251). In Proceedings of the twenty-first annual conference of the cognitive science society Mahwah. NJ: Erlbaum

Rehling, J., Lovett, M., Lebiere, C., Reder, L. M., & Demiral, B. (2004) Modeling complex tasks: An individual difference approach. In proceedings of the 26th Annual Conference of the Cognitive Science Society (pp 1137-1142) August 4-7. Chicago. USA

Salvucci, D. D., Chavez, A. K., & Lee, F. J. (2004) Modeling effects of age in complex tasks: A case study in driving. In proceedings of the 26th Annual Conference of the Cognitive Science Society (pp 1197-1202) August 4-7. Chicago. USA

Salvucci, D. D., Boer, E. R., & Liu, A. (2001) Toward an integrated model of driver behavior in a cognitive architecture. Transportation Research Record. 1779.

Urbas, L., Schulze-Kissing, D., Leuchter, S., Dzaack, J., Kiefer, J., Heinath, M. (2005) *Programmbeschreibung D2-Drive-Aufmerksamkeitstest [Manual for D2-Drive Test of Attention]* Berlin: ZMMS

Simple Object System (SOS) for creating ACT-R environments: A usability test, a test of the perceptual system, and an ACT-R 6 version

Robert L. West (robert_west@carleton.ca)
Department of Psychology, Institute of Cognitive Science. Carleton University
Ottawa, Ontario, Canada

Bruno Emond (bruno_emond@uqah.uquebec.ca)
Institute for Information technology
National Research Council Canada. Ottawa. Canada

Josh Tacoma (joshua.tacoma@gmail.com)
Institute of Cognitive Science. Carleton University
Ottawa. Ontario, Canada

SOS (Simple Object System) for ACT-R 5 is system for creating environments that ACT-R can interact with. SOS is meant as a low fidelity, mock up system. That is, it is designed to be easy to learn and quick to use SOS uses an object-based approach to create worlds composed of objects, similar to ACT-R chunks It can be used to mock up any sort of environment and can also be used to create mental objects that function as the contents of modules (in the ACT-R 6 sense of module)

ACT-R 6

The ACT-R 6 architecture appears to be well suited for SOS Hopefully. we will have a fully functional version of SOS for ACT-R 6 available by the time of the workshop

Usability Test

To evaluate our claim that SOS is easy to learn. we ran a usability test using graduate students enrolled in a one-semester seminar on ACT-R at Carleton University None of the students had previously used ACT-R. Near the end of the course, four students were identified as being competent at building basic ACT-R models (i.e., simple models with no environments) All four had previous computer programming experience and one had previous Lisp experience. They attended one class on SOS (about 2 hours) and then attempted to create an SOS environment for the model they were developing. All four were able to create basic SOS environments for their models with no assistance. Two created complex models. One of these featured complex interactions between objects in the world; the other featured complex processing of objects by a module. Another student. the third author (and Lisp programmer), augmented the Lisp code to enhance the perceptual abilities afforded by SOS, according to ideas discussed in the course but not implemented.

Perceptual System

In previous versions of SOS the ACT-R agent had perfect perceptual command over its environment. All objects that matched its retrieval requests were found. If there was more than one that matched then one of them was chosen at random. SOS now allows the user to set the salience of objects to a percent chance that they will be found by a retrieval request If more than one object is selected then SOS chooses one at random. This makes SOS sensitive to both the number of distracting items and the similarity of the items. To test the system we modelled the data from Fleetwood and Byrne (2002) on icon search. The results show that the SOS system is quite powerful in terms of matching this type of data.

Figure 1 displays the original data from Fleetwood & Byrne (2002) showing the effect of set size and icon quality on search times Each set of icons has a target icon. One third of the distracter icons visually match it but have different text. The other two thirds differ visually The visual differences were large, medium or small This corresponded to good, medium and poor quality icons. Figure 2 shows data generated using the SOS perceptual system. In this case a single salience factor was assigned to cover the entire distracter set for each condition (i.e., we combined visual and text differences). The target salience was set to 1 and the good, medium and poor icon distracter conditions were set to overall salience levels of 25, .35. and .5 respectively Figure 3 shows some refinements to this model. In this case, as in the original experiment, one third of the icons in each condition were set to the same salience level as the target icon. To get the curve we also assumed that increasing the number of distracters increased the salience of the visually distinct icons. We modeled this by increasing the salience factor by a constant for every six icons that were added The factor was set to 5%, 10% and 15% for the high, medium, and low icons Otherwise. we used the same salience levels as above. In theory, the change in salience can be explained by subjects adopting different search strategies for different set sizes.

Although this system is quite flexible in terms of fitting the data it was still constrained in this case by using reasonable estimates of how long it takes to click a target with a mouse and how frequently the visual buffer can be checked (constrained by the production firing rate) However, without having previously established the salience values, this system is not capable of making detailed predictions Its main value is that it can use simple detection data (or reasonable estimates) to set up an ACT-R SOS perceptual system that can capture the stochastic properties of perception across time for a particular task. This is appropriate when the stochastic properties of a perceptual task are important for a model, but the details of how those properties arise is not. If it is important or desirable to understand the role of strategic eye movements and/or attention shifts then the ACT-R PM visual system (Byrne & Anderson, 2001) and the EMMA model of visual attention (Salvucci, 2001) should be used (see Fleetwood & Byrne, in press, for an example using this data).

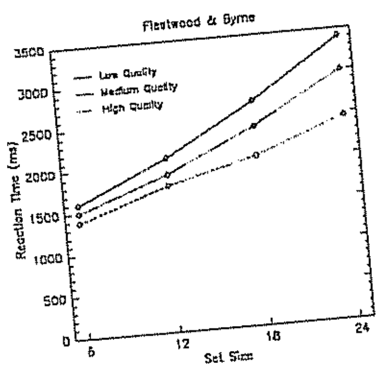


Figure 1 The original data from Fleetwood & Byrne (2002) showing the effect of set size and icon quality on search times.

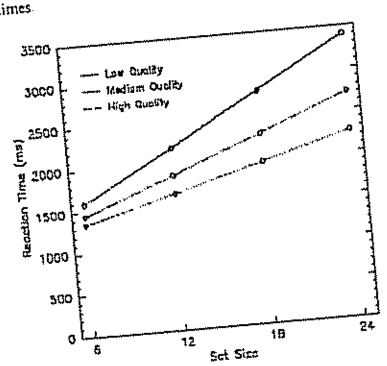


Figure 2. Data generated using the SOS perceptual system using simple assumptions.

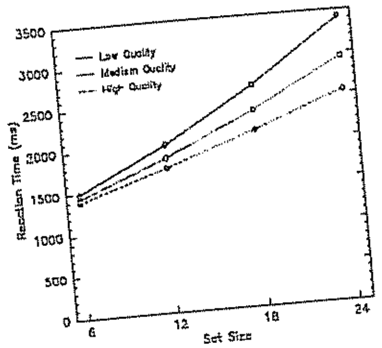


Figure 3 Data generated using the SOS perceptual system using more complex assumptions.

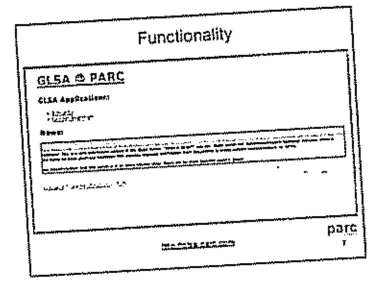
References

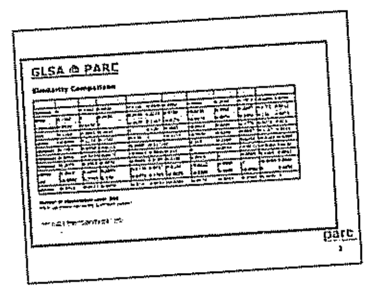
Byrne, M. D., & Anderson, J. R. (2001) Serial modules in parallel: The psychological refractory period and perfect time-sharing. Psychological Review, 108, 847–869.

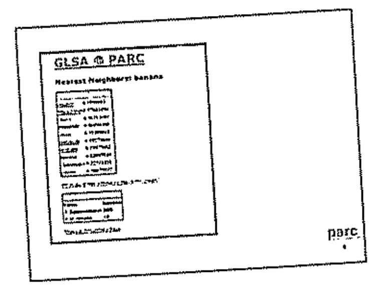
Fleetwood, M. D., & Byrne, M. D. (2002) Modelling icon search in ACT-R. *Cognitive Systems Research*, 3, 25–33

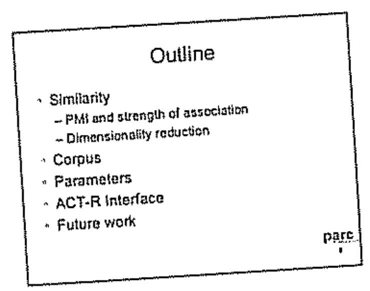
Fleetwood, M. D. & Byrne, M. D (in press). Modeling the Visual Search of Displays: A Revised ACT-R/PM Model of Icon Search Based on Eye Tracking Data. *Human Computer Interaction*.

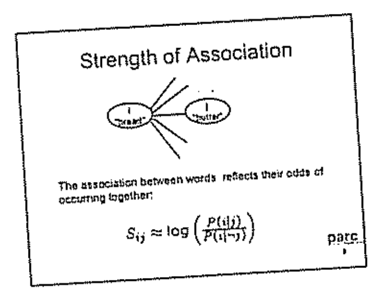
Salvucci, D. D (2001). An integrated model of eye movements and visual encoding. Cognitive Systems Research. 1(4). 201-220











Semantic Similarity Math

$$S_{ij} \approx \log\left(\frac{P(i|j)}{P(i|\neg j)}\right)$$

If $P(i) \approx 0$; $P(\neg i) \approx 1$, then

$$S_{ij} \approx \log\left(\frac{P(i|j)}{P(j)}\right) = \text{PMI}$$

(Pointwise Mutual Information)

Farahat, Pirolli, Markova, 2004; Pirolli, 2005

parc

Information Retrieval

Pointwise mutual information between two words:

$$PMI = I(i,j) = \log\frac{P(i,j)}{P(i)P(j)} = \log\frac{P(i|j)}{P(j)} = S_{ij}$$

In general, pointwise mutual information is the reduction in uncertainty of one random variable due to knowing about the other

(Manning and Schütze, 1999)

parc

Computing PMIs

Estimate probabilities using frequency counts $C(i)$, $C(j)$, $C(i,j)$ of words in a large corpus of N documents

parc

Dimensionality Reduction

Given a corpus V of v words:

1. Build a matrix of strengths of association/PMIs:

$$R_{v \times v} = \langle S_{ji} \rangle$$

2. Reduce the dimension of R to v x k (nr eigen vectors):

$$R' = (V_1 \ldots V_v)$$
$$V_i = (w_1 \ldots w_k)$$

3. Compute similarity using cosine measure:

$$sim(i,j) = \cos(V_i, V_j)$$

parc

Dimensionality Reduction

- Other techniques of dimensionality reduction can be used (e.g., Hellinger metric)
- Dimensionality reduction is a smoothing that takes into account similarity to other terms while measuring the similarity between a specific pair of terms

parc

The Corpus

The first 10 million pages of the Stanford Webbase project (generated by web crawling)

www.microsoft.com 10002
www.google.com 10002
www.w3.org 10001
www.whitehouse.gov 10002
www.apple.com 2
www.epa.gov 10002
www.yahoo.com 237
www.cdc.gov 10002
www.pbs.org 10002
www.un.org 10001
www.access.gpo.gov 2
...

http://dbpubs.stanford.edu:8091/~testbed/doc2/WebBase/

parc

Number of Eigen Values

Too high: overgeneralization

Too low: "overfitting" (i.e., taking too much noise into account)

For several synonymy problem spaces:
n = 150-400 gave best results

parc

Why Not Use Just PMIs?

- GLSA was compared with PMI on several synonymy tests (TOEFL, TSL1, TSL2) (using a different corpus made of New York Times articles)
- It achieved a performance > 70% (~80% for TOEFL and TS2; ~70% for TS1)
- PMI only was consistently worse (65-70%), although still comparable with humans

parc

ACT-R Format

```
(chunk-type meaning)

(defun sym-add-ia-fct (chunk1 chunk2 ia)
  (if ia
      (first (eval `(no-output (add-ia (,chunk1 ,chunk2 ,ia)))))
      nil))

(defmacro sym-add-ia (chunk1 chunk2 ia)
  `(sym-add-ia-fct ',chunk1 ',chunk2 ',ia))

(add-dm (banana isa meaning))
(add-dm (fruit isa meaning))
(add-dm (vegetable isa meaning))
...
(sym-add-ia banana banana 1.0000000)
(sym-add-ia banana fruit 0.08278595)
(sym-add-ia banana vegetable -0.013203774)
(sym-add-ia banana citrus 0.08277088)
...
```

parc

```
; Declare a meaning chunk type
(chunk-type meaning)

; Declare a function that maps GLSA values onto ias;
; it can be modified
(defun sym-add-ia-fct (chunk1 chunk2 ia)
  (if ia
      (first (eval `(no-output (add-ia (,chunk1 ,chunk2 ,ia)
                                       (,chunk2 ,chunk1 ,ia)))))
      nil))

; add words to memory as chunks of type meaning
(add-dm (banana isa meaning))
(add-dm (fruit isa meaning))

; set ias
(sym-add-ia banana banana 1.0000000)
(sym-add-ia banana fruit 0.08278595)
```

parc

Text Format

More convenient if you want to parse it into your own format

```
banana banana 1.0000004
banana fruit 0.08278595
banana vegetable -0.013203774
banana citrus 0.08277088
banana tropical 0.086832026
banana nut 0.035970457
banana grain 0.020718754
banana monkey 0.081152119
banana elephant -0.028544774
banana eat -0.004232715
banana person 0.03531784
fruit fruit 1.0000001
fruit vegetable 0.24325794
fruit citrus 0.18533081
```

parc

Future Work

- Corpus expansion and modification
  - Collect misses and add them periodically to the corpus
  - See what documents need to be removed from the corpus and what documents need to be added
  - Analyze word frequency and decide whether it's representative of the web/person's vocabulary
- TASA corpus as an option
- How does raw PMI compare with GLSA
  - Have a raw PMI server available
- Add word frequency counts
- Add the option to upload a file

parc

Suggestions?

parc

Python ACT-R: A New Implementation and a New Syntax

Terrence C. Stewart <tcstewar@connect carleton.ca>
Robert L. West <robert_west@carleton ca>
Carleton Cognitive Modelling Lab, Institute of Cognitive Science, Carleton University
1125 Colonel By Drive, Ottawa. Ontario. K1S 5B6. Canada

We present a re-implementation of ACT-R and a new syntax for the creation of ACT-R models. This allows for easier development of new sorts of modules and a more gradual learning curve. This short (800 line) implementation provides for all of the core functionality of ACT-R (including production compilation, optimized and non-optimized declarative memory learning, and basic environment interaction). This process has also allowed us to investigate the distinction between the theory of ACT-R and the details of the standard Lisp implementation.

