Title
A System-level Brain Model of Spatial Working Memory

Permalink
https://escholarship.org/uc/item/37b448zj

Journal
Proceedings of the Annual Meeting of the Cognitive Science Society, 28(28)

ISSN
1069-7977

Author
Bond, Alan H.

Publication Date
2006

Peer reviewed

A System-level Brain Model
of Spatial working Memory

Alan H. Bond (alan.bond@exso.com)
National Institute of Standards and Technology,
MS 8263, Bldg. 220, Gaithersburg, Maryland 20899, and
Institute for Neuroscience, UCLA, Los Angeles, California 90095

Abstract

A system-level model of spatial working memory is described, using the author's computer science and logical modeling approach. The mental image which is remembered is located in a lateral parietal area, and is part of a larger distributed representation including object identities and object appearances. The system is driven by plans which are stored in a separate module corresponding to ventral prefrontal and then copied to a planning module corresponding to dorsal prefrontal, where they are executed and sequenced. Mental images are maintained by explicit messages from a currently executed plan. Upon a cue to use a saved mental image for motor output or eye movement, a message from the plan causes the image to be temporarily reinstated allowing it to be used for motor control. We describe a gradual approach to deficits in which a goal attenuates, causing the plan to fall below a threshold. This approach allowed us to match the experimental results of Conklin et al. in which schizophrenic and schizoptypal subjects could carry out a spatial working memory task with a 0.5 second delay but not when there was a 7 second delay.

Introduction

In order to more precisely characterize psychiatric disorders such as schizophrenia, bipolar disorder, autism and ADHD, in recent years there has arisen the concept of *cognitive endophenotype*, which is a particular cognitive ability that may have deficits linked to certain genes. We need to understand these cognitive abilities and their relation to the brain, and we need to understand which underlying processes are shared in common among cognitive functions.

Glahn et al [Glahn et al., 2003] correlated several different components of spatial working memory such as encoding, maintenance, manipulation, time-tagging of visual spatial information, storage capacity and complex motor response, against genetic predisposition to schizophrenia. They concluded that encoding and storage aspects of spatial working memory may be effective endophenotypic markers for schizophrenia.

In this study, we attempted to analyze and understand the underlying mechanisms involved in one cognitive ability, namely spatial working memory. This has been shown to be clearly impaired in schizophrenia. In order to understand it, we developed a system-level model of spatial working memory.

We used a modeling approach of a modular distributed computational architecture and an abstract logical description of data and control [Bond, 1996] [Bond, 1999] [Bond, 2004a], for which we have also analyzed its correspondence to the cortex [Bond, 2004b].

Modeling spatial working memory involved developing mechanisms for a frontal area containing a maintenance process, a posterior area containing mental images, and an integration mechanism involving a simple model of episodic memory, corresponding to the hippocampal complex.

We implemented the model as a computer system and studied normal behavior and then abnormal behavior by introducing different types of component deficit. We specifically obtained a match to the work of Conklin et al [Conklin et al., 2005] on schizophrenic and schizoptypal subjects. For short delays such as 0.5 seconds they performed normally, but for long delays such as 7 seconds they exhibited a clear deficit.

Spatial working memory

We can perhaps define spatial working memory by describing the basic experiment. Different experiments in spatial working memory have been reviewed by Curtis and D'Esposito [Curtis and D'Esposito, 2003] and by Rowe et al [Rowe et al., 2002].

The subject first fixates a central fixation point, then an additional image of a small object appears at a certain spatial location in the periphery, then there is a delay with just the fixation point visible, and then a cue is given and the subject moves either their eye gaze or their hand to the spatial location where they think the small object was. Thus, the idea is that the subject has to remember a certain spatial location for a short time of the order of a few seconds.

In our research, we used a basic experimental design from a recent standard paper by Conklin et al [Conklin et al., 2005]. This has four steps.

1. fixate - there is a cross at the origin and the subject has to fixate it, duration 2000 milliseconds.

