UNIVERSITY OF CALIFORNIA, SAN DIEGO

A Computational Biologically-plausible Model of Working Memory for Serial Order, Repetition and Binding

A dissertation submitted in partial satisfaction of the
requirements for the degree
Doctor of Philosophy

in

Cognitive Science

by

Danke Xie

Committee in charge:

Professor Jeffrey L. Elman, Chair
Professor Garrison W. Cottrell
Professor Virginia de Sa
Professor Harold Pashler
Professor Marty I. Sereno

2009

The dissertation of Danke Xie is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

_____

_____

_____

_____

_____

Chair

University of California, San Diego

2009

DEDICATION

To my mom.

EPIGRAPH

I have not failed. I've just found
10,000 ways that won't work.
– Thomas Edison

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

ACKNOWLEDGEMENTS

Thanks to my advisor Jeff Elman for his invaluable advice and inspiration. His knowledge, profundity and insights gave crucial orientation for the research at various crossroads which were sometimes mystical. His generosity and open-mindedness have made possible the work which took a few unusual paths. If at the end it has collected fruits on oft-deemed dead roads, this would have been impossible were it not for his pioneering spirit and clairvoyance.

I will also thank faculty, colleagues, and peer graduate students at UC San Diego for inspiring interactions, and the Cognitive Science research community across and beyond the country for creating this especially intriguing arena for scientific pursuit.

VITA

2000	B. S. in Psychology *cum laude*, Peking University, Beijing, P.R.China
2008	Ph. D. in Cognitive Science, University of California, San Diego

PUBLICATIONS

Shieh, D. X., & Elman, J. L. (2006). The Divergent-Reconvergent Model of Serial Order Encoding and Retrieval. In Proceedings of the 29th Annual Meeting of the Cognitive Science Society.

Shieh, D. X., & Elman, J. L. (2007). How are Repeated Items Encoded in Working Memory? In Proceedings of the 30th Annual Meeting of the Cognitive Science Society.

Hsiao, J. H., Shieh, D. X., & Cottrell, G. W. (2007). Computational explorations of split architecture in modeling face and object recognition. In Proceedings of the 29th Annual Conference of the Cognitive Science Society.

Hsiao, J. H., Shieh, D. X., & Cottrell, G. W. (in press). Convergence of the visual field split: hemispheric modeling of face and object recognition. Journal of Cognitive Neuroscience.

ABSTRACT OF THE DISSERTATION

A Computational Biologically-plausible Model of Working Memory for Serial Order, Repetition and Binding

by

Danke Xie
Doctor of Philosophy in Cognitive Science
University of California San Diego, 2009
Professor Jeffrey L. Elman, Chair

Current theories accurately view working memory as a multi-component structure, including a phonological store, a visuospatial sketchpad and a central executive. These components have been roughly identified with known brain areas and neuropsychological functions. However, the neural mechanisms for memory encoding and retrieval remain largely controversial, especially the mechanisms for serial order. These open questions warrant courageous attempts to tackle these problems computationally; that is, constructing large-scale biologically plausible computational models to explain detailed memory processes in these components, their interactions and the required control processes.

The thesis is focused on constructing computational models to explain a few critical phenomena and mechanisms in working memory, observed in the Immediate Serial Recall task. The main constraining phenomena for the models are: the primacy effect, the recency effect, the phonological similarity effect in word lists but not in non-word lists, repetition coding by patterns, and position effects in transposition errors (e.g., an item is likely to be transposed with another item at the same position in another group). The modeling method is incremental: it first begins with a biologically plausible model for sequence encoding and retrieval based on the anatomy of the prefronto-basal ganglionic system. Phenomena that confirm this model include the primacy effect, the recency effect, similarity effects and the transposition gradient. This model is then extended with a specialized sequencing mechanism for phonological information, in order to explain the lack of phonoloigcal similarity effect in non-word lists. This Dual Rep-

resentation model, to some extent, can also mitigate the difficulty of recalling repeated items, though it does not explain repetition encoding by patterns. The pattern encoding mechanism and the required binding process are discussed for a later model, which also uses binding to account for the positional effects, where position codes are considered to be temporarily bound to items. At last, it is discussed how the working memory models can be merged with a long-term memory component to transfer sequences stored in working memory into long-term memory.

At last, limitations of the model are analyzed in the Implications and Discussion chapter. The analysis reveals a difficulty in the sequencing of items of different categories by lateral inhibition, because different categories of items are represented in different cortical areas. This result necessitates an additional subsystem that can encode category sequences, because the system needs to encode the category sequence to cue the retrieval of items. For example, a sequence like "Trees are plants" can be recalled better if it is accompanied by a category sequence "N V N". Due to the difficulty with repetitions of the basic cortico-basal ganglia sequencing mechanism, the category subsystem would work best if it is hierarchically organized, where each constituent does not contain repeated categories, or the repetition can be coded away by simple patterns. The category sequence then becomes a hierarchy, for instance, S(N VP(V N))). At this stage, the working memory theory meets linguistic theories. It implies, however, that the phrase-structure syntax should not itself be the ultimate criterion for linguistic acceptability or comprehensibility. If comprehensibility is defined as whether the brain can bind simple concepts into complex concepts following physical laws that govern synchronization, then, the brain should be able to make use of all available information from the sentence, context, long-term knowledge and the even the encoding mechanism. Therefore, an acceptable grammar can include phrase structure rules, and rules concerning the context, repetition patterns, position codes, and even syllabic counts. These implications of the working memory model deserve further exploration.

Further analysis of the model also reveals a need for a specialized subsystem to encode the timing within sequences. It suggests that the cortico-cerebellar loop can be a good candidate for encoding temporal information, in addition to the serial order information encoded by the cortico-basal ganglionic system. A cerebellum-based mechanism for absolute and relative timing is discussed. This work will hopefully help further research about various sequence processing mechanisms of the brain.

Chapter 1

Introduction to Working Memory

The earliest psychological study of memory did not distinguish what we now know as long-term memory (LTM) and short-term memory (STM). This distinction was established around the 1960's when evidence accumulated for the view that human memory is composed of two components: a transient short-term store and a more permanent long-term store rather than a unitary one. A brief history of research that had led to this understanding and the conception of working memory is reviewed below, followed by an introduction of competing theories about the theme of the thesis: the *serial-order* problem.

1.1 Early memory research

Long-term memory was the only subject of early memory research. Researchers were mainly concerned with retention over a long period of time, from a few minutes to days and years. The first scientific research of memory started in 1879, when the German psychologist Hermann Ebbinghaus (1850-1909) applied experimental methods to memory studies. At that time, the experimental method had been used to study sensation, but it was not considered suitable for addressing such "higher mental processes" as memory.

In order to put the high-level mental process to objective study, Ebbinghaus purposefully chose to ignore much complexity, and confined himself to rigorous experimental designs. In his experiments, Ebbinghaus usually used lists of nonsense consonant-vowel-consonant (CVC) syllables as the to-be-remembered material with the purpose of minimizing the effects of previous knowledge. The questions he intensively studied in-

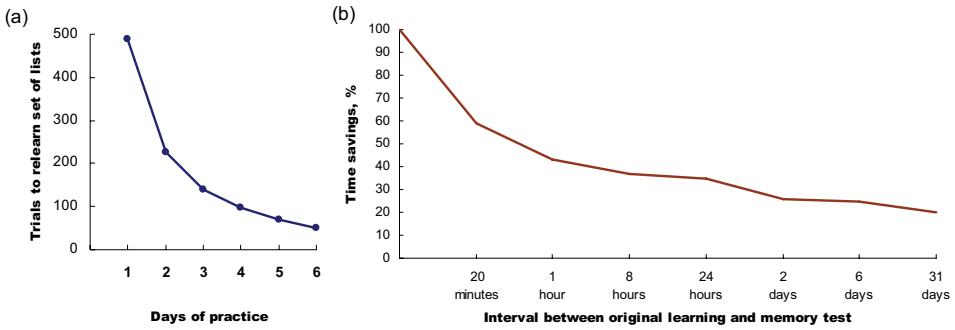


Figure 1.1: Ebbinghaus's learning curve (a) and forgetting curve (b)

cluded how rapidly information is registered in memory and how fast it is forgotten. He devised the *saving method*, which measures the amount of retention by the difficulty to perfectly relearn the material. Using this technique, Ebbinghaus (1885) obtained the well-known *learning curve* and *forgetting curve* (a.k.a. retention curves).

The learning curve (Figure 1.1a) is the number of trials needed to relearn the lists for each day. This number decreases over time, reflecting improved learning. It is noticed that the initially rapid learning rates decrease gradually, exhibiting *negative acceleration*. It was also noticed that if both axes of the plot are log-transformed, the curve becomes linear, revealing a power relationship between learning and practice. Later, it was found that almost all learning functions are close to power functions, known as the *power law of learning* (Newell & Rosenbloom, 1981). The forgetting curve (Figure 1.1b) plots the amount of retention (in percentage) against delay intervals (e.g., from 20 minutes to 31 days). This curve also show negatively acceleration; that is, initial forgetting is rapid but the forgetting rate decreases over time. It is also later established that the curve is closer to power functions than other reasonable alternatives, such as the exponential function (Rubin & Wenzel, 1996; Wixted & Ebbesen, 1991). This generalization is called the *power law of forgetting*.

1.2 Two component view of memory (LTM & STM)

The situation that early memory research mainly concerned long-term learning changed dramatically in the middle 1960's, when accumulating evidence supported a two-component or two-store view that assumes a transient short-term memory (STM) and a more permanent long-term memory (LTM).

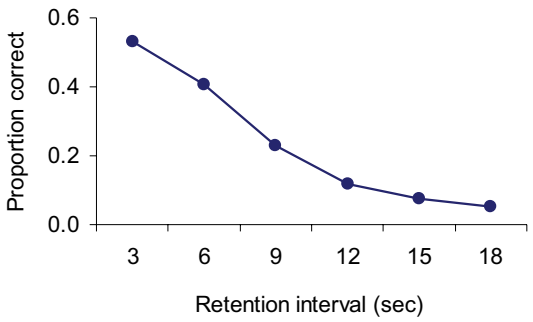


Figure 1.2: Short-term forgetting curve (Peterson & Peterson, 1959)

1.2.1 Memory duration and serial position effects

The first important piece of evidence was J. Brown (1958) and Peterson & Peterson's (1959) findings that a sequence of items well within the memory span will be forgotten within 20 seconds, if the subject is prevented from rehearsing. Peterson and Peterson (1959) showed the subjects three consonants (a CCC trigram, e.g., CQX) and asked them to retain the trigram for 3 to 18 seconds. Though not much forgetting had been expected to occur in such a short period, when subjects were distracted with backward counting by threes (e.g., 723→720→717...), the trigram presented to them was rapidly forgotten (Figure 1.2). This experiment suggested that the duration of the STM is approximately 20 seconds, and STM is distinct from the LTM.

The two components (STM and LTM) show up more clearly in the free recall task. In free recall, subjects are presented with a list of items, usually one at a time, and then are asked to recall them in any order. Typically, recall for early items (primacy effect) and recent items (recency effect) in the list are recalled better than items in the middle of the list, as shown in Figure 1.3. The serial position effects were made well-known by (e.g., Murdock, 1962), though they were first discovered by Nipher in 1878. More phenomena concerning Immediate Serial Recall (ISR) will be reviewed in Chapter 2.

1.2.2 Atkinson & Shiffrin's (1968) model

The two-component view of memory was crystallized as a mathematical and computational model by Richard Atkinson and Richard Shiffrin (1968). Atkinson and Shiffrin published a theory in 1968 that captured the then current wisdom about the

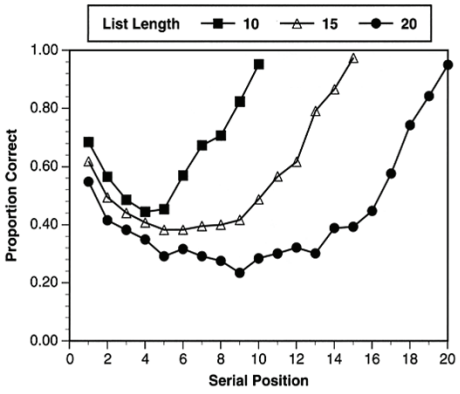


Figure 1.3: The serial-position curve in free recall (Murdock, 1962)

nature of human memory. It was the first precise theory of human memory which was expressed mathematically as well as simulated on computers.

Figure 1.4 shows the basic structure of the model. It has three components. External input first arrives at the *sensory register* through perceptual processes. Then, attended information can enter the *short-term memory*. In the short-term memory, there are several slots (often specified as 4). Each slot can store one element of information, whose duration is about 20 seconds without rehearsal. With rehearsal, information can stay as long as needed. From working memory, the stored information may transfer into long-term memory. The theory posits that when the STM is full, an old item can be displaced if the subject takes in a new item.

This model could account for the shape of the serial position curve. According to this model, the recency effect appears because the last few items are mostly likely to be maintained in the short-term store, because they are not displaced by other items. The primacy effect is explained as early words have more chances to be rehearsed and transferred into the long-term memory. The idea that more rehearsal leads to better long-term retention was indeed demonstrated by Rundus (1971).

Although few researchers still believe in the original theory to this date, its recent variations and new models such as SAM (Search of Associative Memory, Gillund & Shiffrin, 1984; Raaijmakers & Shiffrin, 1980) and REM (Shiffrin & Steyvers, 1997) remain highly influential. These models are now more focused on LTM retrieval tasks, such as recognition, free and cued recall, perceptual identification and lexical decision (Shiffrin, 2003). They are also used (though less commonly) to model short-term effects

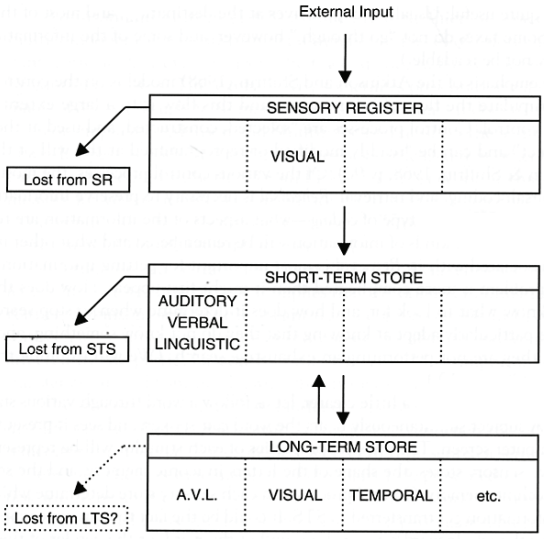


Figure 1.4: Atkinson and Shiffrin's (1968) two-store model of short-term and long-term memory

such as priming (Huber, Shiffrin, Lyle, & Ruys, 2001).

1.3 Working Memory

As the two-store model became accepted and developed, it faced new challenges. Atkinson and Shiffrin (1968, 1971) assumed that the short-term memory acted as working memory, necessary for learning, information retrieval, and the performance of various cognitive tasks. According to the theory, patients with a grossly impaired short-term memory are expected to have other cognitive problems. However, there were patients who appeared to have a normal long-term learning and surprisingly few cognitive handicaps despite short-term memory impairments. The structure of working memory appeared to require further investigation.

Baddeley and Hitch (1974) did experiments to further clarify the role of STM in various tasks, including verbal reasoning, prose comprehension and long-term learning, all of which relied heavily on working memory. The idea is the following: if STM does act as the working memory, then occupying it with a load of digits would interfere with the performance of these working memory tasks. The dual-task experiments showed the following pattern: as the concurrent digit load increased, performance declined, but the degree of disruption fell far short of predicted. Therefore, the results suggested that the

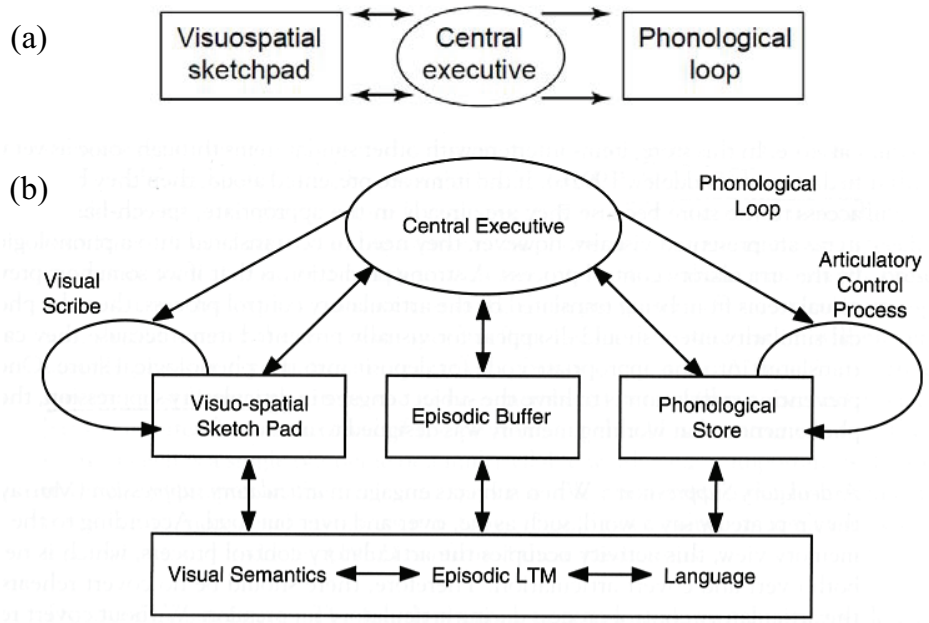


Figure 1.5: Working memory

short-term memory might not be a unitary system as assumed by Atkinson and Shiffrin's (1968) model. There seems to be a considerable amount of the working memory outside the portion utilized by the digit memory span task.

Baddeley and Hitch (1974) proposed a three-component model of working memory. It is shown in Figure 1.5a. The model consists of three components, the *phonological loop*, which holds and manipulates speech-based information, the *visuospatial sketchpad*, which performs memory and manipulation functions for visual and spatial information, and the central executive, which is a control system that coordinates the two slave systems but has no storage capacity. Recently, a fourth component has been added to the model, the *episodic buffer* (Figure 1.5b). The episodic buffer is assumed to be capable of integrating information from a variety of sources, including other slave systems and the long-term memory (Baddeley, 2000).

The phonological loop is the most intensively studied component in Baddeley's model. It lies closely to the traditional concept of short-term memory, and has been implemented in computational models that account for serial recall (Burgess & Hitch, 1999, 1992; O'Reilly & Soto, 2001). According to Baddeley, the phonological loop is assumed to have two components, a *phonological store* that can hold acoustic or speech-based information for 1 to 2 seconds, coupled with an *articulatory control* process, analogous to

inner speech. Information in the phonological store can be refreshed by the articulatory control process. The articulatory process can also convert visually presented materials, such as words or nameable pictures, into phonological forms and register them phonologically.

The phonological loop is a highly successful theoretical construct. It can account for many findings, such as the *acoustic similarity effect* – worse recall of acoustically similar items (Conrad, 1964), the irrelevant speech effect – worse recall of visually presented items at the presence of irrelevant speech (Colle & Welsh, 1976; Salame & Baddeley, 1990, 1982), the word-length effect – longer words are recalled worse (Baddeley, Thomson, & Buchanan, 1975), and articulation suppression – suppression of subvocal rehearsal removes the word-length effect (Baddeley, Lewis, & Vallar, 1984). In-depth understanding of these effects would require knowledge about the representations and mechanisms of serial order at various levels of representation (e.g., lexical and phonological levels). These questions will be addressed by computational models of the phonological loop in this research.

1.4 The serial-order problem

In his famous 1951 paper "The Problem of Serial Order in Behavior", Karl Lashley stressed the importance of the *serial order problem*. The serial order problem refers to how sequences of behaviors are produced, and how the serial order of items is stored and retrieved in a memory system. The serial-order problem has been considered fundamental to understanding the brain and cognition. Lashley proposed that serial-order production involves parallel activation of a set of actions which themselves do not contain the serial order; rather, superimposed on them is an independent ordering mechanism that selects which response, among those activated, to produce at each time. According to Lashley, the serial order is the outcome of the selective mechanism. However, this problem couldn't be fully answered at that time. Lashley wrote *"The real problem, however, is the nature of the selective mechanism by which the particular acts are picked out in this scanning process and to this problem I have no answer"*.

Soon after the publication of this 1951 article, the "cognitive revolution" got underway, giving rise to theoretical linguistics (Chomsky, 1957) and the computer metaphor of the mind. In these subjects, the serial-order problem is solved technically. Theoretical linguistics, which only concerns itself with the competence of serial order but not

with its performance, avails itself of such formal objects as ordered sets, strings etc. as primitives, from which to build descriptions of grammars and other abstract objects (Houghton & Hartley, 1996). In artificial intelligence and computer science, ordered data can be sufficiently stored in serially addressed memory slots or data structures (Knuth, 1997). In such contexts, serial order *per se* does not appear to be of any problem at all. The availability and power of those mathematical or computational entities sufficed to obscure the fact that we have no neuropsychologically grounded theory of serial order for the brain.

Fortunately, with the development of cognitive science, neurophysiology and connectionist modeling the serial-order problem has once again been taken seriously. In recent 20 years, many models and theories have been proposed (Page & Norris, 1998; R. N. A. Henson, 1998; Burgess & Hitch, 1999, 1992, 2005; O'Reilly & Soto, 2001; Melamed, Gerstner, Maass, Tsodyks, & Markram, 2004; Farrell & Lewandowsky, 2002). The following summarizes a few basic ideas in these theories.

1.4.1 Chaining theory

The chaining theory assumes that serial order is stored by a 'chain' of associations between successive items. The items are retrieved by stepping along these associations in a process called chaining. Each item triggers the activation of its successor via associative links.

This theory is probably the oldest approach to serial order. Its simplest form assumes only pair-wise associations between adjacent elements of a sequence. Figure 1.4.1a shows the representation of the sequence V-E-R-Y. This model has difficulty producing sequences with repeated items (e.g., E-V-E-R-Y), because some items (e.g., E) may be followed by two or more different successors (V and R), making the chain ambiguous. To solve this problem, Wickelgren (1969) suggested the use of context-sensitive tokens, with which the sequence E-V-E-R-Y can be represented without repeated items (Figure 1.4.1b). However, the use of tokens fails to capture any relations between different instances of the same type. Another serious drawback of the simple chaining model is that it cannot recover from errors. A single error in the middle of the sequence can break the chain and fail the recall altogether. This problem is not common in humans.

A solution to the repetition and error-recovery problems is to allow multiple items to cue the next one. This solution has been implemented with connectionist

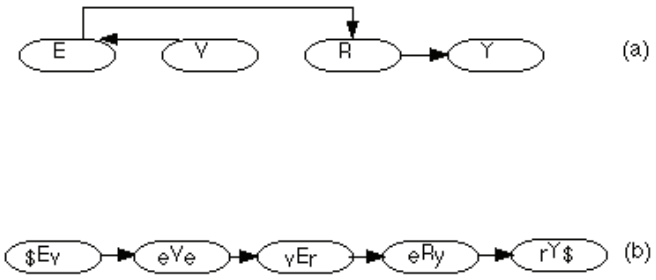


Figure 1.6: The basic chaining model (a) and Wickelgren's context-sensitive chain (b) that solves the problem of repeated items

models that are trained to learn sequences (e.g., Cleeremans, 1993; Jordan, 1986). Neural network architectures that are suitable for this purpose include the Simple Recurrent Network (SRN, Elman, 1990) and Jordan's network (Jordan, 1986). In these models, each item is cued by a compound representation of all previous items. This history representation allows repeated items to be disambiguated, and allows the network to recover from single errors because the history can keep the recall process on track.

These neural networks mainly concern about how the sequences are learned into and reproduced from the long-term memory. They are not sequential working memory models *per se*. To study sequential working memory with neural networks, we would need to look at neural network models of working memory (e.g., Botvinick & Plaut, 2006; O'Reilly & Soto, 2001). These models use sustained activities (with or without chains) to represent contents in working memory. The models to be studied in this thesis belong to this category. They share some aspects as Botvinick and Plaut and O'Reilly and Soto's models, but have important differences, and new mechanisms motivated by brain anatomy.

1.4.2 Positional theory

The positional theory assumes that each item is coded for its position in the sequence, and the list is recalled from position cues. The most complete computational models of positional theory are Burgess and Hitch's (1999) Phonological Loop model and Henson's (1998) Start-End model.

The main mechanism in these models is motivated by the evidence that positional coding can be observed in sequential working memory tasks. For example, when the to-be-remembered sequence is subgrouped (e.g., 732-908), it is found that items at

the same within-group positions are more likely to be transposed with each other than with items at other positions. For example, in the above sequence 7$\underline{3}$2-9$\underline{0}$8, the digits "3" and "0" are more likely to be transposed with each other than the pair of digits "3" and "9". This phenomenon suggests that items are coded for their positions. However, there are still challenging problems facing the positional theory. They include 1) what is the form and nature of the positional code, 2) how is it neurophysiologically generated? and 3) how is a positional code associated with an item (neural plasticity vs. dynamic binding vs. spatial coding)? It is also desirable to clarify how positional theory can be integrated with other serial-order mechanisms. These problems will be discussed in Chapter 5.

1.4.3 Ordinal theory

The ordinal theory assumes that serial order is implicitly represented by the relative magnitude of some continuous variable. For example, Grossberg (1978) proposed that order is represented by the activation strength of items in memory, with the first item 'strongest' and the last item 'weakest'. The order of items can be retrieved by iteratively selecting the strongest item and suppressing it. This mechanism is also known as Competitive Queuing Houghton (1990).

Competitive queuing is an appealing mechanism, which has been shown to reproduce a large scope of human data, especially memory error patterns (Page & Norris, 1998). It has also been considered to be a convergence of many working memory models in the recent 20 years (Farrell & Lewandowsky, 2002). What is more encouraging is that the activation gradient in this theory has been observed in the prefrontal cortex during sequencing tasks (Bullock, 2004). However, difficulties facing this theory include 1) the difficulty to represent and retrieve repeated items, and 2) the difficulty in specifying the exact neural mechanism for this model.

This thesis proposes a novel neural mechanisms for the ordinal theory (Chapter 4) and a group of mutually complementary solutions to the concomitant repetition problem, and discusses the relations between this novel neural mechanism and other serial order mechanisms (Chapter 5).

1.5 Conclusion

Current theories accurately view working memory as a multi-component structure, including a phonological store, a visuospatial sketchpad and a central executive. These components have been roughly identified with known brain areas and neuropsychological functions. However, the neural mechanisms for memory encoding and retrieval remain largely controversial, especially the mechanisms for serial order. These open questions warrant courageous attempts to tackle the problems computationally; that is, by constructing large-scale biologically plausible computational models to explain detailed memory processes in these components, their interactions and the required control processes. These problems will be addressed in this thesis.

Chapter 2

Experimental Background

2.1 Working memory characteristics

Major theoretical developments in working memory are always motivated by ingenious and informative experiments. This chapter summarizes benchmark phenomena about working memory in general, and the phonological loop in particular. Many of these phenomena will be used to motivate and evaluate the models developed in the following chapters. Others of them are provided here to provide a background for theoretical discussions throughout the thesis.

2.1.1 Effects on the phonological loop

Capacity of phonological loop

Baddeley proposed that the phonological loop is capable of storing about 2 seconds worth of information (Baddeley et al., 1975). It was supported by the experiment that measured subject's recall span of words of a certain number of syllables (e.g., 3.5 words for 3-syllable words), and the reading rate of these words (e.g., 1.8 words/sec). The phonological loop capacity is estimated as

$$\begin{aligned} \text{phon. loop capacity} &= \frac{\text{recall span}}{\text{reading rate}} \\ &= \frac{3.5 \text{ words}}{1.8 \text{ words/sec}} \approx 2 \text{ sec.} \end{aligned}$$

The capacity is limited by how far back the loop can remember hearing a word. In immediate serial recall tasks, the functioning of the loop is most robustly observed with

short lists (within 6 items). In long-lists, the loop may only apply to the recency given its limited capacity (Jones, Hughes, & Macken, 2007).

Word length

The word length effect is a robust phenomenon that serial recall performance declines as the word length increases (Baddeley et al., 1975). This effect supports the conceptualization that information in working memory is dynamically refreshed by subvocal rehearsal, and the articulation speed limits the rate at which the information can be refreshed.

Articulation suppression & word length

It is well known that 'articulatory suppression' by the articulation of irrelevant items during list presentation, impairs recall (D. J. Murray, 1968), removes the word length effect of visual lists (Baddeley et al., 1975). It also removes the word length effect for auditory lists when suppression is continued throughout recall in addition to list presentation (Baddeley et al., 1984). These effects support the views that phonological coding is used for working memory, and articulatory suppression can prevent subvocal rehearsal of the phonological information and prevent auditory recoding of visual information.

Phonemic similarity

Phonemic similarity is the robust effect that lists of phonemically similar items are recalled worse than lists of phonemically dissimilar items (Conrad & Hull, 1964). The effect is also known as the Phonological Similarity Effects (PSE). Interestingly, recent studies found that the effect can be removed or reversed when the to-be-recalled lists are nonwords instead of words (if the nonwords have low associative values for long-term memory) (Lian, Karlsen, & Winsvold, 2001; Lian, Karlsen, & Eriksen, 2004). This phenomenon supports spatial coding of pure phonological information, and motivates the phonological store model in Chapter 5.

Articulatory suppression & phonemic similarity

Articulatory suppression during presentation and recall removes the phonemic similarity effect for visual but not auditory lists (Baddeley et al., 1984). This phenomenon

is again consistent with the assumption that rehearsal is necessary for visual stimuli to access the phonological store, whereas auditory stimuli access it directly.

2.1.2 Serial-order effects

Serial position curve

The curve of correct item recall against the serial position (the 'serial position curve') has a bowed shape exhibiting both primacy and recency (Murdock, 1962). It is known that the recency effect diminishes as the decay between presentation and recall increases (Glanzer & Cunitz, 1966; Postman & Phillips, 1965). This can be explained by fast decay of sensory traces which encode recent items, such as the echoic memory or the precategorical acoustic store (PAS, Crowder & Morton, 1969).

Presentation modality, suffixes and phonemic similarity

Auditory presentation of stimuli increases the recency effect (Crowder, 1972; Penney & Blackwood, 1989). An auditory suffix removes this effect (Crowder & Morton, 1969), but visual (graphemic) suffix has little effect on lists presented in either modality (Hitch, 1975; Morton & Holloway, 1970). This can also be explained by assuming that recency effect is mediated by echoic memory or PAS.

Temporal grouping

"Temporal grouping" by the insertion of pauses during presentation typically improves performance by reducing order errors (Ryan, 1969). Such grouping leads to 'scalloped' serial position curves (showing primacy and recency within each group), and it has a greater effect for auditory lists than visual lists (Frankish, 1985). The greatest effect is observed when lists are presented in subgroups of three items (Ryan, 1969). Temporal grouping is not directly modeled in this thesis, but it is seen as resetting positional coding as discussed in Chapter 5.

Errors

The majority of errors in immediate serial recall of familiar items are 'order errors' rather than 'item errors' (Aaronson, 1968; Bjork & Healy, 1974). These order errors tend to involve transpositions of neighboring items (Healy, 1974), and between

phonemically similar items (Conrad, 1964). The locality of transposition errors is modeled in Chapter 4. Transposition of phonemically similar items is considered to be an effect of phonological cuing of item representations, as discussed in Chapter 5.

Serial order intrusions and position-based transpositions

"Serial-order intrusions" occur, in which an item from a previous list is recalled as an error at the same position in the current list (Conrad, 1960). A similar type of error is seen in sub-grouped sequences, where items at the same position of the subgroups are swapped (e.g., 2$\underline{4}$9-8$\underline{0}$3 → 2$\underline{0}$3-8$\underline{4}$9). This type of error has been used to argue for the associations between positional code and items which are required by the positional theory of serial order (R. N. A. Henson, 1998, 2001). These types of error are discussed in Chapter 5 in the discussion of positional coding.

Chapter 3

The Phase Neural Network

Computational modeling of neural mechanisms and human behaviors requires a simplification of the target to be modeled. Different views on what is important for neural processing have generated neuron models and neural network models with different levels of details, and different interfaces with other neurons or networks. Most neuron models represent either the firing rate of the neuron, or the timing of spikes. With the recognition that neural synchronization is probably as important as neural activities in terms of firing rates or spikes, this chapter introduces a new type of neuron model, which represents two types of information to interact with other neurons: a firing rate and a phase. The goal is to build a computational element by which information about both firing rates and neural synchronization can be efficiently and accurately received, processed, and output. Paradoxically, this new type of artificial neurons more accurately represent cortical information processing than individual firing-rate neurons or spiking neurons. In this chapter, this new artificial neuron model is first introduced, with a discussion of its biological relevance. Next, a framework is designed for using this new type of artificial neurons to model high-level cognition, termed the Phase Neural Network (PNN). It entertains common neural organizations used in artificial neural networks, such as layers and projections, but also introduces a few less common structures, such as controllable gates, monitoring signals, and adaptive actions, into the framework. These features greatly facilitate the modeling of complicated neural processes, especially high-level cognitive processes in which working memory, attention, cognitive control are involved. The PNN framework will be used to implement all models in this thesis.

In what follows, I will first briefly review the basic form of firing-rate neurons

and spiking neurons, discuss the problems of binding and neural synchronization (Gray, 1999; Singer, 1999; Shastri, 1999; Malsburg, 1981, 1995), and introduce the new neuron model and the PNN framework.

3.1 Background

3.1.1 Overview of firing-rate neurons

Neurons in the brain receive input from a large number of other neurons, integrate the information, and generate a response. Using firing-rate neurons, the basic form of cortical information processing can be well-captured. When combined with learning algorithms which adjust the weights of interconnections between neurons, neural networks can be a very powerful tool for understanding neural information processing, including perception, learning, memory and development of cognitive functions, and can applied to solving real-world learning and data analysis problems.

The following describes a generic form of firing-rate neurons that has been used in various neural network models (McCulloch & Pitts, 1943; Rosenblatt, 1958; Rumelhart & McClelland, 1986).

1. **Firing-rate representation**. The activity of an ith neuron is represented by its firing rate a_i. The firing rate is typically a continuous value, representing how strongly a neuron is active; or a binary value representing an all-or-none response.

2. **Net input**. The net input x_i is the total input from all afferent (incoming) neural connections. It can be generally represented by a function $f_x(\mathbf{x}, \mathbf{w}_i)$ with two variables: $\mathbf{x} = (x_1, x_2, \ldots, x_n)^T$ is the vector of the activities of all input neurons, and $\mathbf{w}_i = (w_{i1}, w_{i2}, \ldots, w_{in})^T$ are their respective connection weights. Typically, a neuron also has a bias θ_i, which is the "default" input to the neuron, when there is no external input. The negative bias models the activation threshold of the neuron, and a positive bias can correspond to a baseline activity. Formally, the bias θ_i can be in incorporated in the weight vector as a weight w_{i0} paired with a constant unit input $x_0 = 1$.

3. **Output**. The output a_i is defined by a *transfer function* of the net input: $a_i = f(x_i)$.

Typically, the net input function f_x computes the weighted sum of the input activations (3.1), though other net input functions can also be used (e.g., a Radial basis function, Yee & Haykin, 2001).

$$\begin{aligned} f_x(\mathbf{x}, \mathbf{w}_i) &= \mathbf{x}^T \mathbf{w}_i \\ &= \sum_j x_j w_{ij}. \end{aligned} \tag{3.1}$$

The transfer function $f(x)$ models the relation between net input and the output of the neuron. The relation can be either linear or nonlinear. For example, they can model all-or-none firing (3.2)

$$f(x;\theta) = \begin{cases} 0, & x < \theta \\ 1, & x \geq \theta, \end{cases} \tag{3.2}$$

(thresholded) linear output (3.3)

$$f(x;\theta) = \begin{cases} 0, & x < \theta \\ x - \theta, & x \geq \theta, \end{cases} \tag{3.3}$$

and sigmoidal nonlinear output (3.4)

$$f(x) = \frac{1}{1 + e^{-x}}. \tag{3.4}$$

These neurons can flexibly represent different types of information integration as well as different output properties of the neuron. The major limitation is that they do not represent the spike timing information, and may be limited to tasks where spike timing is not crucial. This may be generally true for cortical processing of one type of information, though parallel processing of multiple types of information may require additional neural mechanisms than the firing-rates to enable a binding process (discussed later). The advantages of firing-rate neurons include efficient information representation and processing, flexible network architectures (single- and various multiple-layer models), and strong ability to learn from experiences. It is also "friendly" to mathematical analysis based on its continuous output-input functions, and it facilitates in-depth understanding of both the behavior of the computational model, and neural processes in the brain. Further, recurrent neural networks can process highly complicated information, such

as sequential prediction, language learning, and general dynamic processes of neural processing (Elman, 1990; Rodriguez & Elman, 1999). For a more complete review of neural networks, the reader can refer to the books (Elman et al., 1996; Rumelhart & McClelland, 1986; Haykin, 1998).

3.1.2 Overview of spiking neurons

Compared to firing-rate neurons, there is no single generic form for modeling spiking neurons. They vary greatly in terms of 1) how the model describes spiking dynamics of each neuron, 2) how the neurons are connected, and 3) how biologically detailed the neurons are. In terms of the spiking dynamics, spiking models can be roughly categorized as activity-based models which describe the membrane potentials, and phase models which abstractly describe the periodic behavior of neurons. Examples of activity-based models include the Hodgkin-Huxley model (Hodgkin & Huxley, 1952), the integrate-and-fire model (e.g., Bressloff & Coombes, 2000), the spike-response model (Gerstner & Kistler, 2002), and efficient canonical models (Izhikevich, 2003, 2000, 2004). Phase models are also called oscillators or rotators. They are widely used to study periodic oscillations and synchronization in neuroscience, physics, chemistry, biology and applied mathematics (Strogatz & Mirollo, 1988; Kanamaru & Sekine, 2003; Nishikawa & Hoppensteadt, 2003; Borgers & Kopell, 2003). In terms of the way neurons are connected, there are pulse-coupled neurons which explicitly generate action potentials or spikes to influence other neurons (Mirollo & Strogatz, 1990; Goel & Ermentrout, 2002), and phase-coupled models which directly influence other neurons' phases (Strogatz & Mirollo, 1988; Kanamaru & Sekine, 2003).