ACT-R is the most extensively developed, widely used, and carefully examined architecture for modelling human cognition that we have Its successes are broad and startling, and it is a exemplar of the sort of theory that cognitive science strives toward. However, due to various factors, ACT-R has a rather harsh barrier to entry. Some of this difficulty is due to its complexity: learning any system that attempts to describe human cognition is naturally going to involve a certain degree of effort However, depending on the user's background, the architecture itself can have an impact on how understandable it is In particular, people with a strong background in Lisp typically have a much easier time, especially when running experiments with ACT-R, or collecting data from multiple runs of an ACT-R model Thus, one way to make ACT-R accessible to a wider audience is to implement it in other languages.

Our project is a complete functional reimplementation of ACT-R with this in mind. We have the following goals: (1) To confirm that ACT-R is doing what we think it is doing. A complete reimplementation of software is often used in industry to confirm functionality in this way. (2) To investigate the distinction between the theory of ACT-R and an implementation of ACT-R. It is possible that certain aspects of ACT-R are due more to the implementation choices than to the theoretical commitments (3) To make it easier for ACT-R researchers to investigate modifications and additions to it. This is one of the goals of ACT-R 6 but it still requires an extensive knowledge of Lisp Making ACT-R available in more languages will help this process

In creating this new version, we also have an opportunity to change the syntax of ACT-R. The current syntax is heavily embedded in its Lisp roots, and this can be a significant barrier to entry for new users. Therefore, we are taking this opportunity to develop an alternate syntax which fits well within the new implementation, and will be more familiar to people with a procedural/object-oriented programming background (e g. C++, Java)

Implementation

Our reimplementation is in the Python language It was chosen due to the first author's success using it in a graduate course to teach non-programmers to develop connectionist and evolutionary computational models of cognitive systems (Stewart, 2004) The language is often described as 'executable psuedo-code' due to its goal of having a syntax that is as clear as possible both for writing and reading Significant effort has gone into making it suitable for both beginner and expert programmers, and it supports an elegantly subtle transition from procedural to functional programming. It is also freely available, Open Source. widely ported, and has a comprehensive built-in library

Importantly, Python "supports all of Lisp's essential features except macros" (Norvig, 2000) This gives the full power of Lisp, but a constrained syntax. This syntax has been carefully designed for fast development, clarity, and ease of learning. For a full discussion and comparison between the two languages, see (Norvig, 2000). Our experience has been that academics with no programming background are able to read and understand Python code, and that this accessibility makes them more likely to develop computational models within their own research

Syntax

Unlike the jACT-R project (Harrison, 2005), which uses Java to process models written in the standard ACT-R syntax, we write ACT-R models as normal Python code. This makes it seamless to interact with other Python software for defining experiments, new modules, and other models It also leads us to a different, but identically expressive, syntax

As a sample, here is the Lisp version of the increment-sum production from the addition model in ACT-R Tutorial 1, followed by the Python ACT-R version:

```
(P increment-sum
  =goal>
    isa          add
    count        =count
    sum          =sum
  =retrieval>
    isa          count-order
    first        =sum
    second       =newsum
==>
  =goal>
    sum          =newsum
  +retrieval>
    isa          count-order
    first        =count
)

def incrementSum(goal='add ?count ?sum',
                 retrieval='count-order ?sum ?newsum'):
    goal(sum=newsum)
    retrieval('count-order ?count ?')
```

The first two lines of the Python version define the LHS, which matches on both the goal buffer and the retrieval buffer. The last two lines are the RHS, and show the modification of a slot in the goal buffer and a retrieval request. From the Python point of view, this is a function definition, and the arguments and default values for the function are used as the LHS, while the body of the function is used as the RHS. Our Python ACT-R system extracts this information from the production and uses the matching rules to determine when it will fire

Python ACT-R supports the same functionality as the '–' and '=' (and the new ACT-R 6 '?') matching on the LHS, and the '–', '=', and '+' commands on the RHS

Chunk Syntax

The most controversial change is that chunks in Python ACT-R do not have named slots The main reason for this is to reduce a confusion we have frequently encountered Many people learning ACT-R have difficulty remembering that slot names do not have semantic value. Furthermore, determining a good name for a slot can be difficult if the slot is used in different ways in different productions.

Since slot names are not part of the theory of ACT-R, but are rather there for the convenience of the modeller, we have chosen to investigate what happens if we identify slots by position, rather than by a slot name We have found two interesting positive benefits of this approach First, it eliminates the need to keep track of both slot names and variable names (which is confusing in the common situations where the slot name and the variable name are identical, as in something like 'first =first') Instead, we use bound variables as temporary slot names within a particular production.

Second, it makes the creation of chunks with many slots much less convenient. Although it has been argued that there should be only 7 ± 2 slots in a chunk, it is still common to see models with a larger numbers This is not easily noticeable when one examines just the productions, since each production may only use a few slots. However, in Python ACT-R, since slots are identified by position, you must explicitly say in the production that certain slots should be ignored This seems to form an interesting soft constraint on the number of slots it is practical to use.

Chunks in Python ACT-R also do not have a chunk-type, or a name There is thus no need for the (chunk-type ...) declarations Instead, chunks are an ordered list, usually represented as a line of text with spaces separating the elements (although they can be any arbitrary list)

Lisp ACT-R:

```
(chunk-type count-order first second)
(a ISA count-order first 0 second 1)
```

Python ACT-R:

```
'count-order 0 1'
```

Left-Hand-Side Syntax

Where Lisp ACT-R uses the '=buffername>' syntax to identify what buffer we are referring to on the LHS, Python ACT-R uses a series of default function arguments.

Lisp ACT-R	Python ACT-R
`=goal>`	`goal='`
`=retrieval>`	`retrieval='`
`=whatever>`	`whatever='`

To specify a matching pattern for a buffer, we use a similar syntax as when we defined the chunks. This is a text string that uses spaces to separate the slots. Since there are no slot names, order matters. Python ACT-R uses a '?' to indicate variables (much like Lisp ACT-R uses '=')

Lisp ACT-R

```
=goal>
  isa add
  arg1 =num1
  arg2 =num2
  sum nil
```

Python ACT-R

```
goal='add ?num1 ?num2 nil'
```

Since order matters, slots that are unimportant for the match cannot be left out. Instead, a '?' is used to indicate that this slot is unimportant.

Lisp ACT-R

```
=goal>
  isa add
  arg1 =num1
  sum nil
```

Python ACT-R

```
goal='add ?num1 ? nil'
```

If a variable is used in two (or more) slots, then that forces both slots that use the variable to have the same content (exactly as in Lisp ACT-R)

Lisp ACT-R:

```
=goal>
  ISA      add
  sum      =sum
  count    =count
=retrieval>
  ISA      count-order
  first    =sum
  second   =newsum
```

Python ACT-R:

```
goal='add ?sum ?count'
retrieval='count-order ?sum ?newsum'
```

To indicate that a slot should not match, Python ACT-R uses a '!' This can be combined with the '?' for variables, giving the following possibilities:

Lisp ACT-R	Python ACT-R
`- sum  seven`	`!seven`
`- sum  =sum`	`!?sum`

To do more than one match on the same slot, you can combine these together

Lisp ACT-R

```
  sum   two
  sum   =sum
- sum   three
- sum   eight
- sum   =other
```

Python ACT-R

```
two?sum!three!eight!?other
```

Right-Hand-Side Syntax

In Python ACT-R, the RHS of a production is implemented as the body of the function being defined. This means that any arbitrary Python code can be written for the RHS However, this would be the equivalent of abuse of Lisp ACT-R's !eval! command So, for normal models, the RHS should be restricted to commands which are all buffer and module requests

Inside the RHS, you have access to all of the bound variables from the LHS These are treated exactly like normal Python variables

Lisp ACT-R	Python ACT-R
`!output! (=num1)`	`print num1`

Modifying a particular slot in a buffer is done using temporary slot names Any bound variable can also be used to refer to the slot that the variable is bound to By setting a new value for this variable, we actually modify the slot itself.

Lisp ACT-R

```
(p set-sum
  =goal>
    isa add
    count =count
    sum =sum
==>
  =goal>
    sum =count
)
```

Python ACT-R

```
def setSum(
      goal='add ?count ?sum'):
   goal(sum=count)
```

To put a new chunk into a buffer, or to make a request of a module, the chunk is specified in the same manner as in the LHS Bound variables can be used, and a '?' in a request indicates slots that are not important to match on

Lisp ACT-R

```
+goal>
  isa    add
  sum    =num1
  count 0

+retrieval>
  isa    count-order
  first =count
```

Python ACT-R

```
goal('add ?num1 0')

retrieval('order ?count ?')
```

One open question is whether this syntax should also be used for modifying slots in buffers If so, it may be possible to consolidate buffer changes ('=') and module requests ('+') into a single type of RHS request.

Creating a Full Model

The following is the full source code for the ACT-R Tutorial Unit 1 counting model We start by importing our ACT-R library, which gives Python access to the ACT-R system we have written

```
import actr
```

The model definition is contained within a single Python class This class can then be used to create multiple instances of that model It explicitly specifies what modules exist within this model, and what they are called. Note that it is completely possible to have multiple declarative memory systems, if desired. Adding a new buffer/module is as simple as adding a new line in these declarations It is also possible to design new modules and add them here, as long as the module conforms to a basic set of rules (it must be able to indicate the contents of its buffer, and it must be able to respond to requests) Note that we have chosen to have exactly one buffer per module

```
class Count:
  goal=actr Buffer
  retrieve=actr BasicMemory
  production=actr BasicProduction
```

To simplify the initialization of declarative memory, a large set of chunks can be created like this:

```
memory='''count 0 1. count 1 2.
          count 2 3. count 3 4.
          count 4 5. count 5 6.
          count 6 7. count 7 8,
          count 8 9. count 9 10'''
```

Next, the productions are defined They make use of the named modules created previously For each production, we specify its name, the LHS matching rules, and the actions to take on the RHS Note that the RHS is simply the body of a normal Python function, which means that any valid Python code can be used (such as the print command)

```
def start(
        goal='count-from ?start ?end starting'):
   retrieve('count ?start ?next')
   goal('count-from ?start ?end counting')

def increment(goal='count-from ?x !?x counting',
                 retrieve='count ?x ?next'):
   print x
   retrieve('count ?next ?nextNext')
   goal(x=next)

def stop(goal='count-from ?x ?x counting'):
   print x
   goal('count-from ?x ?x stop')
```

Now that the model has been defined, we can create it by giving it to the ACT-R system Multiple models can be created, as can multiple instances of the same model All communication between these models is done outside of the Python ACT-R system

```
model=actr ACTR(Count)
```

Once the model has been created, it is possible to use any RHS command to explicitly set buffer values This is most useful to set the goal. but can also be used to add chunks into declarative memory

```
model goal('count-from 2 5 starting')
```

Finally, we can run the model A model can be run for a set period of virtual time, a certain number of steps. or until there are no pending productions

```
model run()
```

The system can also create log files as expected

```
0 000 focus='count-from 2 5 starting'
0 000 'start' Selected
0 050 'start' Firing
0 050 focus='count-from 2 5 counting'
0 100 retrieve='count 2 3'
0 100 'increment' Selected
0 150 'increment' Firing
2
0 150 focus='count-from 3 5 counting'
0 200 retrieve='count 3 4'
0 200 'increment' Selected
0 250 'increment' Firing
3
0 250 focus='count-from 4 5 counting'
0 300 retrieve='count 4 5'
0 300 'increment' Selected
0 350 'increment' Firing
4
0 350 focus='count-from 5 5 counting'
0 350 'stop' Selected
0 400 'stop' Firing
5
0 400 retrieve='count 5 6'
0 400 focus='count-from 5 5 stop'
0 400 No events left to process
```

All of the system parameters are available for modification at any time, or can be set when instantiating the model

```
model=actr.ACTR(Count,retrievalThreshold=-0 5)
model params retrievalThreshold=-0 5
```

Current Status

Since the Python ACT-R modules are explicitly added to a particular model, it is easy to develop new models and make modified versions of existing models. We have also chosen to break the standard ACT-R modules up. This means that the optimized declarative memory module is actually a different module than the non-optimized version (although the one is, of course, based on the other). We currently have working versions of the following modules:

Declarative Memory
- *BasicMemory*: no chunk activations, matching requests return the first chunk found. Same as setting (:ESC F) in Lisp ACT-R
- *FastMemory*: optimized activation learning. Same as setting (:ESC T :OL T) in Lisp ACT-R.
- *FullMemory*: non-optimized activation learning. Same as setting (:ESC T :OL F) in Lisp ACT-R