2. note image - there is now also an asterix at a peripheral location, duration 200 milliseconds.

3. delay, maintain image - back to just the cross, duration either 500 milliseconds or 7000 milliseconds.

4. cue, move hand to where the asterix was, duration 5000 milliseconds.

Our modeling approach

Our general brain modeling approach. In the

last few years, we have conducted a series of studies and models concerning problem solving [Bond, 2002b], episodic memory [Bond, 2005b], natural language processing [Bond, 2005a], routinization [Bond, 2006], and social relationships [Bond, 2002a]. For this project, we have begun integrating all of these mechanisms into a single system which we call our dynamic model.

Biological information-processing principles
The basic principles of our design are derived from the biology of the neocortex:

1. Each neural area stores and processes data of given types characteristic of that neural area; data items are of bounded size.

2. To form systems, neural areas are connected in a fixed network with dedicated point-to-point channels.

3. Neural areas are organized as a perception-action hierarchy.

4. Neural areas process data received and/or stored locally by them. There is no central manager or controller.

5. All neural areas have a common execution process, which constructs data items.

6. All neural areas do similar amounts of processing and run at about the same speed.

7. There is data parallelism in communication, storage and processing. Processing within a neural area is highly parallel. Parallel coded data is transmitted, stored, and triggers processing. Processing acts on parallel data to produce parallel data.

8. The data items being transmitted, stored and processed can involve a lot of information; they can be complex.

9. The neural areas act continuously and in parallel.

The realization of a system-level brain model using logic programming

Modules. A system-level brain model is a set of parallel modules with fixed interconnectivity similar to the cortex, and where each module corresponds to a brain area and processes only certain kinds of data specific to that module.

Data items, and their storage and transmission. We view all data streams and storage as made up of discrete data items which we call *descriptions*. We represent each data item by a logical literal which indicates the meaning of the information contained in the data item. An example data item is `position(adam,300,200,0)` which might mean that the perceived position of a given other agent, identified by the name "adam", is given by (x,y,z) coordinates (300,200,0). In order to allow for ramping up and attenuation effects, we give every data item an associated strength, which is a real number. Stored data items are ramped up by incoming identical or related data items, and they also attenuate with time, at rates characteristic of the module.

Processing within a module. We represent the processing within a module by a set of left-to-right logical rules [1] which are executed in parallel. A rule matches to incoming transmitted data items and to locally stored data items, and generates results which are data items

which may be stored locally or transmitted. Rule patterns also have weights, and the strength of a rule instance is the product of the matching data item weights and the rule weights, multiplied by an overall rule weight.

A rule may do some computation which we represent by arithmetic. This should not be more complex than can be expected of a neural net. The results are then filtered competitively depending on the data type. Typically, only the one strongest rule instance is allowed to "express itself", by sending its constructed data items to other modules and/or to be stored locally. In some cases however all the computed data is allowed through. One cycle of the model corresponds to about 20 milliseconds.

Uniform process. The uniform process of the cortex is then the mechanism for storage and transmission of data and the mechanism for execution of rules.

Perception-action hierarchy. Modules are organized as a *perception-action hierarchy*, which is an abstraction hierarchy with a fixed number of levels of abstraction.

Differences between our approach and others. Our approach differs from present-day cognitive models such as ACT-R, SOAR, 4CAP and Kintsch's comprehension model, in that:

1. It is a parallel model, with multiple modules running in parallel.

2. Its basic unit of data is the chunk, or structured packet of information, with chunks being constructed, stored and transmitted by operations of the model. Chunks are accessed associatively from memories in modules.

3. It corresponds to brain architecture, with modules corresponding to brain areas, connectivity corresponding to the connectivity among corresponding brain areas, and data types corresponding to those processed in corresponding brain areas.

4. The dynamics consists of distributed and coordinated sets of processes, for real time control, for plan elaboration and for episodic memory creation and use.

5. The computational method is based on logic programming [Kowalski, 1974] and there is an underlying theory of these models which provides formal semantics and completeness and convergence properties [VanEmden and Kowalski, 1976].