In this section, I will only briefly discuss two types of spiking or oscillatory neurons that are closely related to the current work. For more detailed analysis of spiking neuron models, the reader can refer to the references (Gerstner & Kistler, 2002; Izhikevich, 2000).

3.1.2.1 Integrate-and-fire models

The integrate-and-fire model can be seen as a representative example of the large class of activation-based spiking models. It is one of the most widely used spiking neuronal models. Because of its simple form, it is ideal for analytically deriving theoretical results (e.g., Song, Miller, & Abbott, 2000), as it will be used to derive part of the

results in the proposed neural model.

The membrane potential $u(t)$ of the integrate-and-fire model neuron is described by

$$\tau_m \frac{du}{dt} = -\gamma u(t) + I(t) \tag{3.5}$$

where τ_m is the time scale of the membrane potential of the neuron, γ is the decay rate, and $I(t)$ is the input current. The input current can include external inputs, as well as spikes generated by other neurons. Because neurons communicate by spikes, this is a pulse-coupled neural network.

In integrate-and-fire models, the *form* of an action potential is not described explicitly. Rather, only the *timing* of an action potential is defined as the time when the membrane potential reaches a threshold ϑ. At the same time, the membrane potential is reset to the resting potential u_r. Formally, the firing time $t^{(f)}$ is defined as

$$t^{(f)} : u(t^{(f)}) = \vartheta. \tag{3.6}$$

Immediately after $t^{(f)}$, the potential is reset to the resting potential $u_r < \vartheta$,

$$\lim_{t \to t^{(f)}, t > t^{(f)}} u(t) = u_r. \tag{3.7}$$

In computer simulations, the reset of the membrane potential can be written as

$$\text{if } u \geq \vartheta, \text{ then } u \leftarrow u_r. \tag{3.8}$$

The integrate-and-fire model can be all-of-none, deciding whether an input is strong enough to induce a spike, and it can generate regular spike trains with constant input. But it should be clear from its regular behavior that it cannot generate complicated firing patterns, such as bursts. This type of spiking behavior can be generated by biologically detailed models, such as the Hodgkin-Huxley model (Hodgkin & Huxley, 1952), and its efficient canonical models (Izhikevich, 2000, 2004).

3.1.2.2 Phase models

Phase models are an abstraction of the time-dependent neural activity by assuming that the oscillatory behavior of a neuron, or a local neural circuit can be entirely described by a single phase variable (see Figure 3.1). The connection between neurons

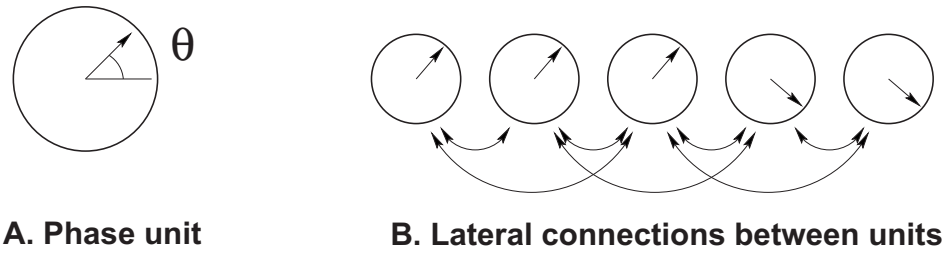


Figure 3.1: Phase Models. A) The oscillatory state of a neuron or a local neural circuit is completed described by a single phase variable θ. B) The binding of features is represented by groups of oscillators separated according to their phase values. Adapted from Wersing (2000).

is typically defined in terms of the phase variables, for example: (however, some phase models also pulse-coupled, e.g., Borgers & Kopell, 2003)

$$\dot{\theta}_i = \omega_i + \varepsilon_i + \sum_j w_{ij} \sin(\theta_j - \theta_i), \tag{3.9}$$

where θ_i denotes the phase of a neuron i, ω_i is the constant natural oscillation speed, ε_i provides a noise term to desynchronize different groups, and w_{ij} is the connection weight from the neuron j to neuron i. The synchronization of two oscillators is represented by a sufficiently small difference in their phase values. To get an intuition of this equation, it can be seen as encouraging synchronization of two neurons if their connection weight is positive, and encouraging desynchronization if the weight is negative. This is due to the phase-coupling term $w_{ij} \sin(\theta_j - \theta_i)$. If an oscillator j is ahead of oscillator i, that is, $\sin(\theta_j - \theta_i) > 0$, and the weight $w_{ij} > 0$, then the phase change term $w_{ij} \sin(\theta_j - \theta_i) > 0$ makes θ_i rotate faster to catch up with θ_j. Similarly, it can be seen that when the weight $w_{ij} < 0$, the same dynamics desynchronizes the oscillators (i.e., it makes them oscillate with different phases).

3.1.2.3 Case study: controlled synchronization

The synchronization pattern discussed above is fairly general for oscillators. That is, positively coupled oscillators tend to synchronize, and negatively coupled oscillators tend to desynchronize. This is true of both pulse-coupled neurons (Mirollo & Strogatz, 1990) and phase-coupled neurons (Strogatz & Mirollo, 1988). This process can occur automatically from the neural dynamics, Eqs. (3.5)-(3.9). For the brain, this automatic synchronization process corresponds to bottom-up processes, such as auto-

matic perceptual grouping or feature binding (Gray, 1999), which can rely on connections between feature neurons without top-down influences. However, the brain also has top-down selective attention which can actively bind or synchronize neurons that are otherwise desynchronized. Can these models accommodate the controlled synchronization process?

The answer is probably that the slower pulse-coupled models can, but phase-coupled models suffer a "squeezing" effect, explained below. Suppose three oscillators have the following ordering in the phases:

$$\theta_1 < \theta_2 < \theta_3.$$

They are currently desynchronized. Now, a top-down process (i.e., selective attention) selectively binds the features 1 and 3, represented by the phases θ_1 and θ_3. This binding process will gradually decrease the differences of θ_1 and θ_3, making θ_1 shift towards θ_3, and θ_3 shift towards θ_1. The problem is, neither θ_1 nor θ_3 can cross θ_2, but can only push it towards the other. At the end, not only are θ_1 and θ_3 bound, but all three of them. This side-effect is not neurophysiologically realistic, because neurons are coupled by action potentials. The phase-coupling effect of neuron is *not* continuous but intermittent, allowing a phase to easily cross another phase without causing the "squeezing" effect.

Because controlled synchronization needs to be accommodated in a general neural processing model unlike more passive physical or chemical counterparts[1], the use of the stable and efficient oscillators might be an unsafe choice. It appears that we would be forced to use pulse-coupled neurons for modeling high-level processing including attentional and cognitive control. Solving this problem with a more efficient neuron model would be a goal for this chapter.

3.1.3 The binding problem and synchronization

Sensory, cognitive and motor processes result from parallel interactions among large populations of neurons distributed among multiple cortical and subcortical structures. Binding, in the general sense, is the process responsible for functionally linking this distributed activity (Gray, 1999; Singer, 1999). In the visual system, the problem corresponds to identifying the relationships among features in an image so that objects

[1]Interested reader can refer to Chapter 6 Section 6.5 for a discussion of the relation between passive physical interactions and active consciousness control.

can be recognized. For example, when a red rectangle and a blue triangle are simultaneously present in the visual scene, neural activities representing all the four features "red", "blue", "rectangle" and "triangle" are active. For correct perception, the relationships among them must be unambiguously represented. How do neurons represent the relations between features? An intriguing theory is that the binding relationships are represented by temporal correlation of neural activities. That is, the neurons representing features of the same object have correlated activities, but neurons representing different objects have uncorrelated activities. This theory is known as the temporal correlation hypothesis of binding, or the synchrony theory of binding (Malsburg, 1981, 1995; Gray, 1999; Singer, 1999; Shastri, 1999). Figure 3.2 shows an example of the use of neural synchrony to represent binding of some arbitrary features A and B. The binding is represented by the synchronization of the neural firings of the features A and B, which are out-of-phase with other features that are not bound with them.

The synchrony theory of binding has several attractive properties. Compared to convergent coding of an object which means each object is represented by a small collection of neurons (by single neurons in the extreme), this theory can avoid the combination problem. That is, it does not need to assume that neurons which can detect all combinations of features always exist. When a combination is new, it can be represented by temporally correlated neural activities rather than by pre-wired connections. The correlated activities on the one hand, provide a short-term representation of binding, and on the other hand, can facilitate long-term learning of this pattern for future recognition of the object. Although the combination problem can be mitigated by using population coding, that is, using an activity pattern rather than a single neuron to represent an object, population coding has a well-known problem of superposition. When multiple patterns are present, it is not possible to distinguish one pattern from another (Malsburg, 1981, 1985).

3.2 The Phase Neurons

The major goal in this chapter is to develop a new type of artificial neuron model, which can be efficiently used to model binding by neural synchrony. Because firing-rate neurons do not provide temporal information of neural activities, computational models which study binding usually used spiking neurons or oscillators. As seen above, spiking neurons generate detailed spike trains or represent instantaneous spiking

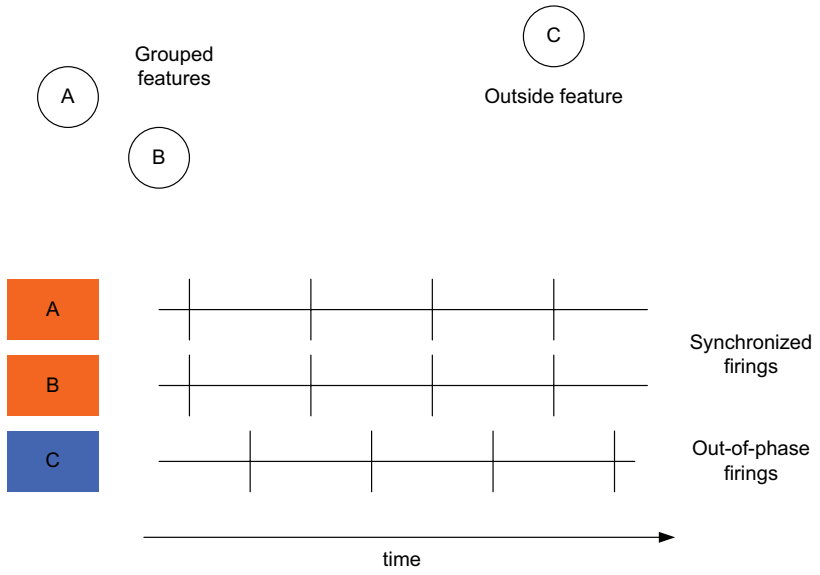


Figure 3.2: An example of binding by synchrony

dynamics, so they usually require longer time to simulate. However, a special property of binding by synchrony is that what is most important is not the specific spike times, but whether neurons are synchronized or not. With only this requirement, it is possible to develop an efficient neuron model which has the efficiency of firing-rate neurons, but can accurately describe neural synchronization and desynchronization.

In this section, I will first describe the mathematical form of the model. The goal is simple: develop a generalized firing-rate model, which can be reduced to the generic firing-rate neurons (as summarized in Section 3.1.1), but can also be extended to modeling several benchmark behaviors of spiking neurons and oscillators, that is:

1. Synchronization between excitatorily connected neurons.
2. Desynchronization between inhibitorily connected neurons.
3. Formation of synchronization groups in a network with both types of connections.

3.2.1 Representation of a computational unit

As analyzed above, phase-coupled oscillators cannot support controlled synchronization, and pulse-coupled networks only provide intermittent representations of bindings when the neurons fire. These properties are seen as disadvantages of current spiking models and oscillators for modeling the binding process. The design of the new

neural model aims at a simplistic computational unit, which provides only sufficient information to represent the binding of features, but no details of the oscillatory phase or membrane potentials. Based on this rationale, each computational unit is taken to model a cortical column, rather than a spiking neuron, because synchronized oscillation is a property of neural populations. Under synchronized population oscillation, individual neurons are often found to spike irregularly (Miikkulainen, Bednar, Choe, & Sirosh, 2005).

To efficiently test for synchronization, the easiest way is to represent the status of the neuron by an integer phase, such that synchronized units have the same phase while desynchronized units have different phases. This approach is taken in this work with biological justification. The following describes the computational unit used in this work.

Each computational unit in the system models a neural population which have oscillatory activities. Two quantities are used to describe the status of the unit: a firing rate a_i measuring how active the neurons are, and a phase-label θ used to describe its synchronization with other units. In this work, a higher-level abstraction of the oscillatory behavior of neural population is desired. The possible values of the phase are integers

$$\theta_i \in \{1, \ldots, \theta_m\}. \tag{3.10}$$

This abstraction representation of phase does not oscillate by itself, though it assumes that the modeled neural population has an oscillatory phase. The phase can be seen as a vehicle for testing whether units are synchronized. Such a test of synchronization can be simply made as whether $\theta_i = \theta_j$. The goal of constructing such a neural model, each represented by a pair (a_i, θ_i), is to find a systematic way of calculating the states of all units in the system, including the firing rate a_i and the phase-label θ_i. Because oscillatory phases are not used in this work, the phase-label is referred to as the phase, for short.

The biological meaning of equal phases in this network, $\theta_i = \theta_j$, can be seen as the neural populations i and j oscillate with approximately the same phases. Conceptually, one can imagine a phase equality tolerance τ_θ, which is used to judge whether two neural populations can be represented by the same phase label. If the spiking times t_i and t_j of two neurons differ by at most τ_θ, they are considered to have the same

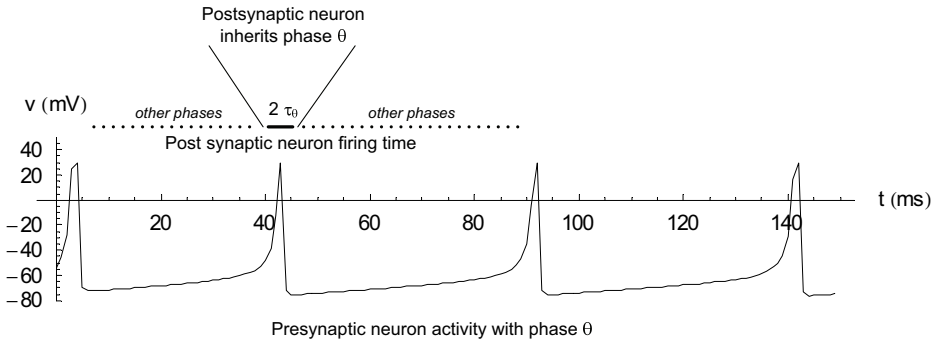


Figure 3.3: Imaginary phase equality threshold τ_θ decides equality of pre- and post-synaptic phases

phase $\theta_i = \theta_j$, otherwise they are considered to have different phases $\theta_i \neq \theta_j$. In this chapter, the activity and phase propagation rules will comply to this interpretation. For example, Figure 3.3 shows if the post-synaptic spike falls in the time window $2\tau_\theta$ around the pre-synaptic spike, it should inherit the pre-synaptic phase θ; otherwise, it should be assigned a different phase $\theta' \neq \theta$.

A seeming contradiction in representing a computational unit by a firing rate and a phase is that, for individual neurons, the two quantities cannot vary independently, and the number of phase oscillation cycles is the firing rate. However, the difference here is that the computational units are neural populations or minicolumns containing about 100 excitatory and inhibitory neurons (Mountcastle, 1957). This type of neural population has two levels of oscillatory behaviors: the spiking behavior of neurons, and the overall oscillation of the population, and they are not always correlated.

The difference can be clearly demonstrated with a computational simulation of a population of interconnected excitatory and inhibitory neurons (Figure 3.4). The plot in Panel A shows spikes of individual neurons, which are organized as a population oscillation, with alternating periods of dense and sparse spikes. The frequency of the population oscillation can be determined from the highest peak in the power spectrum (Panel B). As the input to the neural population is increased, it can be seen that the firing rate of the population (Panel C) increases linearly, while the population oscillation saturates in a small range (Panel D). This phenomenon is exploited in this model. Units are assumed to correspond to neural populations oscillating with the same frequency (so that they can be synchronized), while their firing rates can change freely.

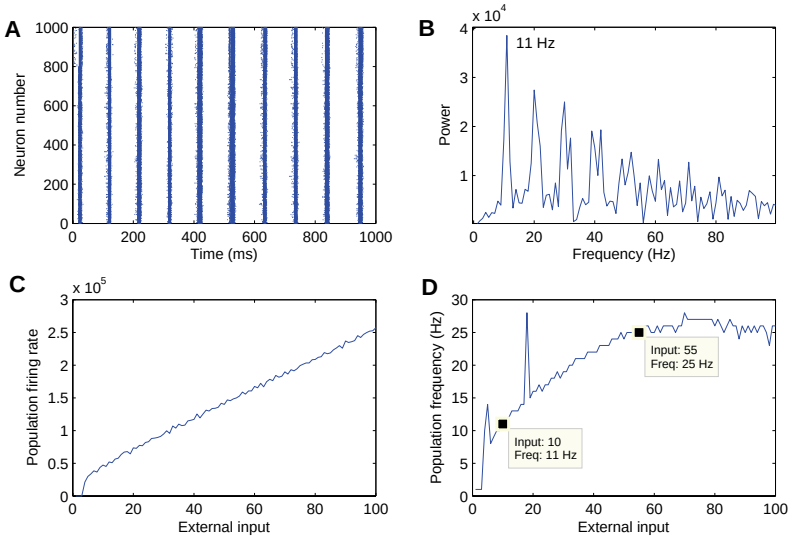


Figure 3.4: Population firing rate and oscillation frequency

3.2.2 Phase operations

An important constraint in the model is that the phase values for any neurons θ_i and θ_j can only be tested for equality. The difference between two phases $\theta_i - \theta_j$ is not meaningful and should not be assessed. For instance, if $\theta_i = 2$ and $\theta_j = 5$, all we know is that θ_i and θ_j are different and the neurons are desynchronized. The phase difference between the two neurons is not necessarily greater than another pair of neurons with $\theta_i = 3$ and $\theta_j = 4$, since the values should be construed as labels. If exact phase differences or spike timing differences are crucial for an application, more detailed spiking neuron models would be desired. By confining the use of phases to only the equality operator, this model runs as efficiently as typical firing-rate neurons, while supporting synchronization, binding and a basic form of STDP without increasing any computational complexity.

3.2.3 Activation propagation

The manner in which neural activation propagates in this neural network is similar to other firing-rate neural networks. The net input of each neuron is first computed by summing the weighted activities from input neurons. Then, an activation level is computed from the net input. In fact, exactly the same activation dynamics can be used

for this type of neurons. However, what is special here is that the input activities may come from units with different phases. It is known that some neurons are highly sensitive to the timing of its inputs, and only respond when the inputs are highly correlated, that is, with the same phase. This type of neurons are known as coincidence detectors (Azouz & Gray, 2000). With the representation of input phases, this neuron is ready to capture the "coincidence detection" property. It is assumed that input with different phases have different effects on the target neuron. The major phase which drives the phase of the target neuron will have the strongest effects, while other phases may have degraded effects. This property is modeled in this way: first, predict the phase of the target neuron ϕ^* by the strongest input, that is, $\phi^* = \arg\max_\phi |x_{i,\phi}|$. Then, weigh the input of different phases ϕ by a function $\beta(\phi, \phi^*)$. The function β can be flexible defined by the modeler to selectively model phase-insensitive neurons or coincidence detectors.

Specifically, each unit i has a net input x_i, an activation level a_i, a phase θ_i, and two parameters, the bias b_i and the decay rate γ_i. The net input x_i is calculated by first separately calculating inputs with different phases $\phi \in \Theta$:

$$x_{i,\phi}^{t+1} = \delta(\phi, \theta_i^t)(1 - \gamma_i)\, a_i^t + \sum_{j \in S_i} \delta(\phi, \theta_j^t)\, w_{ij}^{act}\, a_j^t, \tag{3.11}$$

where the first term is the self-recurrent input, S_i is the set of all units connecting to unit i, and w_{ij}^{act} is the *activation* weight from unit j to unit i. Then the net input x_i is obtained by mixing the phase-specific inputs $x_{i,\phi}$ by a mixing function $\beta(\phi, \phi^*)$:

$$x_i^{t+1} = \sum_\phi \beta(\phi^*, \phi)\, x_{i,\phi}^{t+1} \quad \text{where } \phi^* = \arg\max_\phi |x_{i,\phi}|. \tag{3.12}$$

The phase weighting function β has the form (discussed below):

$$\beta(\phi^*, \phi) = \delta(\phi^*, \phi)\, c_1 + [1 - \delta(\phi^*, \phi)]\, c_2, \tag{3.13}$$

where c_1 is the weight for the most active phase-specific input, and c_2 is the weight for other phases. Specifically, if $c_1 = c_2$, the unit ignores input phases as conventional additive units. If $c_1 > c_2$, the unit is a coincidence detector, and is more sensible to inputs synchronized with the strongest phase ϕ^*. With the net input x_i, the activation a_i of the unit is simply modeled by a linear function of the input, though other functions can be used:

$$a_i^{t+1} = x_i^{t+1} + b_i. \tag{3.14}$$

3.2.3.1 No phase-selectivity

When the β function is chosen to be identity function (3.15), the neuron has no phase selectivity. Its activity dynamics reduces to a generic firing-rate neuron and it simply sums up all weighted inputs.

$$\beta(\phi^*, \phi) = 1. \tag{3.15}$$

The network thus has the familiar form:

$$a_i^{t+1} = (1 - \gamma_i)\, a_i^t + \sum_{j \in S_i} w_{ij}^{act}\, a_j, \tag{3.16}$$

which is also known as Hopfield network model (Hopfield, 1984), or the additive Grossberg model (Grossberg, 1988). When the decay rate is set to $\gamma_i = 1$, the network further reduces to a simpler neural network without self-recurrent connections. Thus, it can be seen that the neuron model presented here can be seen is a generalization of firing-rate neurons. With proper setup, they can model traditional firing-rate neural networks as well as exercising its flexibility and capability in modeling binding and synchronization.

3.2.3.2 Strong phase-selectivity

This neuron can be configured to be coincidence detectors. To one extreme, it can only be sensitive to the dominant phase ϕ^*. In this case, we can set

$$\beta(\phi^*, \phi) = \delta_{\phi^* \phi}. \tag{3.17}$$

In this neuron, the maximum phase-specific input has complete influence on the unit. Inputs with other phases are filtered out. This extreme case might not be biologically plausible, unless the neuron can only integrate inputs from a very narrow time window. For two neurons to be considered as having different phases, their spike timings must differ by a conceptual threshold τ_θ as described above. Therefore, to map this model unit to a biological neuron, it corresponds to a neuron which has an integration window of spikes narrower than τ_θ. The theoretical oscillations of the model correspond to 20-70 Hz gamma oscillations of the cortex (Gray, 1999; Singer, 1999; Borgers & Kopell, 2003). Because spikes with different phases need to be enclosed in one gamma cycle (˜25 ms), the phase equality threshold τ_θ needs to be a fraction of the cycle and it can thus be roughly understood as about 5 ms. This is at the same time scale of cortical coincidence detectors (Azouz & Gray, 2000), so that the theoretical threshold τ_θ can be biologically implemented, though graded phase selectivity is still a safer assumption (defined below).

3.2.3.3 General phase-sensitivity

Because the only operator defined between two phases θ and ϕ is their equality or $\delta_{\theta\phi}$, the most general form of phase sensitivity for this model is

$$\beta(\phi^*, \phi) = \delta_{\phi^*\phi}\, c_1 + (1 - \delta_{\phi^*\phi})\, c_2, \tag{3.18}$$

where the weight for the dominant phase is c_1 and that for other phases is c_2. For a coincidence detector which is not so crisp as described by Eq. (3.17), it can be modeled by this version of β function by setting $c_1 > c_2$. The actual values depend on the width of the neuron's integration window. Because this model is intended for efficient high-level cognitive modeling of binding, some precision of the biological mechanisms is selectively lost by design. A modeler should check that the model neuron works in a reasonable way compared to biological neurons, and set up the parameters correspondingly. In practice, this is not difficult, because most parameters can be specified qualitatively. For example, a general condition, such as $c_1 > c_2$, may suffice.

3.2.4 Phase propagation

If the design of the above activation propagation rules is based on the rationale of capturing different levels of phase selectivity, then the design of phase propagation rules is fundamentally stricter. The phase propagation rule must predict the target neuron's phase well enough to capture benchmark phenomena of synchronization and desynchronization, mentioned above, that is: excitatorily connected neurons tend to synchronize, inhibitorily connected neurons tend to desynchronize, and a interactive network should be able to form multiple synchronization groups. Mathematically deriving the phase propagation rule is difficult, because the unit model corresponds to a cortical column which contains many types of interacting excitatory neurons and inhibitory neurons. A detailed mathematical model of such a cortical column and its interaction with others is difficult to analyze. The propagation rule presented here is first phenomenologically designed, in that it aims at capturing the above synchronization phenomena with the simplest mathematical form. The hope is that a simplistic model which can capture the real neural properties is likely to be a valid approximation of the biological mechanisms. A derivation of the rule is given in the appendix under simplified conditions (Section A.1).

The phase propagation rule is described below. Similar to the activation propagation, the phase-specific inputs are first computed for all phases $\phi \in \Theta$:

$$\tilde{x}_{i,\phi}^{t+1} = \delta(\phi, \theta_i^t)(1-\gamma_i)\, a_i^t + \sum_{j \in S_i} \delta(\phi, \theta_j^t)\, w_{ij}^{phase}\, a_j^t, \tag{3.19}$$

which has the same form as (3.11), but the weight is w_{ij}^{phase} instead of w_{ij}^{act}. Then, a probabilistic distribution for possible new phases is computed.

The pre-normalized phase probability mass $\tilde{p}_i(\phi)$ and normalized phase probability mass $p_i(\phi)$ are:

$$\tilde{p}_i^{t+1}(\phi) = \alpha_i + [\tilde{x}_{i,\phi}^{t+1}]_+^p, \quad p_i^{t+1}(\phi) = \frac{\tilde{p}_i^{t+1}(\phi)}{\sum_{\theta \in \Theta} \tilde{p}_i^{t+1}(\theta)}, \tag{3.20}$$

where $\alpha_i > 0$ is a noise term that determines random phase perturbation, and $p \geq 1$ determines the strength of the dominant phase (in the work, $p = 2$), and $[\cdot]_+$ is a rectifier defined as: $[x]_+ = x$ if $x > 0$ and $[x]_+ = 0$ if $x \leq 0$.

The phase is then chosen as

$$\theta_i^{t+1} = \phi \text{ with probability } p_i^{t+1}(\phi). \tag{3.21}$$

3.2.5 Weights

For flexibility, the weights for activation propagation w_{ij}^{act} and phase propagation w_{ij}^{phase} are assumed to be different. This allows some connections to have selective effects in activation and phase control. In the current work, they are assumed to be related to a general weight w_{ij} from unit j to unit i by two factors κ^{act} and κ^{phase}:

$$w_{ij}^{act} = \kappa^{act}\, w_{ij}, \quad w_{ij}^{phase} = \kappa^{phase}\, w_{ij}. \tag{3.22}$$

The factors κ^{act} and κ^{phase} may depend on the projection in which the weight belongs, since different neural pathways may have different neural transmitters and neurochemical effects.

3.2.6 Hebbian learning

Recent experiments have demonstrated that synaptic plasticity depends on the temporal ordering of presynaptic and postsynaptic spikes, that is, the spike timing difference $\Delta t = t_{post} - t_{pre}$. Typically, the synapse is strengthened when the presynaptic spike shortly precedes the postsynaptic spike by tens of milliseconds ($\Delta t > 0$), and it is

weakened when the presynaptic spike shortly follows the postsynaptic spike ($\Delta t < 0$) (Bi & Poo, 1998; Markram, Lubke, Frotscher, & Sakmann, 1997; Abbott & Nelson, 2000).

In the PNN framework, the relative temporal ordering of spikes is not explicitly modeled. It does not directly accommodate the full spectrum of STDP. However, it can capture a simple form of STDP, that is, when neurons fire in synchrony, their connections are strengthened, but when they fire out of synchrony, their connections may be weakened. This is arguably a very powerful way of modeling binding. The learning rule is the following (Left-hand side is of time $t+1$ and right-hand side is of time t):

$$w_{ij} \leftarrow w_{ij} + \lambda^+ \delta(\theta_i, \theta_j) a_i a_j + \lambda^- [1 - \delta(\theta_i, \theta_j)] a_i a_j, \tag{3.23}$$

where a_i is the post-synaptic activity, and a_j is the pre-synaptic activity, and λ^+ and λ^- are learning rates. For the above effect of synchrony, the parameters can be set qualitatively as $\lambda^+ > 0$ and $\lambda^- < 0$. For Hebbian learning regardless of spike timing, as firing-rate neurons, the parameters can be set as $\lambda^+ > 0$ and $\lambda^- > 0$.

To constrain the boundaries of the weights, normalization can be performed on the new weight vector $\vec{w}_i = (w_{i1}, w_{i2}, \ldots, w_{in})$, where the norm $\| \cdot \|$ can be the L1- or L2-norm:

$$\text{if } \|\vec{w}_i\| > 1,\ \vec{w}_i \leftarrow \vec{w}_i / \|\vec{w}_i\|. \tag{3.24}$$

Negative weights can also be optionally eliminated by weight boundaries $[w_{min}, w_{max}]$.

An interesting property of the learning rule is that although it does not capture timing information, it can still capture *causality* information, which Hebb (1949) correctly predicts to be crucial for determining the direction of synaptic changes. In Hebb's conjecture, when a neuron *causes* another neuron to fire, their connection is strengthened. In STDP, the causality is represented in spike timing difference, which signals the synaptic changes. In PNN, the causality, instead, is represented by phase equality $\delta(\theta_i, \theta_j)$, which is used to determine synaptic changes. Though the surface mechanisms are different, the learning in PNN may capture main properties of STDP by this first-order approximation.

3.2.7 Other learning rules are still applicable

Because the PNN is a generalized firing-rate neural network (and can be reduced to a generic firing-rate network), learning rules that have been used for firing-rate neurons

can still be used in the PNN under the condition: $\beta(\phi^*, \phi) = 1$. Because typical firing-rate learning rules do not concern about the phase, the phase propagation rules can be applied over existing firing-rate propagation rules to predict which neurons are synchronized and which neurons are desynchronized. This is a nice property because this allows the new framework to smoothly extend the existing framework of firing-rates neurons. It can still use existing learning rules such as different formulations of Hebbian learning (Bienenstock, Cooper, & Munro, 1982; Oja, 1982) and backpropagation (Rumelhart & McClelland, 1986). In a simulation below, it will be shown that how the PNN phase propagation rule can be combined with the BCM rule (Bienenstock et al., 1982) in a segmentation task.

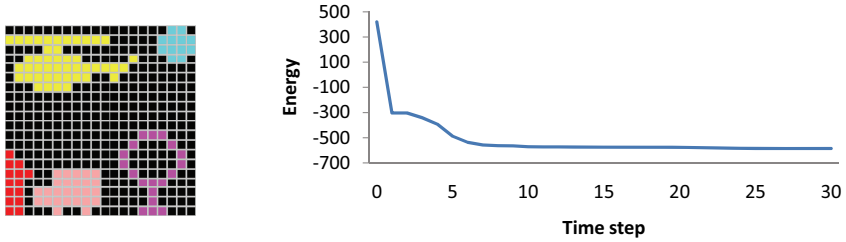
3.2.8 Demonstrations of required properties

From the above discussion, it can be seen that the crucial extension of the PNN framework on firing-rate neurons is its phase propagation rule, and a mechanism to model neurons with phase selectivity. The phase propagation rule is phenomenologically designed to allow efficient modeling of binding with firing-rate neurons, because binding is an important cortical function. The properties, which this neuron is expected to have, are 1) a representation of causality so that the Hebbian learning rule can validly approximate STDP, 2) a good prediction of neural synchronization based on neural connectivity. The latter has been characterized as excitatory connections lead to synchrony while inhibitory connections lead to desynchrony. All these properties are successfully incorporated in the model. The mathematical proofs are given in the Appendix.

Here, the following simulations demonstrate the required synchronization property, and the a correspondence between the PNN Hebbian learning rule and the STDP learning rule.

3.2.8.1 Simulation

The synchronization and desynchronization properties are demonstrated in the following simulation. This simulation mainly uses the PNN phase propagation rules, while the demonstration of the PNN Hebbian learning is left for the next demonstration. The network consists of a single layer of units, which have all-to-all connections. During training, random objects (one or two at a time) are presented to the layer, while the lateral connections are trained with the BCM rule, a weight-constrained Hebbian rule



A. Segmented image B. Energy curve

Figure 3.5: Synchronization/desynchronization simulation

(Bienenstock et al., 1982). The BCM rule can efficiently produce excitatory connections among correlated features, and inhibitory connections among uncorrelated features, resulting in a network for testing synchronization properties. In contrast, the PNN Hebbian learning rule is more suitable for modeling phase propagation between neural layers, as opposed to lateral connections. After forming the positive and negative connections, the synchronization properties of the PNN neurons can be readily tested. After training, features of the same object have excitatory connections (~ 0.1), and features of different objects have inhibitory connections (~ -0.05). Then the synchronization/desynchronization behavior is tested by activating all objects simultaneously. The result shows a perfect match between neuron's behaviors and the required properties. After several cycles of processing, features with excitatory connections are synchronized, and features with inhibitory connections are desynchronized from each other (Figure 3.5A). The process is fast as shown in the energy curve of the network (Figure 3.5B), where the energy is defined as

$$E = -\sum_{i,j} \delta(\theta_i, \theta_j) w_{ij} a_i a_j.$$

This energy function E measures the mismatch between phases and weights. It is increased if a pair of units have the excitatory weights but different phases, or they have inhibitory weights but the same phase. By directly minimizing this energy function, the network can also be shown to have the above synchronization properties (Wersing, Steil, & Ritter, 2001). Here it can be seen that the energy function is automatically minimized by the phase propagation rules. The PNN model, thus, can be seen as a simple alternative for the energy-minimization process. This property can make the simulation faster and easier to implement by software and hardware. It also provides an alternative way to model binding without the use of spiking neurons.

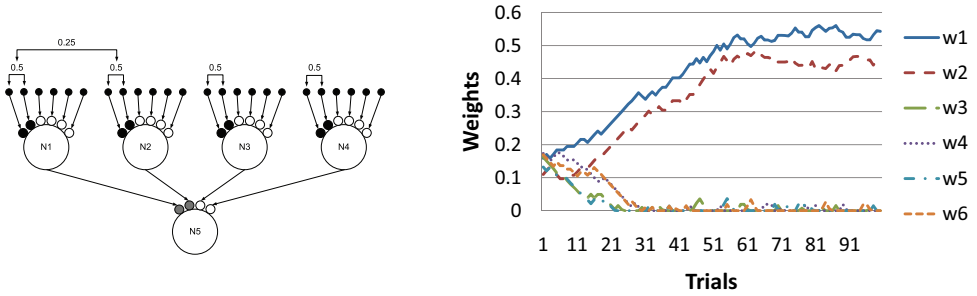
3.2.8.2 Synchrony detection

With the Hebbian learning rule (3.23)-(3.24), I experimented with a small 2-layer network of phase neurons with the configuration shown in Figure 3.6A. The test has often been used to verify the basic STDP properties of spiking neurons (Bofill-i-Petit & Murray, 2004). For a spiking neural network, typical stimulus is to present the first layer with spike trains in which a pair of input (inputs 1 and 2) to each neuron have a temporal correlation of 0.5. Further, the pairs of inputs to Neuron 1 (N1) and Neuron 2 (N2) have a second-level correlation of 0.25. The typical result is that each of the first-layer neurons selectively responds to the correlated inputs 1 and 2, and the second layer neuron N5 selectively respond to the correlated neurons N1 and N2.

I replicate the test performed in (Bofill-i-Petit & Murray, 2004) on the phase neurons. Because of the representations in phase neurons, the inputs become static patterns where first two inputs to each unit are synchronized with a probability of 0.5 (represented by the same phase), and the first pairs of inputs to N1 and N2 are synchronized with a probability of 0.25. Each input pattern is clamped at Layer 1 for three time steps to allow activity propagation. Units are phase sensitive, with parameters $c_1 = 1$, $c_2 = .5$. Learning rates for the first layer are $\lambda^+ = .01$ and λ^-=-.01; and second layer: $\lambda^+ = .001$ and $\lambda^- = -.001$. Both layers use L1-normalization. Simulation results show that the phase neurons have similar synchrony detection properties as spiking neurons (Bofill-i-Petit & Murray, 2004). Layer 1 units typically detect the correlated pattern of inputs 1 and 2, encoded in the growing weights w1 and w2 (Figure 3.6B). The secondary correlation between N1 and N2 can be detected by the secondary unit N5, whose first two weights would increase, while others decrease. However, from this demonstration, the rate of detecting the secondary correlation is not perfect. In some cases, if the first-level synchrony is not detected in time, the secondary neuron will learn other patterns. Therefore, this demonstration shows that the PNN Hebbian learning rule has the potential of detecting synchronized firings as STDP rules, but for higher performance of this network in pattern detection, more analytical work on refining the learning rules may be needed.