Productions
- *BasicProduction*: no learning of production weights Same as setting (:PL F) in Lisp ACT-R
- *PGCProduction*: learning of production activations using the PG-C rule. Same as setting (:PL T) in Lisp ACT-R
- *CompilingProduction*: PG-C learning along with basic production compilation. Same as setting (:PL T :EPL T) in Lisp ACT-R

Other Modules
- *Buffer*: A simple buffer with a module that does nothing. Same as the Goal buffer in Lisp ACT-R.
- *SOS*: An implementation of the Simple Object System used for creating environments for ACT-R (West. Emund. and Tacoma. 2005)

The system does not currently support partial matching, spreading activation, or direct linking of chunks, but these are all planned additions We are also evaluating the possibility of an ACT-R/PM module

Open Design Questions

As this project has developed, a number of questions have arisen as to the approach we should take These are situations where the theory of ACT-R may be being influenced by the standard Lisp implementation For example, it was easier in the Python implementation to have any chunk that was cleared from any buffer to automatically be merged into declarative memory, rather than that happening for just the goal buffer We changed this to conform to Lisp ACT-R, and then discovered that in ACT-R 6 there is strong consideration given to having this happen to all buffers. We therefore think it is worthwhile for us to describe similar design decisions that arose during our work on this project, so as to get feedback from the ACT-R community as to whether these are legitimate possibilities

Production Compilation

Our current production compilation system creates a new production with a RHS that is simply the two previous RHSs put together This means that. in the standard case of a retrieval request and a match on that request, the retrieval request is still made, instead of being bypassed, as in Lisp ACT-R. For the most part, this works out to be functionally the same as the more elaborate compilation system, and has the side effect of automatically dealing with the situation where a retrieval request is used by two different productions Note that this is very different from the approach in ACT-R 6 of having a retrieval buffer automatically cleared (unless it is told not to) when it is made use of

Goal Buffer

The goal buffer in Python ACT-R is not a special buffer As already described. all buffers automatically add their contents to declarative memory Furthermore, the goal buffer is not required on the LHS of Python ACT-R productions Interestingly, this allows for *reactive* productions, which respond to information in a non-goal-driven manner

Furthermore, since the names of buffers are not fixed in Python ACT-R, we have sometimes found it clearer to have a 'focus' buffer instead of a 'goal' buffer This is accomplished simply by giving it a different name when creating the model

It is also possible to have multiple buffers of this sort. although we have not yet found practical uses for this

Chunks

Python ACT-R chunks are much simpler than Lisp ACT-R chunks Lisp ACT-R chunks are full objects in their own right, allowing for inheritance and arbitrary numbers of slot values, whereas our chunks are simple ordered lists This is a very different way of looking at chunks. but seems consistent with the core ACT-R theory It does, however, mean that the elements within the slots of our chunks are not currently guaranteed to also be chunks themselves (they could be text strings, for example). although they can behave as chunks if needed

Availability

The source code is freely available under the GNU General Public License, and can be downloaded from the project website at <http://ccmlab.ca/actr> It is currently a total of 800 lines (22kB)

References

Harrison, A M (2005) jACT-R. Available at <http://simon.lrdc pitt.edu/~harrison/jactr html>

Norvig, P (2000) Python for Lisp Programmers. Available at <http://www.norvig.com/python-lisp html>

Stewart, T.C (2004) Teaching Computational Modelling to Non-Computer Scientists Sixth International Conference on Cognitive Modelling.

West, R.L., Emund, B., and Tacoma, J (2005) Simple Object System (SOS) for creating ACT-R environments: A usability test, a test of the perceptual system, and an ACT-R 6 version 12th Annual ACT-R Workshop.

New approaches for detecting workload and stress

Markus Guhe, Wayne D. Gray, & Michael J. Schoelles

Cognitive Science Department
Rensselaer Polytechnic Institute

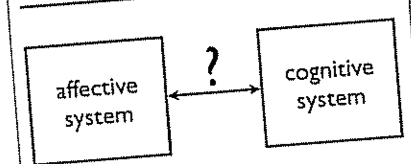
CogWorks

Overview

- Measuring workload and stress
- Tasks
  - Auditory 2-back task
  - One-digit addition task
- ACT-R models
- New approaches
  - DBN (*Dynamic Bayesian Nets*)
  - fNIR (*functional Near Infra Red imaging*)

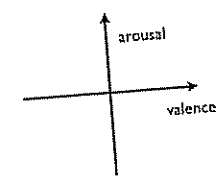
Rensselaer | Cognitive Science

Affective and cognitive system



Rensselaer | Cognitive Science

Two-dimensional affective model



Rensselaer | Cognitive Science

Workload and stress

- Workload – what is it?
  - task load?
  - № of mental/cognitive operations?
- Assumptions
  1. *Task load correlates with mental workload.*
  2. *High mental workload causes stress.*
- Stress affects performance
  - outcome (accuracy)
  - reaction time
  - (stress can also facilitate performance)

Rensselaer | Cognitive Science

Measuring workload and stress

- Measures
  - subjective (eg, questionnaires like NASA-TLX)
  - performance (eg, reaction time)
  - physiological (eg, galvanic skin response)
- No direct and reliable measure
- Our approach: combining multiple measures

Rensselaer | Cognitive Science

The tasks

- We use comparatively simple tasks to set a user's task load
  - single column addition task
  - auditory 2-back
- Task load → mental workload → stress

Rensselaer Cognitive Science

Math task

    3
+   8
 1 2

Incorrect   Correct

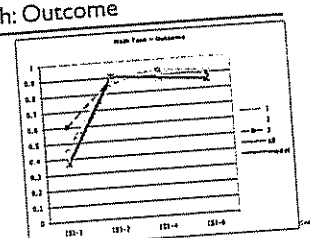
Rensselaer Cognitive Science

ACT-R model

- Look at addend 1 [3]
- Look at addend 2 [8]
- Look at sum [12]
- Retrieve addition fact from memory [3+8=11]
- Decide on answer [incorrect: 11≠12]
- Search for button on screen [incorrect]
- Move mouse to button
- Click mouse button

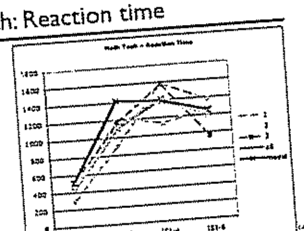
Rensselaer Cognitive Science

Math: Outcome

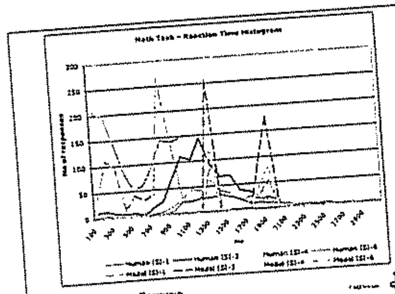


Rensselaer Cognitive Science

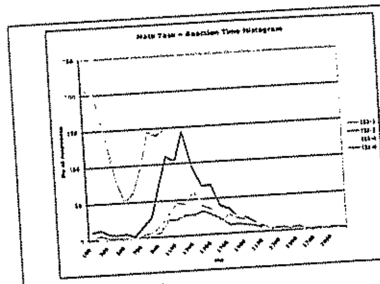
Math: Reaction time



Rensselaer Cognitive Science

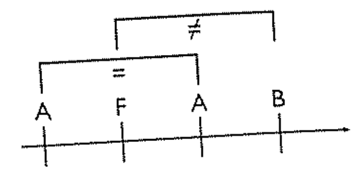


Rensselaer Cognitive Science



Rensselaer Cognitive Science

Auditory 2-back task



Rensselaer Cognitive Science

ACT-R model

- Perceive letter 1 store it [A]
- Perceive letter 2 store it [F]
- Perceive letter 3 store it [A]
- Retrieve letter 1 from memory [A]
- Compare letter 1 and letter 3 [A = A]
- Search for button on screen [equal]
- Move mouse to button
- Click mouse button

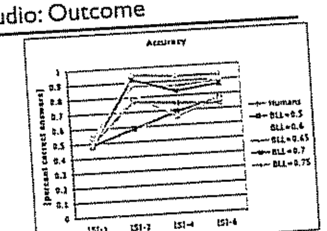
Rensselaer Cognitive Science

Parameters

- We varied
  - BLC
  - BLL
- Best results with
  - OL = nil
  - BLC = 1.0 (because OL = nil)
  - BLL = 0.7
  - digit-detect-delay = 100 (default 300)

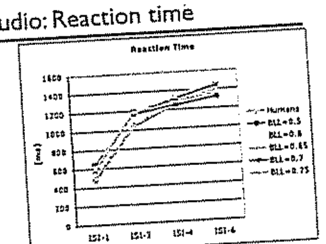
Rensselaer Cognitive Science

Audio: Outcome



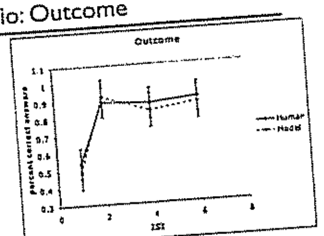
Rensselaer Cognitive Science

Audio: Reaction time

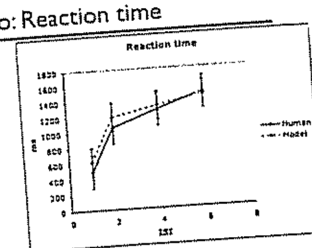


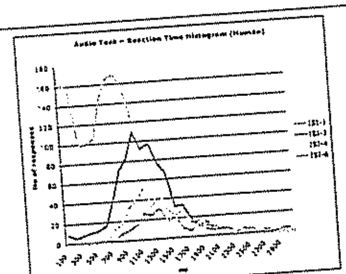
Rensselaer Cognitive Science

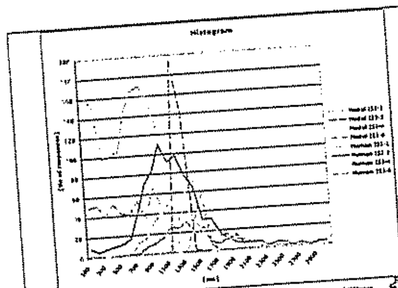
Audio: Outcome



Audio: Reaction time



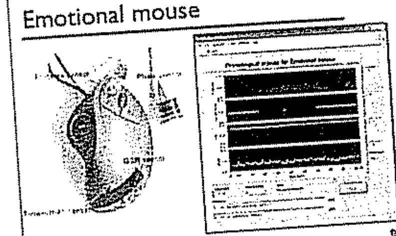




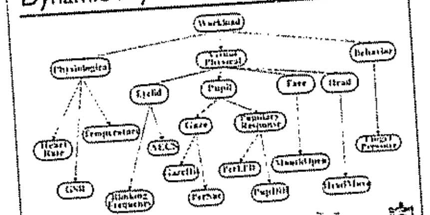
Monitoring facial expression

Monitoring facial expression

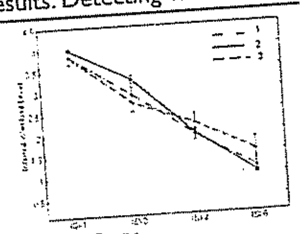
Emotional mouse



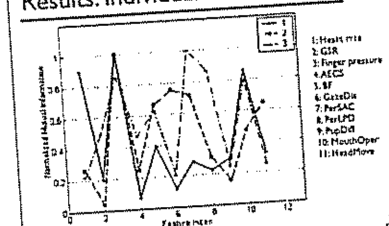
Dynamic Bayesian Net



Results: Detecting workload



Results: Individual differences

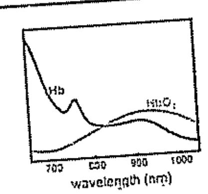


Hemoglobin and NIR light

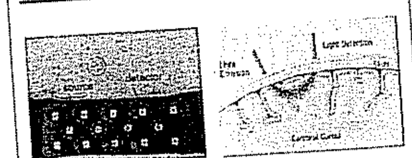
Absorption rate of infrared light by hemoglobin

Hb: deoxygenated hemoglobin

HbO_2: oxygenated hemoglobin

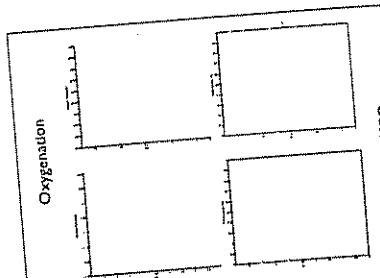


fNIR for brain imaging



Wearing the fNIR headband

Rensselaer | Cognitive Science



Rensselaer | Cognitive Science

Conclusions (1/2)

- Detecting workload and stress is difficult
- Combining multiple measures
- ACT-R models explain task performance
- New methods for measuring workload and stress
  - Dynamic Bayesian Nets
  - functional Near Infra Red Imaging (fNIR)

Rensselaer | Cognitive Science

Conclusions (2/2)

- In ISI-1 users are stressed (!)
- Despite equal outcome ISI-2 to ISI-6 differ (RT DBN)
- Expression of workload/affect/stress differs with individuals
- Individuals may put different effort into performing the tasks

Rensselaer | Cognitive Science

Thanks to ...

CogWorks:
Wayne Gray
Mike Schoelles
DBN:
Qiang Ji
Zhiwei Zhi
Wenhui Liang
fNIR:
Birsen Yazici
Il-Young Son

Rensselaer | Cognitive Science

Questions

?

Rensselaer | Cognitive Science

CogWorks

Interactive Behavior at the Sticking Point:

The Curious Persistence of Apparendy Inefficient Interactive Routines

Chris Sims & Wayne D. Gray

ACT-R Workshop
Treiste, Italy
15-17 July 2005

Rensselaer | Cognitive Science

Overview of Research

- Focused on the acquisition and transfer of interactive routines
- Specifically, how the design of an interactive device influences the interactive routines that people acquire when learning to use a new device
- & how these interactive routines transfer when the original design of the interactive system is modified

Rensselaer | Cognitive Science

Approach

- Combination of
  - Experimental Research
  - High-Density Data Collection
  - Computational Cognitive Modeling

Rensselaer | Cognitive Science

Theoretical Issues

- Understanding the control of cognition
  - Predict the cost-benefit structure that emerges from embodied cognition. interacting with the designed environment. to accomplish a given task

Rensselaer | Cognitive Science

How will we know when we are there?