Incidentally, the concept of a "rule-based model" is not well founded since all known formulations of computation, including functional, automata theoretic, and logical, can be put in the form of rules, and small changes in the form of rules may lead to large changes in computational properties. Neither is a logic programming model "symbolic" since not all models of these logics contain symbols as individuals, and indeed most do not.

Our approach differs from neural network approaches in using an abstract method of description, so that information is represented by abstract chunks and processing by rules which describe the processing of chunks. It also differs from the abstract neural models of Cohen and Braver [Braver et al., 1999] in using complex data items and matching, and in using complex computation and control within each module.

---

[1] Our logical rules are clauses, i.e. they have no nesting, but are made up of a disjunction of possibly negated literals

Our approach is complementary to neural network approaches, and it should be possible, for a given abstract model, to construct corresponding neural network models.

Our approach to the design of a spatial working memory model

Our model is implemented as a set of intercommunicating brain modules that run in parallel.

Our solution to the extension of our model has involved:

(i) generalizing our planning module to use learned plans represented as data items, so that rules competitively reconstruct data items in response to their current situation,

(ii) adding a mental imagery module which stores and represents mental images in terms of image elements and their spatial relationships, and

(iii) adding a module corresponding to the hippocampal complex, and which receives data from cortical areas and constructs a representation of the current mental event.

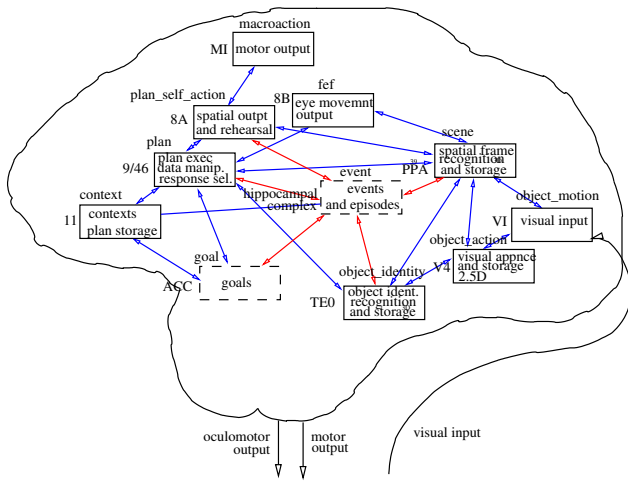Figure 1 diagrams the design of the basic spatial working memory model.

Figure 1: Outline diagram of an initial model

Mental images. There is by now considerable experimental evidence for a network of modules in the human brain which process visual information. A distributed cognitive model of mental imagery was developed by Kosslyn [Kosslyn, 1994] which did not correspond to neuroanatomical areas but did define functionalities that may be present in an imaging store, as well as a propositional representation in a semantic store. Functions involved in imaging included image construction and image gathering, image processing functions such as translation, zooming and so on, and image attention functions.

We decided to use a simple distributed representation for mental images, for future development, so there are four modules altogether. The first two we already have been using before, object_motion is the basic input

module for visual information and object_action computes basic spatial relations among perceived visual objects. Then we added two new modules, one for object identity ("what") called object_identity, corresponding to the ventral temporal areas, and one for spatial layout ("where") called scene, corresponding to lower parietal areas.

Plans. Plans are represented in a form which should be derivable from experience of action, learning by doing. This form allows a mixture of short sequences (3, 4 or 5 steps) and embedding of one subplan within another. So this corresponds to the form of episodic memory in organizing experiences of sequences of mental states.

The form of plans is such that they could be learned either by learning by doing, or by being instructed. These mechanisms were not implemented for this project.

Plans are stored in the context module, corresponding to ventral prefrontal, and evoked by the existence of goals communicated from other modules.

Plan steps are a bit like rules but they are data, and they are executed by executing rules in the planning module, corresponding to dorsal prefrontal. The actual rules in the planning module are general rules which execute plan steps.