3.2.9 Summary

The design and verification work has shown that PNN is a useful extension to firing-rate neurons. It enjoys the efficiency and the diversity of work about firing-rate



A. Two-layer network structure B. Weights for N1 (labeled from left to right)

Figure 3.6: Hierarchical synchrony detection (see text for more details)

models. In addition to the firing rate, the phase neuron has mechanisms for modeling binding, synchronization and a basic form of STDP. Therefore, this simplistic neuron model may be very suitable for modeling high-level cognitive processes, where selective attention, cognitive control, and interface between short-term and long-term memory are involved. In the next section, I will further develop the modeling tool, and discuss organizations of these neurons in a neural network.

In summary, the unit activity and phase dynamics of the phase neurons are modeled by the following equations. The variables and constants are listed in Table 3.1.

Table 3.1: PNN variables and parameters

Symbols	Description
$x_{i,\phi}$	The phase-specific *activation* input of unit i for phase ϕ
x_i	The effective input of unit i
$\tilde{x}_{i,\phi}$	The phase-specific *phase* input of unit i for phase ϕ
a_i	The activation of unit i
γ_i	The decay rate of unit i
b_i	The bias of unit i
α_i	Phase noise parameter for unit i
w_{ij}	The general weight from unit j to i
κ_p^{act}, κ_p^{phase}	The activation and phase weight factors for a projection p
w_{ij}^{act}, w_{ij}^{phase}	The activation and phase input weights $w_{ij}^{act} = \kappa_p^{act} w_{ij}$, $w_{ij}^{phase} = \kappa_p^{phase} w_{ij}$
$\lambda_{ij}^{+}, \lambda_{ij}^{-}$	Bidirectional learning rates for weight from i to j

The phase-specific *activation* inputs:

$$x_{i,\phi}^{t+1} = \delta(\phi, \theta_i^t)(1-\gamma_i)\, a_i^t + \sum_{j \in S_i} \delta(\phi, \theta_j^t)\, w_{ij}^{act}\, a_j^t, \tag{3.25}$$

The effective net input:

$$x_i^{t+1} = \sum_{\phi} \beta(\phi^*, \phi)\, x_{i,\phi}^{t+1} \quad \text{where } \phi^* = \arg\max_{\phi} |x_{i,\phi}|. \tag{3.26}$$

The phase weighting function β for calculating effective input:

$$\beta(\phi^*, \phi) = 1 \qquad \text{No phase sensitivity} \tag{3.27}$$

$$\beta(\phi^*, \phi) = \delta_{\phi^* \phi} \qquad \text{Extreme phase sensitivity} \tag{3.28}$$

$$\beta(\phi^*, \phi) = \delta_{\phi^* \phi}\, c_1 + (1 - \delta_{\phi^* \phi})\, c_2 \qquad \text{Parametrized sensitivity} \tag{3.29}$$

The activation a_i:

$$a_i^{t+1} = x_i^{t+1} + b_i. \tag{3.30}$$

The phase-specific *phase* inputs:

$$\tilde{x}_{i,\phi}^{t+1} = \delta(\phi, \theta_i^t)(1-\gamma_i)\, a_i^t + \sum_{j \in S_i} \delta(\phi, \theta_j^t)\, w_{ij}^{phase}\, a_j^t, \tag{3.31}$$

The prenormalized phase probability mass $\tilde{p}_i(\phi)$ and normalized phase probability mass $p_i(\phi)$:

$$\tilde{p}_i^{t+1}(\phi) = \alpha_i + [\tilde{x}_{i,\phi}^{t+1}]_+^p, \quad p_i^{t+1}(\phi) = \frac{\tilde{p}_i^{t+1}(\phi)}{\sum_{\theta \in \Theta} \tilde{p}_i^{t+1}(\theta)}, \tag{3.32}$$

The phase:

$$\theta_i^{t+1} = \phi \text{ with probability } p_i^{t+1}(\phi). \tag{3.33}$$

The Hebbian learning rule (time $t+1$ on LHS and time t on RHS):

$$w_{ij} \leftarrow w_{ij} + \lambda^+ \delta(\theta_i, \theta_j) a_i a_j + \lambda^- [1 - \delta(\theta_i, \theta_j)] a_i a_j, \tag{3.34}$$

$$w_{ij} \leftarrow w_{ij} / \|\mathbf{w}_i\| \quad \text{normalization} \tag{3.35}$$

3.3 The PNN Framework

To model high-level cognition, the basic phase neurons designed and verified above need to be organized into neural networks. Besides the typical organization structures, such as layers and projections, some other useful modeling structures are also included in the phase neuron network (PNN) framework. They include efficient ways to process the output of a layer by filters (e.g., the winner-take-all filter), the delay lines, and a special mechanism to model cognitive control, termed the Central Executive as in Baddeley's working memory model (Baddeley, 1986), which monitors and coordinates different components of a network.

3.3.1 Layers and filters

Neurons are typically organized in layers. In this framework, some commonly used processing on layers is abstracted and implemented as *filters*. A filter takes the activation levels a_i of the layer, and produces the layer's output o_i. Some commonly used filters are described below.

1. Winner-Take-All (WTA) Filter: Only the most active unit can pass this filter and output its activity. Although winner-take-all can be implemented by inhibitory lateral connections, this filter is faster and can simplify this commonly used process in modeling.

2. Multiple-Winner-Take-All (MWTA) Filter: This filter selects units that are the most active within a local neighborhood. If the neighborhood is large enough to cover the whole layer, this filter is identical to the above WTA filter. The neighborhood shape can be defined as a square with a size or a circle with a diameter.

3. Phase Selection Filter: This filter only allows activity with a specific phase to pass through. The reference phase comes from the output of one or more phase neurons.

The functionality of each of these filters can be implemented by a detailed network of phase neurons. The use of them can accelerate simulation and make the model easier to understand, but they do not add mechanisms that neural networks cannot implement.

3.3.2 Projections

A projection is a group of connections connecting one layer to another. In this framework, the basic forms of projections that can be made between two layers are: topographical projections (one-to-one connections), divergent projections (one-to-many connections), convergent projections (many-to-one connections), and all-to-all connections. After a projection is specified, its weights can be set to zero or randomly initialized, and they can be later adjusted by learning rules.

Delay time The time steps for a neural activation to go through a projection can be changed in this framework, allowing modelers to study the effects of transmission delays and synchronize output from different neural pathways. For example, perfect reproduction of bird-songs requires precise timing in motor sequences, and it is believed to be done with a neural circuitry that adjusts the precise delay time of motor elements by imitating a learned song template (Abarbanel, Gibb, Mindlin, Rabinovich, & Talathi, 2004; Nottebohm, 2002). In a multilayer neural network, the outputs of different layers may also need to be synchronized for further processing, the synchrony can be conveniently controlled by the delay time parameter of a projection.

3.3.3 Monitoring signals and actions

In a large-scale interactive neural network, such as the working memory model, the results of different components need to be monitored, and used to signal other components to cooperate. For example, the recoding of a visually presented letter in a phonological form requires the coordination of the visual buffer and an articulatory component. In the PNN framework, such complicated coordination is formulated by signal/action mappings. Layers can generate monitoring *signals* informing the status of the a layer, such as the existence of above-threshold activation. Binary state units, which are phase units having one of the two values, 0 and 1, can also provide signals about their own states. These signals are processed by the *Central Executive*, which issues a set of actions which can act upon layers, projections or binary state units (Figure 3.7). The common actions for layers, projections and binary state units are ON and OFF actions, which inhibit or enable (activate) them in order to control the behavior of the interactive network. The signal/action maps can be learned through reward signals, and can also be specified by the modeler. It turns out that this mechanism is very effective and flexible

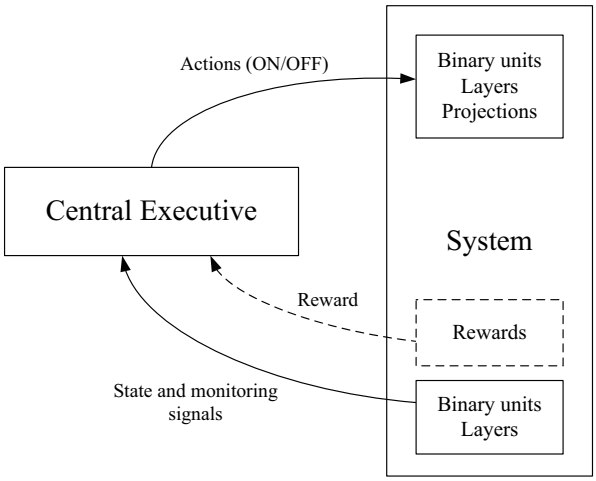


Figure 3.7: Signal-action controller

in controlling complicated network interactions. The working memory models studied in Chapter 5 will rely on this formulation of the central executive.

3.4 Conclusion

A general-purpose neural network framework, the Phase Neural Network (PNN), is developed in this chapter. It extends the firing-rate neurons to have abilities to correctly represent and simulate changes of neural synchronization. This allows the network to efficiently model high-level cognition including binding. The PNN framework also provides convenient facilities to coordinate large-scale interactive neural networks. The control mechanism uses a Signal-Action mapping (S/A mapping) to perform most control functions in the neural network model. This framework provides a coherent way to model working memory and other cognitive functions.

Chapter 4

The Divergent-Reconvergent Model of Sequential Working Memory

In Chapter 1, we have seen that working memory research has converged on two serial order mechanisms: activation gradient coding (the ordinal theory), and position coding (the positional theory). This chapter proposes a noval but plausible neural substrate for activation gradient coding, based on convincing neuroanatomical and physiological data. Then, the neural mechanism is simulated by computational models, whose performance is compared to human performance in the widely used working memory probing task: immediate serial recall (ISR). As discussed later, activation gradient coding is one of serveral serial order mechanisms that are used in ISR. Other mechanisms will be studied in later chapters.

4.1 Methodological principles

Computational models can be highly useful for gaining insight into biological mechanisms. Before they can be trusted, such models must be validated against biological data, to make sure their structures and processes are realistic. In construction of models in this thesis, a guiding principle is to track biological plausibility of all studied models. Practically, three levels of biological plausibility are introduced to evaluate and categorize the proposed models. They are *strict*, *plausible,* and *functional.* The *strict*

level means all components in a model are based on known neural substrates, and they satisfactorily match its target neural anatomy and physiology. This is a high level of biological plausibility, by which the model can be seen as providing a computational account for the functions of the target biological system. The next *plausible* level means that the model has specified a neural network that is likely to be developed in the brain. To qualify this level of biological plausibility, the following Constraints are used:

1. The model can specify the layers and biologically feasible projections between and within layers: topographical, divergent and convergent.

2. The model can specify initial weights of a projection by random distributions (Gaussian or uniform), but cannot specify weights for individual connections.

3. The model can specify a biologically plausible learning algorithm to modify the weights based on well-known environmental inputs.

4. The model can be a *plausible functional* model. A plausible functional model can specify weights of individual connections, but it has to correspond to a functionally equivalent model satisfying Constraints 1-3 above.

The plausible functional model substitutes for a more complex model while avoiding arbitrariness. It is mainly used to expedite simulation and to facilitate functional understanding of a complex model. The lowest degree of biological plausibility in this work is termed *functional*. A functional model has to possess the target functionality and produce required benchmark phenomena, however, it does not need to satisfy Constraint 4, that is, its weights may not be fully specified by a known biologically plausible learning algorithm or based on known neural anatomy. A functional model might be a correct model, given that it is constrained by phenomena and it may be supported by a new learning algorithm and/or neural substrates discovered later.

4.2 Biological background

This section reviews brain structures that are involved in working memory and sequencing. They turn out to converge on two brain areas: the frontal cortex and the basal ganglia. Below, the structures of the two brain areas and the ubiquitous cortico-basal ganglionic loops will first be reviewed, then their currently known functions are summarized.

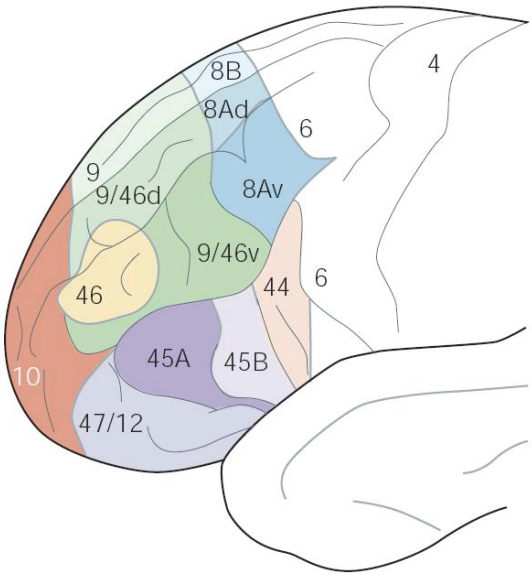


Figure 4.1: The prefrontal cortex (Source: Ramnani & Owen, 2004)

4.2.1 The prefrontal cortex

The prefrontal cortex is the anterior part of the frontal lobes of the brain, as shown in Figure 4.1. It can be subdivided into ventrolateral (Brodmann's areas 44, 45 and 47), dorsolateral (DLPFC; Brodmann's areas 9-10, 46), and anterior prefrontal areas (Brodmann's area 10). It is widely accepted that the prefrontal cortex plays crucial roles in working memory, attention and executive control. Experimental findings so far suggest that the activities in the ventrolateral, dorsolateral and anterior regions can be roughly attributed to the *updating/maintenance* of information, the *selection/manipulation/monitoring* of that information, and the *selection of processes/subgoals*, respectively (Fletcher & Henson, 2001), though it should also be noted that PFC is a complicated brain region which is hard to split along simple lines (Miller, 2000).

Among abundant evidence showing a prefrontal role in working memory, there are several characteristic results from single-unit recordings and fMRI studies. A signature property of PFC activity in an animal working memory task is the sustained neural activity between the stimulus presentation and the consequent response (Fuster, Bauer, & Jervey, 1982). The strength of prefrontal activity during information encoding has also been found to correlated with the probability of a successful subsequent retrieval (Rypma & D'Esposito, 2003). Brain imaging shows that prefrontal activity is correlated with

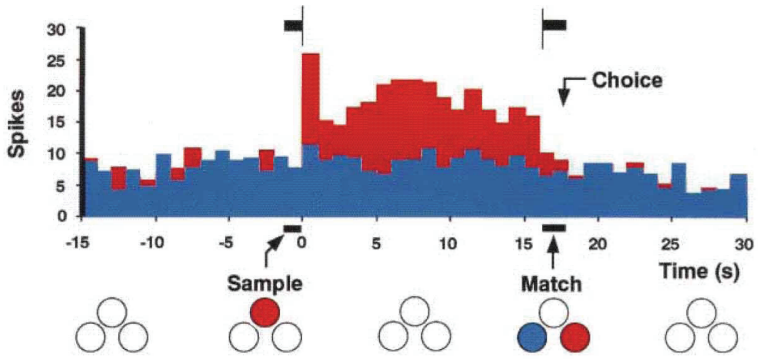


Figure 4.2: Sustained activity in inferotemporal cortex during maintenance of information

maintenance or manipulation of information (Cohen et al., 1997; Courtney, Ungerleider, Keil, & Haxby, 1997; Smith, Jonides, Marshuetz, & Koeppe, 1998), and its activation strength is proportional to working memory load (Callicott et al., 1999; Howard et al., 2003). Although neurons in other parts of the cortex, such as the inferotemporal cortex (Fuster & Jervey, 1982), have similar delay-period activity as PFC, the participation of PFC neurons are probably necessary. The working memory maintenance mechanism has been hypothesized to depend on the long reciprocal connections, or "loops", between the prefrontal cortex and sensory/perceptual areas (e.g., the inferotemporal cortex). A connectivity diagram is shown in Figure 4.3. This view on working memory maintenance mechanism has been supported by experiments, which show that if either component of the network (the inferotemporal or the prefrontal cortex) is disabled, the activity of the other is reduced and the correct performance of the task is dramatically impaired (Fuster, Bauer, & Jervey, 1985). Another important prefrontal connectivity likely involved in working memory functions is the loops between the prefrontal cortex and the basal ganglia, reviewed in the next section.

4.2.2 The basal ganglia

The basal ganglia are a collection of nuclei deep to the white matter of cerebral cortex. They consist of the caudate, the putamen, the nucleus accumbens, the globus pallidus (GP, or pallidum), the subthalamic nucleus (STN), the ventral tegmental area (VTA), the substantia nigra pars compacta (SNc) and pars reticulata (SNr). The caudate, putamen and nucleus accumbens together are collectively known as the *striatum*. The globus pallidus has an internal segment (GPi) and an external segment (GPe). Rela-

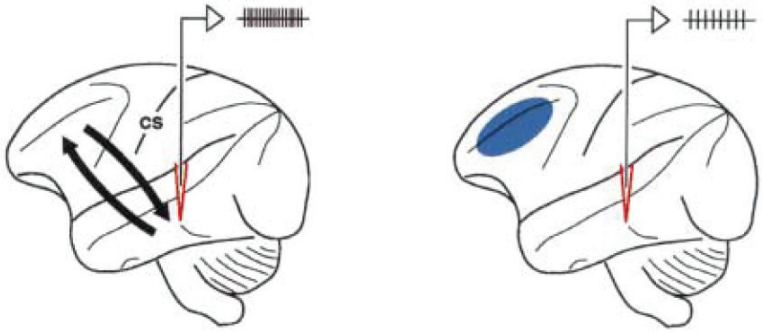


Figure 4.3: Long reciprocal connections for maintaining neural activity (Source: Fuster, 1997)

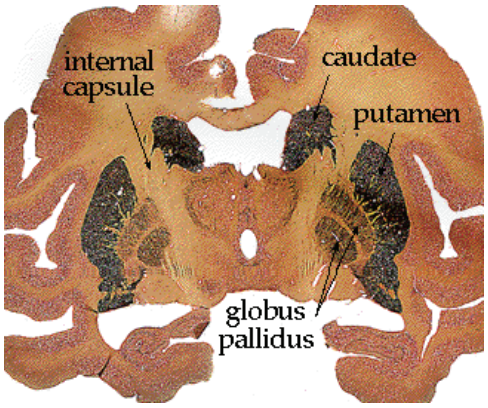


Figure 4.4: A section through monkey basal ganglia, stained for acetylcholinesterase (AChE). The striatum (caudate, putamen) with high concentration of the neurotransmitter Acetylcholine (ACh) are stained, while the globus pallidus (GPi & GPe) remains fairly pale. (Source: Molavi, 1997)

tive positions of main basal ganglia structures can be seen in the section image in Figure 4.4. The basal ganglia as a whole receive input from the cortex, and provide a feedback to it (via basal ganglionic influence on the thalamus). The striatum is the *input* station of the basal ganglia, receiving input from the whole cortex. GPi and the substantia nigra pars reticulata (SNr) are the *output* station of the basal ganglia, collectively referred to as GPi/SNr because they contain cytologically similar neurons and are commonly regarded as one functional structure. They project back to the cortical area where the input is originated (but see Joel & Weiner, 1994). The loop is known as the cortico-basal ganglionic loop, and several other names: the cortiostriatum loop or simply the basal ganglia loop.

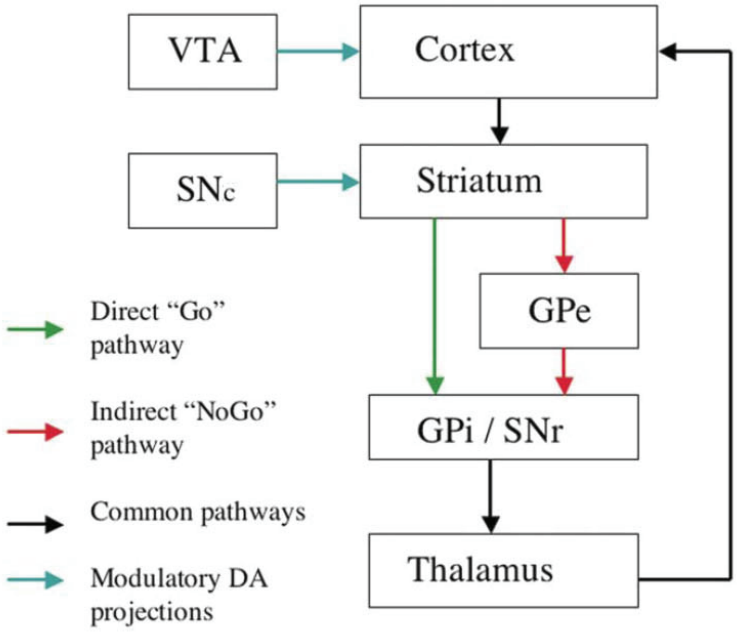


Figure 4.5: Diagrammatic representation of the circuitry within each cortico-basal ganglionic loop (Source: Seger, 2006)

Within the basal ganglia system, the input station, striatum, sends efferents to the output structures GPi/SNr via two distinct pathways: a direct pathway and an indirect pathway (Figure 4.5). Through the direct pathway, the striatum directly inhibits the output structures. Because the output structures tonically inhibits the thalamus which projects to the cortex, the overall effect of the direct pathway is excitatory. The polysynaptic indirect pathway to the GPi is an opposing pathway of activity. This pathway originates in the striatum with inhibitory projections to the GPe. The GPe then inhibits the STN through another level of inhibitory connections. At last, the STN excites the GPi. In sum, the indirect pathway exerts an opposing excitatory effect on GPi, which results in inhibition of the thalamus. For this reason, the direct pathway is also known as the "Go" pathway since it facilitates expression of responses, and the indirect pathway is known as the "No Go" pathway, inhibiting expression of responses (Seger, 2006).

Alexander, DeLong and Strick (1986) first pointed out the correct anatomical structure of the cortico-basal ganglionic connections. In contrast to previous understanding, these loops were found to be a set of parallel structurally and functionally separate circuits which originate in distinct cortical areas, target different areas of the striatum,

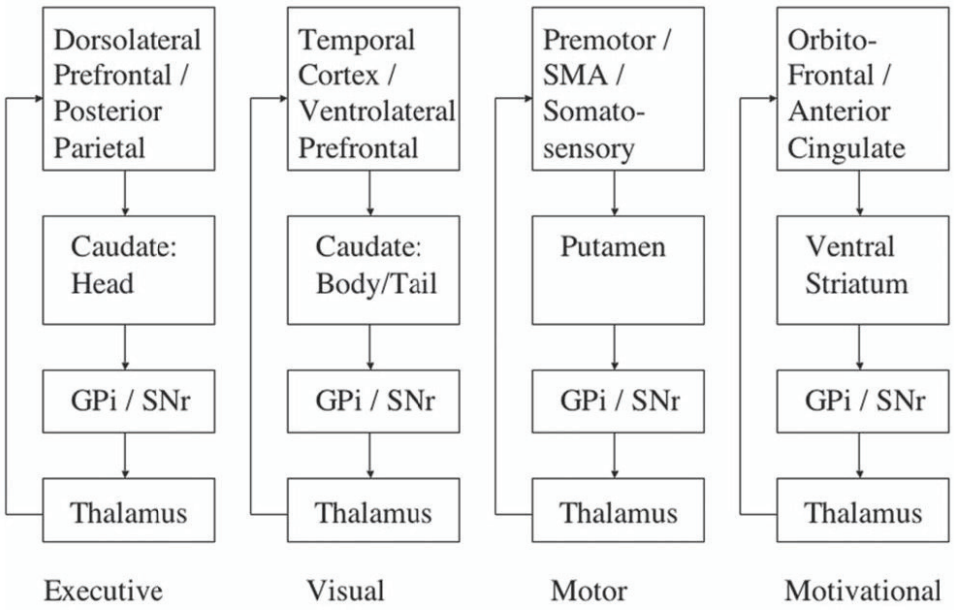


Figure 4.6: Frontal-Basal ganglionic loops (Source: Seger, 2006)

and feed back to the cortical area where the circuit originates. The original five loops identified by Alexander et al. (1986) are later joined by new discoveries to form a total of nine separate loops (Middleton & Strick, 2000). One way to categorize them is by the functions of the cortical areas which they involve. In this way, the nine loops loops can be grouped as *executive*, *visual*, *motor* and *motivational* loops, as shown in Figure 4.6. Each circuit in the figure represents a collection of loops performing a similar function. For example, the motor loops include three loops, originating in the premotor, supplementary motor area (SMA), and the somatosensory cortex, respectively. It is believed that these loops complement the functions of the cortical areas they interact with, however, their exact functions and mechanisms are not yet fully understood.

The traditional view is that the basal ganglia are only involved in the control of movement. This view has been constantly challenged in the recent 10-20 years (Seger, 2006; Middleton & Strick, 2000; L. L. Brown, Schneider, & Lidsky, 1997). Various sources of data indicate basal ganglia's involvement in *nonmotor* operations. Anatomical studies that trace the connectivity between the cortical areas and the basal ganglia clearly indicate that several basal ganglia loops involve cognitive areas of the cerebral cortex. For example, the *executive* loop is originated in the dorsolateral prefrontal area (DLPFC), which is an important area for working memory. The *affective* loop is originated in

the orbitofrontal cortex, which subserves motivational and reward-based learning. Further, single-unit recording experiments show that the activity of neurons within many locations of the basal ganglia is more related to cognitive or sensory operations than to motor functions. For example, in monkeys which are trained to reproduce a stimulus sequence after a delay (similar to human immediate serial recall), increased neural activity is recorded in the globus pallidus during the "delay" period, but not the behavioral "movement" period. This suggests basal ganglia's role in working memory. More specific studies have shown that the activity pattern observed in the basal ganglia is very similar to that observed in prefrontal cortex during delay periods (Owen, 1997). Third, deficits in basal ganglia can cause nonmotor disturbances. Early Parkinson's disease which affects the sensorimotor portion of the striatum is associated with largely motor symptoms, but early Huntington's disease which affects associative portions of the striatum is associated with cognitive disturbances (Middleton & Strick, 2000). Severe Huntington and Parkinson diseases which affect cognitive portion of the basal ganglia can impair the production of syntax (Lieberman, 2000). At last, the deficit FOXP2 gene can also lead to impairment in the production of speech sequences due to dysfunctioning basal ganglia (Lai, Gerrelli, Monaco, Fisher, & Copp, 2003). Therefore, it is probably the case that the basal ganglia do not merely involve in motor functions, but participate in various cognitive functions as well. Current views about the functions of these loops are discussed in the next section.

4.2.3 Functions of the cortico-basal ganglionic loops

Many functions have been attributed to the cortico-basal ganglionic loops. They include long-term memory functions, such as long-term sequence learning and production (Seger, 2006; Berns & Sejnowski, 1998; Aldridge & Berridge, 1998; Aldridge, Berridge, & Rosen, 2004; Cromwell & Berridge, 1996), and the association between stimuli and responses (Seger, 2006), as well as short-term memory functions. The proposed short-term memory functions include gating working memory updates (Frank, Loughry, & O'Reilly, 2001; Gruber, Dayan, Gutkin, & Solla, 2006), and short-term sequence planning and encoding (Middleton & Strick, 2000; Beiser & Houk, 1998). The latter function, short-term sequence planning and encoding, is supported by several lines of evidence. First, a portion of striatal neurons exhibit increased delay-period activity as PFC neurons (Alexander, 1987). Further, basal ganglia legions affect working memory similarly to PFC lesions,

especially in people's ability in short-term sequencing and planning (Lieberman, 2000; Owen, 1997).

Crucial but unknown questions are: How exactly do basal ganglia participate in short-term sequencing? Do they play roles in sequence encoding or retrieval? How are sequences encoded and retrieved? If basal ganglia do play a role in the processes, what behavioral characteristics do they have? Can they explain human behavior? These questions are answered in the rest of this chapter.

4.3 The Divergent-Reconvergent model

The Divergent-Reconvergent model proposes a novel neural mechanism of sequence encoding and retrieval. This neural mechanism, to a great extent, matches with the neuroanatomy and physiology of the basal ganglia. In this section, the idea is first described, following by discussion of its neural substrate in the cortico-basal ganglia loops. Then, computational models are constructed to verify this idea. It is hypothesized that this neural substrate subserves the widely used working memory task: immediate serial recall (ISR). Simulated results will be compared to human performance in this task.

4.3.1 The idea

In one cortico-basal ganglionic loop, the main input station of the basal ganglia, the striatum, receives inputs from one area of the neocortex, and sends processed information, through GPi, back to the neocortical area where it receives input from. It has puzzled investigators for years that there are millions of projection neurons in the primate striatum, but they project to a much smaller set of neurons in the basal ganglia output nuclei, the internal segment of the global pallidus (GPi). Graybiel et al. suggested a divergent-reconvergent connection structure, where the input from the neocortex first divergently project to the striatum, and then reconverges in the GPi (Graybiel, Aosaki, Flaherty, & Kimura, 1994). They proposed that the diverged input feeds into multiple "local experts" performing distinct computational tasks, and their results converge in a later processing area, where the results are selected by a gating network, to generate a single output of the network.

Here, we propose a different function that the divergent-reconvergent architecture may also perform - the sequencing of motor and cognitive elements. Depending on which cortical area the input comes from, the input elements could be perceived objects

(letters, digits or words), planned movements, or cognitive operations (e.g., substeps in a plan, and planned chess moves). In this theory, the divergent-reconvergent architecture encodes a sequence of the input items into a spatial pattern by the interaction between the cortex and the basal ganglia, especially the working memory area DLPFC, and pre-motor and motor areas in movement sequencing, and then the same neural network can sequentially sequentially decode or retrieve the buffered elements.

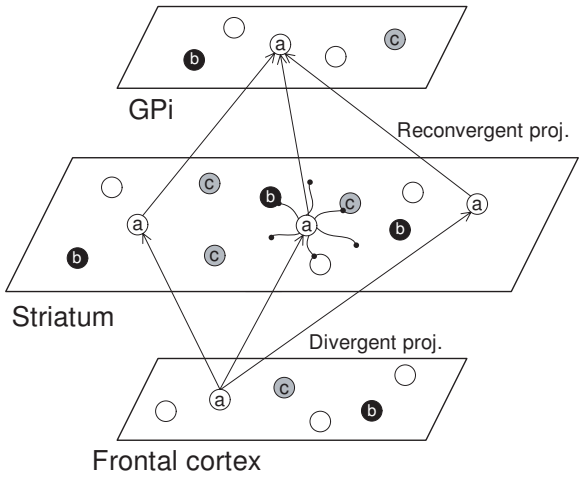


Figure 4.7: The divergent-reconvergent scheme

Figure 4.7 shows the divergent-reconvergent connectivity pattern which can support sequence encoding and retrieval. Units in the frontal cortex represent input items. They can be movements represented in the motor cortex, or cognitive working memory items maintained in the prefrontal cortex (PFC). Each input item activates a set of units that are spatially distributed in the striatum. This set of units, then, convergently activates an output unit in the internal segment of the globus pallidus (GPi). With this connectivity, the network remaps an input unit to a corresponding output unit, via a distributed internal representation. Here, let's consider a variation of this network. What if recurrent loops and lateral inhibition are added to the striatal units? An interesting property emerges: presented items can be actively maintained in the striatum (discussed later), and when a sequence of items is presented to the system, early ones would inhibit a proportion of striatal units representing late items, which will make the distributed representations of late items "weaker" than those of earlier ones. In this way, an activation gradient where early items are more active than late ones can be produced in the striatum. The activation gradient is represented in the GPi, where exist

units that sum up the striatal activity for these items. The GPi itself or a downstream area may then perform a competitive selection process on the active items in GPi to iteratively retrieve items sequenced in the striatum.

4.3.2 Biological support

The above conceptual idea matches very well with neural anatomy of the cortico-basal ganglionic loops. First, the main neurons in the striatum, spiny neurons, are inhibitory neurons with extensive axon collaterals. They may directly exert lateral inhibition on neighbors. This *direct* lateral inhibition, though some experiments have found them to be weak (Tepper, Koos, & Wilson, 2004; Jaeger, Kita, & Wilson, 1994), has been computationally demonstrated to be highly beneficial for sequence encoding (Beiser & Houk, 1998). Thus, this mechanism should not be immediately dismissed. Alternatively, there is a *forward inhibition* mechanism. There are inhibitory aspiny interneurons scattered near a spiny neuron, which receive the same cortical input. These fast-spiking inhibitory neurons may inhibit other spiny neurons (Kawaguchi, 1997), and functionally act as lateral inhibition for the spiny neuron.

Second, let's look at a possible neural substrate for competitive selection. In the model, items' distributed representations in striatum are first converted to localist representations, and then the most active item is selected form the localist representation. This scheme matches very well with neurophysiology. In fact, neurophysiological recordings suggest that GPi neurons are probably performing dimension reduction on its striatal input (Bergman et al., 1998), which is exactly what is needed. Then, the reduced representations can propagate, via the thalamus, back to the PFC. On this pathway, the dimensions can be further reduced. Thus, in the PFC, a localist representation of the activation gradient is expected to be found. Evidence for this has been found, as reviewed in Bullock (2004). Though it is not required by the model, it is worth noting that if GPi neurons can perform WTA locally (Berns & Sejnowski, 1998), sequence decoding may also occur in the basal ganglia. However, this would not explain why an activation gradient is found in the PFC.

The last feature in the above model is that the neurons which exert lateral inhibition can maintain their activity. This is absolutely untrue for the inhibitory striatal spiny neurons. However, their activity can be maintained by continuous input from the prefrontal cortex, which we have seen to have sustained activity during delay period

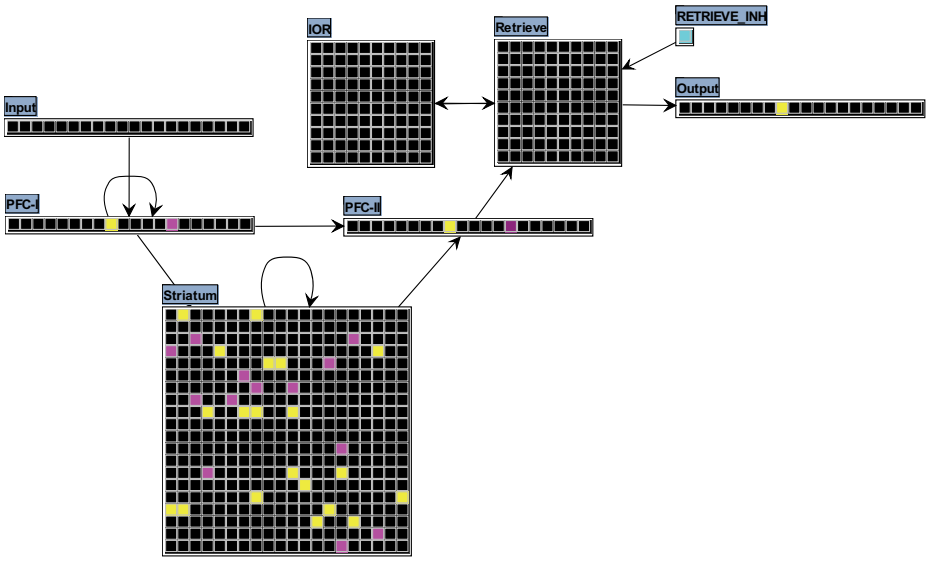


Figure 4.8: The architecture of the Divergent-Reconvergent model

of working memory (Fuster et al., 1982, 1985; Fuster, 1997; Chafee & Goldman-Rakic, 1998). In addition, this model can also coherently explain some mysterious features of the loops, which in turn provides additional support to the model. The discussion can be found later in Section 4.3.7.

4.3.3 Network architecture and dynamics

The goal of the first model (Figure 4.8) is to verify the concept and observe its basic characteristics in immediate serial recall. This model uses localist representations for items. Noises and interferences in the model are removed, and collectively represented by a single noisy input to the retrieval process. This model embodies one cortico-basal ganglionic loop that involves the DLPFC working memory area. It is embodied by the three layers: PFC I, Striatum, and PFC II. Two intermediate structures from the Striatum to the PFC II – the globus pallidus (GPi) and the thalamus – are omitted, because they can be abstractly modeled by an excitatory projection.

Input layer. The input layer receives input to the network. It represents any cortical area which sends itemized information to the prefrontal cortex working memory area. This layer contains N_{item} units, each representing a distinct item. Presentation of an item activates a unit for one time step.

PFC I layer. This layer actively maintains input items. Each of the N_{item}

units receives input from one unit in the input layer (one-to-one connections). The short-term memory property of this layer is implemented by exponential decay, with a decay parameter γ for all units. The rate of decay in this layer is to be determined by parameter fitting against human data.

Striatum layer. This layer is the center of serial order encoding as explained above. Units have strong lateral inhibition on neighbors. Each of the $N_{striatum}$ units arranged in a square layer receives excitatory input from exactly one PFC I unit, and has lateral inhibition on $\eta = 4$ neighboring units (except at the borders). The assumption that each layer receives input from one PFC I unit is a simplification. In the real striatum, there are neurons which receive input from multiple PFC neurons. However, this type of neuron is common in the striatum (Aldridge & Berridge, 1998), and they are the only type of neurons needed for a basic serial order encoding mechanism. The hypothesized serial order encoding mechanism, possibly obscured in the cortico-basal ganglionic loops, is more readily understood and evaluated when this subset of neurons are singled out.

PFC II layer. This layer receives reverberating neural activity from the striatum via the thalamus (the thalamus itself is omitted). There are two points worth noting. First, this layer, though technically separated from PFC I, models the same area as PFC I. It corresponds to the idea that the basal ganglionic loop feeds back to the same area as it receives input from. However, this layer represents a different layer in the 6-layer cortical organization. This separation is a simplistic way to help the model stabilize the reverberating neural activity from the PFC-BG positive feedback loop. The brain also faces the same problem, and may overcome it by a similar laminar separation (efferent neurons to the BG are in Layer V, and afferent neurons are in Layer IV). But the brain may also use other mechanisms, such as inhibitory interneurons (Houghton, 1994). This model can be seen as a simple approximation of a complex stabilization mechanism. Based on the commonality of location, PFC I is assumed to topographically project to PFC II (one-to-one connections with small weights $w = .01$).