- When our models, interact with the same software as our human subjects, and show the same patterns of acquisition and transfer of interactive behavior
- & when our confidence in these models is so great that we do not bother collecting empirical data

Rensselaer | Cognitive Science

CogWorks

Calendar Task

Chris Sims

Rensselaer | Cognitive Science

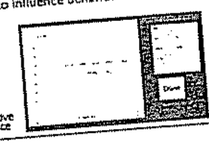
Design

- Participants must navigate from the current date to a target date for an appointment that must be programmed into the calendar
- On each trial, two strategies are available:
  - Navigate a single day at a time (Day View)
  - Navigate an entire week at a time (Week View)
- The utility of each strategy varies as a function of the distance that must be navigated

Rensselaer | Cognitive Science

Small Changes in Design Produce Large Shifts in How Device is Programmed!

- Like the DVD, the Calendar is a simple device
- Varied the design to produce tiny changes in the costs of different steps in interactive behavior
- Increment/decrement in costs would be considered insignificant by most designers – i.e., these changes would not be expected to influence behavior



Rensselaer | Cognitive Science

But . . .

- Unlike the DVD
  - Interface of the Calendar Task required the use of different strategies for different programs (i.e. for setting reminders for different future dates)
  - Hence, all subjects in all conditions use both major strategies – navigating by day (day view or DV) AND navigating by week (week view or WV)

Rensselaer | Cognitive Science

The Tipping Point

- For close dates we expect all Ss to use DV
- For far dates we expect all Ss to use WV
- Where is the tipping point? How does it vary by condition!
- How is it affected by transfer!

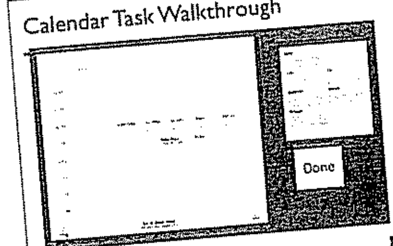
Rensselaer | Cognitive Science

Modeling Training Only

- To date, modeling has focused on capturing the training data, not the transfer data
- Hence, we are only presenting the training data and the model for the training data

Rensselaer | Cognitive Science

Calendar Task Walkthrough



Rensselaer | Cognitive Science

NOTE

- In this report we focus on navigation time and navigation strategy
- From first access of the target date information (info window) to arrival at the target day (calendar window)

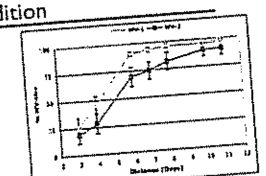
Rensselaer | Cognitive Science

Experimental Design

- Within-Ss: 7 distances
  - 3, 4, 6, 7, 8, 10, & 11 days from current day
- Between-Ss: varied the cost of using the Week View by adding lockout times to the WV button
  - Either 1 or 3 seconds (WV-1, WV-3) depending on group
- 7 trials per block, 10 training blocks, & 5 transfer blocks
  - WV-1 to WV-1
  - WV-3 to WV-1
- WE ONLY DISCUSS THE TRAINING BLOCKS

Rensselaer | Cognitive Science

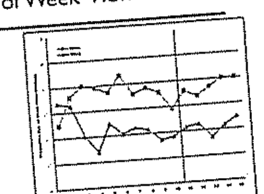
Use of Week-View by Lockout Condition



- Use of WV increases as the distance of the appointment increases
- Varies significantly by condition (1s lockout [WV-1] vs 3s lockout [WV-3])
- Transition appears gradual and not step-like, even within individual subjects data

Rensselaer | Cognitive Science

Use of Week-View over Blocks



- Strategy use does not change after transfer!!!!

Rensselaer | Cognitive Science

Modeling Challenges

- A surprisingly difficult task to model in ACT-R
- Issue
  - Relative utility of using week-view vs day-view varies with distance
  - But ACT-R can only assign one utility to a given strategy

Rensselaer | Cognitive Science

Modeling Approach (training data)

- Since we were interested in strategy selection, we constructed two hybrid human/ACT-R models
- Use human empirical navigation times to determine perceptual-motor costs of each strategy
- Use ACT-R's default PG-C to optimize strategy selection over these costs
- Hence, these are simulated ACT-R models

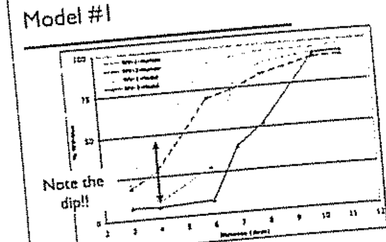
Rensselaer | Cognitive Science

Model #1

- Separate productions for each strategy at each distance
  - 2 strategies (day-view or week-view) x 7 distances = 14 strategy productions
- Learns to optimize strategy selection for each possible distance
- Model was run 200 times per condition

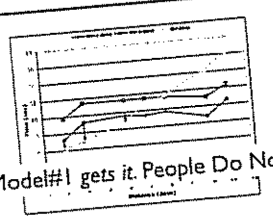
Rensselaer | Cognitive Science

Model #1



Rensselaer | Cognitive Science

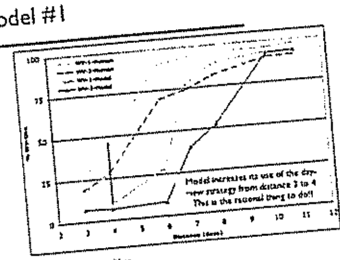
Human Strategy Navigation Time



Model#1 *gets it*. People Do Not

Rensselaer | Cognitive Science

Model #1



Rensselaer | Cognitive Science

Model #1: A Model of Near-Optimal Performance

- But people are not optimal on this task
- Poor qualitative fit not an aberration, but a rational adaptation to the properties of the environment
- Model has two separate strategies for each distance.
  - Overly sensitive to small costs that people ignore

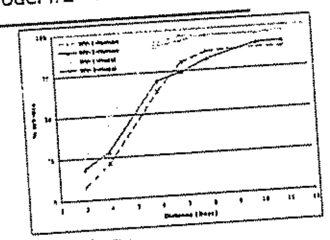
Rensselaer | Cognitive Science

Model #2: Near & Far

- Instead of representing each distance as a separate productions, model #2 simply classifies distances into near or far
- Results in 4 productions (2 strategies x 2 distances)
- Adds one parameter: a theshold distance for classifying distances as near or far
- Each model subject ran with a different threshold, drawn from a normal distribution (mean and stdev fit to data)
- Model was run 200 times

Rensselaer | Cognitive Science

Model #2 - Near & Far



Rensselaer | Cognitive Science

Modeling Conclusions

- Optimizing the *Physical Task Environment* requires two separate strategies for each distance
- In the *Functional Task Environment* distance is represented as *near* versus *far*
- *Functional Task Environment* shows a tradeoff between performance and complexity of representation
  - *Just because its out there, doesn't mean that it is in here!*

Rensselaer | Cognitive Science

Teaser

- Transfer results are extremely interesting and were unexpected to us as experimenters and as modelers (who was the idiot who claimed that models could never produce unexpected behavior!)
- Have to wait until Chris Sims completes his Masters Thesis for the full story on this

Rensselaer | Cognitive Science

General Conclusions

- Stable, suboptimal performance seems reasonable for this task environment
- Need to quantify costs of acquiring more precise strategies

Rensselaer | Cognitive Science

Problems & Challenges for Modeling

- State representation is an open-problem in ACT-R: typically hand-coded or tweaked until model fits the data
- How does representation develop through exposure to a task!
- Interested in quantifying the tradeoffs in different representations

Rensselaer | Cognitive Science

Conclusions Across Studies

- DVD
  - What seemed to be stable, suboptimal performance turned out to be a change in the *functional task environment* due to experience
- Calendar
  - *Functional task environment* represented distance as near vs far and optimized the results of that representation
  - Points to need to understanding cost-benefit tradeoffs in acquiring fine distinctions among state representations

Rensselaer | Cognitive Science

General Conclusions

- Functional task environment is not the same as the physical task environment
- We are making progress on understanding how the functional task environment emerges from cost-benefit tradeoffs of embodied cognition interacting with the physical task environment to accomplish a set of goal-driven tasks
- Emerging issue: understanding the cost-benefit tradeoffs in acquiring fine distinctions among state representations

Rensselaer | Cognitive Science

Thank You

Rensselaer | Cognitive Science

ACT-R in the Brain

Yulin Qin & John R. Anderson

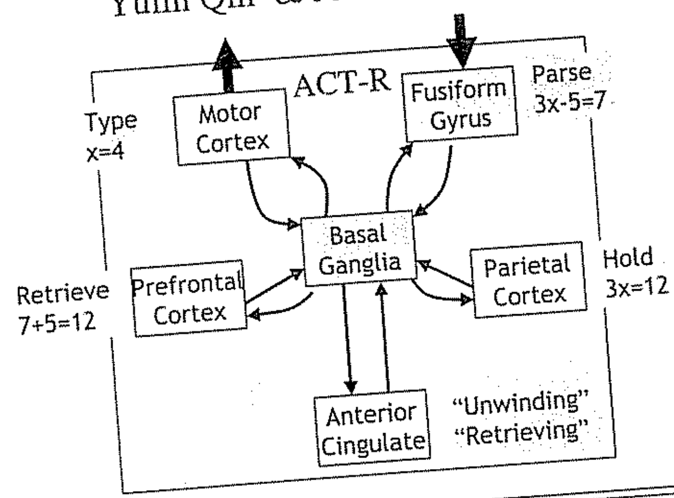


Illustration of the Four Conditions of the Experiment

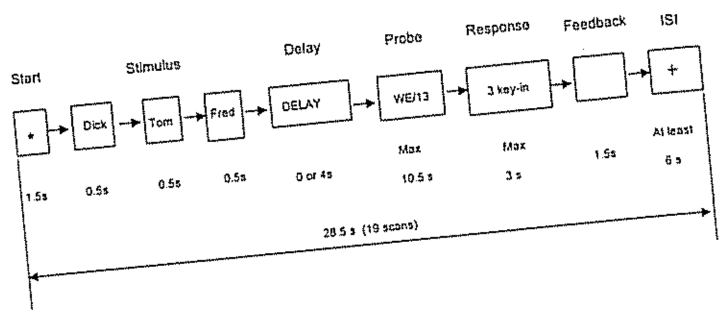
Associations:

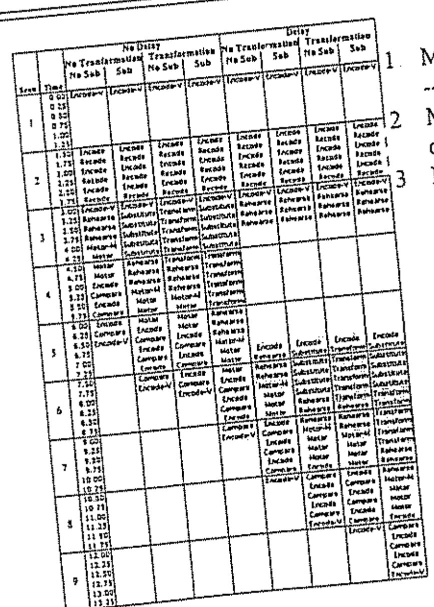
AT is associated to 23; BE to 24

Dick to Index; Fred to Middle; Tom to Ring

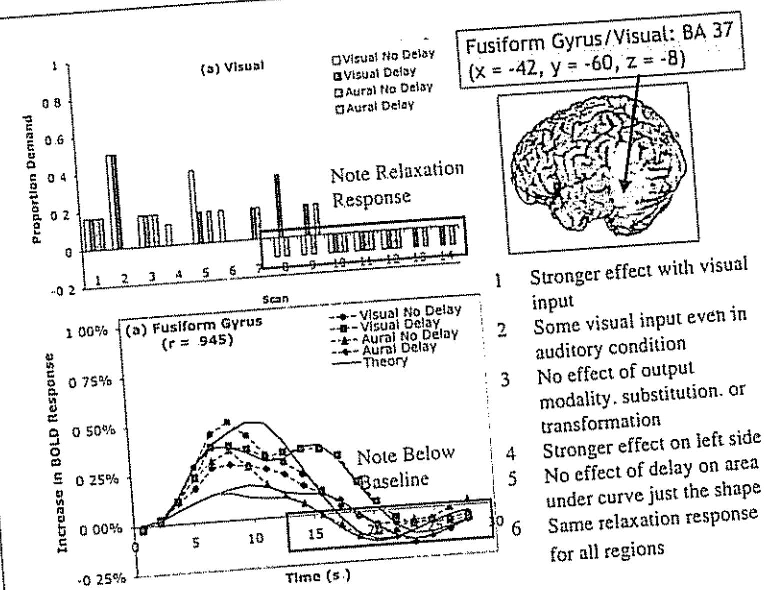
	No Transformation	Yes Transformation
No Substitution	**Stimulus:** Tom Dick Fred 24 **Response:** Ring-Index-Middle	**Stimulus:** Tom Dick Fred 23 **Response:** Ring-Middle-Index
Yes Substitution	**Stimulus:** Tom Dick Fred BE **Response:** Ring-Index-Middle	**Stimulus:** Tom Dick Fred AT **Response:** Ring-Middle-Index