We call plans contexts. A context is then a set of context descriptions. There is a head context which is triggered by a goal and determines the sequence of steps, and then there are four contexts representing these steps.

We show below the plan we used for the basic spatial working memory experiment, Figure 2 shows the structure of the plan into a head and the set of plan steps.
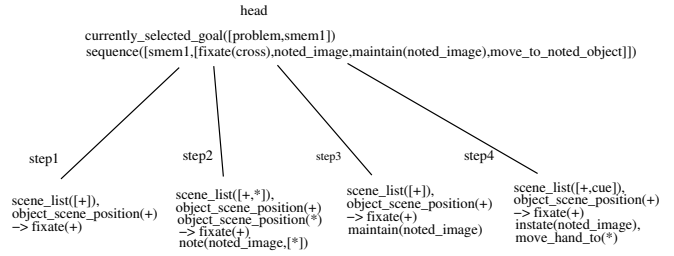

Figure 2: Diagram of structure of plan for spatial working memory

The plan is actually represented and stored in associative memory, as five data items, each data item being a single logical term of the form: context(key,if_part,then_part,provided_part,weight_part).

The execution of plans. In execution, steps are selected based on the current perceived image, which is taken from scene, as well as their order in the sequence. A step gets activated when its conditions are true and provided it is next in the current sequence.

A context step has a key which describes it and when it is being executed this key is asserted. A context key actually refers to the step and to its current parent, for example [noted_image,smem1] is the key for plan step noted_image evoked from the head with key smem1.

During the execution of a context, there will usually

be two such keys in existence, one for the head, or parent, and one for the step being executed. Thus, during the second step of the plan, the cecs, currently evoked contexts, are [smem1,top] and [noted_image,smem1], and during the third step they are [smem1,top] and [maintain(noted_image),smem1].

Figure 3 shows a general case of a plan made up of several nested and sequenced plan steps. It shows the activation of step1212 and the control memory at that time consisting of a set of cec expressions.
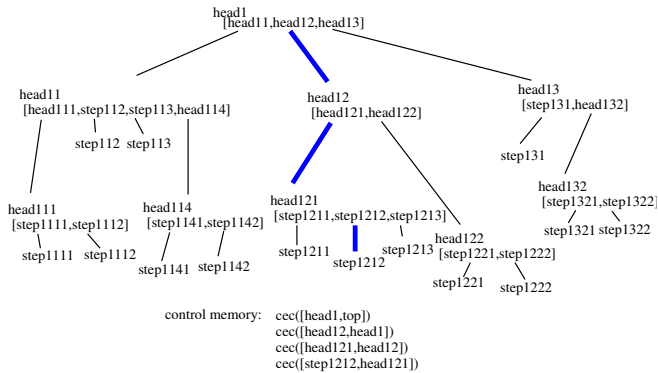

Figure 3: The memory of execution of plans

The cecs and the current sequence give us the information to maintain the orderly execution of plans. These data items form a short term memory for the execution of plans. This memory is of course in the store of the plan module.

Incidentally, cecs are continuously re-evoked every cycle, independently and in parallel, and if any cec is not evoked then this can cause termination of the current step at any level. This allows a plan step to succeed or fail at any level. If it terminates at a higher level, all the cecs below will not be re-evoked, since they can only be evoked if their parent exists. If the last step in a sequence terminates then this leaves just the head of the sequence, which then itself terminates.

Episodic memory. We chose to use a very simple version of episodic memory, corresponding to the hippocampal complex, and following our published ideas on episodic memory [Bond, 2005b]. The main idea is to form representations of instantaneous mental events and then to form episodes in nested groups of no more than 4 events or episodes. The system computes an associative key for each event or episode that allows its unique retrieval from the store of the episodic memory module. The contents of this store is usually called a *cognitive map*.

In representing the current episode in episodic memory, we are currently taking the current episode key to be simply the set of these active cecs, so this is a path through the context nesting to the current context being executed. Thus in the above example the current episode key is [[smem1,top], [noted_image,smem1]].