Retrieval layer. This winner-take-all layer represents the competitive selection process. The winner-take-all function is technically modeled by a WTA filter (Section 3.3.1). After an item is retrieved, its activity propagates from this layer to the Output layer. At the same time, this activity propagates topographically to the IOR layer, which inhibits the retrieved item. During list presentation, this layer is inhibited by the Recall Inhibitor, which is deactivated to allow recall after list presentation. The

size of this layer is not crucial as long it can hold representations for all items. In this simulation, 100 units are used. If a topographical relation between this layer and the PFC II layer is assumed, it can also be modeled by a layer with N_{item} units.

IOR layer. This layer is the substrate of inhibition-of-return (IOR). It has a short-term memory for retrieved items, and inhibits them at the Retrieval layer (topographical projection). Because gradual decay of the inhibition is not found in some experiments (Duncan & Lewandowsky, 2005), the decay parameter of units in this layer is simply set to 0. The connection weights from Retrieval layer to this layer are set to $w = 1$ and the backward inhibitory weights are set to $w = -10$.

Output layer. This is layer can be seen as representations of motor programs responsible for actualizing retrieved items. It receives input from the Retrieval layer (one-to-one), and can be seen as activating a downstream sequence production system to execute movements required to produce an item.

4.3.4 Training

This model requires a minimal amount of training, and it can be done with only Hebbian learning. During the training phase, the network only receives individual items but not sequences. Each item is only presented in isolation 3 times. When an item is presented, the network does the following learning. In the PFC II layer, the representation unit for this item learns the Striatum pattern. A unit in the Retrieval layer detects this PFC II activity, forming a representation of the item in the Retrieval layer, which is in turn, associated with the active motor unit in the Output layer. The total number of trials is small: 20 (items) $\times$ 10 (passes) = 200 trials.

The following explains the parameters used in training. Each training trial lasts for 5000 ms (50 time steps). The learning algorithm is normalized Hebbian learning in the PNN framework. They are applied to the three layers: PFC II, Retrieval and Output. These weights are initialized to small random values and are bounded within $[0, 1]$. The learning rate is set fairly high: $\lambda^+ = .1$, and $\lambda^- = -.1$. The negative learning rate λ^- for desynchronized units can prevent undesired associations between different items. For example, when two items A and B are simultaneously stored in the Striatum and PFC II (normally happens in sequential memory), the neuron representing A in PFC II may be wrongly associated with both A and B in the Striatum, due to Hebbian learning. One way of technically solving this problem is to turn off learning during the testing phase.

Another way, which is more biologically plausible, is to assume that different items are desynchronized, so that a spike-timing-dependent Hebbian rule won't associate them. The latter is represented by the negative parameter λ^- in the Hebbian learning rule in PNN. The desynchronization of items is done by a lateral inhibition in PFC I.

A special treatment is taken on the lateral inhibition in the Striatum. It is observed that when a Striatum unit inhibits a neighbor which happens to encode the same item, the mutual inhibition between the two units causes oscillation, when the item is activated. This adds a source of noise to the system. In this basic model, it is assumed that such within-item inhibition can be removed by Hebbian learning based on their simultaneous activation. This results in a system which is more deterministic and easier to understand. The perturbing effect of the such oscillations is, instead, collectively modeled by a random noise input to the Retrieval layer.

At the end of training, every item should reliably activate one PFC II unit, one Retrieval unit and one IOR unit. This completes the association process that links different components of the system together. It is notable that although the network is not trained to memorize sequences explicitly, this ability automatically emerges after it has learned to recognize items in isolation.

4.3.5 Simulation 1: Basic serial recall

This simulation tests the basic capability of the model to encode and retrieve sequences. Lists are presented to the model, followed by immediate serial recall after presentation. During the testing phase, no further learning occurs. Immediate serial recall is a consequence of the network's architecture, once it has learned to recognize items in isolation.

The item set used in this simulation is $N_{item} = 20$ symbols, each represented by one distinct unit in the Input layer. The Striatum layer of the network has $N_{striatum} = 3600$ (60×60) units. The dimensions of the Retrieval and IOR layers are 10×10. To introduce some errors, Gaussian noises are injected into the inputs of units in the Retrieval layer. The noise has a zero-mean, and a standard deviation σ.

Materials are lists of 1 to 10 items, randomly drawn the set of 20 items without repetition. For each length, there are 800 lists. This simulation does not address the problem of item similarity. Items containing multiple features and different degrees of similarity will be studied in Simulation 2 (Section 4.3.6).

Simulation results show clear list length effects (Figure 4.9), the primacy and recency effects (Figure 4.10), and the correct characteristics of transposition errors (Figure 4.11). In the model, list effects occur because 1) for longer lists, there are more active items competing for output at each point during recall; this results in higher rates of error; 2) late items in longer lists are represented by fewer striatum neurons, and the lower activity also causes higher error rates under noises. Figure 4.9 shows list length curves under different noise levels σ. It can be seen that this model qualitatively replicates the non-linear S-shape characteristics of humans (Crannell & Parrish, 1957). Quantitatively fitting a neural network to human data is generally difficult, but it can be done with an analytical version of the Divergent-Reconvergent model (Section 4.4).

The *primacy* effect occurs in this neural network model because early items are more active than late items. The higher activation levels, and greater distances between competing items may these items more distinctive albeit random noises. The *recency* effect however occurs simply from a fact that the final item has fewer transposition errors than the penultimate item, because it can only be transposed with earlier items but not later ones. This effect surfaces statistically, but it does not reflect all factors which may contribute to it, such as the phonological coding or auditory coding of the final item (Crowder & Morton, 1969; Beaman & Morton, 2000), and the higher activation of recent items due to less decay. More detailed study on the recency effect, and its interactions with modality, articulatory suppression and the suffix effect are discussed with fuller model (Chapter 5).

The upper panel of Figure 4.10 shows the transposition gradients produced by the Divergent-Reconvergent model for each output position (the noise level σ = .03). The six points in each gradient indicate the proportion of responses from input positions 1 to 6, from left to right. The peaks of the transposition gradient are the correct responses, and the slopes indicate decreasing numbers of transpositions as the transposition distance increases e.g., Estes (1972). However, this model, unlike some associative models, does not require directly coded position information to account for the observed transposition gradients. These gradients can emerge from the competitive selection nature of remembered items. If an item is not retrieved at the right position, it tends to be retrieved in a close position after it, because its activation level is higher than remaining items. This explanation is shared by the Primacy model (Page & Norris, 1998). Real errors due to position coding and associations are discussed in Section 5.5.1.

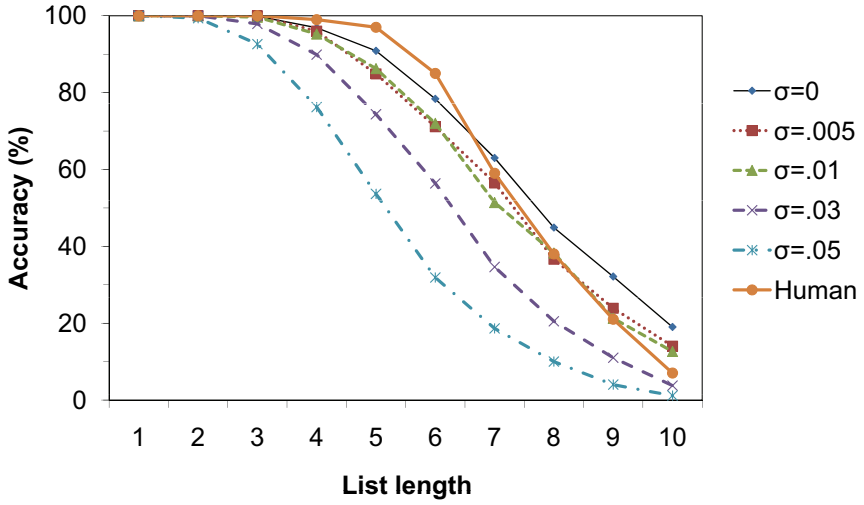


Figure 4.9: List length effect in the Divergent-Reconvergent model under the influences of different noise levels σ. The human data are replotted from R. N. A. Henson (1998), which are averaged across the limited and unlimited letter conditions of Crannell and Parrish (1957); see text for details.

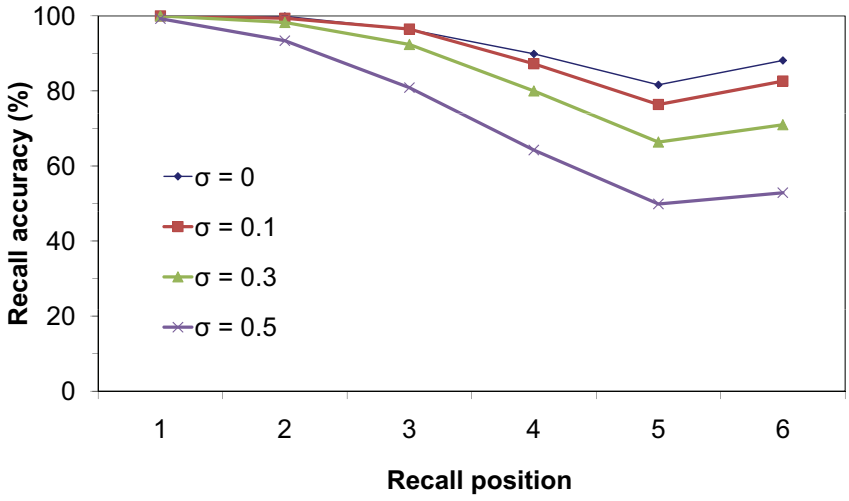


Figure 4.10: Serial position curves in the Divergent-Reconvergent model

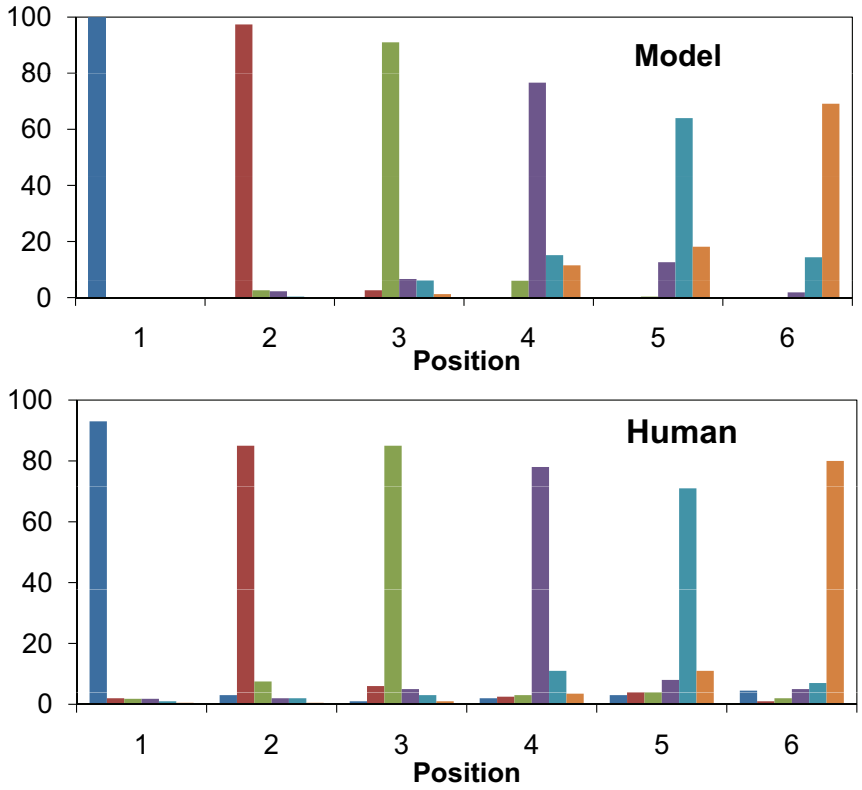


Figure 4.11: Transposition gradient in the Divergent-Reconvergent model

The list-length effect, the primacy and recency effect, and the transposition gradients support the Divergent-Reconvergent model as a plausible model for working memory.

4.3.6 Simulation 2: Exploring similarity effects

Typical working memory models based on competitive queuing (CQ) and activation gradients adopt localist representation for the items (Page & Norris, 1998; Burgess & Hitch, 1999). Because localist representations do not represent item similarity, these models have used an additional process to account for the similarity effects found in both the auditory and the visual modalities (Conrad, 1964; Logie, Della Sala, Wynn, & Baddeley, 2000; McNeil & Johnston, 2004). The common practice in working memory is to assume a two-stage retrieval process. In the first stage, an item is selected, and then it is passed to a second stage where it may be replaced by a similar item (R. N. A. Henson, 1998; Page & Norris, 1998; Burgess & Hitch, 1999).

In this neural network model, however, it is possible to take advantage of distributed representations to represent similarity between items. A first issue concerning the representations of items is whether their features are represented in the PFC, or are only represented in the sensory cortex. The former choice represents a symbolic view of items in the working memory, where the item representation is devoid of sensory content itself but can serve as a reference for its content. Similarity effect can be seen as arising when the item-level representation interacts with sensory representations outside the main serial order mechanism. Alternatively, it may be assumed that items in the PFC working memory system are inherently represented by features. The similarity effect can arise within the serial order mechanism itself. The latter possibility is studied in this simulation.

Specifically, two questions are to be answered. The first is whether lists with confusable items are more difficult to retrieve. The second is whether the current architecture is adequate to account for the following phenomenon. In lists containing phonologically confusable items (e.g., BHPYX), is this pair of confusable items is more likely to be transposed with each other, while other items are kept in their correct positions? This effect is a typical finding in lists containing phonologically confusable items (Baddeley, 1968).

In this simulation, a new set of 20 items are used, each containing 5 features. Each of the first 16 items contains 5 completely distinct features (Items 1-16). In addition, there are two pairs of similar items: a pair sharing 1 feature (labeled A and B), and a pair sharing 2 features (labeled X and Y). This item set requires 97 different features. The materials are 3 types of lists, 800 lists for each type. The Dissim type contains 5 distinct items. The Sim1 and Sim2 types contain 3 distinct items plus either of the above two pair of similar items, respectively.

A few modifications to the neural network model are made to accommodate the new changes. The sizes of the Input, PFC I, II layers are changed to the number of features: $N_{features} = 97$. The Striatum layer is larger in order to represent more features: $N_{striatum} = 6400$ (80×80). The sizes for the Retrieval and IOR layers are still 100, and the size of the Output layer is still 20 representing 20 items.

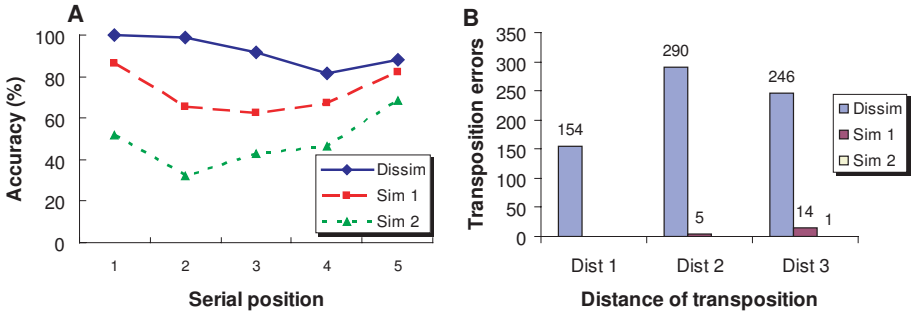


Figure 4.12: Similarity effects on accuracy and transposition error distributions

4.3.6.1 Similarity effect on list accuracies

Figure 4.12A shows the serial position curves of the 3 types of lists, where a clear similarity effect can be observed. Lists with more higher confusability induce more recall errors. From Panel B, it can be seen that more transpositions errors have occurred, and the distances by which items are displaced from their original positions are longer in confusable lists. The effect occurs because activation levels of items sharing features are less distinctive than those not sharing features: the shared features cannot contribute a difference to the items' activation levels. This property is consistent with the fact that people have greater difficulties recalling lists with confusable items than lists with non-confusable ones (Baddeley, 1968; McNeil & Johnston, 2004). Therefore, the answer to the first question is *yes*.

The second question is, is the pair of confusable items more likely to transpose with each other than with other items? In human experiments, this phenomenon has commonly been found in phonologically confusable letters. These letters tend to transpose with each other, while keeping non-confusable letters in the right positions (Baddeley, 1968). Surprisingly, this property is not found in this model. When a pair of confusable items is included, their transposition was found in a few instances: for example, "Y 6 1 X 5" was occasionally recalled as "X 6 1 Y 5". However, this type of error is accompanied by increased transposition errors between the confusable items and non-confusable items. Thus, the characteristic "saw-tooth" pattern, in which confusable items have common transposition errors while non-confusable items are intact, did not emerge in this model.

4.3.6.2 Discussion

What does the result suggest? Does it immediately mean that the model is wrong? Several considerations are given below. First, this serial order model is partially motivated by the anatomy of cortico-basal ganglionic loops. As discussed before, this mechanism may apply to different types of elements, including motoric, perceptual, conceptual and linguistic elements. Though it is true that phonologically encoded lists exhibit obvious sawtooth-like serial position curves, when the lists contain alternatively confusable and non-confusable items, this phenomenon is not necessarily true of lists containing similarity in other domains, such as vision and semantics. Traditionally, visual similarity was not believed to affect serial recall, because working memory encoding was thought to be phonological. However, recent studies have changed this opinion by showing clear visual similarity effects when phonological coding is reduced by articulatory suppression (Logie et al., 2000). In Logie et al.'s study, visually similar items are found to worsen list recall. However, no evidence of the sawtooth-pattern in visually similar lists has been reported. In contrast, semantic similarity has a different pattern, instead of being confusable, it is found to have a facilitative effect on *item* recall, and no detrimental effect on order recall (Saint-Aubin & Poirier, 1999). Therefore, the similarity effect seems to be modality dependent, and thus, it appears to be acceptable for this model, not to commit to any specific type. This model may correctly not exhibit this effect by itself.

Further analysis of the similarity-based difficulty reveals the same issue as repeated items. That is, repeated activation of a feature does not induce a new representation. This difficulty is an intrinsic one for competitive queuing (CQ) and arguably also for the brain. The brain can certainly recall lists with repeated items, but the performance is known to be worse than lists without repeated items (Jahnke & Bower, 1986). However, the brain's solution to the repetition problem appears to be non-trivial (consider the binding error of repeated items, such as 233 recalled as 223, or 858 as 585). This issue is left for the Repetition Encoding and Chunking model (discussed in Section 5.5.1). The problem of accounting for the sawtooth pattern in addressed in Chapter 5.

4.3.7 Mysteries of the basal ganglia

The Divergent-Reconvergent model provides a functional interpretation of the cortico-basal ganglionic loop. Based on the new functional considerations, it explains

some unusual physiological characteristics of the basal ganglia. These seemingly unrelated physiological characteristics, in turn, provide independent support for the correctness of the model.

4.3.7.1 Inhibition, inhibition, inhibition

Approximately 80% of the striato-pallidal (from the striatum to the GP) and pallido-thalamic neurons (from the GP to the thalamus) are GABAergic. This prevalence of inhibition is uncommon in the central nervous system. There haven't been many satisfactory explanations for this phenomenon. However, the Divergent-Reconvergent model may offer a striking explanation: the prevalence of inhibition is necessary for the brain to perform online sequencing, including perceptual, cognitive, linguistic and movement sequences. According to the Divergent-Reconvergent model, the central mechanism of sequencing is inhibition from early items to late items, but not the reverse. This requires *strong* inhibition that turns neurons off from future activation. Lateral inhibition in the cortex itself is too weak for this purpose. Therefore, the brain would need an area where neurons have stronger lateral inhibition for sequencing. This area, according to this theory, is the striatum. This explains why the striato-pallidal neurons are inhibitory. Then, why are pallido-thalamic neurons also inhibitory? What is the use of this double-inhibition connectivity pattern? Several interesting propositions have been proposed by researchers (Frank et al., 2001; Berns & Sejnowski, 1998). The Divergent-Reconvergent model also provides an answer. The striatal output encodes the strengths of serially ordered items. Early items have stronger output than late ones. However, because this output is inhibitory, it cannot be selected by a winner-take-all competitive process. As a way to fix the problem, it can be first converted to a comparable amount of excitatory output. This conversion can be perfectly done by the GP and the thalamus. The thalamus has tonic excitatory output, which is tonically inhibited by the GP. Now, the GP, to the amount that it is inhibited by the striatum, will release the thalamus let it generate a comparable amount of excitatory output. In this pathway, inhibition is converted to excitation proportional to the strength of the inhibition. The above theory coherently explains the prevalence of inhibitory neurons in the striatum, and the functions of the double-inhibition pathway. This does not exclude other theories, because a brain structure likely has multiple functions.

4.3.7.2 UPs and DOWNs

Another mystery explained by this model is the unusual property that the striatal spiny neurons have two states: an UP state in which the neuron is readily activated by cortical input, and a DOWN state in which it is extremely difficult to activate (C. J. Wilson, 1995). The Divergent-Reconvergent model also provides an interesting explanation for this. Functionally speaking, there are two types of striatal neurons, *free* neurons which can be fired by incoming activation, and *off* neurons which are inhibited and won't fire upon activation. The UP and DOWN states of the striatum neurons can make this mechanism work more robustly. The two states of neurons may improve the stability of the serial order mechanism.

Another notable physiological characteristic of the basal ganglia is its high concentration of acetylcholine (ACh). The striatum has the highest concentration of ACh in the brain for unclear reasons. However, it has been indicated that a major function of ACh is to stabilize the activity state of the spiny neurons (Di Chiara, Morelli, & Consolo, 1994). In other words, ACh helps keep spiny neurons active when they are already active, and helps keep them inactive when they are inactive. This physiological property also increases the stability of the serial order mechanism.

The model can even explain why the brain has multiple specialized mechanisms to increase the stability of striatal neurons. As mentioned above, the striatal neurons do not have self-recurrent excitatory connections, but its sustained activity is maintained by continuous input from the cortex. The projection distance, and the oscillatory nature of cortical output, means that the continuous input is not very stable. The striatum has to rely partially on its local mechanisms to fill in the gaps. Therefore, it makes perfect sense to have evolved UP and DOWN states, and have demanded the highest concentration of ACh in the whole brain.

These mysterious characteristics of the basal ganglia all appear to have a coherent story in the Divergent-Reconvergent model. It is highly probable that this model has discovered an important neural mechanism that has not been discovered before.

4.4 The mathematical Divergent-Reconvergent model

The neural network version of the Divergent-Reconvergent model has shown a good qualitative fit to human data. Because the performance of a neural network model

is collectively determined by its architecture, parameters (layer sizes and learning rates) and training experiences, quantitatively fitting the model is generally infeasible. But fortunately, the neural mechanism represented by the Divergent-Reconvergent model is clear enough to enable an analytical form of the model, which may be fitted quantitatively. The mathematical form of the Divergent-Reconvergent model is derived and tested in this section.

4.4.1 A mathematical model of the striatum

The striatal activation for an item depends on two parameters: the proportion ρ of striatal neurons activated by one item, and the inhibition factor η (how many neighbors each unit inhibits). In the neural network version of the Divergent-Reconvergent model, the parameters $\rho = 1/N_{item}$ and $\eta = 4$. Let's see if we can use these parameters to calculate the number of neurons representing each item in the sequence, without simulating the whole neural layer.

For the first item, we know that the number of neurons it activates is simply

$$n_1 = \rho N_{striatum}, \tag{4.1}$$

and the number of neurons they inhibit is $\eta\rho N_{striatum}$, occupying a proportion h_1 of the whole layer:

$$h_1 = \frac{\eta\rho N_{striatum}}{N_{striatum}} = \eta\rho.$$

For the second item, we know that some proportion of its neurons are inhibited by the first item, this proportion is the same h_1 if items are homogeneously distributed in the Striatum. Thus, the number of *free* neurons that can be activated by the second item is

$$n_2 = \rho N_{striatum}(1 - \eta\rho). \tag{4.2}$$

The number of neurons activated by other items can be similarly calculated. Comparing the results (4.1) and (4.2), we can see that they are all relative to the size $N_{striatum}$, thus a better way of representing these quantities is using proportions of the Striatum layer, that is, making a transform: $A_i \equiv n_i/N_{striatum}$. The model A_i is a dimensionless model, regardless of the actual size of the Striatum layer. It can be used to replace the Striatum layer in the Divergent-Reconvergent model, and simulate an

infinite number of neurons! The formula for A_i, the proportion of units representing the i^{th} item, is given below:

$$A_i = \begin{cases} \rho & i = 1 \\ \rho \prod_{j=1}^{i-1} [1 - \eta A_j] & i > 1, \end{cases} \tag{4.3}$$

Using this model, the bottleneck in the Divergent-Reconvergent model, the large Striatum layer, can be reduced to simple calculations. It also allows for parameter fitting because the parameters ρ and η can be easily changed, without changing the architecture of the neural network.

The analytical model is simulated by substituting a mathematical Striatum layer into the Divergent-Reconvergent model. The mathematical Striatum layer has N_{item} units. The activation of each unit a_k can be conceptually seen as

$$a_k = \max(A_i, x_k),$$

where x_k is the net input from the PFC I layer, and A_i is the activation limit calculated based on the proportion of *free* neurons. This model provides a good approximation of the Striatum layer in the Divergent-Reconvergent model. Their performances are compared later.

4.4.2 An analytical form of the activation gradient

The activation gradient A_i specified as in equation (4.3) is defined recursively. It is not easy to see what property it has. However, in fact, this activation gradient has an interesting and simple form, shown below. Taking the ratio A_{n+1}/A_n, we have

$$\frac{A_{n+1}}{A_n} = \frac{\rho \prod_{i=1}^{n} (1 - \eta A_i)}{\rho \prod_{i=1}^{n-1} (1 - \eta A_i)} = 1 - \eta A_n.$$

Multiplying A_n to the RHS, we get an iterative form of the activation gradient

$$A_{n+1} = A_n(1 - \eta A_n).$$

Then by rearranging the terms, we have

$$A_{n+1} - A_n = -\eta A_n^2. \tag{4.4}$$

This is a simple difference equation on A_n. It can be approximated by a differential equation:

$$\frac{d\tilde{A}}{dn} = -\eta\tilde{A}^2, \quad \tilde{A}(1) = \rho.$$

The solution, which can be used to predict the activation gradient without iterations, is

$$\tilde{A}(n) = \frac{\rho}{1 + \rho\eta(n-1)}. \tag{4.5}$$

The activation gradients generated in the Divergent-Reconvergent model (pure neural network version), the mathematical model of the Striatum based on (4.3), and the approximate function (4.5) are shown in Figure 4.13. The parameters used in this demonstration are: $\rho = 1/20$, and $\eta = 4$. It can be clearly seen that the mathematical model of the Striatum fits the neural network model very well. Because the recall process depends on the Striatum layer only by the activation gradient it generates, this mathematical Striatum layer (20 units) has exactly the same effect as the neural network model with 3600 (60×60) units. This mathematical model is used to replace the Striatum layer in later Divergent-Reconvergent models in this work.

In the successful Primacy model (Page & Norris, 1998), the activation gradient is generated by a linear function, since no mechanistic theory was available to inform about its nature. In this work, with the probable neural substrate identified, we can better predict the nature of the activation gradient, and describe it with a potentially more accurate function. The function (4.5) is a good description of the mathematical nature of the activation gradient generated in the striatum, based on the proposed theory.

4.4.3 Simulation 3: Quantitative fit of human data

After replacing the Striatum layer in the Divergent-Reconvergent model with the mathematical model, we obtain a new implementation of the model (Figure 4.14). The mathematical version helps verification of the Divergent-Reconvergent model, since it allows for a quantitative fit to human data by optimizing parameters. One addition to this model is a recall control unit (REC_CTRL), which is used to temporarily pause the retrieval process after an item has been retrieved. This models the time for an item to be output. In later simulations, it will be used to capture the effect of articulatory speed. In the parameter estimation process below, it is simply used to add a 200 ms delay after each retrieval to simulate the articulation of a digit or a letter.

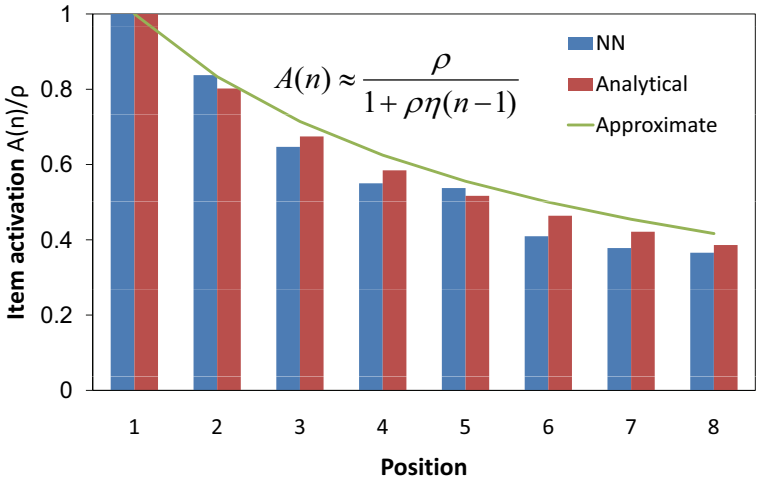


Figure 4.13: Activation gradients in the Divergent-Reconvergent model

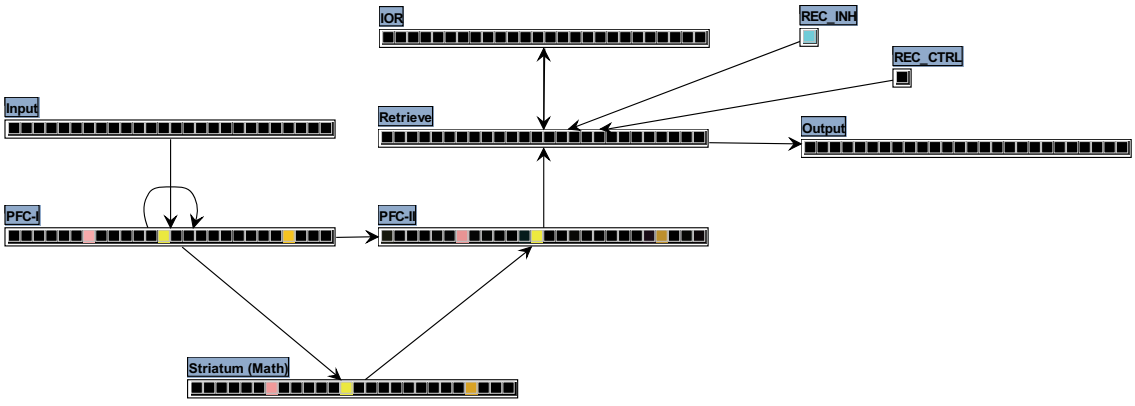


Figure 4.14: The architecture of the mathematical Divergent-Reconvergent model

The goal of this simulation is to better support the model as a valid model for immediate serial recall, by *quantitatively* fitting human data. The five parameters of the Divergent-Reconvergent model are shown in Table 4.1. The two parameters ρ and η determine the shape of the activation curve. The noise level σ determines the level of random noises added in the Retrieval layer. The activation timescale T determines how long short-term memory traces last. Specifically, it describes the time for PFC I activity to decay to the threshold level θ. Items whose activations are below the threshold θ cannot be retrieved. For each parameter, a range of possible values are determined by pilot experiments as shown in Table 4.1. The optimization process considers about 4 - 6 levels of values for each parameter. This results in a search grid of about 10,000 combinations. For faster fitting, the parameters are separated in two groups (ρ, η, σ) and (T,θ), and are fit alternately for 3 cycles. This usually results in a good fit. The goal here is not to fit human data as closely as possible (because they have noises), but to show that the model can reproduce them within reasonable ranges.

Human data are selected to represent typical characteristics found in human immediate serial recall tasks, including a list-length curve (Crannell & Parrish, 1957), and a serial position curve (R. N. A. Henson, 1986). The two curves are fitted independently. The best parameters for both data sets, which happen to be the same, are: $\rho = .04$, $\eta = 4$, $\sigma = .04$, $T = 90$ sec, and $\theta = .4$. Figure 4.4.3 shows the result of fitting list-length effect, and Figure 4.4.3 shows the result of fitting the serial position effects. Both results suggest a good fit of the Divergent-Reconvergent model to the human data.

A potential problem, however, is that it cannot be ascertained that all the serial recall performance can be attributed to the neural substrate of the Divergent-Reconvergent model. The neural substrate, the prefronto-basal ganglionic loop, is only one component in the working memory system. According to Baddeley's model, there is at least one important component – the phonological store – which is crucial for serial recall. By fitting human data without this important component, it is likely that the capability of the model and its neural substrate is somehow exaggerated. To better constrain this neural substrate, special measures are taken in the next section.

4.4.4 Parameter estimation

One goal of data fitting is to verify whether a computational model can quantitatively reproduce human data, as is achieved above. Another goal is to use the human

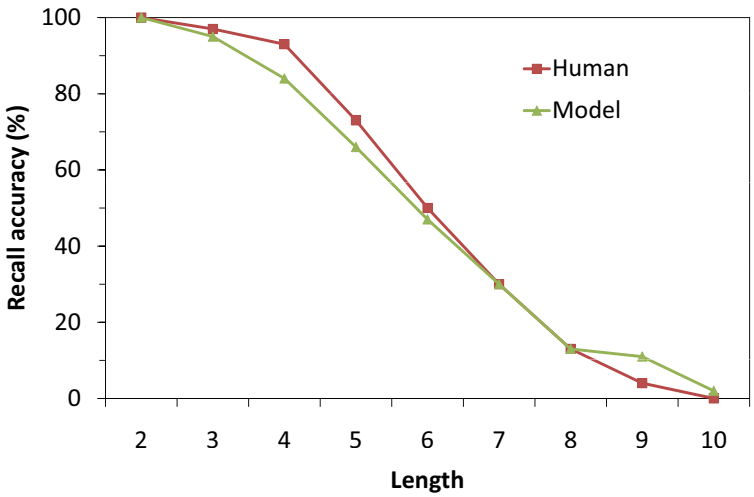


Figure 4.15: List-length effect in the DR model (parameters optimized). The human data are replotted from R. N. A. Henson (1998), which are averaged across the limited and unlimited letter conditions of Crannell and Parrish (1957); see text for details.

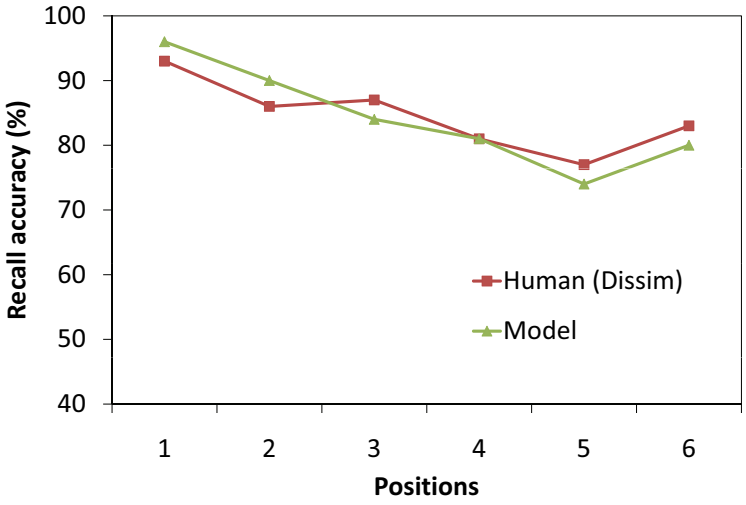


Figure 4.16: Serial position effects in the DR model (parameters optimized). The human data are from the recall of visually presented nonconfusable letters ('Human', adapted from Exp. 1, R. N. A. Henson, 1986); see text for details.

Table 4.1: Parameters in the Divergent-Reconvergent model

Param.	Description	Search range	Adopted values
ρ	Divergent density	.03 - .06	.04
η	Lateral inhibition factor	2 - 4	2.5
σ	Std. Dev. of Gaussian noise	.03 - .08	.06
T	Activation timescale	60 - 120 sec	90 sec
θ	Retrieve threshold	.1 - .4	25%

data to reduce the degrees of freedom of the model, so that it can 1) best predict human data in new experiments, and 2) correctly describe the capability of its neural substrate. The second objective, however, is not easy to achieve. This is because there are multiple working memory components that simultaneously contribute to the performance of any working memory tasks. In immediate serial recall, for instance, at least three components can be involved under some conditions: the item-level mechanism (in the prefronto-basal ganglionic loop), the phonological store (in the parietal lobe) (Baddeley, 2003), and the auditory sensory memory (in the temporal lobe) (Buchsbaum & D'Esposito, 2008).

How can we define human data which best characterize the neural substrate of the Divergent-Reconvergent model? A rationale used here is assuming when the brain can use multiple memory components for retrieval, it can learn to choose the *best* combination or them. Unlike many working memory models, phonological coding is not taken as "confusing" and detrimental to an otherwise good recall. Rather, it is taken to be "improving" recall of a weaker component, if phonological coding is distinctive. Using this rationale, the parameters of the Divergent-Reconvergent model should be fitted against data with minimal phonological coding. The ideal choice would be visually presented lists with articulatory suppression. In practice, it turns out to be similar to using visually presented items that are phonologically confusing (a serial position curve with 6 data points) (R. N. A. Henson, 1986), and visually presented long lists (a serial position curve of 8 data points) (Page & Norris, 1998). Both conditions have to some degree reduced the effectiveness of phonological coding, though it is not totally eliminated. After fitting the model to the 14 data points, its digit span[1] is measured. It is found that the digit span 4.23 falls in the range of human digits spans 4-5 under articulation suppression (Chincotta

[1]The digit span is measured as follows. Lists with increasing lengths are presented and recalled. Each length is tested with two sequences. The digit span is operationalized as the last correctly recall length, after the model (or human) makes two consecutive errors (Chincotta & Hoosain, 1995).