19 Scans of 1 5 seconds Each

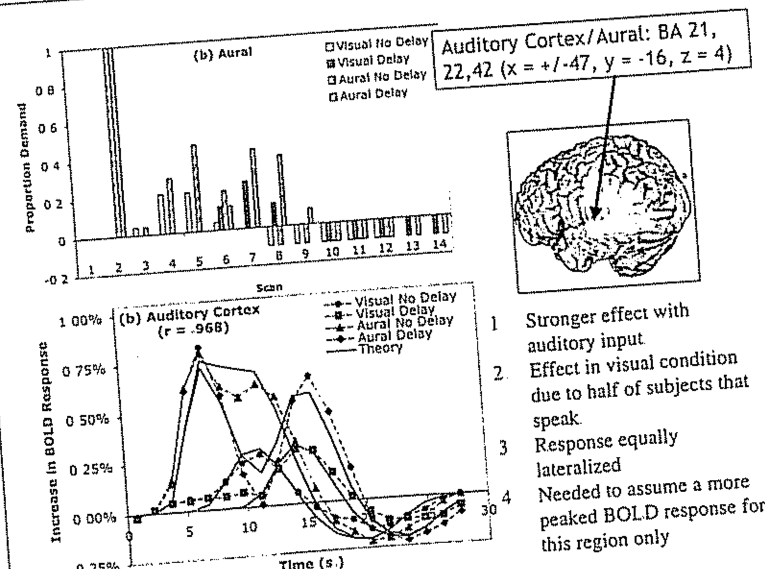




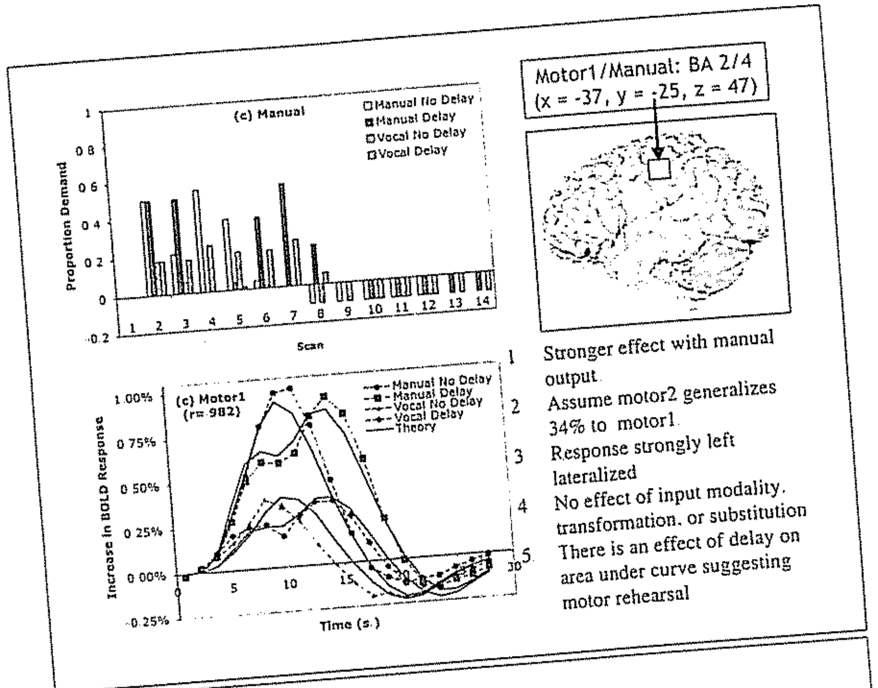
1. Manipulated modality of input --visual versus aural
2. Manipulated modality of output -- manual versus vocal
3. Looked at
   (a) Fusiform gyrus (left)
   (b) Auditory cortex (both)
   (c) Motor1 -- manual (left)
   (d) Motor2 -- vocal (both)
   (e) Posterior parietal (left)
   (f) Prefrontal (left)
   (g) Anterior Cingulate (left)
   (h) Caudate (right)

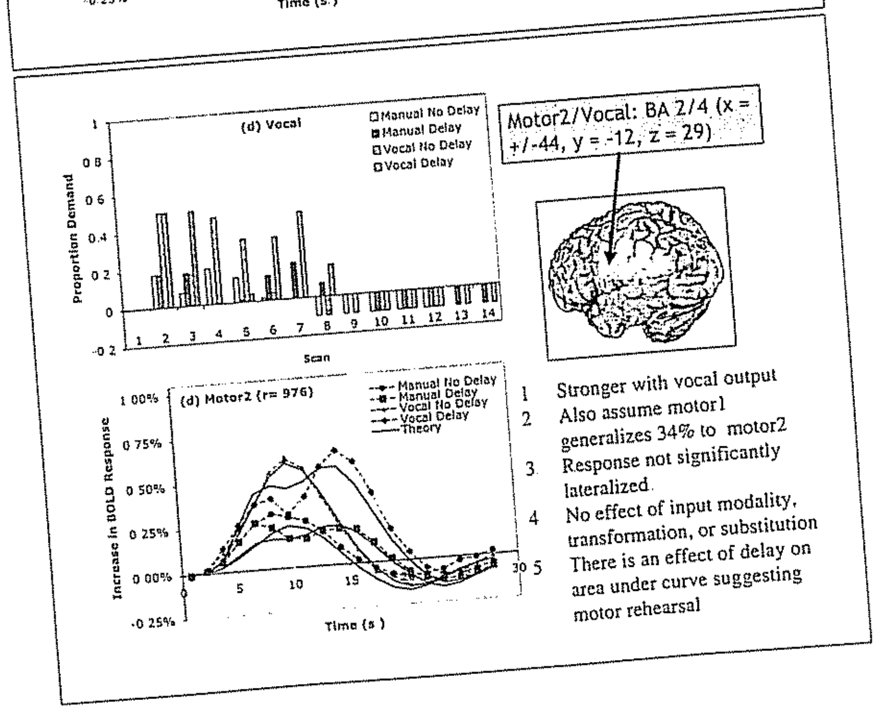


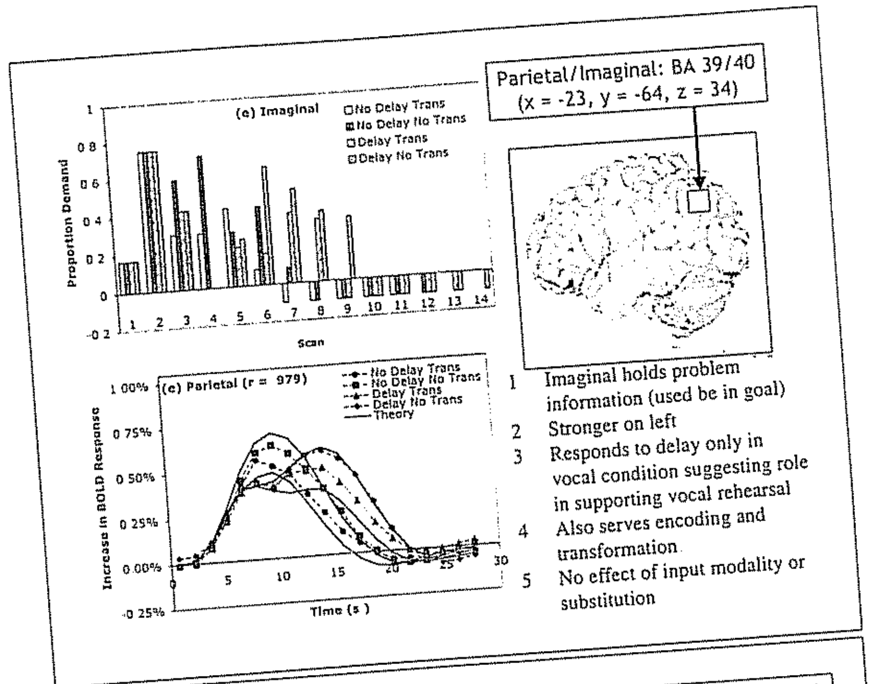
1. Stronger effect with visual input
2. Some visual input even in auditory condition
3. No effect of output modality, substitution, or transformation
4. Stronger effect on left side
5. No effect of delay on area under curve just the shape
6. Same relaxation response for all regions

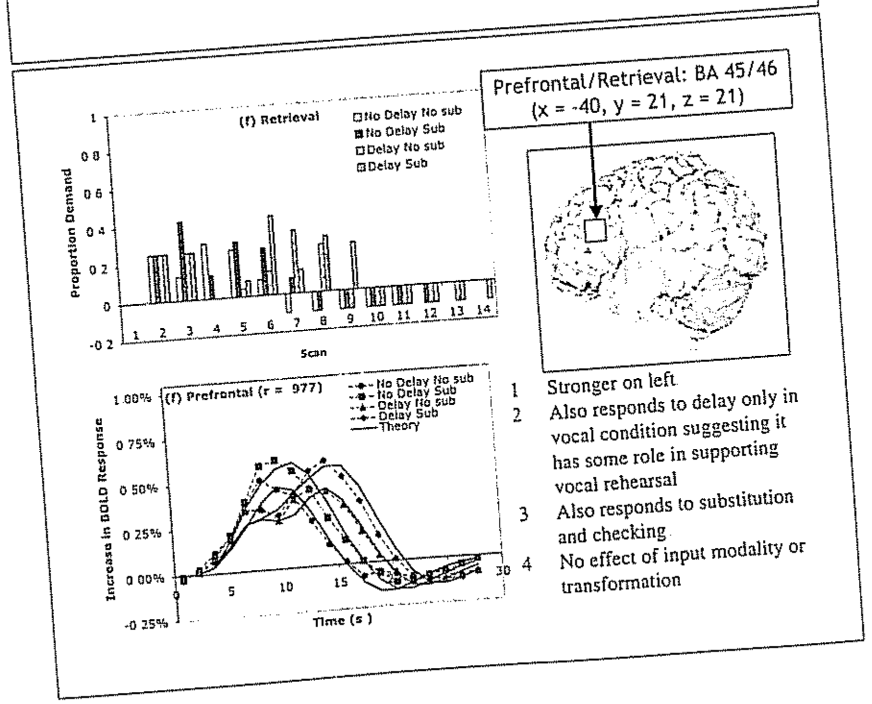


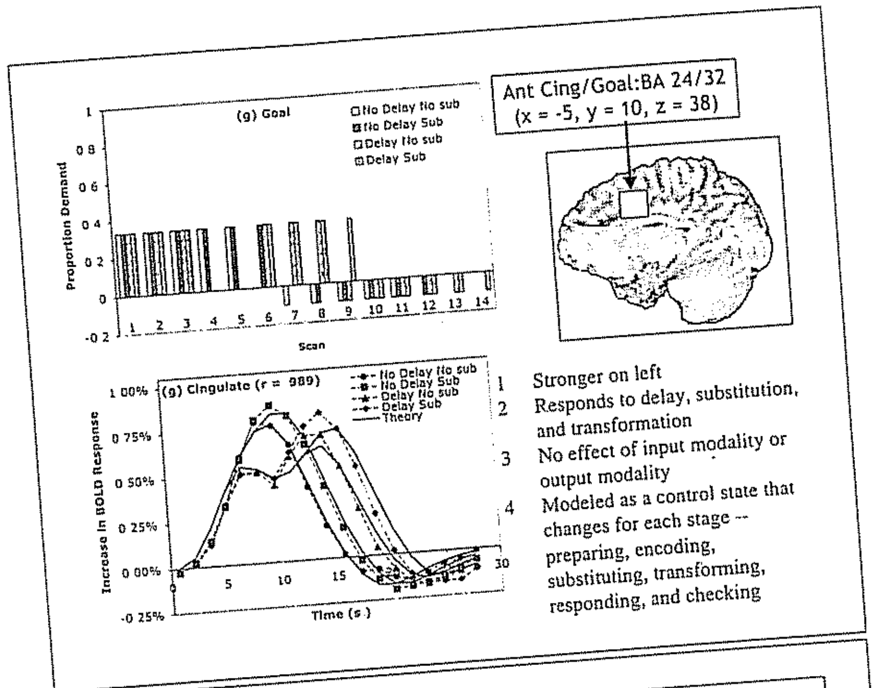
1. Stronger effect with auditory input
2. Effect in visual condition due to half of subjects that speak
3. Response equally lateralized
4. Needed to assume a more peaked BOLD response for this region only

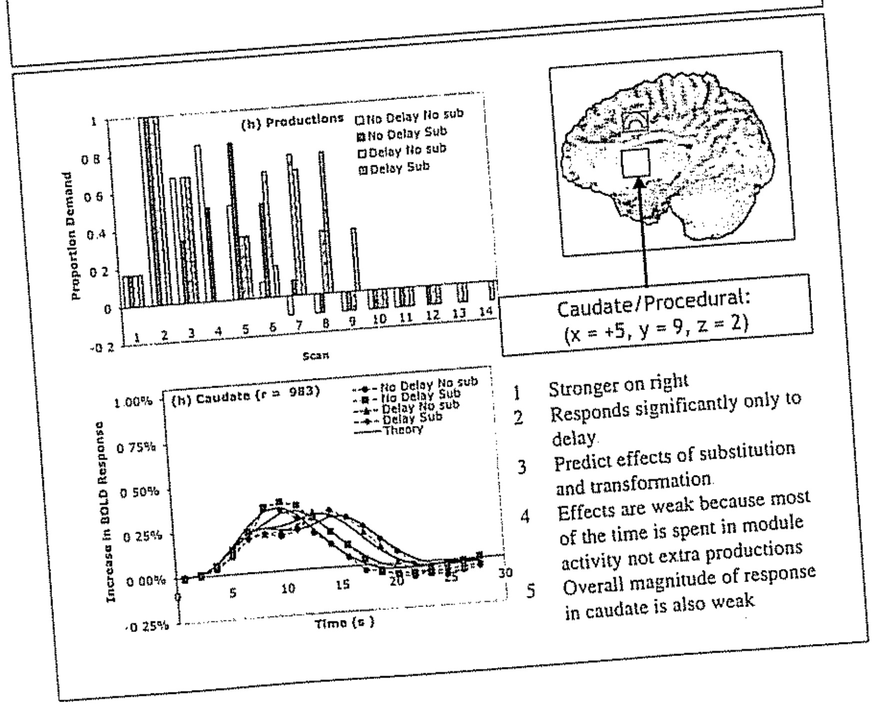

Motor1/Manual: BA 2/4
(x = -37, y = -25, z = 47)
(c) Manual
Manual No Delay
Manual Delay
Vocal No Delay
Vocal Delay
Proportion Demand
Scan
(c) Motor1 (r= 982)
Manual No Delay
Manual Delay
Vocal No Delay
Vocal Delay
Theory
Increase in BOLD Response
Time (s)
1 Stronger effect with manual output
2 Assume motor2 generalizes 34% to motor1
3 Response strongly left lateralized
4 No effect of input modality, transformation, or substitution
5 There is an effect of delay on area under curve suggesting motor rehearsal