Noting, maintaining and using mental images.
In our model, plan steps send messages to the scene module to perform certain operations on mental images stored there. Basically, we need to make a note of an image of the scene to be remembered during the note phase, then we need to maintain this image and stop it from attenuating away during the maintain phase and then we need to reinstate the remembered image during the action upon cue phase.

We concluded that under normal circumstances there was a flow of continuously changing perceived images through the visual system. However, some salient images, or images that are explicitly noted, will have constructed more precise, detailed or complete representations which will be labeled with an associative key. Such a key enables the system to store a set of different descriptions of different aspects of a given scene, and to be able to retrieve them using the associative key that all these related descriptions contain. In the 1981 version of Kosslyn's model [Kosslyn, 1981] he has a LOAD operation which basically grabs a new image from the visual system, so this is related to our own idea. Then there can be several mental images stored in the mental imagery module, as well as the ongoing instantaneously perceived visual image. The stored mental images will attenuate fairly rapidly. A given image can be prevented from disappearing by maintaining it with constant retrieval and reconstruction. Also of course images can be captured as components of episodic memory and then stored in long term memory, but this will only occur for a small number of possible images and may take longer.

Every mental image that is constructed has an associative key; we call such constructed images *scenes*. We arbitrarily chose the current episode key as a key for the current mental image, i.e., the one derived from the current percept. Normally, as the scene changes so does the mental image corresponding to the percept, and it is named with the new current episode key. The previous image will attenuate fairly fast and disappear; with the current settings this takes about 3 cycles.

Given this approach, we were able to precisely define noting, maintaining and reinstating of images:

(i) Noting the current mental image creates a new scene which is labeled by a name sent from the planning module, instead of a name derived from the current episode key.

(ii) Maintaining a named mental image consists of simply executing a rule which recognizes and reconstructs its components.

(iii) Instating a named mental image. When we come to act using the stored mental image, we instate the noted image to become the current mental image and its spatial properties are then used by the planning and motor hierarchy to execute the desired motor action. This temporarily deemphasizes the percept which is continuously being refreshed from the input visual stream.

The spatial working memory model as implemented

The current system. All of the above mechanisms have been designed and implemented and the system will successfully carry out the basic spatial

working memory experiment. The system was programmed in Sicstus Prolog and the BAD language. The BAD language and manual can be found at http://www.exso.com/bad.html.

Predicted imaging files. The program allows one to write to a file all the activation values for all modules for all cycles. We developed an energy measure for the activation values used in visualization. The predicted images were produced by (i) allocating Brodmann areas to each module, (ii) using standard Talairach coordinates for the voxels in each Brodmann area, and (iii) using the energy consumption value for a given module as the value for every voxel in its corresponding Brodmann area(s). We used the afni imaging software, and could write out files in a standard format suitable for input to afni, for particular times as desired. We show afni images for the 126th cycle, in the maintain noted image phase, in Figure 4, where we have chosen samples corresponding to sagittal, axial and coronal views for each time, and for two different geometric positions, we use Talairach coordinates:

(i) sagittal 30, axial 10 and coronal -43, which is the plan module or Brodmann 10 and 46

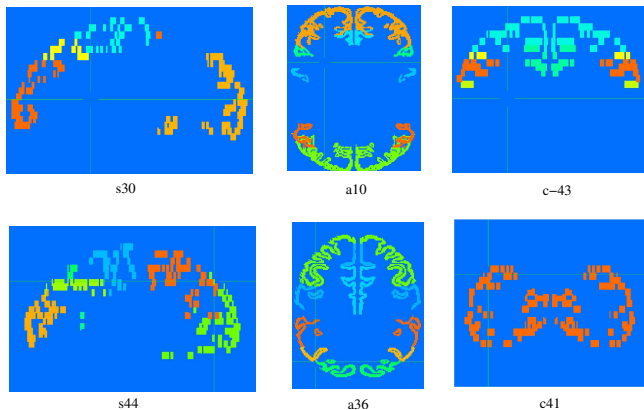(ii) sagittal 44, axial 36 and coronal 41, which is the scene module or Brodmann 40.