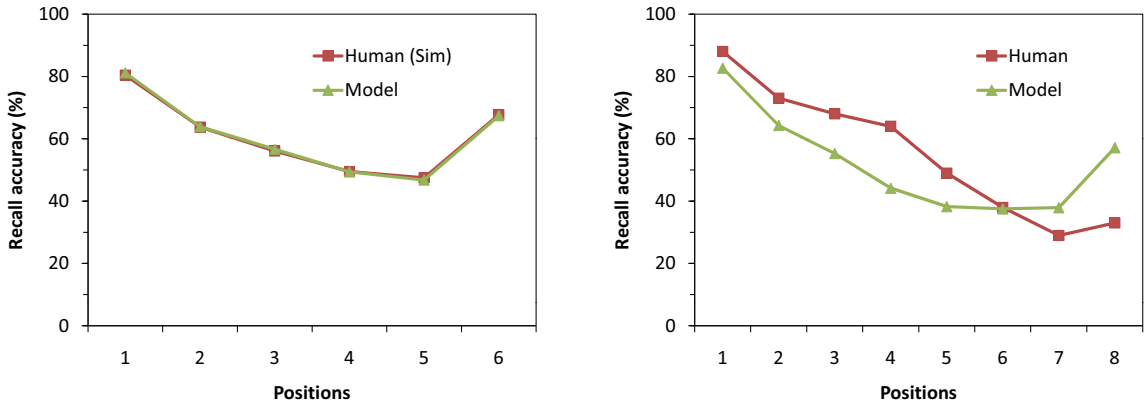


Figure 4.17: Serial position effect data for parameter estimation

& Hoosain, 1995). This suggests the model has captured the amount of contribution of its underlying neural substrate, but does not account for the portion contributed by the phonological store. These set of parameters are adopted for the Divergent-Reconvergent model as shown in Table 4.1.

The distance from the performance to the full capacity of working memory with phonological coding allows for a smooth incorporation of a phonological store. A phonological store can be seen as providing better memory performance by the intrinsic serial nature of the phonological coding. This treatment can avoid a puzzling question as in a two-stage model. A two stage model often assumes that a correctly retrieved item must go through a second stage, in which it is replaced by a wrong phonologically similar item. This model does not need to assume a two-stage process. Further, by situating the phonological store differently, it can account for some additional data, such as the lack of phonological effect in nonwords, discussed in the next chapter.

4.5 General Discussion

4.5.1 Processing limits of the Divergent-Reconvergent model

A foreseeable limitation of the Divergent-Reconvergent model is that it provides an effective way of sequencing items belonging to the same category (such as letters, words, nouns and verbs), but it may have more difficulty in sequencing items from different categories. This is because representations of items in different categories may be located in very distant brain areas. It is known that cortico-basal ganglionic loops

involving different brain areas are anatomically separate, and do not have rich lateral inhibition required by this serial order mechanism. How can the sequencing of these items be explained? At first, this prediction of sequencing difficulty may be true of the brain. The lack of lateral inhibition across different brain areas explains why short-term memory requires a centralized mechanism in the dorsolateral prefrontal cortex (DLPFC) and makes use of the fronto-basal ganglionic loop (Owen, 1997). In this way, information from different modalities may be sequenced by the same cortico-basal ganglionic loop. This is not to say, however, that any items represented in the same loop can be well sequenced. It still depends on how close these representations are. In the author's opinion, the Divergent-Reconvergent model best matches the processing in one module of the striatum (Graybiel et al., 1994), a part of one of the five macroscopic loops proposed by Alexander et al. (1986). The lack of lateral inhibition between modules (e.g., foot and hand) can very well explain why mixed foot and hand movements are harder to sequence than foot or hand movements alone.

This analysis has a crucial implication on language processing. That is, it is very likely that the brain has a difficulty in sequencing words from different categories, such as verbs and nouns. Neuropsychological studies suggest that nouns are represented in the anterior *temporal* regions, but verb are represented in the left inferior *frontal* regions (Goodglass, Klein, Carey, & Jones, 1966; Damasio & Tranel, 1993; Daniele, Giustolisi, Silveri, Colosimo, & Gainotti, 1994). Words from the two classes are probably not represented in the same cortico-basal ganglionic loop. Even considering their centralized prefrontal representations, these word classes are too big to allow effective all-to-all lateral inhibitions. This opens a discussion on sequencing beyond the limits of the Divergent-Reconvergent model. I propose three possible solutions of this issue:

1. One possibility is that each word is represented by a set of features. Each feature belongs to a smaller feature set, where it can be sequenced against other features in the same set. With such distributed representations, the serial order of the words can be indirectly encoded by sequencing their features. This feature-based model, however, still cannot sequence words which do not share features. For the basic Divergent-Reconvergent model, the author thinks, this is an ultimate limitation. If words do not share any features that can be sequenced by lateral inhibition, they cannot be sequenced by this model.

2. An alternative neural mechanism for cross-category sequencing is phonological cod-

ing. If these items are coded in a phonological form, their serial order can be maintained by the inherent sequential nature of phonological codes. A drawback of this mechanism is that phonological information quickly decays if it is not rehearsed in about 2 seconds (Baddeley, 1986).

3. Yet another plausible mechanism which can circumvent this problem is a superimposed category sequence over the word sequence. In this theory, items of different categories are separately sequenced in different Divergent-Reconvergent models. Over these models, there is a category-label sequencing mechanism which remembers the sequence of category labels. The remembered category-label sequence can be used to select the correct Divergent-Reconvergent models from which the next word can be retrieved. By hierarchically combining the category-label model, and the category-specific models, sequences containing words from arbitrary classes can be encoded and retrieved.

The last solution, in my opinion, is the most appealing. There is evidence that serial order retrieval is based on categories. For example, in recall errors of nonwords, consonants are most frequently transposed with other consonants, and vowels are most frequently transposed with other vowels (Hartley & Houghton, 1996). A model which explains the above category-based errors assumes syllable templates, such as CVC, which bias retrieval to the required phonemic categories. The mechanism can be seen as a special case of the category-label memory component. A crucial issue faced by the category-label component is, however, not that the category is too big, but it is too small. Such sequences inevitably contain many repeated elements. The basic Divergent-Reconvergent model has zero-tolerance on repeated items. It indubitably loses the representation of all but the first instance. Solutions to this problem are discussed in Section 5.5.1. Theoretical implications of this mechanism on the syntax of natural languages are discussed in Chapter 6.

4.5.2 Corticostriatal projections and long-term learning

An assumption being kept so far in the Divergent-Reconvergent model is that each item activates a comparable number of neurons in the striatum. This is an ideal condition for the model to work: all items start with equal preferences for output, and their only differences come from their presentation order. In some conditions, however, the equality of item strengths in the striatum cannot be assumed, but needs to be

adjusted. The first situation is when some items have stronger cortical activity (e.g., due to saliency or rich features), their striatal representations are not equal to other items. The different "prior" strengths may greatly degrade the accuracy of serial order encoding. There must be a way with which the brain can fix this problem.

Another situation is when some prior ordering has to be learned by the subject. For example, in animal studies of serial order, animals are often trained to respond to simultaneously presented items in a predefined order. The animal has to learn what the order is by trial-and-error. The task, known as *simultaneous chaining*, also requires different "prior" ordering of items. How can the Divergent-Reconvergent model be applied to these situations? The key to the problem is to recognize that the model needs to learn the strengths of corticostriatal projections, based on feedback on its performance. Such reinforcement learning is known to be a main function of the striatum. It is known that the midbrain dopaminergic (DA) neurons, which encode unexpected rewards (Schultz, Apicella, & Ljungberg, 1993), project heavily to the striatum. Accordingly, it has been believed that the striatum controls and learns sequential behaviors and simple reactions from trials and errors (Barto, 1995; Marsden, 1980; Graybiel et al., 1994). The Divergent-Reconvergent model can fit into this bigger picture of long-term learning in the striatum. The idea is illustrated in Figure (4.18), which represents different modes of learning concerning sequential processing. The Divergent-Reconvergent model (Panel A) requires equal connections from items represented in the cortex and the distributed representations of items in the striatum. Simultaneous chaining, a form of long-term sequence learning, requires learning the serial order by the connections (Panel B). At last, complicated sequential prediction and production processes require learning to map contextual representations to future responses (Panel C). The last mode is best represented by the SRN (Elman, 1990).

4.6 Conclusion

It has been demonstrated with computer simulations that the Divergent - Reconvergent model indeed possesses the capability to encode and decode sequences. It is the first *serial order* short-term memory model which is fully based on known neural mechanisms. It is surprising that this neuron-level model can account for human performance at the behavioral level. Based on the ubiquitous existence of cortico-basal ganglionic loops, the brain may use the serial order mechanism represented by this model

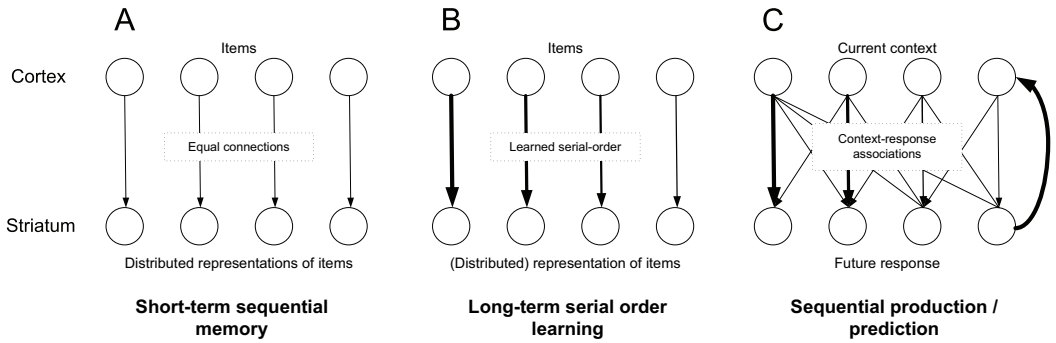


Figure 4.18: Three modes of corticostriatal learning

to encode the serial order of various types of information and movements. This model may provide a fairly satisfactory though primitive answer to the *serial order* problem which has puzzled researchers for decades. By the biological plausibility criteria discussed at the beginning of this chapter, this model can be rated as a *strictly biologically plausible* model.

In summary, the Divergent-Reconvergent model provides a neurally plausible account for the brain's serial order mechanism. However, it has not addressed many important issues, such as the recall of non-words, the mechanisms of temporal grouping, and the encoding and recall of item repetitions. They are addressed in the next two chapters.

Chapter 5

The Dual Representation Model of Phonological Loop

The serial order mechanism studied in the Divergent-Reconvergent model strictly requires that the to-be-remembered lists consist of recognizable items without repetitions. If a list includes a nonword, which cannot be recognized as an item, or the list contains repeated items, the model cannot accurately recall the list. If the model does reflect the limitations of its neural substrate, it would necessitate additional neural mechanisms to remedy these problems. In this chapter, an important component of working memory, the phonological store, is added to the Divergent-Reconvergent model, in order to address the problem of nonwords and repetitions in working memory. It will also be evaluated by its ability to reproduce phonological effects, such as the word length effect, the similarity effect, and the suffix effect. The need of incorporating a fully functioning phonological store into working memory models is also suggested by other researchers (e.g., Burgess & Hitch, 1999). This chapter addresses this challenge.

5.1 The big picture of two interactive components

The basic idea explored in this chapter is that working memory for serial recall contains two main components. The two components are an *item store*, and a *phonological store*. The goal is to computationally simulate the two components and identify a reasonable way in which they interact. At the conceptual level, a dual component structure is common in psychological models, such as Baddeley's (1986) working memory

model, and the dual-coding theory (Paivio, 1986). Both of them postulate that visual and phonological information are processed in different channels but both are used to represent and organize knowledge. However, this does not mean that all different neural processing must involve distinct processing routes. Neural networks with learning can very well integrate two modes of processing in a single route.

The first component of this model, the Item Store (IS), is based on the Divergent-Reconvergent model. It corresponds to the prefronto-basal ganglionic loop. The contents are discrete percepts, as well as other discrete mental elements from various domains including words, objects, concepts and movements. For two main reasons, the contents are not supposed to include analog imagery, acoustic information or phonological information. The first reason is the difficulty for the activation gradient mechanism to represent excessive repetitions. Further, it is also known that phonemic similarity is different in the recall of nonword lists (Lian et al., 2001, 2004). This suggests that phonologically coded nonwords may rely on a different neural mechanism from that of the item level.

The second component of this model is the Phonological Store (PS). The concept is almost the same as Baddeley's phonological store model. The main difference is that this PS model explicitly excludes item-level information. As other computational working memory models, such as Burgess & Hitch's (1999) phonological loop model, this model has a dual-representation structure: an item-level representation and a phonological representation. what's new in this model is that the phonological store is fully functioning, in the sense that it can encode and retrieve the serial order of phonemes, rather than phonemes without orders. This advancement enables direct simulation of the rehearsal processes underlying working memory. With the new phonological store, the Dual Representation model can consistently account for the phonological similarity effect, the word-length effect, the irrelevant speech effect, the auditory suffix effect, and the interactions with modality and articulatory suppression.

A limitation of this model is that it has not implemented the visuospatial sketchpad. Visually presented items are assumed to directly activate item-level representations in the Item Store. Without a visuospatial sketchpad, this model does not directly account for visual similarity effects (Logie et al., 2000) , but a basic form of the visual similarity can be explained as a general property of the Divergent-Reconvergent model (see Section 4.3.6). Further, this model has a formal definition of the Central Executive. It can systematically simulate the control process of multiple interactive components.

5.1.1 The Central Executive

According to Baddeley (1986), the central executive is postulated to be responsible for the selection, initiation, and termination of processing routines (e.g., encoding, storing, retrieving). However, it hasn't been adequately modeled by computational models. In this work, I propose a formulation: the Central Executive is a mapping from Signals to Actions (an S/A mapping). The Signals include *external* signals such as an instruction to recall (or stop, repeat), and *internal* monitoring signals which represent detected state changes in the system (e.g., increased activity of a layer or full loss of activity of a layer). In the PNN framework, each layer, projection or a binary state unit can be optionally inhibited an "OFF" action, or enabled/activated by an "ON" action (see Figure 3.7). Neither external and internal signals have *a priori* meanings to the system, until they are mapped to Actions by the central executive. It turns out that this simple formulation is adequate for all neural network models in this work.

For example, it has been used to control the Retrieval Inhibitor (RETRIEVE_INH, also referred to as the Recall Inhibitor or REC_INH) in the Divergent-Reconvergent model (Chapter 4). To make sure that the Retrieval Inhibitor inhibits the retrieval process during list presentation, and enables recall when needed, the modeler only needs a small program (an S/A mapping), and to insert START and RECALL instruction to the stimulus list. The Central Executive can do the control based on the program:

```
START -> RETRIEVE_INH.ON
RECALL -> RETRIEVE_INH.OFF
```

This control program for the basic Divergent-Reconvergent is trivial, but the Central Executive model turns out to be a powerful way to simulate the coordination of the interactive components in the Dual Representation model. An S/A mapping allows the modeler to conveniently coordinate the model, and clearly display the control protocols used. For the working memory, it also allows for implementation of different rehearsing strategies, which has been a difficulty in working memory modeling. Ideally, the S/A mapping should be the result of learning (especially reinforcement learning). In this work, the learning process is not modeled yet, and the control programs represent the modeler's view on memory processes and interactions, as in other working memory models.

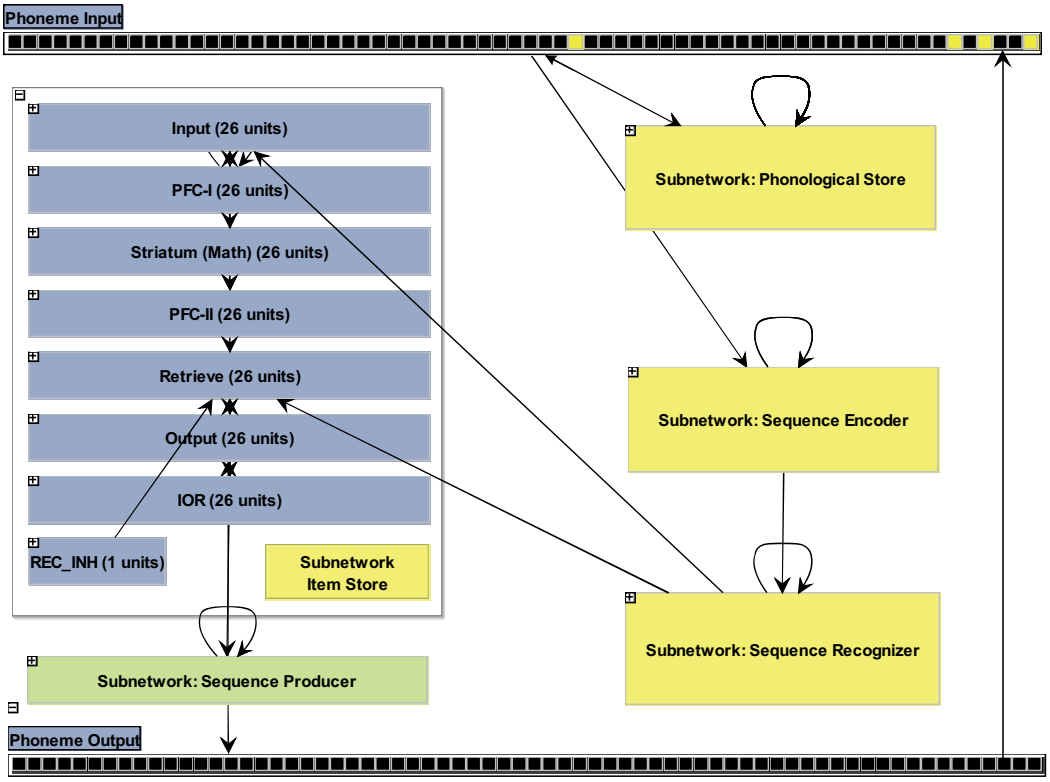


Figure 5.1: The Architecture of the Dual Representation model

5.2 The Dual Representation model

The main goal of this model is to explore a new architecture of the well-known phonological loop model. As discussed above, the new architecture includes an amodal Item Store and a Phonological Store. Its architecture is described below with some detail.

5.2.1 The architecture

The Dual Representation model (Figure 5.1) is an interactive system, implemented as a large-scale neural network. It consists of an amodal item store (represented by the Divergent-Reconvergent model), and a phonological system consisting of the Phonological Store, the Sequence Encoder and the Sequence Recognizer. The phonological store is analogous to a "tape", on which phoneme sequences can be written and retrieved. The Sequence Encoder encodes a sequence into a spatial pattern, and the Sequence Recognizer recognizes the spatial pattern as an item. Detailed implementation of this model is given later in Section 5.3. The way that the item store and the phonological

store interacts is summarized as the following:

1. Phonological→Item. A phoneme sequence retrieved from the phonological store, can be recognized by the sequence recognizer. The recognized item is forwarded to the item store (at the Input layer). At the same time, item activations by the phonological form can influence the retrieval process, and bias retrieval towards the activated items. This mechanism generates the phonological effect: phonologically similar items are more likely transposed with each other because these items are commonly activated.

2. Item→Phonological. The output of the item store, can be converted to a phoneme sequence by the Sequence Producer. The phoneme sequence can feed back to the Phoneme Input layer. This process refreshes the content in the phonological store, and is a standard rehearsal process in Baddeley's model.

The first type of interaction (phonological→item) is a new but straightforward account for the phonological similarity effects. This is new because this mechanism is only feasible when the phonological store is fully implemented. Alternatively, a two-stage process can be used to produce phonological similarity effects. Their differences are discussed later.

5.2.2 Training

Training of this model is fairly simple despite the complexity of its architecture. As the Divergent-Reconvergent model, training only involves learning individual items in isolation (e.g., 26 letters). In each training trial, one item is phonologically presented to the Phoneme Input layer, and the corresponding unit in the sequence recognizer is activated. The sequence encoder uses its recurrent connections to encode the sequentially presented phonemes, and the sequence is associated with the sequence recognizer. The Divergent-Reconvergent model (the item store) is trained as in Chapter 4 at the same time by associating corresponding items in all layers.

5.2.3 Recall strategies

After training, a challenging problem faced by the Dual Representation model is about how to control the recall process. This is because under different experimental conditions, information available is different. With visual presentation under articulatory suppression, the information is only represented in the item store – the Divergent-

Reconvergent model alone. With auditory presentation or visual presentation without suppression, the information is represented in both stores. For recall from either one or two stores, coordination and control are required.

Let me first exemplify the real challenges of coordination, and then present a relatively simple scenario, which is adequate for most serial recall tasks. The real challenge is when information is incomplete for either one or both stores. For example, there is a list item that the subject cannot pronounce (e.g., a seen foreign letter). The letter may be represented in the item store, but it is not represented phonologically. How can the subject best recall the sequence? Because of the superiority of phonological coding, he may recall from the phonological store until the strange letter. Then he has to pause and use the item store alone, and then continues with phonological retrieval. This example shows that there is an optimal control problem that is obviously out of the scope of this work. However, it stresses two issues: a need to match up the item code and the phonological code, and a need to control of the retrieval focus or a combination of information sources. The first problem of associating the item code and phonological code refers the capability of accessing the phonological code from an item code, and the opposite, accessing the item code from the phonological code. This problem is discussed in details in Section 5.3.1, but it is not fully implemented in this model due to its complexity. The second problem, the control of the retrieval focus on either the item or phonological store, however, is a simpler problem and can be reasonably approached. It provides a starting point for studying the control process in working memory, and covers a few typical experimental conditions, explained below.

Recall from the item store (visual words under suppression) This recall strategy focuses on the item store. It applies to the scenario where the sequence is minimally phonologically coded, but is mainly represented in the item store – the Divergent-Reconvergent model. The recall strategy is thus identical to a Divergent-Reconvergent model alone. When phonological coding is absent, this is an optimal strategy.

Recall from the phonological store (auditory nonwords) This strategy applies to the scenario where nonwords which have no associations in the long-term memory are only coded phonologically. Their recall is identical to retrieval from the phonological store alone. It does not involve interaction between the two stores either. This strategy is optimal when item coding is not available.

Recall from both (auditory and visual words w/o suppression) This is an more complicated recall scenario. The sequence has two *complete* representations in the two stores (the scenarios of incomplete representations are not addressed by the models in this work, discussed later). To achieve the optimal recall results, the recall mechanism must aligned the item code and the phonological code during retrieval. When representations are complete, the two stores can be respectively accessed the contents in synchrony to achieve alignment. The mechanism used in this work is to start the retrieval from the phonological store, and to feed item activations to the item store. A retrieval is made when two sources of item activations are merged. The merge occurs when the phonologically retrieved sequence is a recognized as a complete item.

The Central Executive model provides a principled way of simulating all the three recall strategies above.

5.2.4 Rehearsing strategies

During the presentation phase or the delay period before recall, the subject may subvocally rehearse the memorized material for better maintenance. In order to model the process of rehearsal, we need to specify two what is rehearsed and when it is rehearsed. For the *what* question, there is experimental support for *cumulative rehearsal*, or the rehearsal of the accumulated list so far (Page & Norris, 1998). Although it is difficult to systematic investigate participants' covert rehearsal strategies, experiments with overt rehearsal or instructed covert rehearsal are consistent with the idea that *cumulative rehearsal* is optimal and usual (Ferguson & Bray, 1976; Palmer & Ornstein, 1971).

The next question is *when* the material is rehearsed. They can be rehearsed whenever there is an adequate pause between presented items. But do subjects rehearse in every pause? No. Rehearsal frequency (about 50%) can be lower than the number of items (O'Neill & Douglas, 1996). It appears that subjects don't rehearse if it is not needed. In this model, an economical strategy is adopted. A rehearsal starts "just in time" before the information is forgotten. For (relative) easiness of modeling, an ongoing rehearsal is assumed to run to its completion, and it is assumed not to start if other phonological processes are underway, for example, during the presentation of an auditory item. Further, rehearsal is assumed not to be interrupted by a new input item. If a rehearsal has started, the new item has to wait in sensory memory, a level below

working memory. Because sensory memory is not fully modeled here, the Simulator is used to help buffering the waiting inputs. A simple protocol is used between the Simulator and the neural network: the Simulator will check the input layer and delay the presentation of the next item if a previous item has not been cleared by the neural network; if the activation of an input item has been passed to the next layer, the neural network is responsible for clearing the input layer. With this protocol, the neural network can fully rehearse the material without interruption and does not lose the next input.

The reason for go to great lengths in modeling the rehearsal process is because rehearsal is a central process in working memory, but it hasn't been systematically modeled before. Here, by adopting an optimal strategy, *just-in-time cumulative complete* (JITCC) rehearsal, the model can automatically practice rehearsal without further assumptions. Hopefully, this will provide a systematic account for effects where rehearsal or suppression of rehearsal is involved.

5.3 Subnetwork models

This section discusses mechanisms of various neural processes in the Dual Representation model. It is separate from the architecture discussion above because detailed implementation are not crucial for the behavioral characteristics of the model if the implementation satisfies a few macroscopic properties. In summary, the macroscopic properties are: the phonological store encodes time by a spatial dimension, and the sequence encoder reflects sequence similarity. With the two properties, the model can encode both words (by both the item and phonological stores) and nonwords (by the phonological store alone), and it will exhibit the observed phonological similarity effects as well as other effects. Below, I will present an implementation of these neural components, in the hope that they can shed some light on future studies of the neural substrate of the phonological loop.

5.3.1 The "tape" model of the Phonological Store

Representing serial order along a spatial dimension is not a new idea, but it's an idea that needs to be treated carefully. The idea appeared very early in Atkinson and Shiffrin's (1968) short-term memory model (see Chapter 1) and was also assumed by Conrad (1965). In these early models of serial order, items are assumed to be stored in boxes representing different position, however, the "box" model has been criticized

on at least three aspects. First, it is hard to determine for the model how many boxes are needed. Second, it does not account for swapping errors between nearby positions (R. N. A. Henson, 2001). Third, the model has to duplicate codes for each position, which can be a problem for a large number of items.

Though the "box" model is probably invalid at the item level, the idea of encoding time by a spatial dimension might be valid at the phonological level (c.f., the grid model, Crowder, Harvey, & Routh, 1983). This is because phonological coding only involves a small family of phonemes. The capacity of the model can also be described by time (about 2 seconds) rather than by an unspecific number of boxes. Further, empirical data support the model's prediction that nonword serial-recall based on pure phonological coding lacks the PSE (Lian et al., 2001), if nonwords are phonologically coded along a spatial dimension.

An earlier model that addressed the serial order problem of phonemes is the phonological loop model (O'Reilly & Soto, 2001). O'Reilly and Soto's model consists of a series of stripes, which can be selectively accessed to store or retrieve phonemes at different sequence positions. The mechanism that controls which stripe can be accessed is assumed to be the basal ganglia gating mechanism. In this research, the author explores a similar idea. The neural layer for sequence memory is assumed to be selectively accessible as in O'Reilly and Soto's model. The selectivity, here, is assumed to be controlled by a sweeping wave, or a neural activity that sweeps across a neural layer. In an abstract level, the sweeping wave can be seen as the output of the basal ganglia gating mechanism, but there are other possibilities. It may also be generated by spontaneous spread of neural activity or a winner-take-all process (explained below), as observed in the cortices (e.g., Kleinfeld & Waters, 2007; Ferezou et al., 2007). More specifically, the shape of the sweeping wave can either be a point activity moving along one dimension, or a circular activity spreading out from a center (see Figure 5.2). In this model, information content is assumed to be encoded by an dimension orthogonal to the propagation direction of the wave. This information can either be tonotopic (encoding frequency) (Cheung, 2005; Talavage et al., 2004) or phonological (encoding phonemic features). Based on the above mechanisms, a tape-like sequential memory can be constructed below.

The architecture of the model is shown in Figure 5.3. Input is presented to the 1-dimensional *In* vector, which encodes features of the speech sound. This model remains open about the details of features. On the one hand, the traditional phonological store

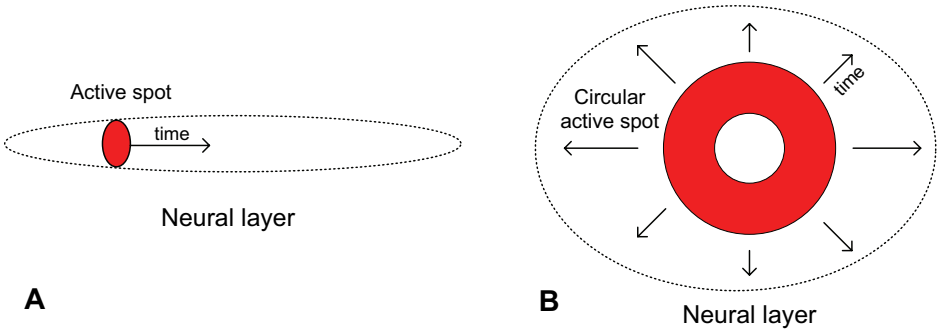


Figure 5.2: Sweeping waves as a selector for neural access. A) A simple sweeping wave can be generated by an active spot moving along one direction. B) A circular sweeping wave can be generated as an active spot spreads out from the center.

model (Baddeley, 1986) suggests that the features are phonological. On the other hand, it is suggested from other evidence that the content is coded acoustically (Jones et al., 2007). Because the key aspect of the model is that a sound sequence can be encoded by a spatial dimension, the model is not confined to either type of representations. The 1-dimensional *In* vector is spatially expanded into a 2-dimensional *Input* layer, which topographically projects to the *Tape* layer, the core of the phonological store. Here, information can be maintained for about 2 seconds. The projection from the Input layer to the Tape layer is gated by a moving wave W. The wave W is assumed to be a directional wave of neural activity.

During retrieval, the content in the *Tape* layer is propagated topographically into the Output layer gated by another moving wave R. The wave R can be seen as a retrieving index for the information stored on the tape. For the system to function, there are several (difficult) requirements for the wave R:

1. It should be able to find the beginning of the speech sound on the Tape layer (serial retrieval).

2. It should be able to skip blank or silent regions of the tape (time compression).

3. It should be able selective the sound of *any* item on the tape (selective retrieval).

These functions are within people's capabilities with auditory memory, and should be possessed by the system. Though diverse and difficult as they might appear, fortunately, there is a parsimonious solution to all the problems above.

The first two functions of the wave R, finding the beginning of the sound information and skipping silent regions, can be fulfilled by one single neural mechanism –

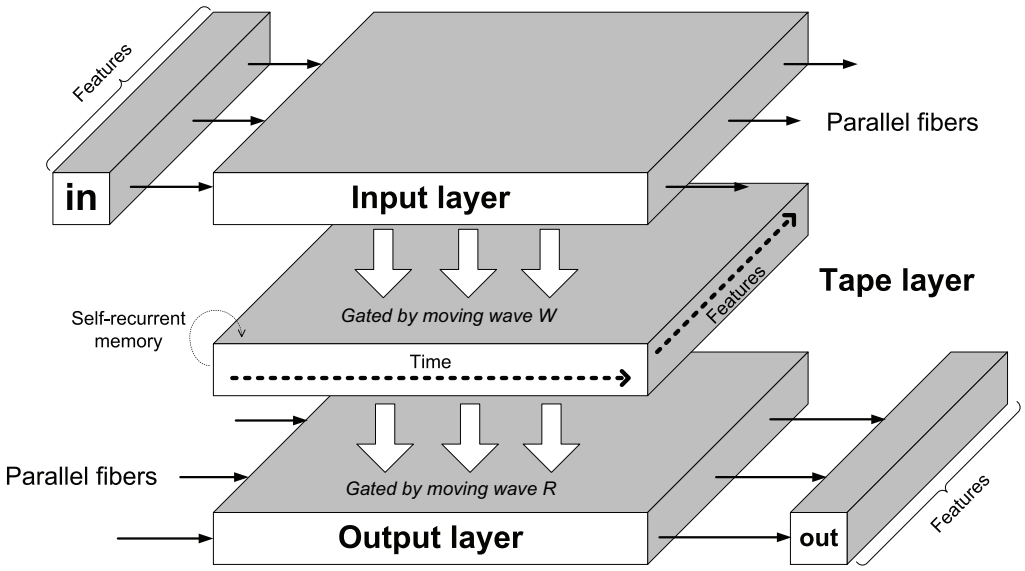


Figure 5.3: The phonological store model.

a *thickness gradient* of a neural layer, shown in Figure 5.4. In the example, the Tape layer contains three active items and there are pauses between them (e.g., the words "THREE [pause] FOUR [pause] FIVE"). For retrieval, the wave R should be generated at the beginning of the first word (shown by the circular dot in the Wave-R layer). This can simply be done by a winner-take-all process, because the activity at the beginning of the first word is magnified by the thickness of the gradient layer. Then, the wave can simply follows along the activation gradient to retrieve the rest of the sound. When the wave R hits a silent region, it may automatically skip it because of the winner-take-all principle.

The third function, selective retrieval of the sound of an item, is also supported by the same architecture. To select the sound of an item, all that's needed is to increase the activity of all features of this item in the Tape layer. For example, to search for the sound of the item "FOUR". We can simply increase the activity of all the features /f/ and /o:/ and /r/ in "FOUR" in the Tape layer. Though this also increases the activity of other items, such as "FIVE", the correct item "FOUR" will receive the greatest activation boost, and will be selected by the winner-take-all process.

Therefore, we can see that the single mechanism based on a *thickness gradient* is very powerful. It can simultaneously support serial retrieval, blank skipping, and content-based retrieval. Due to its parsimony, it may represent a valid mechanism for

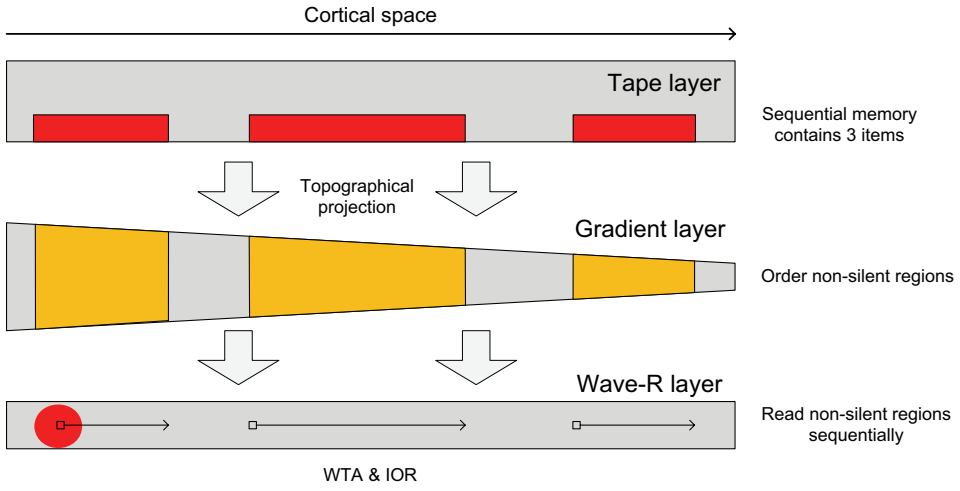


Figure 5.4: Sequential reading of nonsilent regions of phonological store.

acoustic and/or phonological memory. Interestingly, this model is partially supported by anatomical and physiological data of the auditory cortex. The *thickness gradient* has been found in specific layers in monkey's auditory cortex (Galaburda & Pandya, 1983). The computational mechanisms illustrated here might help understand the neural circuits in the auditory cortex.

5.3.2 The sequence encoder/recognizer network

The sequence encoder is a recurrent network that can maintain a representation of a presented sequence. Though many neural network models can be used to encode sequences, such as the Simple Recurrent Network (SRN, Elman, 1990) and other recurrent networks (e.g., Schulz & Reggia, 2004), in this research, I would like to explore a novel neural network that has a lesser known property, the *counting property*.

Below, I will first introduce the counting property, and then use it to construct a sequence encoder model. The sequence encoder can tokenize repeated items, and form different representations for different sequences with repetitions. At the same time, it also maintains the similarity relations between different sequences, so that it can produce the phonological similarity effects in the working memory model.

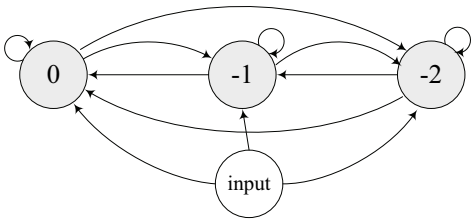


Figure 5.5: Schematic counter/tokenizer. All connections have the weight 1. The values in the neurons represent their biases. Activation functions output 1 for positive inputs, and 0 otherwise.

5.3.2.1 Counter/tokenizer

Consider the neural network in Figure 5.5. This group of neurons is fully connected with weights 1. The biases are 0, -1, and -2, respectively, and the activation function is binary, outputting 1 for positive values and 0 for other values. By a quick inspection, it can be seen that the first input (with a strength 1) can only activate the first neuron, because other neurons have negative biases. When this neuron remains active by its recurrent connection, it can increase other neurons' biases by 1, making them easier to fire. The second input can now activate the second neuron, and so on. Similarly, this network can also count down. This illustrates a mechanism with which a neural network can count. Does the brain use any mechanisms similar to this for counting?