Motor2/Vocal: BA 2/4 (x = +/-44, y = -12, z = 29)
(d) Vocal
Manual No Delay
Manual Delay
Vocal No Delay
Vocal Delay
Proportion Demand
Scan
(d) Motor2 (r= 976)
Manual No Delay
Manual Delay
Vocal No Delay
Vocal Delay
Theory
Increase in BOLD Response
Time (s)
1 Stronger with vocal output
2 Also assume motor1 generalizes 34% to motor2
3 Response not significantly lateralized
4 No effect of input modality, transformation, or substitution
5 There is an effect of delay on area under curve suggesting motor rehearsal


Parietal/Imaginal: BA 39/40
(x = -23, y = -64, z = 34)
(e) Imaginal
No Delay Trans
No Delay No Trans
Delay Trans
Delay No Trans
Proportion Demand
Scan
(e) Parietal (r = 979)
No Delay Trans
No Delay No Trans
Delay Trans
Delay No Trans
Theory
Increase in BOLD Response
Time (s)
1 Imaginal holds problem information (used be in goal)
2 Stronger on left
3 Responds to delay only in vocal condition suggesting role in supporting vocal rehearsal
4 Also serves encoding and transformation
5 No effect of input modality or substitution


Prefrontal/Retrieval: BA 45/46
(x = -40, y = 21, z = 21)
(f) Retrieval
No Delay No sub
No Delay Sub
Delay No sub
Delay Sub
Proportion Demand
Scan
(f) Prefrontal (r = 977)
No Delay No sub
No Delay Sub
Delay No sub
Delay Sub
Theory
Increase in BOLD Response
Time (s)
1 Stronger on left
2 Also responds to delay only in vocal condition suggesting it has some role in supporting vocal rehearsal
3 Also responds to substitution and checking
4 No effect of input modality or transformation





From Emotion to Memory:
An ACT-R view on the Somatic Marker Hypothesis

Andrea Stocco (stocco@units it)
Danilo Fum (fum@units it)
Department of Psychology,
University of Trieste, Italy

Introduction

The Somatic Marker Hypothesis (SMH; Damasio, 1994) is probably the most important contemporary theory of emotions According to the hypothesis, the neurological substrates of the emotions are the perceived immediate bodily reactions to environmental stimuli, which can be sensed through internal representations that are continuously updated in the sensory regions of the brain These somatic representations are conveyed, through sensory pathways, to a convergence area in the orbitofrontal cortex Within this region, they are associated with other representations conveying contextual information In this way, the emotional reactions become somatic markers for the previously encountered stimuli that elicited them

Once formed, somatic markers may be reactivated when the organism faces situations similar to the ones that induced the markers The organism is then already pre-alerted and pre-disposed to react properly, and unconsciously biased towards certain behaviors

The Iowa Gambling Task

Most of the empirical evidence supporting the SMH comes from experiments performed with a paradigm known as the Iowa Gambling Task (hereafter IGT: Bechara, Damasio, Damasio, & Anderson, 1994) This task was developed to capture, within a laboratory situation, some important aspects of real-life decision making: uncertainty about the future, lack of perfect information, and the trade-off between immediate and postponed rewards.

In the Iowa Gambling Task, participants are asked to repeatedly select a card from an array of four decks, labeled *A*, *B*, *C* and *D* Each selection always results in an immediate positive outcome Decks *A* and *B* carry bigger wins, while C and D lead to smaller monetary rewards

Unpredictably, however, a win may also be immediately followed by a subsequent negative outcome These penalties are arranged so that selecting from *A* and *B* ("bad decks") will produce an overall loss of money Therefore, the advantageous strategy is to select from *C* and *D* ("good decks"), that yield an eventual profit.

Normal participants usually start selecting from the bad decks, but end up performing significantly more selections from the good ones More interestingly, selections from the bad decks are predicted by greater anticipated increases in the skin conductance response (SCR) than selections from the good ones (Bechara et al , 1996; 1997) Since these reactions appear before participants acquire conscious knowledge of the task (Bechara et al , 1997), they were originally taken as evidence for an implicit mechanism of somatic markers that was sensing the bad strategy

Conversely, patients with lesions in the orbitofrontal cortex (OFC) do not show any SCR increase while performing the task, and, correspondingly, they remain stuck to the bad decks, unable to switch to the good ones

The original interpretation for such results has been questioned by other researchers The main point of the debate is how exactly emotions affect higher-level cognition Tomb, Hauser, Deldin & Caramazza (2002) pointed out that SCR responses may be dissociated from bad decks by varying the scheduling of losses Similarly, Maia & McClelland (2004) showed that good performance in the task is accompanied by explicit knowledge of the underlying structure, casting doubts on the supposedly unconscious knowledge carried by the markers The striking difference between healthy subjects and patients, however, is harder to frame, since it implies a specific decision-making inability in a category of patients whose cognitive skills are reported to be preserved (e.g., Eslinger & Damasio, 1985).

Recently, Fellows & Farah (2005) have proposed that the cause of patients' impairment may be an inability to acquire new stimulus-reward links once preliminary associations have been learned They were successful at showing that patients' impairment disappears when no reversal of previous expectation is required

A computational model

In Fum & Stocco (2004) we proposed a revision of the Somatic Marker Hypothesis that was grounded on a functional integration of emotion and memory We put forward a model that could replicate the basic experimental results. The core of the model was the ACT-R. declarative memory system, rewritten in Lisp and provided with special routines to implement a memory-sampling decision process within the IGT

Emotion and memory

The main tenet underlying the ACT-R. theory is that human cognitions is adaptive, and that the retrieval of information reflects the probability of occurrence of events in the environment (e.g., Anderson & Schooler, 1991).

However, sometimes uncommon events need to be recalled fast and not to be forgotten in spite of their rarity This is vital when such information is of valuable biological importance Since larger baseline activation interferes with the learning of new facts, the most rational solution is to have vital information on rare events to be easily recallable by means of large associative strengths with environmental cues. Such strengths should reflect the biological value of the information itself. As a result, relevant events may be recalled promptly in the context they are more likely to occur, without cluttering working memory as if they were constantly active.

The representations of the organism inner state provide an effective way of encoding the immediate biological value of and event, which can be also used to evaluate its associated cues. In this sense, Damasio's theory is both attractive and convincing.

In our model, neither internal somatic states nor emotions are modeled directly. Their computational counterpart, however, is their emotional impact, which is calculated for each outcome, stored, and eventually used to reinforce immediate associations between cues and events

This associative value is added to the interassociative strength S_{ij} between chunks, which is calculated according to the frequency-based Bayesian estimates as described in the equations of Anderson & Lebiere (1998)

This additional associative factor is mediated by the orbitofrontal cortex, which is also thought to play a role in the active maintenance of somatic information in working memory The contribution of the orbitofrontal cortex is expressed through a new parameter, η

Core features of the model

In ACT-R, when the proper goal is attended, the activation of related chunks is given by the sum of their base-level activation and the spreading component

$$A_i = B_i + WS_{ji}$$

We simply added a third factor that was proportional to the experienced emotional value of the event encoded in j:

$$A_i = B_i + WS_{ji} + nV_i$$

The term V_i is the emotional appraisal of the fact encoded in chunk i, and is the output of the processing of different subcortical regions—most notably the amygdala and the basal ganglia. These regions are known to be sensitive to the magnitude and frequency of rewards, and anatomically project to the OFC In case of monetary values, the emotional impact is obviously related to their numerical magnitude, and was calculated as $V_i = \log(i) / \log(\max(i))$

It may be noted that the two contextual components look similar Indeed, they both reflect the activity of two prefrontal areas (dorsolateral and ventromedial) and perform similar functions over different contents, following a general rule in the prefrontal cortex (Goldman-Rakic, 1996; Schoenbaum & Setlow, 2001).

Other computational models

Computational approaches to emotion have been attempted several times. Most notably, Rolls (2000) has proposed an autoassociator network model of the role of orbitofrontal cortex in dealing with emotionally-charged information. This approach is functionally very similar to ours Wagar & Thagard (2004) have put forward another neural model of cognitive-affective integration It is much detailed in mimicking existing neural circuits, but we regard some of its mappings as questionable

Within the ACT-R. community, Roman Belavkin has previously dealt with this topic (e.g., Belavkin, 2003) Belavkin explicitly linked the role of emotion with goal value (G) and noise in goal activation (τ) in ACT-R. Our approach is rather different, but we certainly share the common view that the main computational role of emotion is to allow further processing of relevant information, although we prefer to obtain this by means of implicit retrieval of associated declarative information Furthermore, we also share the view that the basic mechanism is to be recollected within the subsymbolic part of ACT-R, although its effects may be manifest on the symbolic side

Simulations

Because of the term ηV_i, normal participants are more likely to recollect negative outcomes that followed that own choices. When the η parameter is set to zero, the model mimics the behavior of orbitofrontal patients (see Figure 1). In this damaged version, it is completely attracted by positive outcomes, whose baseline activation shadows the negative drawbacks and hinders the spontaneous process of recalling (and re-experiencing) the aversive results

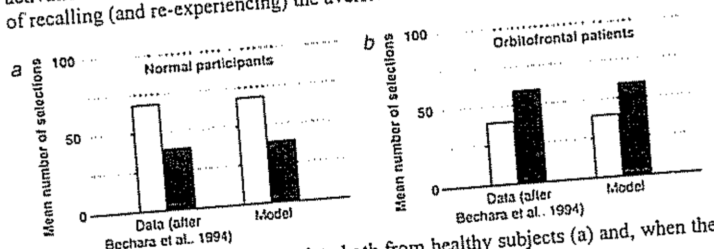


Figure 1: The model reproduces data both from healthy subjects (a) and, when the η parameter is set to zero, of OFC patients (b)

Implicit and explicit processes

Bechara et al. (1997) argued that the effect of somatic markers is entirely implicit: they drive behavior without humans being aware of their action

We take a different stance In our model, an automatic and implicit process is required to associate choices with their outcomes, and automatic and implicit is the activation of such information when a particular choice is being attended for evaluation. However, once it has been retrieved, that piece of information is fully explicit and available to conscious processing. This makes possible for a person to select certain options even when they are associated with largest penalties—and larger SCR increases, as in the experiment by Tomb et al (2002)

More simulations

In addition to its declarative memory store and the orbitofrontal linking mechanism, our model requires other components performing computations A common way of testing the hypothesized functions of such modules is to disable them and compare our impaired model's performance with that from patients having a functionally corresponding lesion

The amygdala is know to play a role in the appraisal of frequency and magnitude or rewards (Zalla et al , 2001), and, in particular, to be involved in the processing of fear In our model, this immediate appraisal of outcomes is performed by the fuction returning the V value We altered it to return zero for any of the negative outcomes, and were able to obtain a pattern of choices that is similar to what was obtained by Bechara et al (1999) Results are reported in Figure 2a

Emotion and working memory in decision making

Bechara et al (1998) reported an apparent double-dissociation: OFC patients performed normally on working memory tasks but poorly on the IGT; on the contrary, patients with lesions in the dorsolateral part of the prefrontal cortex exhibits severe impairments in working memory but scored normally on the IGT The authors suggested that decision making may rely on emotional circuits only, and be dissociated from working memory

Although based on a functional integration of emotion and memory, our model could reproduce this exact pattern of results A working memory disorder was introduced by reducing the W parameter, and then having the model run the Gambling Task Our simulated results closely resemble the original data The rationale underlying our results is that IGT is not an intensive working memory task A limited amount of attentional resources is required to sample outcomes from memory but, even if this resource is limited, their relative differences on the V value are sufficient to correctly estimate the possible drawbacks from the risky cards

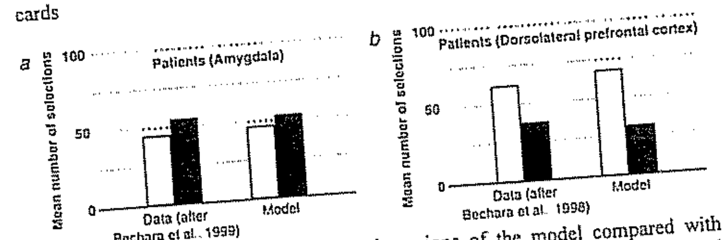


Figure 2 Performance of two damaged versions of the model compared with performance from patients with lesions in the amygdala (a) and in the dorsolateral prefrontal cortex (b)

Conclusions

With our model, we have addressed the issue of the relation between emotion and cognition within the ACT-R approach of the adaptive character of human cognition We have shown that it can reproduce the basic experimental findings reported by Damasio and co-worker Finally, we have further test our proposed model, and shown that it can also account for other neuropsychological impairments

References

Anderson, J R & Schooler, L J (1991) Reflections of the environment in memory *Psychological Science*. 2, 396-408

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

Bechara, A., Damasio, A. R., Damasio, H., & Anderson, S. W. (1994) Insensitivity to future consequences following damage to human prefrontal cortex. *Cognition, 50*, 7–15

Bechara, A., Damasio, H., Damasio, A. R., & Lee, G. P. (1999) Different contributions of the human amygdala and ventromedial prefrontal cortex to decision-making. *Journal of Neuroscience, 19*, 5473–5481

Bechara, A., Damasio, H., Tranel, D., & Damasio, A. R. (1997). Deciding advantageously before knowing the advantageous strategy. *Science, 275*, 1293–1295.