Figure 4: Predicted imaging files for cycle 126, in the maintain noted image phase

Results for the time course of energy consumption. Figures 5 and 6 show the time course of energy consumption by the plan module (red and uppermost), the scene module (blue and second), the context module (purple and third) and the goal module (green and lowest). The ordinate is a measure of the instantaneous energy consumption. The abscissa is time and is in discrete units which are cycles, i.e., 20 millisecond increments, the marked divisions being at 50 cycle intervals, from 0 to 500 or from 0 to 900 depending on the duration of the experiment. We can see the spurts in energy at the main transition points between phases, at cycles 10 (fixate), 111 (note), 126 (maintain) and 146 (move hand). We also see the imaging system staying active as it visually tracks the movement of the hand, after which the system falls into a rest state.

Modeling the spatial working memory deficit in schizophrenia

Systematic analysis of the effect of different lesions. We investigated in what ways the system could be compromised. By examining the dependencies of one data type on others, we could systematically determine all the effects of lesioning each component of the model. **Graded deficits due to attenuation.** We were able to obtain a graded deficit in spatial working memory performance by allowing the goal to attenuate faster in time. For normal performance the attenuation rate for goals was set to a very small amount, corresponding to extinction in 1000 cycles, whereas for the pathological case we set it to attenuate faster, corresponding to extinction in about 400 cycles. This resulted in the pathological case being able to carry out the task with a 0.5 second delay but not for a 7 second delay.

Figure 5 shows the energy curves for the 0.5 second delay case, with the normal and pathological systems behaving very similarly. Figure 6 shows the energy curves in the 7 second delay case. The graph shows the reduction of the goal and context energy, and it also shows that the move hand operation did not occur. This approach also reduces the activation level of the frontal planning area, which agrees with some experimental findings.
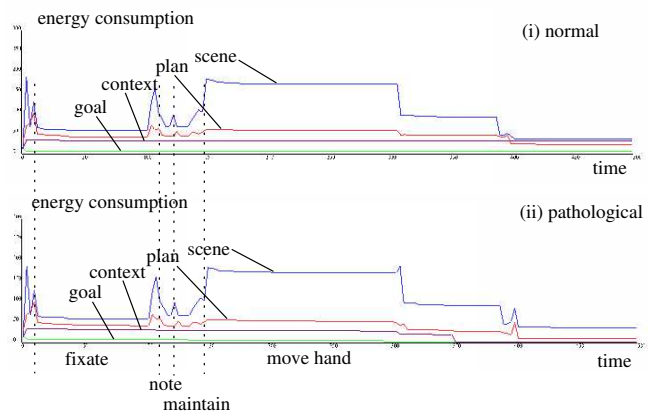

Figure 5: Time course for 0.5 second delay experiment. Time from 0 to 10 seconds, or 0 to 500 model cycles
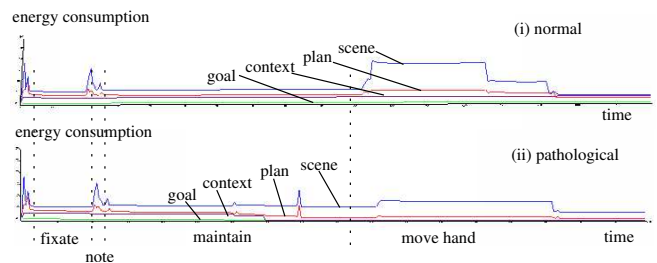

Figure 6: Time course for 7 second delay experiment. Time from 0 to 18 seconds, or 0 to 900 model cycles

Summary and conclusions

We have shown how a general system-level model of the brain can be used to model the brain mechanisms involved in carrying out spatial working memory experiments. This involved the development of mechanisms for mental imagery, for planning and for episodic memory. We also concluded, tentatively, that the most general way to obtain the deficits observed in schizophrenia is by attenuating the goal description, which then leads to the attenuation of the plan being executed.