Though the above model requires exact biases, it can be shown that the *counting property* is present in neural assemblies even with random biases. As shown in Appendix A.3, the counter is also likely to have a capacity of *four*, similar to a type of animal and human numerical capacities (J. Atkinson, Campbell, & Francis, 1976; Luck & Vogel, 1997; Carey, 1998; Cowan, 2001a). Therefore, it is possible that a similar mechanism is used in the brain.

5.3.2.2 Sequence encoder/recognizer

The Divergent-Reconvergent model can encode any sequence by a unique spatial pattern if it does not contain repeated items, but for sequences with repeated items, the model can only encode the first instance of each item. The idea in this sequence encoder is that neurons have non-uniform biases as in the counter so that they can tokenize different instances of an item, and the sequence encoder as a whole can then encode

sequences even with repeated items. The architecture of the sequence encoder/recognizer network is illustrated in Figure 5.6. The sequence encoder can be seen as an extension of the Divergent-Reconvergent network with the above counter network embedded in the neural assemblies. Above the sequence encoder, a sequence recognizer recognizes the representations of sequences as a perceptron.

Though the combination of a tokenizer and an activation gradient model (the Divergent-Reconvergent model) may serve as a general sequential memory model itself, I choose this mechanism for sequential perception but not sequential working memory *per se*, because tokenization is not adequate to account for many phenomena in working memory. In working memory tasks, it is often found that repetition is encoded by patterns. For example, during recall of repeated items, there is usually a longer pause after a repetition pattern such as XX or XYX(Kahana & Jacobs, 2000). This suggests that people can recognize the repetition patterns and use it to reproduce the sequence, rather than transparently tokenize all instances of the same item and treat them as different entities. Further, there are also repetition errors in typing that are errors when using patterns like XX and XYX, such as "letter"→ "leeter", or "cheese"→"cheses" (Rumelhart & Norman, 1982). Working memory encoding based on the repetition patterns will be discussed later in Section 5.5.1.

5.3.3 Inter-component communication

An interesting problem in a large-scale neural network model is about how to communicate information between different components. This is a new challenge for neural network modeling. An example of the problem is the coordination of the Item Store and the Sequence Producer. The two components work in a hierarchy. The high-level Item Store retrieves a sequence of items or symbols, each of which needs to be expanded by the Sequence Producer as an articulatory sequence. In this process, the Item Store has to wait until articulatory output is done to request the production of the next item. When the Sequence Procedure is articulating an item, the Item Store has to know it's busy and should wait.

Some signaling mechanism is required for the interaction of multiple components, and it is reasonable to assume that the brain has a solution for this problem. In this model, the working hypothesis is that a to-be produced/processed item remains active in the high-level component, while it is being processed by a low-level component.

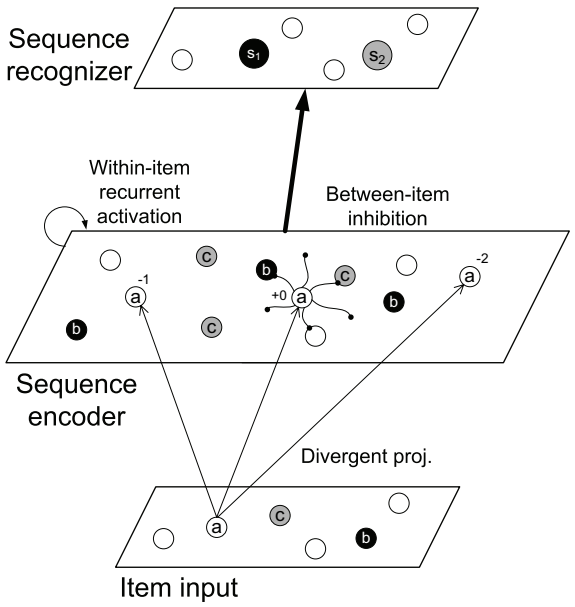


Figure 5.6: Sequence encoder/recognizer. This model is based on the Divergent-Reconvergent model. Groups of neurons in the Sequence Encoder layer form activation gradients that represent the serial order of items. However, neurons in each group (corresponding for one item) have different biases (shown by the signed numbers) giving them the *counter/tokenizer* property explained in the text. The whole network can uniquely encode different sequences even with repeated items (up to a few repetitions). The sequence recognizer is a perceptron.

When the processing is done, the low-level component deactivates this item by backward inhibition. This mechanism is used for two purposes in the Dual Representation model. First, it is used to coordinate the hierarchical articulatory processes as discussed above. Second, it is used to delay sensory input into working memory, while the working memory is performing a rehearsal (Section 5.2.4). Though this idea is a tentative method to enable large-scale neural network modeling, this neural mechanism may nonetheless bear some biological relevance. For example, it is also observed that downstream processing (e.g., shape perception) can reduce neural activity at previous neural layers (e.g., the primary visual cortex) (S. S. Murray, Kersten, Olshausen, Schrater, & Woods, 2002). This also gives some support for the above backward inhibition. Later in this thesis, this mechanism is also used to propose a novel explanation for the well-known Sternberg effect (Section 6.2). It is hoped that this mechanism can receive further verification and investigation in order to understand the interactions between different cortical areas.

5.4 Simulations

5.4.1 Phonological effects on serial order recall

This simulation tests the phonological similarity effect in this model. Four types of materials are used: 1) lists containing phonologically dissimilar letters (Dissimilar), 2) lists containing phonologically similar letters (Similar), 3) lists containing alternatively similar and dissimilar letters (DSDSDS and SDSDSD). The lists are auditorily presented to the model (as a phoneme strings), at a rate of 500 ms per item. During the gaps between items, the model can automatically start a rehearsal process by the above optimal rehearsal strategy. After presentation, the "recall from both" strategy is used to recall the lists.

The serial position curves (Figure 5.7, left) show that the model has clear similarity effects. Specifically, the recall performance for phonologically dissimilar items is the better than recall of phonologically similar items. In lists of alternate similar and dissimilar items (e.g., P U B Y V O), the serial position curve shows a sawtooth pattern, where dissimilar items stand out with higher recall accuracy than similar items. This effect replicates previous human experiments (Conrad, 1964). Human data (R. N. A. Henson, 1986, , Exp. 1) under comparable conditions are plotted in Figure 5.7 (right) for comparison. The parameters in the phonological loop model are listed in Table 5.1. They

are first manually set to enable basic functions of the model, then quantitatively fit to the human data. The parameters for the Item Store are the same parameters as in the Divergent-Reconvergent model.

Discussion: multiple-information interaction

Most working memory models of immediate serial recall store two types of information, item-level information and phonological information. A convergence reached by recent models is a two-stage model of recall, where an item is first retrieved from the item store alone, and then its phonological code is activated and the recall item is re-determined by the activated phonological code. This noise process can yield the effect that phonologically similar items are relatively confusable with each other, while phonologically dissimilar items are substantially subject to confusion (Burgess & Hitch, 1999; R. N. A. Henson, 1998; Page & Norris, 1998). By this process alone, phonological information can only have detrimental effect on recall. Without further mechanisms, this is not immediately consistent with superiority of the auditory modality for serial recall. This simulation, however, demonstrates that there is an alternative way to generate the phonological similarity effect, where phonological information can be seen as having beneficial effects, consistent with the modality effect that auditory encoding is better than visual and semantic encoding for serial recall.

The way that item-level information and phonological information interact in this model is cooperative and parallel, rather than serial as in many working memory models. In a serial account, the phonological similarity effect is produced in two stages: 1) an item is retrieved from the item store; 2) its phonological code is activated and the recall item is re-determined from the activated phonological code. When the process is noisy, the phonological similarity effect is produced, where similar items can be more easily confused than dissimilar items. This simulation shows that two-stage model is not the only solution. With a phonological store where the serial order of phonemes can be retrieved, confusion can arise from the phonological→item mapping process (sequence recognition). In this process, dissimilar phoneme sequences have stronger facilitory effects than similar phoneme sequences, because they are more distinctive from each other. This can also explain for the phonological effects. The difference from the two-stage model is that the phonological information in this model is *facilitory* rather than confusing. The author believes this is a more accurate mechanism because it parsimoniously explains for

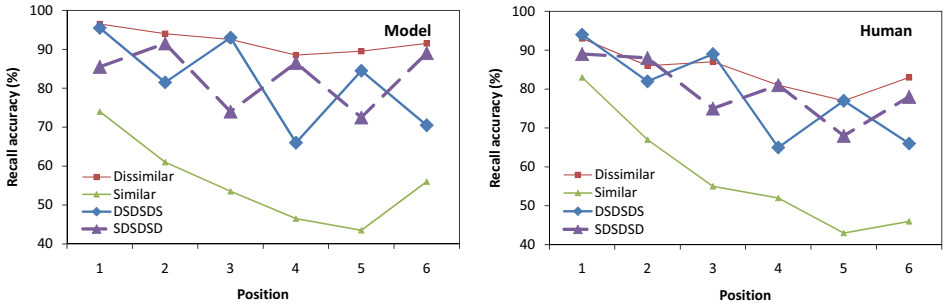


Figure 5.7: The serial position effects in Dual Representation model

both phonological similarity effects, and the superiority of the phonological modality in serial recall. This model is more complicated than a two-stage model, however, the added component, a phonological store encoding the serial order of phonemes, is considered necessary by other researchers too (e.g., Burgess & Hitch, 1999), so this model is subject to criticisms based on parsimony. Rather, with a full neural implementation, it can be seen as an advance in modeling and bears greater biological relevance.

Table 5.1: Parameters of the Dual Representation model

Param.	**Description**	**Values**
σ_p	Std. Dev. of Gaussian noise	.05
w_p	Phonological weight	1
T_p	Phonological timescale	4 sec

5.4.2 Phonological similarity effects in nonwords?

Despite the robustness of phonological similarity effects (PSE) in words, the immediate serial recall of nonwords sometimes reveal a null or reversed effect (Lian et al., 2001, 2004; Karlsen, Imenes, Johannessen, Endestad, & Lian, 2007). These data are valuable for a deeper understanding of the structure of the phonological loop. Here, I summarize two types of nonword studies: *nonword repetition* and *nonword serial recall*. Nonword repetition, repeating a long nonword (e.g., spentonymidderoxing), resembles the serial recall of a word list (Gupta, Lipinski, Abbs, & Lin, 2005; Archibald & Gathercole, 2007). However, *nonword serial recall*, the recall of a list of short nonwords (e.g., kev dob tiv...), does not yield phonological similarity effects, unlike the serial recall of words (Lian et al., 2001, 2004). Why is there a difference? It is possible that the sylla-

bles in a to-be-repeated long nonword are recognizable as items, thus, the recall process resembles word serial recall. However, the nonwords in nonword serial recall do not have long-term associations (a control condition of the experiment) and thus their recall does not resemble word serial recall. It relies only on phonological code but not item code[1].

This explanation can be clearly seen from the Dual Representation model. In this model, whether or not items have long-term representations determines the existence of phonological similarity effects, because PSE does not come from the phonological store itself, but from the interaction between the phonological store and the item store. As explained in the "recall from both" condition in Section 5.2.3, PSE occurs during the recognition process in which a recalled phoneme sequence is recognized as an item. If the phoneme sequence cannot be recognized (not having long-term associations as in Lian et al.'s experiments), the model would have no PSE, consistent with the experimental finding (Lian et al., 2004). However, when the phoneme sequences can be partially recognized, as in a long-nonword consisting of familiar syllables, the model predicts similar PSE as in a word serial recall. This is consistent with findings of nonword repetition (Gupta et al., 2005).

Therefore, the Dual Representation model consistently explains for the difference in nonword repetition and nonword serial recall. The following simulation tests this property. It shows that when nonwords have no long-term memory associations, there is no phonological similarity effects (Lian et al., 2004), but when the items are recognizable (identical to word serial recall), there is detrimental phonological similarity effects. Because this experiment is run on a computational model, it is easy to control the presence and absence of a long-term memory associations. The same set of items can be used for both conditions (words vs. nonwords). In the word condition, the items (10 English digits) are used to pre-train the sequence encoder/recognizer, so the model can recognize them. In the nonword condition, the model is not pre-trained and it can only treat the items as pure speech sound. This procedure perfectly controls for word lengths and phonemic complexities in both conditions. The results of recall accuracies for words and nonwords are given in Figure 5.9. They confirm that though words have clear PSE, nonwords without long-term representations don't, consistent with the findings of Lian et al. (2004).

[1] As mentioned above, the representation of phonological information is controversial. Here, the term phonological code is vaguely used to represent any coding of the sound of words, possibly including acoustic, phonological and articulatory codes.

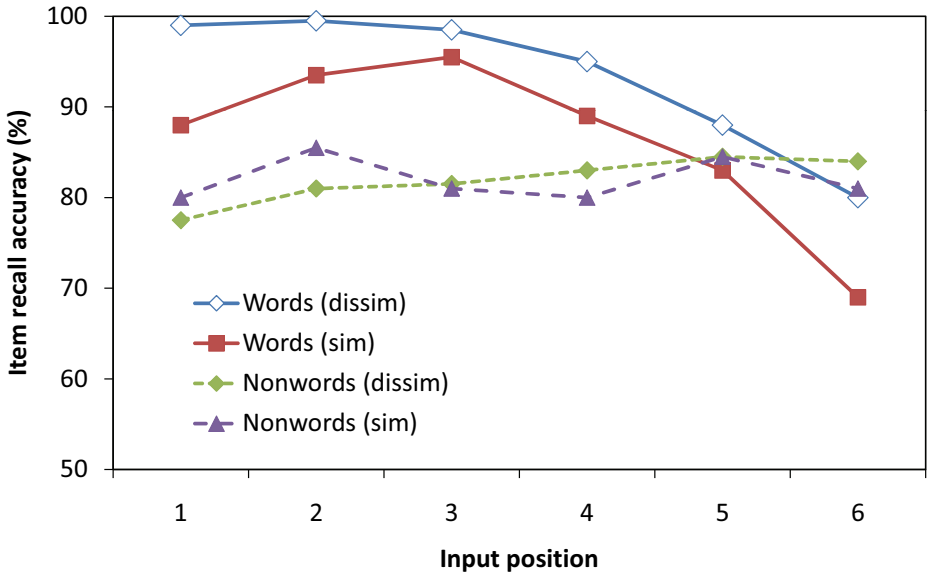


Figure 5.8: Comparison of phonological similarity effects on words and nonwords.

Technically, in this simulation, nonword recall errors are induced by random noises that flip the on/off states of units in the phonological store. The probability of a unit being flipped at each time step is $\sigma_f = 10\%$. The same procedure applies to word recall in this simulation. This controls the amount of random petulance's for the two conditions. In other simulations of the model, for simplicity, recall errors are only induced during the item recognition process (using the parameter σ_p), but the phonological store contents are not subject to random perturbations as in this simulation.

5.4.3 Modality effects

As discussed in the previous chapter, serial recall performance of the Divergent-Reconvergent model or the item store can be improved when it is augmented with a phonological store. The augmented model has a natural modality effect for auditorily presented lists, which would be better recalled than visually presented lists that are not auditorily recoded. This is shown in Figure 5.9. This effect comes from the property that auditorily presented lists are stored in both the Item Store and the Phonological Store, but visually presented lists are only stored in the Item Store. Storage in both stores is recalled better since the two types of information is cooperative and complementary during retrieval. Specifically, the item store has a longer duration but a weaker representation

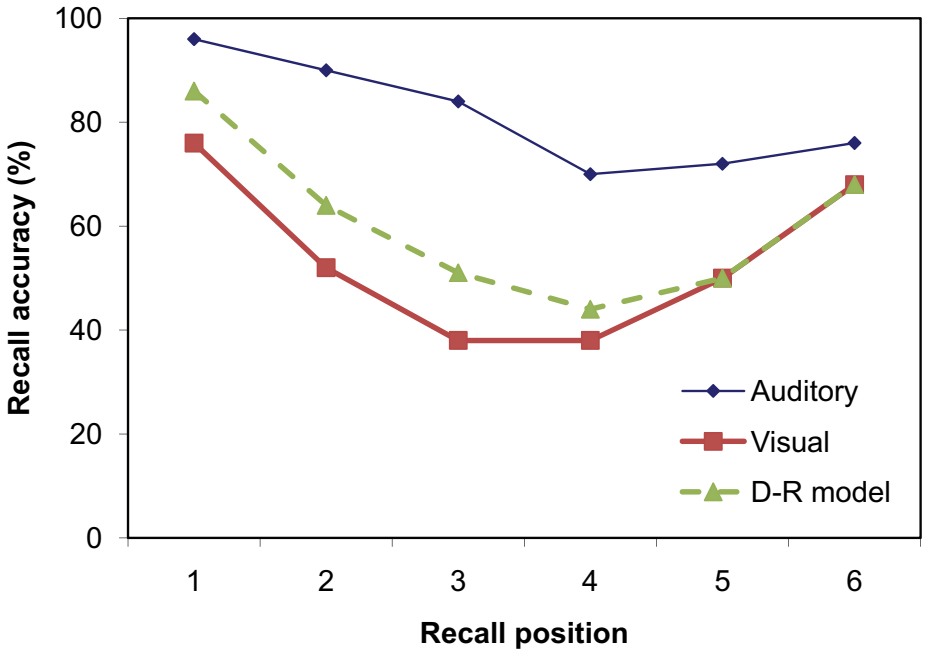


Figure 5.9: Modality effect in the Dual Representation model. Auditorily presented lists are recalled better than visually presented lists (under articulatory suppression). Visually presented lists are recalled a slightly worth than the equivalent D-R model alone, because of perturbation from random noises in the Phonological Store.

for serial-order since it is based on relative activation levels. The phonological store has a shorter duration but a stronger representation for serial-order because serial-order is maintained by a spatial dimension.

With the new phonological modality added to the amodal Item store, the model immediately lends itself to a full repertoire of benchmark phonological loop effects, such as the modality effect discussed above, word-length effects, articulatory suppression, lexicality effects (word vs. nonword) and so on. The advantage of this model with a fully implemented phonological store is that all processes underlying these effects are inherent in the architecture of the model. Detailed processes can be directly observed, and these simulations usually don't require modification of the model, but only requires feeding in the right stimulus, and setting simple control programs (giving instructions). Some of other effects are discussed below.

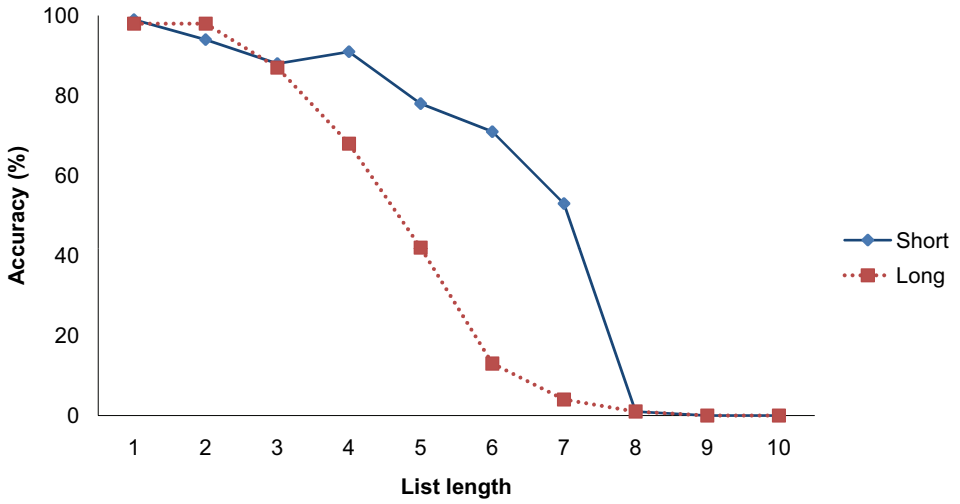


Figure 5.10: Word-length effects on list recall accuracy

5.4.4 Word-length, irrelevant speech and suffix effects

These effects are associated with phonological coding and the rehearsal process in Baddeley's phonological loop model. As a direct implementation of the phonological loop, the Dual Representation model can also exhibit these effects. Figure 5.10 shows the word length effects. Lists of long words are recalled worse than those of short-words. This is because during list presentation and the delay period, the model automatically rehearses the material. Since long words take longer time to rehearse, fewer long words can be remembered. Irrelevant speech which adds additional noises to the phonological representation would degrade recall performance. This is easy to see for the model (simulation omitted).

A disadvantage of this model is that it does not directly account for the suffix effect if it assumes (for technical reasons) that the phonological store has a complete representation of the whole list. In this case, the representation cannot be removed by a single auditory suffix. In order to account for the suffix effect, some improvements of the model are needed (discussed below).

5.4.5 Future improvements

The major disadvantage of this model is that it relies on a strong assumption about phonological representation. That is, it requires a complete phonological representation for all list items in order to synchronize the retrieval from both stores. This

technical requirement turns out to interfere with the model's ability to fit human data. By analysis, it is known that the model has a smaller auditory suffix effect than expected. To model the limited capacity of the phonological store and its susceptibility to the suffix, the model needs to be technically improved. There are two ways to do it. The first is to fully develop the Central Executive, and requires it to learn to handle incomplete phonological representations. This requires an implementation of reinforcement learning in the working memory model. An alternative way to improve this model is to technically modify the phonological store by adding a boundary marker for each phonological item, so that even though the phonological information is decays quickly and makes the representation incomplete, the non-decaying boundary makers can help synchronize the retrieval of the phonological store and the item store, thus dissociating technical requirements from psychological properties of the model.

5.5 Discussion

After developing the large-scale modeling framework for working memory, it is important to see whether this framework can fit into the bigger picture of other working memory theories and models, and handle new challenges. Two major theoretical challenges for the current model are 1) it does not fully handle repetitions in the sequence, 2) it does not encode positions. I will briefly discuss how this model can be improved to address these issues.

5.5.1 Repetition coding

Repetition is a well known problem for any models based on an activation (or primacy) gradient (Page & Norris, 1998; Shieh & Elman, 2006). In these models, an item is typically represented by a single node or a given group of neurons. The presentation of the second instance of an item will increase the activation of these neurons, but this does not correctly represent the two instances (Bradski, Carpenter, & Grossberg, 1994). In order to encode and retrieve repeated items, additional mechanisms are needed. A tokenization process may partially mitigate this issue by allowing the model to encode and retrieve repeated items. This has been addressed in Bradski et al. (1994). In line of the idea of tokenization, I have proposed a new tokenization process based on distributed neural biases (Section 5.3.2).

However, from human performance, we know that different tokens of the same type have more intriguing interactions, which cannot be explained by tokenization alone. Interesting phenomena can be observed from motor short-term memory such as typing[2]. Some typing errors are highly informative for how the brain encodes repeated items, such as misplaced repetitions (e.g., *supper* → *suuper*) and wrong repetition numbers (e.g., *suppper*) (Rumelhart & Norman, 1982). The errors suggest that repeated items are represented by an item code and a separate code for the repetition pattern. The errors can be explained as the repetition code is applied to mistaken items (causing misplaced repetitions), or its activation strength is perturbed by noises (causing wrong repetition numbers). Another error, the alternation reversal error (e.g., *these* → *thses*), also suggests for a special alternating mode, where the letters are produced by an alternating mechanism generating the form XYX. When a wrong letter is produced first (e.g., due to binding error), the output will be the reversed alternation YXY. Further, it is found that when people retrieve sequences with the patterns XX or XYX, there is usually a longer pause after the pattern, suggesting the presence of pattern encoding and reproduction (Kahana & Jacobs, 2000).

These phenomena motivated one of our models (not reported above) which explicitly detects and encodes repetition patterns, and uses them to mitigate the issue of repetitions (Shieh & Elman, 2007). This model is a hybrid model that uses phase neurons to encode activation and binding, but symbolic representations of chunks and hierarchical structures. Future studies of repetition following the this work can involve explaining 1) how repetition is detected by a neural network, 2) how repetition patterns are encoded, and 3) how repetition patterns are reproduced.

Below, I briefly discuss some basic ideas in a more advanced model in this line. Figure 5.11 shows a simple repetition detector. In this detector, the memory layer remembers the last item, and the change detection layer is activated if there is any difference between the current item and the previous item. If there is a difference, the output Repetition Detector will be inhibited. Otherwise, it will be activated by the New Item detector to signal a repeated item. Multiple repetitions (e.g., XXX or X^3) can be encoded by the *counting* property. Complicated patterns (e.g., XYX) requires two of such detectors, alternately updated to memorize the last item. Once a repetition is

[2] According to the theory that different cortico-basal ganglionic loops mediate short-term sequencing of different types of information or movements (Chapter 4), motor short-term memory and digit short-term memory are based on comparable neural mechanisms.

detected, the repetition code (e.g., XX or XYX) can be bound with the item code so that they can be retrieved together later for reproduction of the pattern. For example, in the word 'supper', the repetition code XX or X^2 can be bound with the letter 'p' to encode the doubling 'pp'. In the word 'these', the repetition code XYX with two variables X and Y can be bound with the letters 'e' and 's' to encode pattern 'ese'. The binding process can make use of an additional neural mechanism – a *binder*, which can simply be a neural layer with bidirectional topographical connections with the Item Store. The binder in this model has rich excitatory interconnections. When items (or a variable and an item) are selectively loaded into the binder from the Item Store, they are automatically synchronized by the excitatory connections in the binder, and the synchrony feeds back to the Item Store. With the binder, the repetition code and the item code can be selectively bound (under the control of the Central Executive). Retrieval of a repeated item can also be done within the framework by the Phase Selection Filter which retrieves the item code and the repetition code simultaneously due to their synchrony (see Section 3.3.1 about the Phase Selection Filter) . After the retrieval of the repetition code and bound items, a pattern reproduction process can use a decreasing *counter* to control the number of repetitions, and if the repetition is separated by other items, an additional *counter* can be used to count the items between repetitions. Because a neural implementation of the repetition encoding model is complicated, the details are likely to be learned and are non-uniform for all people. The objective of the above construction process is to show that the basic idea can be feasibly modeled within the PNN framework, and future models would benefit from learning algorithms that adaptively train the networks to perform these tasks.

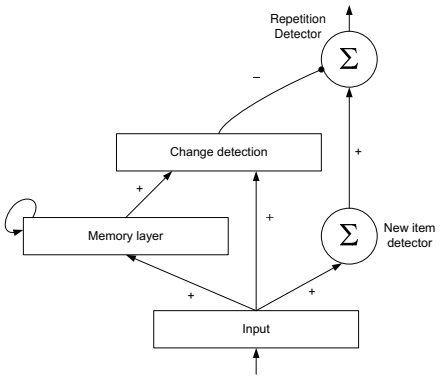


Figure 5.11: Repetition Detector

5.5.2 Position coding

Though positional coding has not been directly modeled in the Dual Representation model, it is crucial to explain several phenomena. These phenomena include transposition errors between the same within-group positions in sub-grouped lists (e.g., 1<u>5</u>4-3<u>9</u>3) (R. N. A. Henson, 1998), and facilitated retrieval of repeated items when they occur at the same within-group positions (e.g., 1<u>5</u>4-3<u>5</u>3) (R. N. Henson, 1998). Some remaining issues about the positional theory include 1) it is yet not clear whether precise positional code exists beyond coarse markers such as *begin, middle*, and *end* (Botvinick & Plaut, 2006); 2) it is not yet known what neural mechanisms generate the positional code; and 3) it is not clear what neural mechanism associates the positional code and the item code (e.g., neural plasticity vs. activity-based mechanisms).

Here, I briefly discuss a mechanism for positional coding, which may avoid some of the above criticisms. The idea assumes that the counting property of neural assemblies provides discrete position codes for early items (e.g., from positions 1 to 4) and a position code the last item (see Appendix A.3 for a discussion about the capacity of 4). Thus, it agrees with the idea of limited (coarse) position coding (Botvinick & Plaut, 2006). The last item is assumed to be marked by an "end" signal. This might result from a general mechanism which informs the system to take further actions when a sequence ends. This set of positional codes (1,2,3,4 and END) is not adequate to handle the serial-order problem alone (e.g., fully retrieve a long sequence), but they can effectively index the items in working memory for easier access. To accommodate temporal grouping, the positional code can be assumed to be reset at the boundary between temporal groups as in other positional theories (R. N. A. Henson, 1998; Burgess & Hitch, 1999). In sum, the above postulation can explain for the position-based errors and facilitation mentioned above, but does not require full positional coding as criticized by some researchers (Botvinick & Plaut, 2006).

On the mechanism of position-item association, the positional code is hypothesized to be initially associated with the items by dynamic binding or neural synchronization, before neural plasticity is changed. In this way, the model can avoid some criticisms concerning the use of a long-term memory mechanism for online working memory storage. Later, long-term transfer of position-item binding allows the model to explain "serial-order intrusions" across trials, in which an item from a previous list may be mistakenly recalled at the same position of the next trial. The transferring mechanism for long-term

memory is discussed in the next section.

Surprisingly, position coding in this theory turns out to be very similar to repetition coding, in that they both involve the activation of a separate code, and the binding of the code to the item. The binding can be done with neural synchronization in the theoretical *binder*. In a pilot model simulated by the author, the above position and repetition coding schemes can indeed be performed by the same mechanism and process: loading the binder with the item code, as well as the position code and repetition code (if any) upon the presentation of each new item. This model can be seen as augmenting the representation of an item with additional features, and binds them together to form an instance of the item. The pilot model shows initial ability to reproduce interesting phenomena: for example, adjacent repetition (e.g., 1<u>55</u>943) and same-position repetition (e.g., 1<u>5</u>4-3<u>5</u>3) can facilitate recall (R. N. Henson, 1998).

From these examples, it can be seen that the current working memory framework can reproduce a large range of phenomena about repetition and positional coding beyond the Divergent-Reconvergent and the Dual Representation model. They provide further support for the validity of these models.

5.5.3 Jackendoff's 4th challenge for cognitive neuroscience

A sequence encoded by an activation gradient can be stored into and retrieved from long-term memory in a straightforward way. It only requires that the activation pattern in the PFC-II layer be converted to an equivalent weight pattern, which can be done by Hebbian learning. However, when the activity pattern in working memory involves binding, long-term storage of this pattern is a difficult episodic memory problem. It's sometimes known as one of the four challenges from Jackendoff (Jackendoff, 2002).

The Jackendoff's challenges proposed in 2002 were those that needed convincing connectionist models to deal with, and are arguably fundamental to any connectionist models for linguistics and cognition. In full, the four challenges are 1) the binding problem 2) the problem of multiple instantiation 3) the problem of variables, and 4) the problem of binding for long-term memory. This thesis has to some extent addressed all the first three questions. The binding problem is addressed by the phase neurons and the PNN framework. The instantiation problem has been addressed by the counting property of neural assemblies earlier in this chapter. The problem of variables is seen as a special case of binding between symbols and values and is addressed in the above

repetition encoding theory.

The fourth challenge concerns the transparency of the boundary between working memory and long-term memory. He argues that linguistic tasks require the same structures to be instantiated in working memory and long-term memory and that the two instantiations should be functionally equivalent (Jackendoff, 2002, , p. 65). Working memory representations are typically implemented using activation levels and (possibly) temporal synchrony, whereas long-term memory is implemented using synaptic connectivity. The disparity of implementation media suggests that it would be difficult to achieve functional equivalence (Gayler, 2003). Here, I propose a model in which equivalence between long-term memory and short-term memory can indeed be achieved. The architecture of the model is shown in Figure 5.12. In this model, an activity pattern with temporal synchrony is represented by the Feature layer. The storage of the activity pattern is first stored by associating this pattern with an episode node in the episodic layer. Further, each of the synchronized group is detected by a coincidence detector (a group node) in the Group layer, and the group node is mutually connected with this group of features and the episodic node. These long-term memory weights can altogether fully represent the activity pattern in the working memory to the precision of binding relations. During retrieval, activation of the episodic node can reactivate all the feature nodes and the group nodes. The group nodes will first be desynchronized by their lateral inhibition, and then the desynchrony propagates to the feature layer and re-organize the activity into the original binding relations. This long-term memory system is adequate to store any activity pattern and their binding relations as required by Jackendoff's 4th challenge. This mechanism may be highly useful for the storage of linguistic information and other working memory information, such as repetition coding and general variable binding. Thus far, we have a connectionist framework which provides consistent solutions for all four of Jackendoff's challenges on connectionist models for linguistics and cognitive neuroscience.

Further, based on the discussion of positional coding and repetition coding above, we can see that the repetition problem faced by the activation-gradient-based model has been very well remedied by multiple complementary mechanisms. These mechanisms are summarized in Table 5.2, including phonological coding of 2 seconds of information, repetition patterns for regular and simple patterns, position coding for short lists (e.g., 3 or 4 items), and long-term chunks for sequences separable into shorter

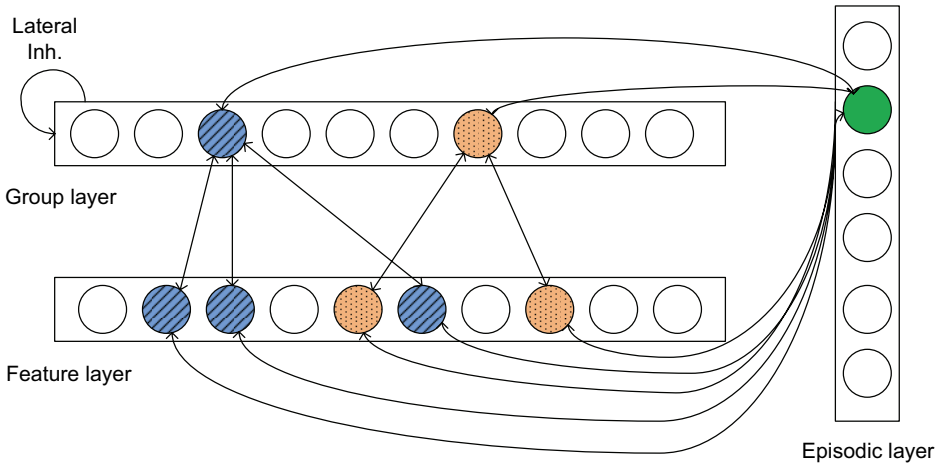


Figure 5.12: Long-term memory of activity patterns with binding

sequences.

In sum, the working memory neural network model has fairly successfully addressed the serial-order problem, the repetition problem and the binding problem. It is hoped that this model can be further refined and extended by other theories and models of working memory, attention, long-term memory and linguistic processing, and that the discussed neurobiological mechanisms may shed light on the neural substrates of working memory.

5.6 Conclusion

With the introduction of a phonological store and required control processes, the working memory model can account for more phenomena. They include PSE in words and the lack of PSE in nonwords, the modality effect that auditory presentation results in better recall than visual presentation (without recoding), and the interaction between articulatory suppression and the modality, word-length, and irrelevant speech effects. All these effects are produced by a coherent model without task-specific modifications.

Further, the model shows that item and phonological codes can interact in a cooperative and parallel way, instead of a serial and competitive way. Further, this chapter proposes a new serial order mechanisms for phonological encoding and retrieval, which is related to the anatomy of the auditory cortex. The modeling framework also allows for the modeling of complicated storage, retrieval, control, encoding (repetition

Table 5.2: Complementary solutions to the repetition problem

Solution	Condition	Limitation
Phonological coding	Information is encoded phonologically	2-seconds duration
Repetition patterns	Repetition in simple patterns	Not for nonwords or complicated patterns
Position coding	Automatically occurs for perceived items	Not for nonwords; imprecise for long lists
Long-term memory	Repeated are in different chunks	Requires learned chunks

encoding, position encoding) and long-term memory processes. These are exciting areas to explore.

Chapter 6

Implications and Discussion

The Divergent-Reconvergent model predicated on the basal ganglia is a serial order mechanism, which does not maintain the timing information in a sequence. It does not encode the precise interval between elements. Further, because cortico-basal ganglionic loops originating in different cortical areas are anatomically separate, it implies a difficulty for sequencing items belonging to different categories of information, if they are represented in distant cortical areas. That is, it may be difficult for the model to maintain the serial order of mixed sequences of words and hand movements, or sequences of mixed hand and foot movements, etc. In the linguistic domain, this may imply a difficulty of sequencing mixed consonants and vowels, or nouns and verbs (if they are adequately separate in brain areas). This chapter discusses these implications of the Divergent-Reconvergent model. It is first suggested that the cerebellum may provide additional fine-grained timing information for sequences to complement serial-order of the basal ganglia system. Second, it is suggested that to remedy the cross-category sequencing problem, a category label sequence may be concurrently encoded in working memory. This issue has several interesting implications, including the inevitable involvement of long-term memory to remember frequent category subsequences, and a possible relation to syntax. At last, it is discussed what implications the working memory model has for consciousness and free will in general.

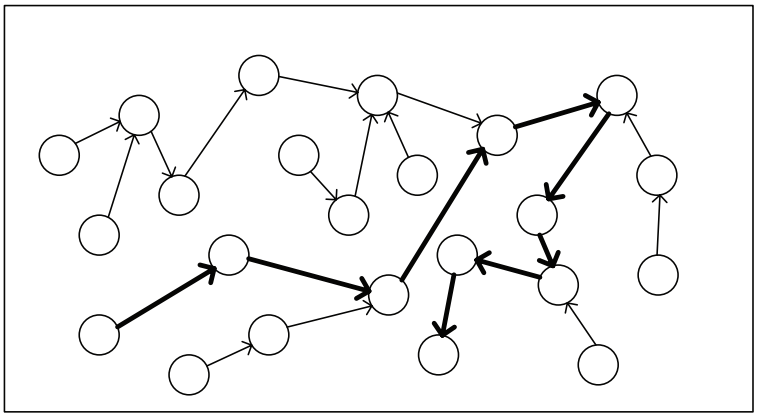


Figure 6.1: Random one-to-one connections form chains

6.1 Cerebellum and timing

The Divergent-Reconvergent model explains the how cortico-basal ganglionic loops encode the serial order of elements in a sequence, however, this model is not good for encoding timing information of the elements. The time of retrieval of the items depends on how fast a previous item is generated, and how the system controls the recall process. This suggests that the cortico-basal ganglia system itself may need to be accompanied by a parallel subsystem to provide better timing information. Given that the the cortico-cerebellar loop runs parallel to the cortico-basal ganglionic loop, and that the cerebellum is crucial for timing (Ivry, Keele, & Diener, 1988), it is possible that we can further pursue how timing information is encoded and retrieved for sequences, by integrating the cerebellum.