Bechara, A., Damasio, H., Tranel, D., & Anderson, S. W. (1998) Dissociation of working memory from decision making within the human prefrontal cortex. *Journal of Neuroscience, 18*, 428–437

Belavkin, R. V. (2001). The role of emotion in problem solving. In *Proceedings of the AISB'01 symposium on Emotion, Cognition and Affective Computing*, pp. 49-57. Heslington, York, England

Damasio, A. R. (1994). *Descartes' Error: emotion, reason, and the human brain*. New York, NY: Gosset/Putnam Press

Fellows, L. K., & Farah, M. J. (2005). Different underlying impairments in decision-making following ventromedial and dorsolateral frontal lobe damage in humans. *Cerebral Cortex, 15*, 58-63.

Fum, D., & Stocco, A. (2004). Memory, emotion, and rationality: An ACT-R interpretation for Gambling Task results. In C. D. Schunn, M. C. Lovett, C. Lebiere & P. Munro (Eds.), *Proceedings of the Sixth International Conference on Cognitive Modelling*. Mahwah, NJ: Lawrence Erlbaum.

Maia, T. V., & McClelland, J. L. (2004) A reexamination of the evidence for the somatic marker hypothesis: What participants really know in the Iowa gambling task. *Proceedings of the National Academy of Sciences, 101*, 16075-16080.

Rolls, E. T. (2000) *The brain and emotion*. Oxford, UK: Oxford University Press

Schoenbaum, G., & Setlow, B. (2001) Integrating orbitofrontal cortex into prefrontal theory: Common processing themes across species and subdivisions. *Learning & Memory, 8*, 134-147

Tomb, I., Hauser, M., Deldin, P., & Caramazza, A. (2002). Do somatic markers mediate decisions on the gambling task? *Nature Neuroscience, 5*, 1103–1104.

Zalla, T., Koechlin, E., Pietrini, P., Basso, G., Aquino, P., Sirigu, A., & Grafman, J. (2000) Differential amygdale responses to winning and losing: *A functional magnetic resonance study in humans*. European Journal of Neuroscience, 12, 1764–1770

Executive Control in Sentence Comprehension: An ACT-R Model of Agrammatic Aphasia

Cristiano Crescentini (crescent@sissa.it)
Cognitive Neuroscience Sector, International School for Advanced Studies
Via Dell'Orologio 6, 34136 Trieste, Italy

Andrea Stocco (stocco@units.it)
Department of Psychology, University of Trieste
Via S. Anastasio 12, 34134 Trieste, Italy

Introduction

Current hypotheses about agrammatism refer to different frameworks. Within the framework of the Government and Binding theory, Grodzinsky's Trace Deletion Hypothesis (TDH) states that agrammatism results from a damage to a specific mechanism connecting the antecedent to its trace (Grodzinsky, 2000). In opposition to the TDH, Piñango (2000) postulates that a processing deficit is at the core of the agrammatic comprehension style. She put forward the Slow Syntax Hypothesis (SSH), which drives attention on the effects of the movement of the grammatical constituents of a sentence, and focuses on a type of movement that provokes a deviation from the canonical order of thematic roles in the surface representation of the sentence. According to this view, lexical activation in agrammatic patients is slower than normal, and therefore they are unable to build the syntactic structure of the sentence quickly enough to prevent semantic linking from emerging and dominating the meaning derivation process. Piñango provides evidence for the SSH exploiting the case of the psychological verbs of the *"Frightened" type* also called *Object-Experiencer* (OE) verbs since they show the spontaneous feature of reversing the order of the thematic role in their active version, showing the Default thematic grid of *Experiencer-Theme* only in the passive form. The behavior with this verbs is the opposite to that with *Subject-Experiencer* (e.g. *"Love", Like"etc.* ES verbs) verbs in which the thematic role of *Experiencer* shows up as the subject of the sentence. In active form, the verbs of this group have a default *Experiencer/Theme* thematic construction. As reported in Grodzinsky (2000), agrammatic patients perform well with the active form of *Subject-Experiencer* verbs and with the passive of the *Object-Experiencer* verbs whereas chance performance is found with passives of *Subject-Experiencer* and with actives of *Object-Experiencer*

A computational model

We postulated that the cause of the slowing of the lexical activation process is an inability to inhibit intrusive lexical information. We tested our hypothesis within the simplified domain of psychological verbs. The model tries to reproduce in the most detail the parsing process, basing it on a previous ACT-R model developed by Lewis (1999).

The crucial step in the model is the retrieval of a thematic grid, which triggers the assignment of roles to the encountered nouns. This retrieval is cued by the processing of specific words, which are either nouns, verbs or words denoting a passive form.

Thematic grids, like any other piece of declarative knowledge in the model, have an associated activation value that expresses their availability to retrieval and reflects the past history of the chunk itself. This base-level activation may be overcome by a contextual component, which spreads from the amount of attentional resources devoted to a specific lexical cue. This amount is ruled by a

single parameter, *W* By maintaining sustained activation of a few elements, this parameter enables working memory and goal-directed behavior (Altmann & Trafton, 2002)
Since the default argument order is *Experiencer/Theme* in the sentences we used, this information is more active Contextual activation is required to overcome it and retrieve the opposite structure, as is in passive forms of ES and active forms of EO verbs With an abnormally lower value of *W*, the contribution of contextual activation is insufficient to enhance the *Theme/Experiencer* grid, letting the default one compete for retrieval. This interference increases the time needed to complete it and the probability of assigning the wrong roles in the semantic representation

Results

We tested our model's syntactic comprehension in a simulated experiment The model was first presented with a study set of 12 sentence, made of six SE verbs and six OE verbs In each category, half of the sentences were in active, and the other half in passive form. The model was tested on a second set of other 12 sentences, made with the same materials of the first ones. As predicted by our account, in the normal version, with a W value of 3 0, model's performance was errorless for each sentence (Figure 1) whereas in the damaged version, with a W of 2 0. model's performance was at chance when the study or the test sentence was either a passive ES or an active EO sentence An examination of model's semantic representation showed that in both cases there was a 50% chance of misrepresenting thematic roles.

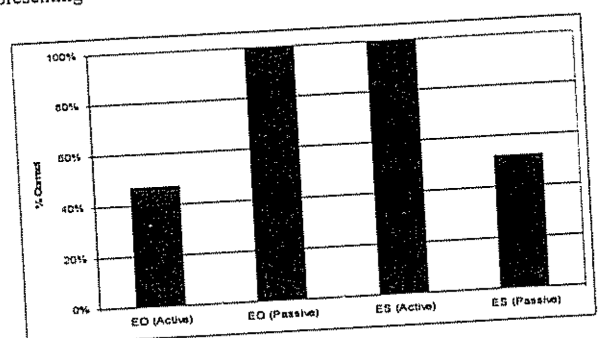


Figure 1: Mean performance for the normal simulation in the comprehension of different sentences with psychological verbs.

Discussion

We presented a computational model that postulates agrammatism as a disorder stemming from the inability of using on-line lexical information to overcome interference amongst competing syntactic elements. This approach is consistent with the time course hypothesized within the SSH. Computer simulations showed that, in virtually damaged conditions, the model could correctly reproduce the behavior of both participants and aphasic patients

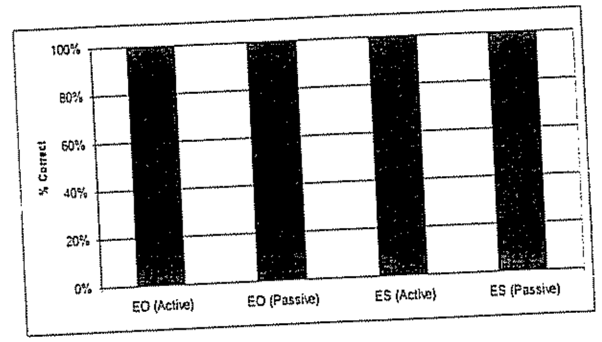


Figure 2: Mean performance for the aphasic simulation in the comprehension of the same materials Results are averaged over 200 simulations in the "aphasic" condition

References

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

Grodzinsky, Y. (2000) The neurology of syntax: language use without Broca's area *Behavioral and Brain Sciences*, 23, 1-71

Lewis, R. L. (1999) *Attachment without competition A race-based model of ambiguity resolution in a limited working memory* Presented at the CUNY Sentence Processing Conference, New York

Piñango, M. M. (2000) Canonicity in Broca's sentence comprehension: The case of psychological verbs In Y. Grodzinsky (Ed.), *Language and the brain: Representation and processing* (pp 327–350) San Diego/London: Academic Press

Neural Correlates of "Expert" Geometry Problem-Solving

Yvonne Kao
Project Advisor: John Anderson

Carnegie Mellon University
Center for the Neural Basis of Cognition
Program in Interdisciplinary Educational Research

Goals and Motivation

- To understand the underlying cognitive processes that take place during geometry problem-solving
- To see how the process of geometry problem-solving is realized in the brain

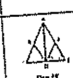
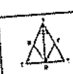
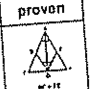
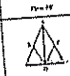
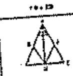
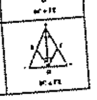
Overview of Study

- Above average geometry problem solvers
- Solve problems of varying difficulty in the fMRI scanner
- See what happens
- Compare to algebra

Geometry Domain

- Plane geometry
  - Reflexivity
  - Vertical angles
  - Parallel lines
  - Isosceles triangles
  - Triangle congruence

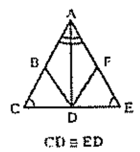
Design

Difficulty x Highlight	1-step	3-step	Cannot be proven
No Highlight			
Highlight			

Procedure

- Training session
  - Self-paced review of geometry concepts
  - 12 self-paced practice problems
  - 36 timed practice problems
- Scanning session
  - 120 timed problems
  - 1 scan every 2 seconds

Trial Structure

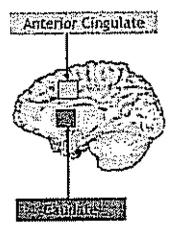
Problem	Feedback	ITI
	Correct! or Incorrect	+
Max 30 seconds (? scans)	1 second (0 5 scans)	16 seconds (8 scans)

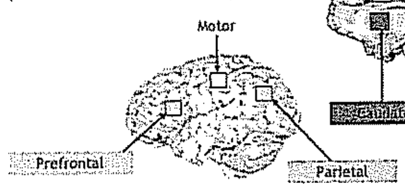
Predictions

- Main effects
  - Difficulty → Accuracy
  - Difficulty → Latency
  - Highlight → Latency

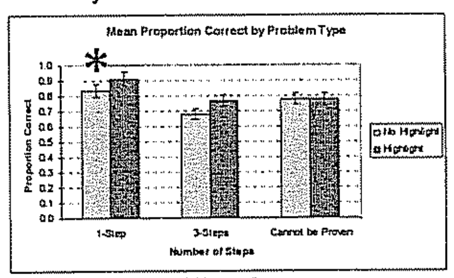
Predictions

- Activity in regions associated with ACT-R modules
- Significantly more RH activation (Parsons & Osherson, 2001)



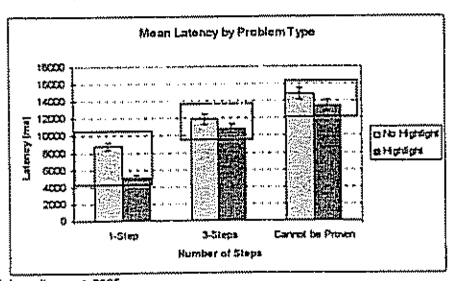


Accuracy



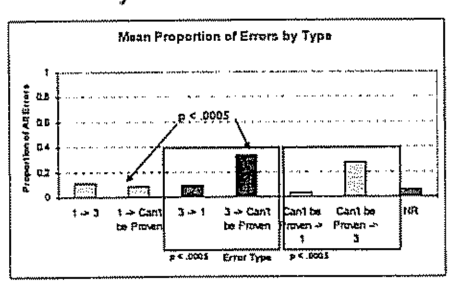
Difficulty: $p < .0005$ Highlight: $p = .027$

Latency



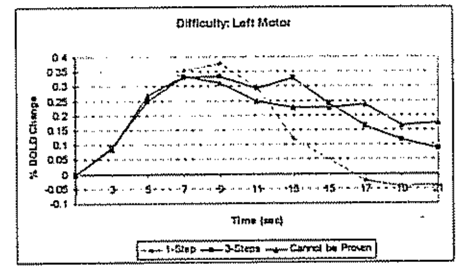
Interaction: $p < .0005$

Error Analysis



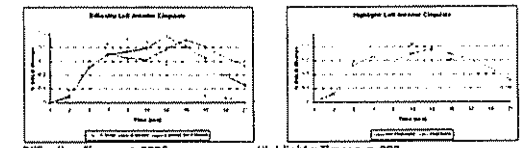
$p < .0005$

Left Motor



Difficulty x Time: $p = .001$

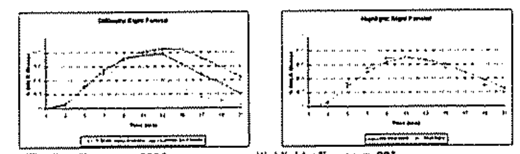
Left Anterior Cingulate



Difficulty x Time: $p < .0005$ Highlight x Time: $p = .002$

- Right side also shows both effects

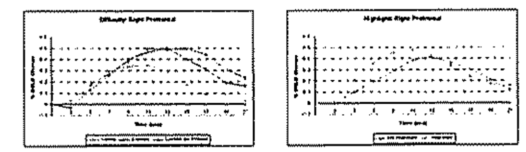
Right Parietal



Difficulty x Time: $p < .0005$ Highlight x Time: $p = .021$

- Left side is slightly greater in magnitude
  - Shows difficulty effect
  - Highlight effect is marginal