Acknowledgments

References

[Bond, 1996] Bond, A. H. (1996). A Computational Architecture for Social Agents. In *Proceedings of Intelligent Systems: A Semiotic Perspective, An International Multidisciplinary Conference, National Institute of Standards and Technology, Gaithersburg, Maryland, USA, Oct 20-23*.

[Bond, 1999] Bond, A. H. (1999). Describing Behavioral States using a System Model of the Primate Brain. *American Journal of Primatology*, 49:315–388.

[Bond, 2002a] Bond, A. H. (2002a). Modeling social relationship: An agent architecture for voluntary mutual control. In Dautenhahn, K., Bond, A. H., Canamero, D., and Edmonds, B., editors, *Socially Intelligent Agents: Creating relationships with computers and robots*, pages 29–36. Kluwer Academic Publishers, Norwell, Massachusetts.

[Bond, 2002b] Bond, A. H. (2002b). Problem-solving behavior in a system model of the primate neocortex. *Neurocomputing*, 44-46C:735–742.

[Bond, 2004a] Bond, A. H. (2004a). A Computational Model for the Primate Neocortex based on its Functional Architecture. *Journal of Theoretical Biology*, 227:81–102.

[Bond, 2004b] Bond, A. H. (2004b). An Information-processing Analysis of the Functional Architecture of the Primate Neocortex. *Journal of Theoretical Biology*, 227:51–79.

[Bond, 2005a] Bond, A. H. (2005a). A psycholinguistically and neurolinguistically plausible system-level model of natural-language syntax processing. *Neurocomputing*, 65-66:833–841.

[Bond, 2005b] Bond, A. H. (2005b). Representing episodic memory in a system-level model of the brain. *Neurocomputing*, 65-66:261–273.

[Bond, 2006] Bond, A. H. (2006). Brain mechanisms for interleaving routine and creative action. *Neurocomputing*, 69:1348–1353.

[Braver et al., 1999] Braver, T. S., Barch, D. M., and Cohen, J. D. (1999). Control and cognition in schizophrenia: a computational model of dopamine and prefrontal function. *Biological Psychiatry*, 46:312–328.

[Conklin et al., 2005] Conklin, H. M., Curtis, C. E., Calkins, M. E., and Iacono, W. G. (2005). Working memory functioning in schizophrenia patients and their first-degree relatives: cognitive functioning shedding light on etiology. *Neuropsychologia*, 43:930–942.

[Curtis and D'Esposito, 2003] Curtis, C. E. and D'Esposito, M. (2003). Persistent activity in the prefrontal cortex during working memory. *Trends in Cognitive Sciences*, 7:415–423.

[Glahn et al., 2003] Glahn, D. C., Therman, S., marko Manninen, Huttunen, M., Kaprio, J., Lonnqvist, J., and Cannon, T. D. (2003). Spatial working memory as an endophenotype for schizophrenia. *Biological Psychiatry*, 53:624–626.

[Kosslyn, 1994] Kosslyn, S. (1994). *Image and Brain*. M.I.T. Press, Cambridge, Massachusetts.

[Kosslyn, 1981] Kosslyn, S. M. (1981). The medium and the message in mental imagery: A theory. In Block, N., editor, *Imagery*, pages 207–258. M.I.T. Press, Cambridge, Massachusetts.

[Kowalski, 1974] Kowalski, R. A. (1974). Predicate logic as a programming language. In *IFIP74*, pages 569–574.

[Rowe et al., 2002] Rowe, J., Friston, K., Frackowiak, R., and Passingham, R. (2002). Attention to action: specific modulation of corticocortical interactions in humans. *Neuroimage*, 17:988–998.

[VanEmden and Kowalski, 1976] VanEmden, M. H. and Kowalski, R. A. (1976). The Semantics of Predicate Logic as a Programming Language. *Journal of tbe Association for Computing Machinery*, 23:733–742.