In this section, we take a special method in constructing the model. First, we start with constructing an arbitrary neural network which can learn timing for a stimulus-response pair. We try to keep it as simple as possible so that it is likely to exist in the brain. Then, we initially verify the model by comparing it with known timing properties in classical conditioning. Finally, we search the anatomy of the cerebellum for a functional equivalent of the model. If it is found, we can adjust the model to match the anatomy.

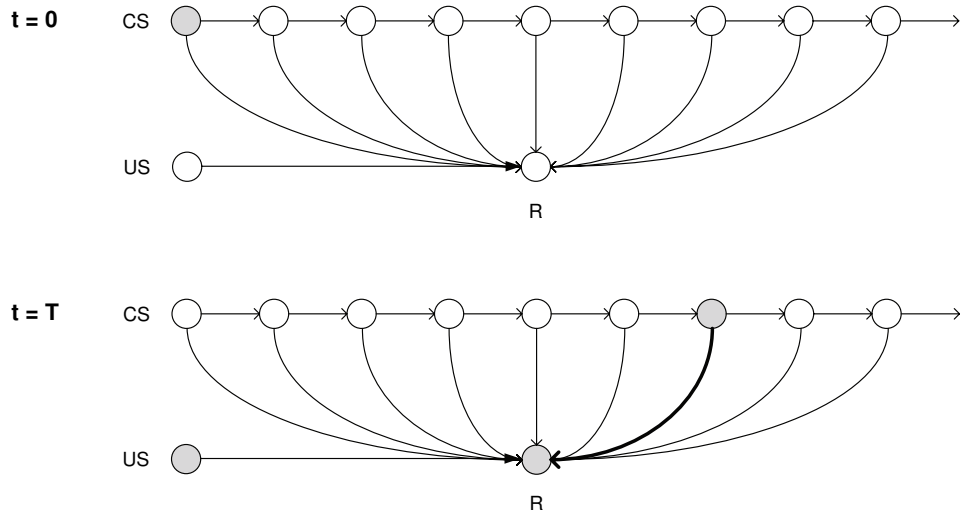


Figure 6.2: Chain model of S-R timing learning

6.1.1 The Random Chain model

Consider a brain area with a large number of neurons, where each one is randomly connected to another as Figure 6.1 shows. A notable emergent property of this area is that from any neuron, a long chain of neurons can be traced before they form a loop. Suppose the number of neurons is N, the probability $P(L, N)$ of having an acyclic chain of at least length L starting from an arbitrary neuron is

$$P(L, N) = \prod_{i=0}^{L-1} \frac{N - i}{N}.$$

For example, suppose there are a million neurons $N = 10^6$, starting from any neuron, the probability of the existence of a chain of a least 100 neurons without a loop is approximately $P(100, 10^6) \approx .995$. With the powerful one-to-one random connection scheme, we can expect to find long chains of neurons. A conceptual model of learning stimulus-response (S-R) timing by a neural chain is shown in Figure 6.2. The model consists of a neural chain, all neurons of which are projected to a response neuron area, where the neuron R represents a response.

The model works in the following way. During stimulus-response (S-R) pairing stage, the conditional stimulus (CS) activates a neuron in the poll, which is represented as the head of the acyclic chain in Figure 6.2. As time passes, the activity spreads along the chain, the activity of the neural chain is at a single neuron which represents the specific

delay time. At time T, the unconditional stimulus (US) activates the response (R) unit. Meanwhile, the synaptic connection from the active chain neuron to the response neuron is strengthened by Hebbian learning. After many times of S-R pairing, the connection will be strong enough, such that when the CS is presented alone, after time T, the response R will be elicited by the chain neuron that fires at time T. As long as transmission time along the chain is stable over time, the neural chain can very well encode the timing of the learned response.

Does this model have any neurophysiological predictions that can be verified? The model does predict two properties: 1) Over-training will decrease the US-R time delay. 2) There is no chronotopic maps in the brain area which encodes time. The first property comes from a side-effect of Hebbian learning in the chain. When the chain is activated multiple times, not only is the neuron with delay T connected with the response neuron, but also the neuron preceding the delay T neuron because its activation shortly precedes the response neuron. After training, this preceding neuron may activate the response R by itself, and thus shorten the CS-R delay time. This property has indeed been found in experiments, where in well-trained animals, the conditional response (CR) occurs before the onset time of the US – known as *anticipatory CR* (Desmond & Moore, 1988). The second property, that there is no chronotopic maps in the brain, is also consistent with current status of physiological studies. No chronotopic maps have been found in any brain areas (Ivry, 1996). This property follows the way in which the neural chain which encodes delay times is formed. This chain is not formed by connecting spatially adjacent neurons on a specific direction, but randomly connecting any pair of neurons. Thus, there is no chronotopic map in this area.

6.1.2 Cerebellar mechanism of timing

Many models of cerebellar processing of time have been proposed. One of the timing mechanisms is the use of a delay line (Zipser, 1986), which is similar to the Random Chain model, but it has only a few neurons in the chain. This type of models makes good explanations why the interval between CS and CR gradually shortens with more training (Desmond & Moore, 1988; Moore, Desmond, & Berthier, 1989), as the Random Chain model. Compared to those delay line models, the Random Chain model is novel in that it also explains how a long chain can be formed to represent time of up to a few seconds. Moore et al. (1989) proposed a cerebellar implementation of the

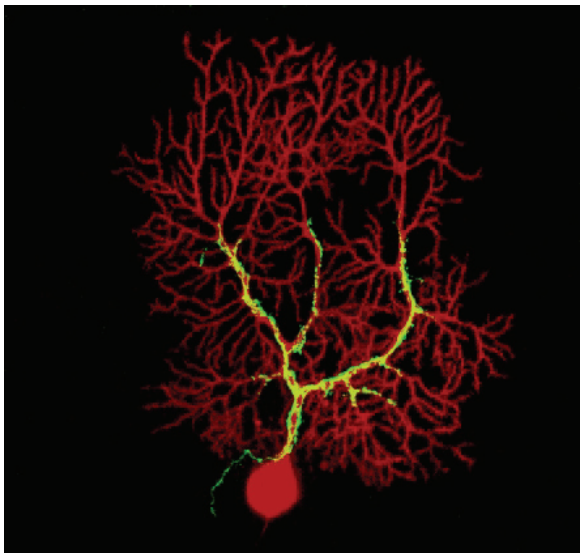


Figure 6.3: Each Purkinje cell receives a synapse from only one climing fiber. A climbing fiber was labeled with a green calcium indicator and a Purkinje cell was labeled with a red dye. (Source: Haugland, 2005)

delay line model. They tentatively assume that the delay line is implemented by a series of neurons in the pontine nucleus, based on the reason that the pontine nucleus is one route by which information from CSs can ascend to cerebellar cortex to form robust conditioned responses. This mechanism is questionable, because neural transmission speed is fast and it is hard to believe that a small nucleus contains a long enough chain of neurons to subserve long delays, unless it can be justified in details. Below, we will proceed with an analysis of the cerebellar anatomy, in search of a more plausible neural underpinning of the theoretical Random Chain.

The crucial requirement of the Random Chain is that neurons have one-to-one connections. This requirement is very strict, because even if a neuron has only two or three connections, within a few synaptic connections, the number of activated neurons will quickly explode. The number of activated neurons in the chain is an exponential function k^n, where k is the number of connections each neuron makes, and n is the number of relays. Suppose the delay of 1 sec requires 100 relays and the connection factor $k = 2$, this model would activate a maximum of $2^{100} \approx 1.27 \times 10^{30}$ neurons, which are already more than the total number of neurons in the brain. Therefore, to make the chain model work, the brain would have to evolve a special type of neuron which makes exactly one connection to another neuron, and it would help if the circuit can reduce

the rate of neural relays. We start by looking for one-on-one synaptic connections in the cerebellum. This turns out to be very easy: the major cerebellar cortical cells – Purkinje cells – each is innervated by a *single* climbing fiber from a neuron in the inferior olive. An image is shown in Figure 6.3. The climbing fiber from an olivary neuron climbs up the dendritic arbor of a Purkinje cell, and makes extraordinarily strong synapses to them, such that a single action potential in a climbing fiber is sufficient to trigger a sodium spike, followed by a short burst of spikelets in the Purkinje cell. The inferior olive, which sends the climbing fibers, has indeed been implicated in various functions, including the timing of movements (De Zeeuw et al., 1998).

According to the Random Chain model, the chain would be implemented in a pathway from the inferior olive to the Purkinje cells, where the one-to-one connection is made, and then the purkinje cells should project back to the inferior olive, where hopefully the connections are also one-to-one. Let's follow the neural connections in the hypothetical loop and analyze the existence of predicted structure. First, the loop structure indeed exists. It is known as the olivocerebellar loop, composed of the inferior olive, purkinje cells in the cerebellar cortex, and the deep cerebellar nuclei, as shown in Figure 6.4. The mammalian inferior olive is composed of the principle olive, the dorsal and medial accessory olives, and several small subnuclei. In general, each olivary subnucleus projects contralaterally to one or more longitudinal zones of Purkinje cells. The Purkinje cells project to a deep cerebellar nucleus, which in turn projects back to the same olivary subnucleus, forming a loop. In addition to the loop connections, the olivary subnucleus also directly send collaterals to the deep cerebellar nucleus. This anatomical unit consisting of a particular Purkinje-cell zone with its specific olivary input together with their innervated cerebellar nucleus, has been named a cerebellar module (Voogd & Bigare, 1980).

Does the olivocerebellar module support the circulation of one-to-one neural activity? It does not appear so because each Purkinje cell innervates multiple neurons. Previous computational work of a similar timing mechanism indeed does not entertain this idea (Kistler & De Zeeuw, 2002), though they assumed that each Purkinje cell innervates only a small number of neurons. That is, the reverberation in the loop is assumed to be divergent. An immediate problem is that with divergence, olivary activity can increase without bound. In their models, Kistler and de Zeeuw introduced an additional shunting inhibitory projection. The co-existence of a shunting inhibition could restore

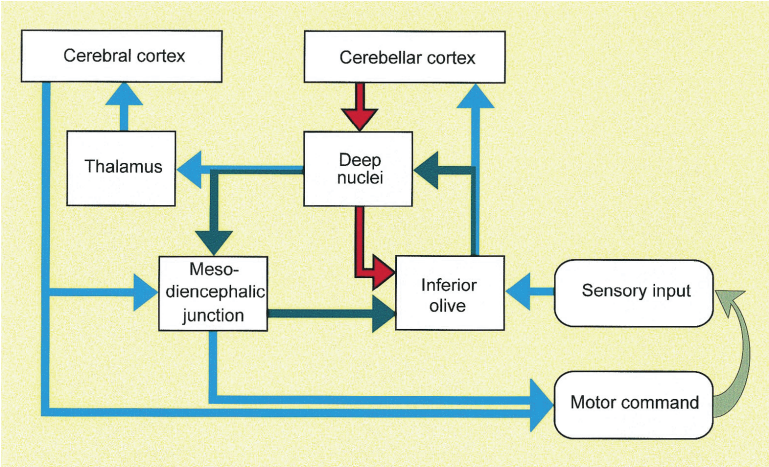


Figure 6.4: The inferior olivary network. The loop composed of the inferior olive, cerebellar cortex, and the deep nuclei is the hypothetical timing loop in which the Random Chain exists. (Source: De Zeeuw et al., 1998)

the stability of the network by forming limit cycles. However, when neural activity is trapped in a limit cycle, it will lose the ability to unambiguously encode time. In their model, they assumed that biological noises play a role in perturbing neural activity trapped in limit cycles. The cost is a rapid loss of information in the loop. They showed that the activity encoding a stimulus is almost completely lost in about 20 iterations. Because activity propagation in the olivocerebellar loop has a latency of about 100 ms, 20 iterations equate an acceptable time-encoding capability of about 2 seconds. Yet, the excitatory divergent and shunting inhibitory divergent projections might not be the only possible stabilizing mechanism. From the Random Chain model, we could expect that one-to-one connections would be better since it maintains the information perfectly. Here, we further seek neural mechanisms which make one-to-one connections possible.

A special property of the olivary neurons made us suspect that some slight divergence may be reduced back to one neuron by a dimension reduction mechanism. This special property is the existence of electrical coupling between olivary neurons by gap junctions. The gap junction is shown in Figure 6.5. A functional significance of the gap junctions is that the output of neighboring olive neurons are perfectly synchronized. This synchronized olivary activity diverges into a larger number of neurons in the pathway, but they are likely to be later reconverged due to the coincidence detection or dimension reduction properties of the neurons (Oja, 1982). After the divergent activity has been reconverged, olivary activation level can maintain stable, instead of being subject to ex-

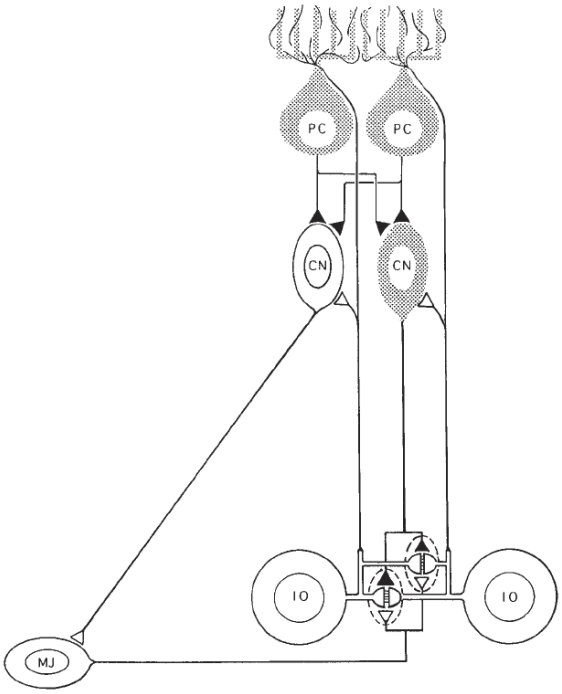


Figure 6.5: Diagram of the neuropil in the inferior olive, and its relation with the cerebellum (top) and mesodiencephalic junction (MJ, left). All olivary spines (half circles) are located within glomeruli (doted circles), and innerated by both an excitatory mesodiencephalic and an inhibitory cerebellar terminal (white and black triagnles, respectively). The olivary axons provide climbing fibers to the Purkinje cells (PC) in the cerebellar cortex and collaterals to both the inhibitory and excitatory cerebellar nuclei (CN) neurons. The inhibitory projection neurons in the cerebellar nuclei project exclusively to the inferior olive (IO), while a substantial part of the excitatory projection neurons in these nuclei innervate the neurons in the mesodiencephalic junction that in turn project back to the inferior olive. The excitatory and inhibitory neuron sin the cerebellar nuclei can receive input from the same Purkinje cell axon. Small lines between olivary spines indicate the dendrodendritic gap junctions by which they are electronically coupled. (Source: De Zeeuw et al., 1998)

ponential increase. In this stable field activity, the detailed pattern of activity can change in a time-dependent deterministic way as the Random Chain mode shows. Compared to the model using shunting inhibition and noises to maintain olivary stability (Kistler & De Zeeuw, 2002), this model can avoid rapid loss of information and can support longer timing intervals. The hypothesis that the gap junctions support time encoding is consistent with the result that the removal of electrical coupling of olivary neurons by deletion of the Cx36 gene adds 10-20 ms error to fine temporal coordination of muscle firing during movement (Placantonakis, Bukovsky, Zeng, Kiem, & Welsh, 2004).

After the comparison with anatomy, it can be seen that the required elements of Random Chain model indeed exist. The model, as planned, can be grounded in real neural anatomy with the following revisions: 1) Each stimulus item at time 0 is not encoded by a single neuron, but a small number of olivary neurons to ensure robustness. 2) This set of olivary neurons activate the same number of Purkinje cells. 3) Purkinje cells project back to the olivary cells in a slightly divergent-reconvergent way. The possible increase of active cells through the loop is reconverged by dimension reduction along the pathway, because the input into the loop is highly synchronized due to electronically coupled olivary neurons.

With the neural mechanisms pinned down at the olivocerebellar loop, we have new interpretations of the functional roles of several other anatomical structures in the cerebellum. *First*, the Random Chain model requires a braking mechanism to stop activity circulation in the loop. This braking mechanism is indeed found in the cerebellum. Over the Purkinje layer, there is a molecular layer with interneurons (stellate and basket neurons) which can strongly inhibit the Purkinje cells. They can be used to stop current activity in the olivocerebellar loop and reset the timing for later learning or execution needs. *Second*, the speed of human motor sequential responses can be adjusted in a relative way. That is, people can speed up or slow down responses while keeping the relative temporal structure of the motor sequence. In the Random Chain model, this can be done by increasing neural transmission speed in the area containing the major chain cells in a uniform way. In the olivocerebellar loop, the parallel fibers may fulfill this functional role (Figure 6.6). They weakly activate a large number of Purkinje cells in a nonspecific way. This can be seen as a mechanism to adjust the execution speed of the timing loop. *Third*, each olivary neuropil[1] receives an inhibitory input from the Purkinje cell

[1]Neuropil consists of unmyelinated neuronal processes (axonal and dendritic) within the gray matter of the central nervous system.

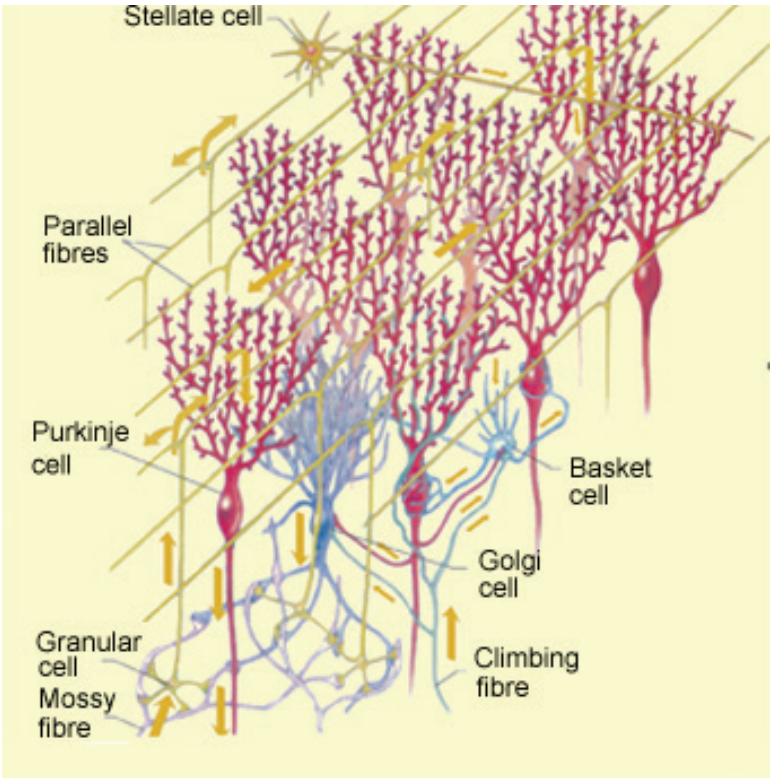


Figure 6.6: Parallel fibers may serve as a speed adjusting element

and an excitatory input from an indirect pathway via mesodiencephalic junction (Figure 6.5). The function of the structure, according to the Random Chain model, would be 1) the direct inhibitory pathway deactivates olivary activity representing the previous time step; 2) the indirect excitatory pathway then updates olivary activity to represent the next time step. From the above analysis, we have shown that the hypothetical model can very well map onto its theoretical neural substrate – the cerebellum.

Compared to the similar olivocerebellar timing model (Kistler & De Zeeuw, 2002), the current model shares some features but also has additional advantages. For example, both Kistler & de Zeeuw's model and this model explain the irregularity of the activity of individual neurons in the olivocerebellar pathway. It had been proposed before that the inferior olive uses periodical synchronized activity of 10 Hz to encode time. A mystery was that no 10 Hz periodic neural activity is found in Purkinje cells, or the deep cerebellar nuclei (Keating & Thach, 1997). Kistler and de Zeeuw's model (2002) explains this fairly well: the olive uses synchronized neural patterns which changes at a rate of 10 Hz to represent time, however, individual neurons in the olive can have a lower firing rate (1 Hz) and can be aperiodic. The Random Chain model shares this property. It does not predict individual neurons to have periodical activity. In the simplest form, an acyclic loop, each neuron fires at most once. In more general random network with loops and cross-talks between different chains, neurons are predicted to fire in an irregular but deterministic way, as Kistler and de Zeeuw's model points out. The additional advantages of the current model are 1) it posits a slightly divergent-reconvergent loop which can, in effect, function as the theoretical one-to-one connection; in this way, it can avoid the vulnerability of simple one-to-one connections, and also avoid the hypothetical shunting inhibition mechanism and its concomitant rapid information loss in the loop. 2) It predicts the existence of other cerebellar structures: the parallel fibers whose role may be to adjust timing speed by shortening or elongating the loop latency; the molecular layer whose role may be to brake and reset the timing loop. The model thus fits best with existing anatomical and neurophysiological evidence.

6.1.3 The Cerebellar Random Chain model

Figure 6.7 shows the improved Random Chain model based on cerebellar anatomy. The inferior olive represents the time with a synchronized pattern consisting of multiple active neurons. This pattern innervates the same number of Purkinje cells, which

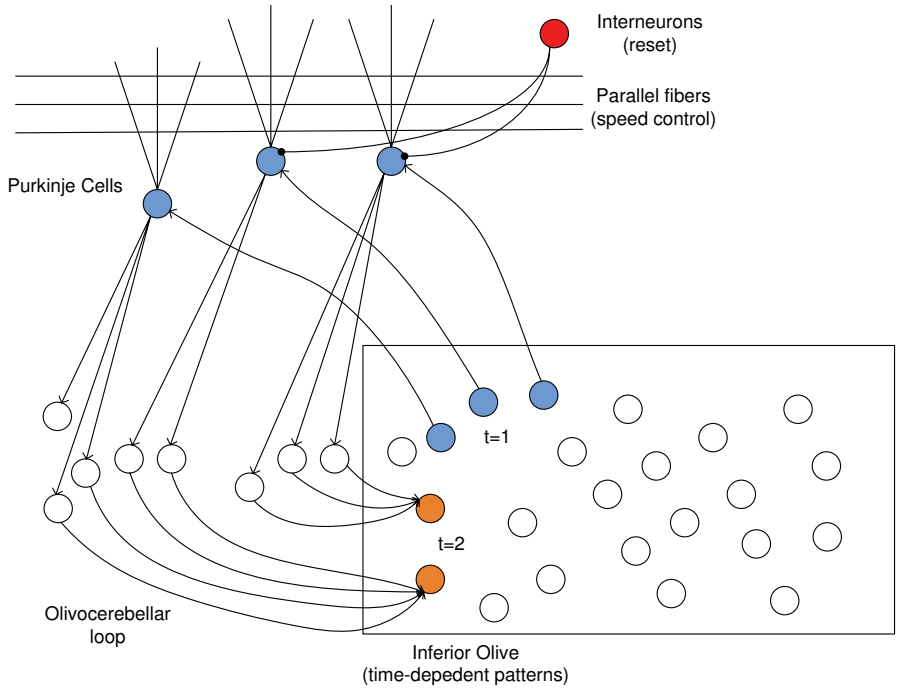


Figure 6.7: Cerebellar Random Chain Model

project to a group of neurons in a slightly divergent manner. The divergent activity is then reconverged by dimension reduction and finally travels back to a small number of neurons in the inferior olive. The exact number of active neurons representing each time step is not crucial. The crucial criterion is that the activity should not diverge without bound, or diminish. In this sense, the hypothetical mechanism – divergence followed by dimension reduction – is better than the shunting inhibition mechanism Kistler and De Zeeuw (2002), as explained above. The time-dependent activity in the Inferior Olive can be used as individual nodes in the above simple Random Chain model, which can be associated with a conditioned response for the brain to learn the timing of a response (or an unconditional stimulus). In some complex behaving situations, the speed of executing learned responses or motor sequences, can be controlled by adjusting the input activity to the parallel fibers. An executing sequence can be stopped by 1) activating the interneurons that inhibit the Purkinje cells to stop the timing mechanism, and 2) activating the No Go pathway in the cortico-basal ganglionic loop to stop the sequencing mechanism (Chapter 4).

6.2 The Divergent-Reconvergent Architecture and Sternberg effect

The working memory models studied in this thesis have been configured to perform the Immediate Serial Recall task. This task has been most commonly used to probe the characteristics of working memory. From the angle of the models in the thesis, this property can be seen as a result of the requirement of minimal control, or minimal Central Executive intervention. Due to the activation gradient automatically formed in the cortico-basal ganglionic loop, this recall scheme basically requires no additional control but to enable the competitive recall process upon a recall signal, and delay the retrieval of a next item while the current item is being produced. With the formulation of the Central Executive as a Signal-Action mapping component, its "program" in the Immediate Serial Recall task is a small set of Signal-Actions pairs (see Section 5.1.1).

However, there are tasks which require more control. In order to show that the architecture used in this work is not restricted to serial recall, it would be nice to show that the architecture can also perform other tasks. Here, I discuss its possible application to the well-known Sternberg task (Sternberg, 1966), and an explanation of the mysterious Sternberg effect. The Sternberg task is the following. A short sequence of symbols are presented for a subject to remember. Then, the subject is presented a probe symbol and is required to judge whether the probe is contained in the sequence. The mean response time is found to increase linearly with the length of the sequence, and there is no difference between "yes" and "no" responses. The result suggests an *exhaustive search* strategy used by the brain. Because it would seem more efficient for the motor response to begin as soon as a positive match occurred (i.e., self-terminating search), some have argued that exhaustive search is implausible. Besides the mystery of exhaustive search, it is also striking that the discovered speed of scanning – 25 to 30 symbols per second – is about four times as the maximum rate of subvocal speech (Sternberg, 1966).

More recently, the fast-speed scanning process has been related to the gamma-band oscillation in the cortex (Jensen & Lisman, 1998). This new model assumes that the content items are associated with peaks of a gamma oscillation. In each cycle of the gamma wave, an item can be compared. This explains the speed of scanning. Further, this model assumes that a theta wave controls when the subject can make a response. It was found that if the phase of the theta wave is reset upon the onset of the probe, or its

frequency is modulated by the number of items, the model can produce characteristics of an exhaustive search.

Here, I propose an alternative explanation of *exhaustive search* based on the working memory architecture studied in this work. The basic story is the following. Before scanning, all items are loaded in the Retrieval layer. Upon a probe, a scan process begins. All the items in the Retrieval layer are allowed to propagate to the Output layer and then to the IOR layer. Because different items are desynchronized, they arrive at the Output layer at different times (in a fast gamma-band frequency). Each retrieved item can be compared with the probe, and if a match is found, a "yes" response is prepared. For reasons explained below, the response is only "prepared" but not immediately released. It is released only after the Retrieval layer containing all list items is fully inhibited by the IOR layer. That is, after all items have been scanned – the exhaustive search scheme.

Why is the "yes" response not immediately released, but waits until the Retrieval layer is clear? The working memory architecture provides a novel but consistent answer to this question. That is, any newly prepared response is not immediately released, but temporarily inhibited until a signal removes the inhibition. This is not an *ad hoc* modification for the Sternberg task, but a control strategy necessarily learned for serial recall. Let's see why this control strategy is learned by the Divergent-Reconvergent model. In this model, after the first item is retrieved and inhibited, the next item is ready to be output. However, at this time, the output system which converts the first item into a motor sequence is busy. The next item *must* be temporarily inhibited until the first item is fully produced. The end of production is signaled by an "empty" Retrieval layer. So the system would have learned this control rule, long before it performs the Sternberg task. Therefore, in the Sternberg task, the next response "yes" or "no", has to wait until the Retrieval layer used for scanning has been cleared. This equals an *exhaustive search* strategy.

This explanation of the Sternberg task, of course, is only a speculation. However, it can be computationally tested in a hierarchical version of the Divergent-Reconvergent model, where a downstream component outputs a sequence for each item in the upstream component, and inhibits the upstream component until the output is done. The scanning process can be controlled by a small program in the Central Executive, and its timing can be modeled by requesting that the PNN framework temporally separate neural ac-

tivation with different phrases (by default, activations with all phrases are produced simultaneous in one time step). This experiment would be a good test for examining the generality of the working memory architecture. A prediction from this model is that with extensive training, the subject, either a monkey or a human, can possibly increase the speed of "yes" responses (and making it a non-exhaustive search), because it can learn a new control program to remove response inhibition earlier than the end of scan. This prediction provides an interesting criterion to distinguish this model and other explanations. Without modifying the exhaustive search behavior, this model also has a nice explanation of why "yes" responses are slightly faster than "no" responses. This could be because a "yes" response is prepared during the scanning process, while a "no" response is prepared after the end of the scanning process. The parallelism of the "yes" preparation and the scanning process can save some time.

6.3 Syntax and sequencing across categories

The Divergent-Reconvergent model predicts that items represented in different brain areas are more difficult to sequence than those in the same area, because cortico-basal ganglionic loops are anatomically separate. This limitation may imply difficulties in sequencing nouns and verbs. Aphasic studies and brain imaging studies suggest that nouns are located in the anterior temporal regions, while verbs are in the inferior frontal regions. An implication is that an additional mechanism may be needed to robustly sequence verbs and nouns.

This mechanism, the author hypothesizes, is concurrent coding of category sequences. For example, the word sequence "Trees are plants", can be recalled better if it is encoded with a category sequence "N V N". During recall, the category sequence can be first reproduced, and the category label is used to select the word class. This hypothesis is consistent with evidence showing that recall errors are often category-bound, where the categories can be word classes or phonological categories (vowels and consonants) (Hartley & Houghton, 1996).

The category sequencing system is hypothesized to be an instance of the Divergent-Reconvergent model, because of the ubiquity of cortico-basal ganglionic loops. A very important implication following this hypothesis is the difficulty with sequences containing repeated items. The Divergent-Reconvergent model will not immediately handle the abundant repetitions in category labels, such as the "N V N" sequence. However, this

can be seen as a driving force for the system to *learn* to use chunks to reduce repetition. For example, the subsequence "V N" can be chunked as a new category VP, if it occurs frequently. After chunking, the repetition is removed and the category sequence becomes "N VP", which can also be shown as a hierarchical tree S(N VP(V N)). This structure is readily interpreted as a phrase structure for the sentence "Trees are plants"[2]. The individual category sequences, VP = V N, and S = N VP, can be frequently reused for other word sequences, thus, may become a common structure to organize words for communication.

At this stage, the working memory theory clearly meets linguistic theories. Here, it is elaborated a little bit about how a category sequencing mechanism in working memory can help develop a stable syntax by learning and memory. Let's assume that a group of individuals will circulate words (that is, repeat messages that they hear), and they prefer word orders which *match* category sequences they have learned. Matching simply means a category sequence S contains the categories of all the to-be-produced words W. For example, for the words $W = \{\text{climb (V)}, \text{trees (N)}\}$, both the sequences $S_1 = \text{VN}$ and $S_2 = \text{NV}$ would match. They can compete for controlling the word order when producing W. It can be imagined that if one category sequence S becomes a little more popular, it will gradually dominate over the other. This is driven by a working memory property: if a circulating message matches a popular category sequence S, it may be remembered better by more listeners, and may be repeated more successfully. This will make S more popular in the group. The magnifying effect can eventually let S dominate over other less popular word orders until it becomes a standard in the whole group. In sum, we can see that a category sequencing mechanism in working memory can unify syntax.

As mentioned above, the category sequencing system is most likely to shape syntax into a hierarchical one driven by the factor: reduction of repetitions and the length. This process can take place by reinforcement learning by performance criteria. That is, category sequences useful for increasing memory performance are reinforced in the individuals, and further gradually become the syntax of the whole group. The uniformity of language syntax doesn't require any language-specific innate rules, but only a not-too-complex memory architecture.

This theory contrasts the belief that each linguistic universal is genetically

[2]Of course, from this toy example, the system has not determined the more general structure S(NP VP(V NP)) for the example sentence.

coded. From this memory and learning theory, how linguistic universals are developed in individuals may be re-analyzed. Because detailed linguistic rules are very unlikely to be genetically transmitted (e.g., through rule-specific language genes), this theory offers new angles in viewing the innateness of syntactic knowledge.

6.4 An alternative criterion for grammaticality

The category sequencing mechanism encourages word orders to conform to a system of phrase structures, which maximizes the chance of re-using learned category sequences and maximizes serial-order memory performance. The category sequencing mechanism resembles an implementation of a context-free grammar. However, it is worth noting that grammaticality in this working memory architecture is not best defined by a formal grammar, but a more general and fundamental criterion.

An important aspect of this work is that working memory not only stores a set of active items, their serial order, but also their binding relations. Thus, communication between two individuals' working memory would entail the transfer of both the items in order, as well as their binding. For example, to communicate the idea that "the rectangle is red", the listener has to bind the concepts "rectangle" and "red". Here, the author adopts the synchronization theory of binding (Gray, 1999; Singer, 1999). According to this theory, the neurons representing "red" and "rectangle" must be synchronized. If the listener can perform the right synchronization process after receiving the sentence, she is said to have comprehended it.

From these premises, an alternative criterion for grammaticality can be defined as follows. A sentence is grammatical if it can adequately inform the comprehender about how to bind the conveyed elementary concepts, given the comprehender's prior knowledge. This criterion emphasizes the condition *correct binding*, but it does not constrain the form of the sentences, as formal grammars do. The relation between a formal grammar and this criterion is that formal grammars, such as the context-free grammar, can help the comprehender perform correct binding with a little extra help – a binding protocol. For example, using the phrase structure rule "Object → Color Shape", and a binding protocol "bind the right-hand-side of any production rule", a listener can correctly bind "red" and "rectangle" after hearing the phrase "red rectangle", because the categories Color and Shape are synchronized, which then synchronize the words "red" and "rectangle".

Let's see an extra example from index grammar. The sentence structure $N_1N_2V_1V_2$ communicates the bindings: N_1V_1 and N_2V_2. This relationship cannot be expressed by a phrase structure rule. But it can be constructed by the following binding protocol: index nouns and verbs separately, and bind those with the same index. The two examples show that the traditional definition of grammaticality can be alternative defined in terms of binding communication. The above criterion is at work here, which can be interpreted as checking whether a binding protocol exists and tells how to bind the elements in a sentence. Of course, if binding protocols are unrestricted, there would be no way to determine grammaticality by this criterion. Thus, binding protocols require a formal specification and restriction. This happens to be feasible in the working memory model, because the binding protocols can be identified to a program in the Central Executive. Such a program already has a formal definition: a set of Signal-Action pairs. The signals and actions are also restricted by the working memory architecture. Therefore, this architecture can potentially support this alternative grammaticality criterion.

Using the working memory architecture, and the formal definition of its programs, this computing machinery can be used to decide grammaticality of human languages. By an analogy with automata deciding acceptability of formal languages, this machinery can be seen as a natural language automaton (NLA), which decides the acceptability of natural languages. It can be programmed to simulate any deterministic automaton or Turing machine, however, it lives with the strengths and weaknesses of human working memory, and has the natural tendency to process natural languages in ways close to humans.

It is worth pointing out that the above criterion of grammaticality has temporarily adopted the view that grammaticality is not equated to meaningfulness. This criterion only requires that the comprehender be informed about how to bind, but it does not care whether the binding can be actualized (i.e., resulting in synchronization). The latter depends on the elements' semantic compatibility and the context. To avoid a theoretical and vague "grammaticality", one can adapt the above criterion to test comprehensibility, which goes: a sentence is *comprehensible* if the comprehender is adequately informed about binding (using the above criterion) and the binding can be *realized* by neural synchronization. The latter judgment can use physical laws that govern the synchronization of oscillators, which are good models of cortical columns. Of course, understanding binding in semantic processing and comprehension would require

a deeper understanding of the brain's semantic representations and processes. However, the idea of developing a formal model which tests natural language grammaticality or comprehensibility by a more general criterion is attractive, and may be a fruitful direction for future research.

6.5 Conditions on synchronization and implications for consciousness

Consciousness is thought to be highly related to working memory. This section deviates a little bit from the systematic methods of studying working memory to discuss some general issues about consciousness. Some views are better situated within a working memory theory as well as within computational models of cognition.

Much of the current debate about consciousness is driven by a distinction proposed by David Chalmers (1996) between the "hard" and "easy" problems of consciousness. The "easy" problem is to understand the nature of the neural events without which there would not be conscious experience, a set of questions with which most neuroscientists are well acquainted, for example: How does the brain bind together into a unified perception the output of various parallel processes occurring in the brain? How is memory stored and retrieved? What are the mechanisms involved in and the function of selective attention? In contrast, the "hard" question asks for a general explanation of consciousness. How is it possible that there are conscious beings in a physical world at all? Why do some neural states result in conscious experience and not others? According to Arciszewski and Akins (2000), insofar as one believes that there is a hard problem of consciousness, pointing towards the kinds of neural states that correlate with certain kinds of conscious events is not an adequate response. Though providing such a correlation is a far-from-trivial task, the "how" and "why" questions would remain open. This section discusses a little bit about both questions, from the point of view of the working memory model studied in this work.