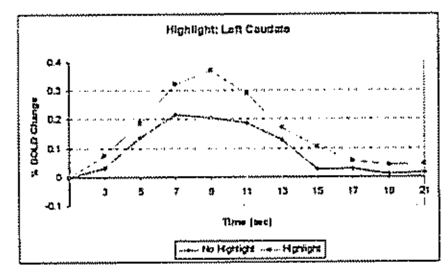
Right Prefrontal



Difficulty x Time: $p < .0005$ Highlight x Time: $p < .0005$

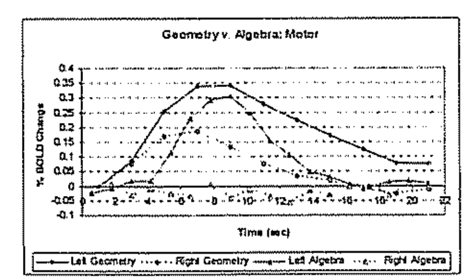
- Left side also shows difficulty effect
  - Less in magnitude overall

Left Caudate

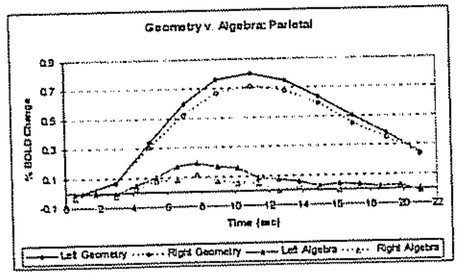


Highlight x Time: $p = .035$

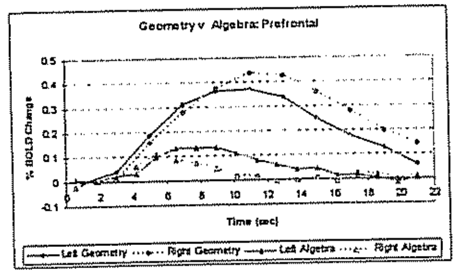
Task Comparison: Motor



Task Comparison: Parietal



Task Comparison: Prefrontal



Conclusions

- 1-step v. 1+ steps strategy
- Sensible results for Difficulty
- Surprising effects of Highlight
  - Accuracy
  - Anterior cingulate, prefrontal, caudate
- Left hemisphere vs. Right hemisphere
  - ACT-R

Future Work

- ACT-R Modeling
  - Diagram Configuration Model
    - Koedinger & Anderson, 1990

Thank You!

Advisor: John Anderson

Committee:
Marcel Just
Ken Koedinger

Jon Fincham
Jennifer Ferris
Pat Gunn

ACT-R 6
Official Release

Dan Bothell
Carnegie Mellon University

Brief History

- Proposed at the 2002 Workshop
  - Concurrently with ACT-R 5's release
- Initial description at the 2003 Workshop
  - Early prototype
  - Claimed a 2005 Workshop release
- Discussion session after ICCM 2004
  - Fleshed out some issues with syntax
- Here it is!
  - Fully functional
  - Used it for the 2005 Summer School

2

What is ACT-R 6?

- The same theory as ACT-R 5
- Rewritten implementation
  - Eliminate unnecessary legacy code
  - Unify/standardize the buffer mechanism
  - Better integration of the Cognitive and Perceptual/Motor components
    - Only one time maintenance mechanism
  - Make the whole system modular
    - Easy to add new components
    - Easy to remove/replace existing ones

3

How similar is it to ACT-R 5?

- Very similar
- Most of the commands are still there
  - reset, clear-all, sgp, p, add-dm, run
- Models look basically the same
- Same equations
  - Procedural
  - Declarative memory
- With basically the same parameters
  - Same defaults and usage
- Same Perceptual and Motor modules

4

Why should I use it?

- It cleans up some issues that can make ACT-R 5 tricky to work with
- It has new features
  - To make things easier for modeling
  - To add some requested capabilities
- It is easier to extend and modify
  - Easier to distribute and combine extensions
- In many cases it is faster than ACT-R 5

5

Things that were cleaned up

- Overall structure
- Buffers
- Declarative memory
- Productions
- Vision module
- Module states
- Production compilation
- Available commands

6

Basic structure

- A central event scheduling system
  - Independent of the theory itself
- A set of modules
  - All treated equally
  - Should each be independent
  - May have one or more buffers as an interface
  - Responsible for scheduling its own events

7

Buffers

- They all work the same
  - Can hold one chunk
  - Relay queries and requests to/from a module
- The chunk is a copy
  - Doesn't exist outside of the buffer until it is cleared
  - Changes are not reflected back to the original chunk
- Essentially chunk creation scratch pads

8

Chunks

- Not just for Declarative memory
- Any module can create/use chunks
- The set of all chunks does NOT equal DM!

9

Declarative Memory

- Holds only the chunks that are added explicitly or those that come from the buffers
- When a buffer clears the chunk merges into DM
  - True for all buffers
- Those merges are the references for base-level learning
  - Not the LHS usage as in ACT-R 5
- Because buffers hold copies DM chunks can't be changed from within a production
  - Previously it was a recommendation

10

General Production Changes

- No LHS Retrievals
- Can't use !eval! in the slot value position
- More rigorous syntax checking
- LHS ordering not important

this will work:

```
(p test
  =goal>
   isa goal
   - value = value
  =retrieval>
   isa fact
   slot  =value
 ==>   )
```

11

Productions LHS

- Only four possible conditions available
  - *=buffer>*
    - Test the chunk in the buffer just like in 5
  - !eval! or !safe-eval!
  - !bind! or !safe-bind!
    - Same as in ACT-R 5
    - Safe versions accepted by production compilation
  - *?buffer>*
    - Query the buffer or its module

12

Buffer queries

- Replaces the *-state buffers
- Syntax

  ?buffer>
  *((-) query value)**

- Either true or false
  - No bindings
- Must all be true for production to match
- Examples

```
?retrieval>          ?visual>
  state   busy         - state   error
  buffer  empty        buffer    =check
```

13

Queries continued

- Every buffer/module must respond to
  - State
    - Values: busy, free, or error
  - Buffer
    - Values: full, empty, requested or unrequested
  - Others can be added by a module writer
    - Modality for the current PM modules for example

14

Production RHS

- Essentially the same operators as in 5
- Removed the obsolete ones
  - !pop!, !push!, !retrieve!, etc
- Standardized the mechanism for all buffers

15

Possible RHS actions

- *=buffer>*
- *-buffer>*
- *+buffer>*
- !eval! and !safe-eval!
- !bind! and !safe-bind!
- !output!
- !stop!

16

RHS actions

- *=buffer>*
- !eval! and !safe-eval!
- !bind! and !safe-bind!
- !output!
  - All the same as in ACT-R 5
  - The safe versions do not inhibit the production compilation mechanism
- !stop!
  - Not actually new, but does work now
  - Generates a break event in the scheduler
  - Terminates the current "run" command

17

RHS -buffer>

-buffer>

- Clears the chunk from the buffer
- That's it!
- Does not result in any action by the module
  - Unlike ACT-R 5 where that could also cause the corresponding module to reset/clear

18

RHS +buffer>

+buffer> isa chunk-type
*{{modifier} [slot | request parameter] value}**

or

+buffer> *chunk-reference*

- Sends a request to the module
  - Implicitly clears the buffer as well
  - Essentially the same as ACT-R 5

19

Vision Module

- Removed the attended slot from visual-location chunks
- Replaced with
  - a RHS request parameter
    +visual-location>
    isa visual-location
    :attended nil
  - A LHS query
    ?visual-location>
    attended nil
- Good because now visual-locations can merge properly without the changing attended slot
- The query can match nil to new but a LHS slot test couldn't

20

Vision Module cont.

- Attention Shifts changed from
  +visual>
  isa visual-object
  To
  +visual>
  isa move-attention
- No longer need the scale slot in visual-objects
- Easier to read in productions
  - The old systems analogy to declarative didn't seem all that helpful

21

Production Compilation

- The same general theory as 5
  - Combine consecutive productions into one
  - Incorporate requested chunks and remove the request
- Mechanism is now split into two distinct steps and applied on a buffer-by-buffer basis
  - Check for possibility of composition
  - Perform the composition
- More robust than the mechanism in 5
  - Slightly more restricted than the 5 mechanism

22

Production Compilation cont.

- Applies to all buffers (even user created)
- Basic mechanism is that there are 4 styles of buffers
  - Goal, retrieval, perceptual, and motor
- Any buffer can be set to any style
- New styles can be added
- Existing styles can be modified for both steps

23

Commands

- Removed some duplicate commands
  - {set-general-base-levels. set-all-base-levels, set-base-levels, setgeneralbaselevels, setallbaselevels. setbaselevels} → set-base-levels
- The PM commands have had the "pm-" removed
  - For example pm-proc-display is now proc-display
- Commands referencing obsolete items removed
  - In particular anything that included wme
- Sgp sets parameters for all modules

24

New Features

- Request parameters
- Declarative finsts
- Sources of activation
- Multiple models
- Strict Harvesting
- P* command

25

Request parameters

- Buffer specific parameters
  - Valid no matter what the chunk-type
  - Always a keyword (which distinguishes it from an actual slot)
- Examples

  +visual-location>
  isa visual-location
  :attended nil

  +retrieval>
  isa any-chunk-type
  :recently-retrieved nil

26

Declarative Finsts

- Cannot modify chunks in DM in a production
- Major reason for changing chunks in DM was to mark them to prevent retrieval
- Now there are automatic markers just like vision
- They are limited in time and number
  - settable with parameters
- Indicated with the request parameter :recently-retrieved
  +retrieval>
  isa fact
  :recently-retrieved nil

27

Sources of activation

- All buffers are potential sources now
- Each buffer has a separate parameter like :ga for the goal buffer
  - :ga defaults to 1
  - All others default to 0
- :mas now also used to enable/disable spreading activation since setting :ga to 0 is not sufficient

28

Multiple Models

- Out of the box ACT-R 6 supports multiple models
- Any number of models can be loaded
- Each has its own set of modules, chunks, and parameters
- Can be run synchronously or asynchronously
  - Determined when loaded
  - Not adjustable afterwards

29

Strict harvesting

- New mechanism of productions
- When a buffer is matched on the LHS of a production it is automatically cleared on the RHS unless there is an =buffer action to keep it around
  - Parameterized so that one can specify which buffers get "strict harvested"
  - Out of the box all but the goal buffer do
- Cleans up issues with
  - References for BLL
  - Production compilation
  - Micro-managing perceptual buffers

30

Experimental addition: P*

- Exactly like p except slot-names can be variablized
  - On both the LHS and the RHS
- Limited variability (for now at least)
  - Will not do any binding – the variable must be bound elsewhere
  - Only one level deep per buffer test

31

Example P* uses

```
(p* search
  =goal>
    isa search
  =retrieval>
   isa strategy
   constraint =c
   value =v
==>
  +visual-location>
    isavisual-location
    =c =v
)
```

```
(p* check
  =goal>
   isa check
   which-slot =s
   which-value =v
  =retrieval>
   isa memory
   =s =v
==>
  )
```

32

Extending via new Modules

- All modules are built the same way
  - Including the defaults
- Can remove or replace any module*
- Placing a file in the modules or tools directory with a lisp name will cause it to be loaded
- Eventually would like to have a database of available modules and tools that people can use
- No "how to" docs right now, but the current modules serve as examples and there is an API doc that describes the available functions

33

Modifying the base modules

- Declarative and Procedural modules are now more user configurable
- All the equations have 'override' hooks like similarity did previously
  - :BL-HOOK
  - :SPREADING-HOOK
  - :PARTIAL-MATCHING-HOOK
  - :NOISE-HOOK
  - :SIM-HOOK
  - :SJI-HOOK
  - :UTILITY-HOOK
  - :UTILITY-C-HOOK
  - :UTILITY-P-HOOK
- Should relieve people of needing to hack the main code

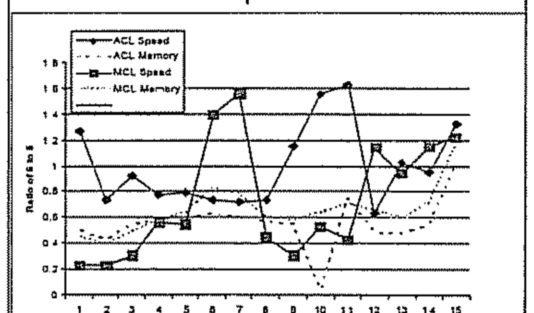
34

Performance Evaluation

- Has not been highly optimized yet
- Used the tutorial models as a benchmark because they touch all the main components
- Used ACL 6 2 on Windows XP and MCL 5 0 on Mac OS X 10 4
  - Need to increase the MCL heap under OS X (ccl::set-preferred-size-resource *heap-size-in-bytes*)
- Basic speed and size comparison
  - Using the time function

35

Comparison



More Information

- The tutorials show the new system in use
- The test models in the distribution are the commented conversion of the ACT-R 5 tutorial models
- User manual included with the docs
  - Still a bit rough, but it is being worked on
- Can always look at the source code
  - A little more structured/spread out
  - Slightly more commented

37

Where can I get it?

- The ACT-R website
  http://act-r.psy.cmu.edu
  - Updated when there are significant changes
- Via Subversion
  - Always the most up to date code
  - Version control software available from http://subversion.tigris.org
  - All files are under version control
    - Including the tutorial, docs, and the environment
  - Available from our server at svn://alba psy cmu edu/usr/local/svnroot/actr6

38