6.5.1 From a grain of fact to look at consciousness

A subliminal stimulus not reaching human consciousness can affect behavior but it is not subject to conscious control. How does this suggest for a theory of consciousness? First, as we have seen that low activation results in oscillations at sub-gamma frequencies.

Only after it reaches a threshold, a stimulus representation can be stably synchronized with other components in the cognitive system, which oscillate in the gamma band. This reverberates the idea that consciousness has to do with gamma oscillations. But because gamma oscillation can happen everywhere, it is not interpreted as consciousness itself, but only as a tool to make consciousness possible. A hypothesis studied in this section defines consciousness as the following.

Hypothesis Consciousness is what is synchronized with a family of control handles in the prefrontal cortex.

From the working memory point of view, consciousness may simply be a side-effect of the evolution of cognitive control. The subjective experience can simply be interpreted as a collection of control handles, with choices about "location, color, shape, etc." The whole collection of the control handles self-organized into a nice menu is our visual experience: a colorful, vivid scene with objects positioned in it (see Figure 6.8). It is quite different from "Computer's menu" (top-left panel). I suspect that the visual experience is mapped on a 3D space in some people, or a 2D space in some other people, explaining why people are differently capable of spatial cognition.

The above self-organization process of control handles (or more conventionally, visual features) can be imagined as follows. We start with the Computer's menu, with different visual features represented separately. Imagine that the green color moves to a colorless tree shape, forming an image of a green tree. Next, the green tree moves to its target location and stays there. As a result, in your imagination, there should be a green tree with a definite spatial location. This resembles our visual subjective experience – all types of visual features are glued together. The holistic representation is a more powerful interface to the world for us to control our perception and behaviors, but this doesn't necessarily change its nature.

This view of subjective experience is not just to repeat the idea that it is a mental representation, or consciousness has a control function. But it operationalizes a two-way connection from a computational working memory model to the subjective experience, and the way back. The model→experience direction is established this way. By the hypothesis that consciousness is the set of units synchronized with a selected family of control handles, we can *determine* which units in the model are included in consciousness. From there, we can use the contents they represent to show the model's subjective experience. Though the way we show it may not be the way we see the real world,

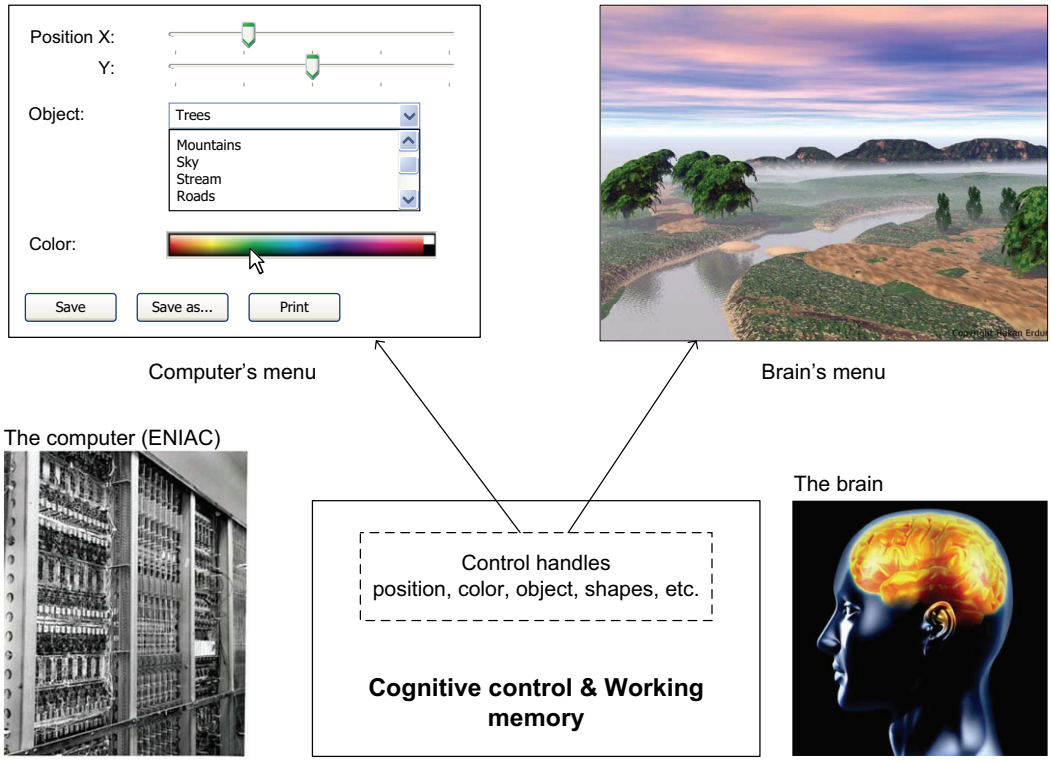


Figure 6.8: The control menu metaphor of consciousness

with the menu metaphor, we can interpret it that way. The reverse experience→model direction can also follow. What we do with the subjective experience, such as directing our attention to an object or imagining a new object, can all be mapped onto a working memory model, by changing either its content or attention focus. With the two-direction mapping established, we can begin to use a computational model to study consciousness.

Within a computational model, those "control handles" are equated to active units in the prefrontal cortex. This PFC-centered view is specific to mammals, and doesn't apply to animals whose working memory is not located in the prefrontal cortex (e.g., reptiles and amphibians). The hypothetical handle units are a little bit different from content units in the working memory model, in the sense that they do not have fixed contents, but can serve as "indexes" to access other contents in working memory. However, similar to other PFC units, they are desynchronized among themselves by lateral inhibition. This allows for grouping of the whole conscious experience, where features can be grouped into different objects by binding with different handles.

By this definition, we can also manipulate the conscious content of the model at any moment by varying what is synchronized with these handle units, and study its behavioral outcomes. We can also lesion the system to see how it effects consciousness. This hypothesis also provides some predictions about human consciousness. It predicts that if PFC is inactivated, there will be no conscious experience. This hypothesis can also be tested with different types of animal behaviors, since there are conscious behaviors resulting from thinking and working memory, and many animal behaviors that are produced without routing to the prefrontal cortex.

The following two sections discuss some implications of this hypothesis about the subjective experience and qualia.

6.5.2 What is the subjective experience or qualia?

Qualia is broadly defined as the subjective experience of stimulus, and narrowly defined as indivisible elements in the subjective experience, such as the redness in red. What generates qualia is an elusive question, because it is not knowable whether the redness in red really exists, since we don't know if the redness in one's experience is the same as the redness in another's experience. If they are not the same, there's no need to discuss a single qualie by itself. It is more tangible to study its relations with other experiences. For example, given the experience or qualia $Q(C)$ of a stimulus

C that is composed of elements A and B, some thing we can do is to study if $Q(C)$ is composite or not. Being composite means there are weights w_1 and w_2, such that $Q(C) = w_1Q(A) + w_2Q(B)$. Being not composite, or incomposite, means that $Q(C) \neq w_1Q(A) + w_2Q(B)$ for any w_1 and w_2. Because for different people, $Q(A)$ and $Q(B)$ are not necessarily the same, we cannot tangibly study them alone. However, we can study their relations with $Q(C)$. If we can manipulate the experience so that for subject X, $Q(C)$ is composite, but for Y, $Q(C)$ is incomposite, we would begin to have a scientific measure of qualia, and a model to explain it. The neural mechanism that can determine whether an experience is composite or incomposite is discussed in the next section.

6.5.3 Conditions on synchronization and consciousness

From the analysis of synchronization of cortical columns (minicolumns), it is known that for cortical neurons to synchronize, they would need a minimal level of activity and excitatory connections. The minimal level of activity puts the minicolumns containing the neurons around a common frequency, with which they can synchronize with other columns. Too little activity would result in inability to synchronize (though it may have priming effects). Based on these basic properties of synchronization, we may predict which part of neural activity in the brain is conscious and which is not. This section discusses this possibility. The above hypothesis of consciousness is first elaborated as follows:

1. Consciousness requires spread of neural synchronization.

2. For an operational definition, a small set of root working memory neurons i in the prefrontal cortex are presumed to have consciousness.

3. The consciousness of neurons i can spread to content neurons in other areas of the brain to build up a bigger conscious experience, if and only if these content neurons can be *actively* synchronized with the root neuron. Active synchronization means that the content neurons are not coincidently synchronized with neuron i, but by reliable mutual (direct or indirect) connections with neuron i, so that they remain synchronized when neuron i changes phase.

4. Each root neuron represents a part of the whole conscious experience. This experience is operationalized as a set C_i of all neurons actively synchronized with the root neuron i.

5. Total consciousness experience at any moment is the union $\cup_i C_i$.

The core of this hypothesis is to define the conditions for a neuron to actively synchronize with a root neuron i. In a spiking neural network (or equivalents), this implies two conditions:

1. **Activity condition**. The neuron must be adequately active. That is, the minicolumn it resides in must be have a population rhythm that is close to that of neuron i. Weak activation of the minicolumn makes it unable to synchronize with neuron i, and remain unconscious.

2. **Connection condition**. The neuron must have mutual connections with neuron i (direct or indirect), so that they can remain synchronized when either one changes phases due to noises or additional input. An intuition here is that synchronization must remain long enough to take a significant effect.

Now we can start to see how the hypothesis works in some conditions for consciousness, as tabulated in Table 6.1. It systematically and computationally explains why sensory perception is conscious, while the priming effect, habitual actions or involuntary muscles are not conscious. Because any input to the prefrontal cortex (outside the prefrontal area itself) must pass the thalamus, it explains why lesions in the thalamocortical system always cause consciousness impairments (Baars, 2005). Dreams are a special form of consciousness. It occurs during REM (rapid eye movement) sleep. REM sleep is a status between conscious and unconscious experiences. If unconsciousness means non-synchronized oscillations, and consciousness means stably synchronized oscillations, REM sleep would be a status of intermittent synchronization. According to the synchronization theory of binding, this would result in abundant binding errors. It is indeed true that binding errors is the most salient character of dreams – it is full of symbolism, which is essentially a mistaken binding of some wrong but similar objects to other parts of the dream. In some people, dreams are colorful, but in some people, they are black and white. This can be seen as partial binding of visual information with color cells left out. Thus, we can see that a simply hypothesis can account for a lot of conscious and unconscious conditions.

In addition, this hypothesis also provides an explanation of why combinatorial input sometimes result in a composite subjective experience, but sometimes results a new subjective experience. For example, there are coldness detectors and warmness

Table 6.1: Examples of conscious conditions

Condition	Consciousness	Explanation
Visual and auditory perception	Yes	Mutual connections between sensory cortex and prefrontal cortex
Briefly primed items	No	Fails activity condition
Heart beats	No	Not controlled by working memory neurons in the prefrontal cortex
Habitual control of motor response	No	Not involving working memory
Result of habitual motor response	Yes	Proprioception enters working memory
Lesions in the thalamocortical complex	No	Prefronto-cortical connections rely on the thalamus
Dreams	Yes	Dreams can be seen as recurring sensory experiences with recombinations

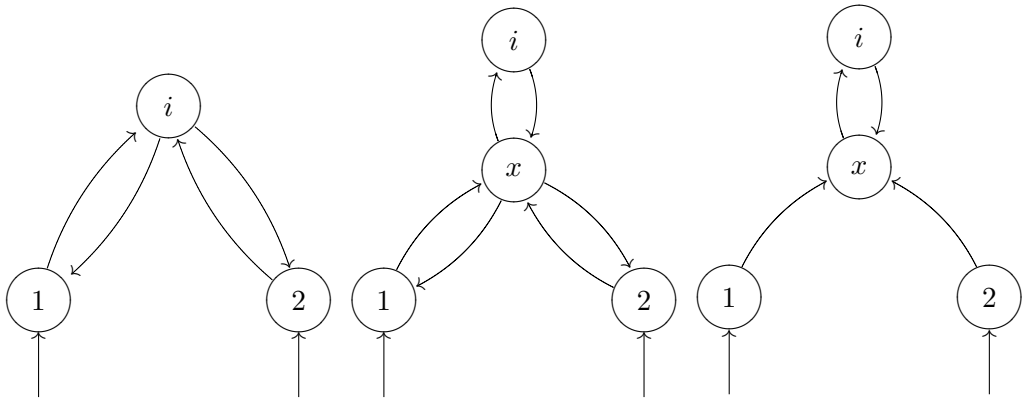


Figure 6.9: Three sensory combination mechanisms

detectors in the skin. When they are individually activated, we experience "coldness" and "warmness", respectively. However, if they are simultaneously activated, we experience neither, but a new feeling or qualie: "heat". The subjective experience resulting from combined input of "coldness" and "warmness" is not composite. Since the hypothesis defines subjective experience, can it be used to explain why this happens? Yes.

The differences can be illustrated by different patterns of sensory input to the prefrontal cortex (Figure 6.9). When there are two sensory inputs, they can be connected with the root neuron i in different ways. In the *additive* model, both features are mutually connected with i. According to the hypothesis, both features are synchronized and experienced. Thus, the subjective experience $C_i = \{1, 2\}$. An example of this is when tasting an ice-cream, one has a combined feeling of coldness and sweetness. In the *structured* model, the experience contains a combination of both features, as well as a group feature, $C_i = \{1, 2, x\}$. An example of this is the perception of a red rectangle. The experience is both the red color and the rectangular shape, as well as a coherent object. The color and shape features and the object x are all experienced. In the *replacement* model, however, the input features lack feedback connections from neuron i, thus, they are not experienced. The subject can only experience the percept x, that is, $C_i = \{x\}$. An example of this is the above perception of heat when "cold" and "warm" sensory neurons are simultaneously activated. Another example of the replacement experience is the perception of the color "purple", when sensory neurons for "red" and "blue" are simultaneously activated. Though people can infer that purple is a combination of red and blue, the experience or qualie of "purpleness" is a different qualie from "redness" and "blueness".

The consciousness hypothesis presented above can explain how different types of subjective experience can result from combinations of sensory inputs. The three models may be general enough to produce all kinds of subjective experiences from sensory inputs, as a single type of logical gates (NOR or NAND) can implement any arbitrary computation. This is a hypothesis left for future verification. Because the PNN has all required properties to model consciousness, it is possible that the working memory simulated in PNN can readily be applied to the study of consciousness. I agree with Baars (2005) that the subjective experience or consciousness is not unique to humans, but can be possessed by animals and even computational models as well.

6.5.4 A deterministic account of free will

The problem that troubles people in believing in free will is that for each state of a physical system, its next state is determined by simple physical laws. How can anything in the universe bear free will if the course of the universe has been fully determined? Phrasing the question in this way obscures the answer. In this section, I demonstrate how a system can exhibit free will, as we will see them as so, under deterministic physical laws.

First, let's discuss what qualifies as a free will that people can talk about. The concept "free will" is probably a subjective experience which people feel they have, but inanimates do not. If people attribute the causality of an object's behavior only to external forces, then they don't think it has free will. If people feel that the subject can choose what to do based on its own thoughts and desires. Then people think it has free will. At the root, people attribute free will to observed subjects if their actions are based on their own decisions. To people's intuition, decision is at the center of free will, rather than an absolute freedom in physical senses, since people cannot examine whether their behaviors are obeying absolute physical laws or not; they only have an intuition of physical laws but not a systematic understanding about it. People don't judge "free will" based on whether the subjects can choose their fate or not either, for the same reason. However, the answer for the latter question, as we will see, is "yes".

Regardless of human intuitions about "free will", the absolute question of whether a subject can have true free will against the determinacy of physical laws is still valid. In the following, it will be shown how people can exhibit true free will, in both the eyes of observers, as well as in deterministic physical laws. For the former, let's take the above assumption that free will is not about absolute "freedom", but it's about control and decision. This is easy to achieve. But true free will in the physical sense requires some special treatment, explained below.

The trickiest question about free will is that the current state of the physical world has only one outcome. As put by Einstein, "God does not play dice". How can a fully deterministic universe allow any free will? Is the future of humanity predetermined and there is no way to change it? Predeterminism of the future is starkly counter-intuitive, but difficult to rebut. On the one hand, it seems to be wrong because people don't seem to have any difficulty in determining what they will do next. A lack of free will doesn't seem to be real. On the other hand, it's also extremely difficult to show how

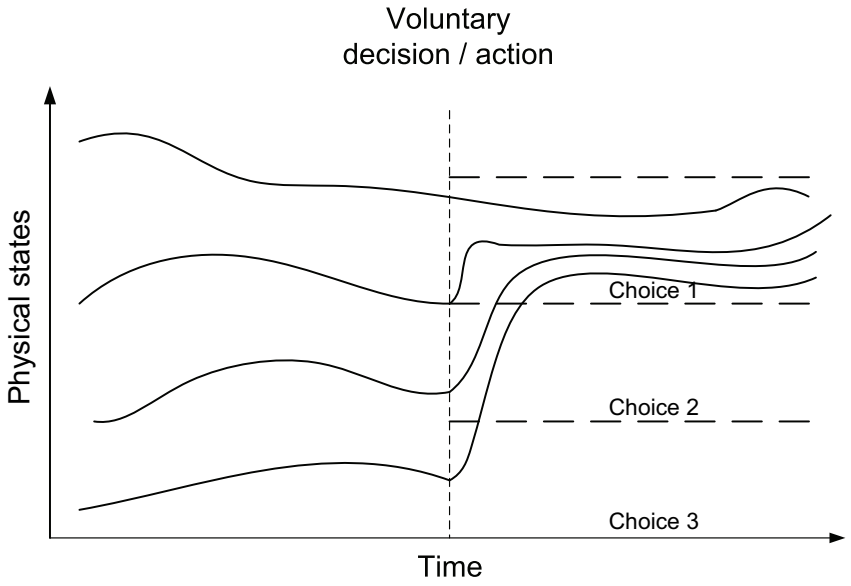


Figure 6.10: The channeling effect of a voluntary decision/action on physical states

people can possibly have free will and obey physical laws at the same time. Some people have put the hope on quantum mechanics, where the states of physical particles are indeterministic. However, indeterminism doesn't give free will (as the above observation criterion would grant), and it is not necessary for free will either, as shown below. This is a truly unusual property.

Let me demonstrate this in three steps. First, it is helpful to realize that a voluntary action can change the physical world in a way that greatly deviates it from a "default" course. This is what I will call the *channeling effect* of voluntary actions, as illustrated in Figure 6.10. The same effect applies to the physical state of the mind or the brain, after a voluntary decision (mental action) is made. The physical state of the mind is dramatically changed due to the outcome of the decision, in terms of changes in neural activities, and electronic and ionic distributions. The decision thus channels all possible previous physical states of the mind into a subset of physical states that possibly follow this decision, labeled "Choice 1" in the figure. In this region, though different previous states remain different, and maintain their physical identity, they can "functionally" be considered as a macroscopic category including all physical outcomes of this decision. Let's refer to this macroscopic category of physical states as a logical state $s_1 = \{\text{all possible physical states after Choice 1}\}$.

Second, it can be shown that the logical states, if chosen appropriately, can fully account for the behavior and the future of the system. This property is not difficult to see: many logical systems whose states are completely determined by their previous logical states can qualify. For example, by knowing the logical state of a computer, we can know its future behavior without knowing any details of its physical states (e.g., in terms of quantities concerning its molecules, atoms and electrons). When a physical system implements a complete logical system, whose behavior can be fully determined by its logical states, and rules governing them, the physical system will have a duality. From a current physical state x_0 of the system, we can use physical laws to predict its future state x_t at any time t. At the same time, from the system's current logical state s_0, we can also fully predict its future logical state s_t at time t. The two representations of the system are consistent, in the sense that no matter what the actual states x_t and s_t are, x_t must be a valid physical implementation of the logical state s_t. The consistency can be seen as, the physical state x_t as a point flies through the physical state space based on true and pure deterministic physical laws, but its trajectory swiftly passes only the regions allowed by the predicted logical states. Therefore, we can sufficiently describe the system in one of the two ways: either as a physical state trajectory, or as a logical state sequence.

Third, although it is extremely hard to describe free will at the physical level, it is not impossible to discuss it and implement it in a computer using the logical states. As discussed above, "free will" is a subjective judgment on a behaving system. If we consider that a subject chooses its own behaviors based on its own desires, and attempts to realize its goal, then we consider that it has free will. Such a criterion can be satisfied by a computing device, without too much difficulty (it doesn't require that the system can pass the Turing test). For example, it can be a program which 1) maintains its current self-generated goal or "desires", 2) tries to act towards the goal, and 3) executes goal-directed actions in the physical world. With these properties, observers should probably attribute free will to the system, if they use the same criterion on the possession of free will by humans and animals.

Does the system, whose free will is generated at the logical level, violate any physical laws? No. The physical laws can deterministically predict the logical state sequence, just as accurately as the logical system would do at the logical level. Thus, the observed free will, in the logical system executing the above program, does not require

any indeterminacy of the physical world. At this point, it should be fairly clear what begins to enable a physical system to exhibit free will is: an implementation of a universal computing machinery (such as a Turing Machine), which imitates human goal-oriented cognition and behaviors as the program discussed above does. Even if one doesn't fully identify it with human free will, the fact that goal-oriented behaviors can freely happen in a deterministic physical system is convincing enough for an acceptance of a considerate amount of free will in humans.

Why does this happen? Why is the result different from our intuition that physical laws are inflexible and unable to support free will? The answer is that only a small subset of physical states has this property. For a physical system to possess free will, it has to be in a few fairly rare states which correspond to a logical system that can exercise free will. Within this logical system, the physical laws only enable the logical processes that underlie free will but not constrain it. Once the state of a physical system comes to the entrance or a "trap" of a logical system, its fate is largely determined by the logical system. The physical system may be "freed" later, if the logical system with "free will" breaks down. The dominance of physical laws and logical laws can alternate, and coexist without contradiction during the duality period. This can be illustrated as generalized channeling effects in Figure 6.11.

As a human being, our physical structure is one of the singularities in the physical world which possesses such a dualistic property. Interpreting free will in this way, humans don't need to resort to any physical indeterminacy to enjoy free will; rather, perfect determinacy of a physical world can best ensure the existence of free will. The physics-humanity duality breaks down with the decease of life. While humans assume duality, we can exert free will and change the physical world, at least, the part which humans either possess or interact with.

The result

People have free will because we can base our decisions and actions on our own goals and strive to achieve them. A computing device may to some degree imitate human decision making and exhibit "free will". Physical determinacy of the mind does not reduce our free will to any significant degree. Free will does not require any indeterminacy of the universe. The physical world can still be viewed as highly deterministic, as Einstein claims: "God doesn't play dice." The universe has only one course. It's not

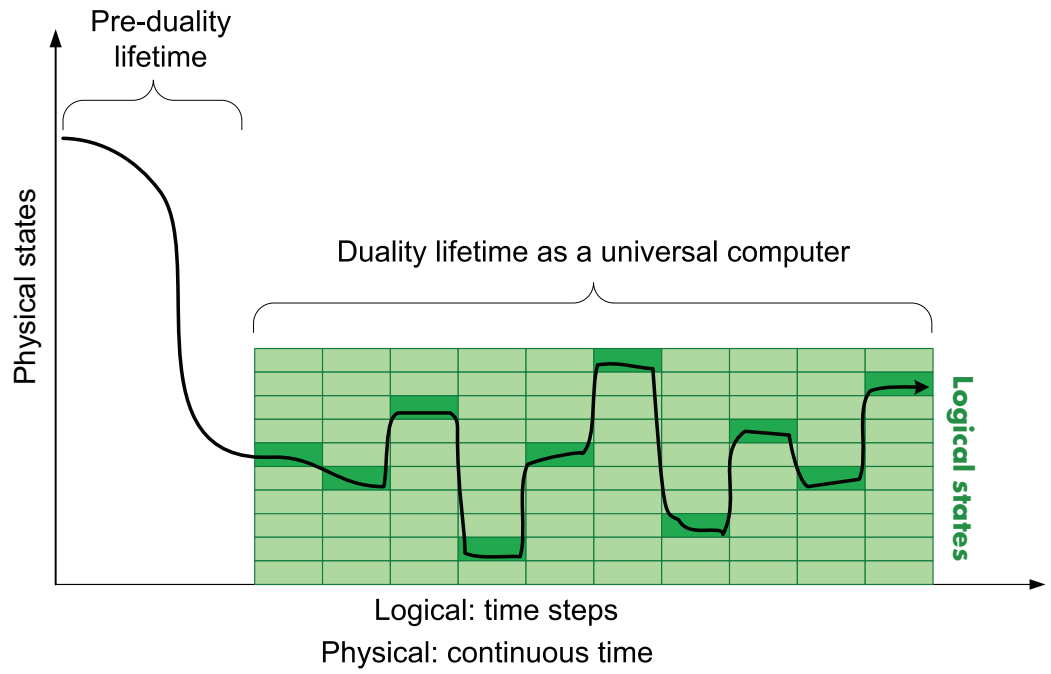


Figure 6.11: Duality of a physical system as a universal computer

predetermined, but is determined by us.

6.6 Conclusion

In this thesis, I followed the theoretical conceptualization of working memory (Baddeley, 1986), made use of computational models to systematically study its neural mechanisms and processes, especially the serial order mechanism and its neural substrate. The models, in many aspects, have been supported by neural anatomical and neural physiological evidence, as well as a range of behavioral effects as observed in human immediate serial recall tasks. With some new in-depth understanding of the brain's serial order mechanisms, this model has implications on the timing function of the cerebellum, and syntactic structures of languages. From a view about neural synchrony in working memory and cognitive control, some thoughts are given on topics about consciousness. It is hoped that this work has provided deeper understanding about the working memory, its neural mechanisms and its implications for people's lives.

Appendix A

A.1 Derivation of the phase dynamics rule

The phase dynamics rule (Section 3.2.4, Eq. 3.21) can be derived from the integrate-and-fire model (Section 3.1.2.1). Without loss of generality, we can look at a unit receiving inputs from a set of units i. Let u be its membrane potential, and t_i^k denote the time of the kth spike of unit i. For notational convenience, define the spike train $\Lambda_i(t)$ of unit i as

$$\Lambda_i(t) = \sum_k \delta(t - t_i^k), \tag{A.1}$$

where the Dirac Delta $\delta(x)$ models a spike at time x.

The input current $I(t)$ to the unit's membrane potential u is modeled as the weighted sum of all incoming spikes, plus a noise current $\varepsilon(t)$, thus

$$I(t) \equiv \sum_i w_i \Lambda_i(t) + \varepsilon(t), \tag{A.2}$$

where w_i is the synaptic weight of the input unit i.

Plugging the current (A.2) into the Integrate-and-fire model (3.5), we have the membrane potential u:

$$\frac{du}{dt} = -\gamma u(t) + \sum_i w_i \Lambda_i(t) + \varepsilon(t). \tag{A.3}$$

In the following, I will derive from the membrane potential model (A.3), the probability $p(\theta_i; \hat{t}_i)$ that the postsynaptic unit would assume the same phase θ_i as the presynaptic unit i after the spike at time $\hat{t}_i$. By definition, the postsynaptic unit has the same discrete phase as a presynaptic unit, if it fires within a threshold time τ_θ (the phase equality tolerance) after the presynaptic unit; otherwise, it should have a different phase.

Without loss of generality, we can assume that all inputs units have distinct phases θ_i (if multiple units have the same phases, they can be combined into a single unit with a larger weight). Let $p(\theta; t)$ denote the probability that the postsynaptic unit assumes phase θ at the next time step t. The probability $p(\theta_i; \hat{t}_i)$ after a spike $\hat{t}_i$ can be written by definition as below:

$$p(\theta_i; \hat{t}_i) = P(u \text{ fires in } [\hat{t}_i + \tau_\theta]), \tag{A.4}$$

The condition for the postsynaptic unit to fire is that the membrane potential u rises above the firing threshold ϑ with an peak increment $\Delta u(t)$ in the time window $[t, t + \tau]$. Thus, the probability (A.4)can be written as

$$p(\theta_i; \hat{t}_i) = P[u(\hat{t}_i) + \Delta u(\hat{t}_i) \geq \vartheta]. \tag{A.5}$$

The peak increment $\Delta u(t)$ is the maximal increment of u as it evolves from time t to $t + \tau$, defined as

$$\Delta u(t) \equiv \max_{t \leq s \leq t+\tau} \int_t^s \frac{du(t)}{dt} dt \tag{A.6}$$

Since the time window τ_θ is typically much narrower than the time scale $1/\gamma$ of the spontaneous decay of the membrane potential u (i.e., $\tau_\theta \ll 1/\gamma$, see Figure 3.3), the spontaneous decay can be ignored during this time interval $[t, t + \tau_\theta]$. The integral above thus mainly depends on the last two terms in du/dt (A.3):

$$\int_t^s \frac{du(t)}{dt} dt = \int_t^s \Big[-\gamma u(t) + \sum_i w_i \Lambda_i(t) + \varepsilon(t) \Big] dt \tag{A.7}$$

$$\approx \int_t^s \Big[\sum_i w_i \Lambda_i(t) + \varepsilon(t) \Big] dt \tag{A.8}$$

$$= \int_t^s \Big[\sum_i w_i \sum_k \delta(t - t_i^k) + \varepsilon(t) \Big] dt. \tag{A.9}$$

Since τ_θ is small, the only presynaptic spike from unit i in the time window $[\hat{t}_i, \hat{t}_i + \tau]$ is the spike at time $\hat{t}_i$ itself (other spikes cannot be in the time window either because they have different phases). In this case, the peak increment $\Delta u(\hat{t}_i)$ is very simple:

$$\begin{aligned} \Delta u(\hat{t}_i) &\approx \max_{\hat{t}_i \leq s \leq \hat{t}_i + \tau} \int_{\hat{t}_i}^{\hat{t}_i + \tau} \Big[\sum_j w_j \sum_k \delta(t - t_j^k) + \varepsilon(t) \Big] dt \\ &= w_i + \Delta\varepsilon(\hat{t}_i), \end{aligned} \tag{A.10}$$

where $\Delta\varepsilon(t) = \max_{t<s<t+\tau} \int_t^s \varepsilon(t)dt$ (here $t = \hat{t}_i$) is the maximal noise in the short time window $[t, t+\tau]$. Below, the shorthand $\Delta\varepsilon$ refers to $\Delta\varepsilon(\hat{t}_i)$.

Plugging (A.10) into (A.5), we have

$$p(\theta_i; \hat{t}_i) = P[u(\hat{t}_i) + w_i + \Delta\varepsilon \geq \vartheta]. \tag{A.11}$$

Since in the situation above, the postsynaptic neuron is so close to its firing threshold ϑ that it can be fired by a single input spike, the distance from u to ϑ must be adequately small. In this case, we can approximate the distribution of u in this small region $[u, \vartheta]$ with a uniform distribution $p_i(x) = 1/(\vartheta - u)$, then we have

$$\begin{aligned}
p(\theta_i; \hat{t}_i) &= P[u(\hat{t}_i) \geq \vartheta - w_i - \Delta\varepsilon] && \text{(A.12)}\\
&= \int_{\vartheta - w_i - \Delta\varepsilon}^{\vartheta} p_i(x)dx && \text{(A.13)}\\
&= \int_{\vartheta - w_i - \Delta\varepsilon}^{\vartheta} \frac{1}{\vartheta - u}dx && \text{(A.14)}\\
&= \frac{1}{\vartheta - u}x\Big|_{\vartheta - w_i - \Delta\varepsilon}^{\vartheta} && \text{(A.15)}\\
&= \frac{1}{\vartheta - u}\left(w_i + \Delta\varepsilon\right) && \text{(A.16)}\\
&\propto w_i + \Delta\varepsilon. && \text{(A.17)}
\end{aligned}$$

The probability $p(\theta_i; \hat{t}_i)$ is closely tied to the time of a spike of unit i. To find the probability distribution for any time t, we take the expected value of $p(\theta_i; t)$. First, observing that when there is no spike from unit i, the probability density $p(\theta_i; \hat{t}_i)$ comes only from the noise term, $p(\theta_i; t) = \Delta\varepsilon(t)$. So we can use the spike train variable $\Lambda_i(t)$ to equivalently write

$$p(\theta_i; t) \propto \Delta\varepsilon(t) + w_i\Lambda_i(t), \tag{A.18}$$

and its expected value

$$\hat{p}(\theta_i; t) \propto E[\Delta\varepsilon(t)] + w_i E[\Lambda_i(t)]. \tag{A.19}$$

Since the expected value of the spike train $E[\Lambda_i(t)]$ of unit i is just its firing rate $a_i r$ (a_i is the output of unit i, and r is a scalar), we have

$$\hat{p}(\theta_i; t) \propto \alpha + w_i a_i, \tag{A.20}$$

where $\alpha = E[\Delta\varepsilon(t)]/r$ is proportional to the expected noise. This is very close to the phase dynamics equation (3.21) that we want to prove.

Generalizing (A.20) to an arbitrary *post*-synaptic unit n whose input of phase ϕ is $\tilde{x}_{n,\phi}$ (in the above situation of one input neuron, $\tilde{x}_{n,\phi} = w_n a_n$), we obtain the equation of the pre-normalized phase distribution ($[x]_+$ is due to the max operation in Δu):

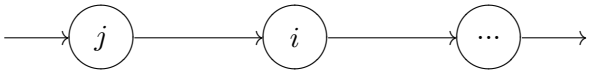
$$\tilde{p}_n(\phi) = \alpha_n + [\tilde{x}_{n,\phi}]_+. \tag{A.21}$$

This has the form of the phase dynamic rule (3.21) used in the phase neurons in Chapter (3). The actually rule used has an exponent p, which is added for its utility in controlling the speed of synchronization, discussed in Section A.2.2[1].

A.2 Proofs of PNN properties

A.2.1 Phase propagation

Multi-unit recordings in the visual pathway have found neural synchronization between different areas along the visual pathway, such as the retina, LGN, and the cortex. Synchronization between consecutive layers can occur due to feedforward and feedback connection (Castelo-Branco, Neuenschwander, & Singer, 1998). We consider phase propagation between layers as a basic property that the new neural network model must have. Here, we verify that PNN does have the property. A theorem is given below for a simple case of connection shown in Figure A.1. It is straightforward to show that it holds under more general conditions too (e.g., $a_i^t > 0$). It is worth noting that phase propagation and thus binding relations in the network are stable across many intermediate layers. It does not suffer spike timing jittering as in networks of individual spiking neurons.



pre-synaptic post-synaptic

Figure A.1: Phase propagation

[1]The missing of the exponent p might also be a result of the simplifying assumptions in the above procedure.

Theorem 1. *Given an inactive unit i receiving input from only another unit j with activity $a_j^t > 0$, then its phase will be equal to that of unit j: $\theta_i^{t+1} = \theta_j^t$, when the noise $\alpha_i \to 0$ is ignorable.*

Proof. The phase-specific phase inputs $\tilde{x}_{i,\phi}^{t+1}$ for phases $\phi \in \Theta$

$$\tilde{x}_{i,\phi}^{t+1} = \delta(\phi, \theta_i^t)(1-\gamma_i)\, a_i^t + \sum_{j \in S_i} \delta(\phi, \theta_j^t)\, w_{ij}^{phase}\, a_j^t = \begin{cases} w_{ij}^{phase} a_j^t, & \phi = \theta_j^t \\ 0, & \text{otherwise.} \end{cases} \tag{A.22}$$

The prenormalized probability mass when $\alpha_i \to 0$:

$$\lim_{\alpha_i \to 0} \tilde{p}_i^{t+1}(\phi) = \lim_{\alpha_i \to 0} \alpha_i + [\tilde{x}_{i,\phi}^{t+1}]_+^p = \begin{cases} [w_{ij}^{phase} a_j^t]_+^p, & \phi = \theta_j^t \\ 0, & \text{otherwise.} \end{cases} \tag{A.23}$$

Therefore, the probability mass $p_i^{t+1}(\phi)$ when $\alpha_i \to 0$ is $\lim_{\alpha_i \to 0} p_i^{t+1}(\phi) = \delta(\phi, \theta_j^t)$, which means $\theta_i^{t+1} = \theta_j^t$. ■

A.2.2 Synchronization and desynchronization

Two basic properties of neural oscillation are 1) synchronization between excitatorily connected neurons, and 2) desynchronization between inhibitorily connected neurons. These properties have been identified by analysis and simulation (Strogatz & Mirollo, 1988; Campbell & Wang, 1996). They form a natural solution of representing multiple objects or visual segments. In the representation, an object is defined by a group of features whose oscillations are synchronized, or a Hebbian assembly, and diffident objects are represented by different groups of neurons whose oscillations are out of phase (desynchronized). This is an elegant representation in that it eschews the need for highly specialized object cells (grandmother cells) whose number is subject to combinatorial explosion, and it nicely utilizes the property that neurons suitably respond to correlated activity to perform object recognition. The two properties can be seen as fundamental for any spiking neural networks. Here, we show that the PNN framework supports both fast synchronization and desynchronization as specified above. Proofs are given below for fully connected networks with uniform weights though the same properties hold for heterogeneous weights. The speed of synchronization depends on the parameter p in equation (3.20). Larger p corresponds to faster synchronization.

Theorem 2. *For a fully-connected set of N units $u \in U$ whose connection weights are uniformly $w > 0$, and activities are constant a, all units will be stably synchronized.*

Proof. Let $s_i^t = |\{u \in U \mid \theta_u^t = i\}|/N$ be the proportion of units with phase i. Since the noises α_u are small compared to unit activities: $\alpha_u \ll a_u$, they can be ignored. Using the propagation rules (3.19)-(3.21), the expected proportion for phase i is $\hat{s}_i^{t+1} = E[s_i^{t+1}] = (s_i^t)^p / \sum_i (s_i^t)^p$. That is, it's the normalized pth power of s_i. When $p > 1$, it can be shown that the largest proportion $s_{max}^t = \max_i s_i^t$, is always expected to increase. That is, the largest synchronized group will increase until all units are synchronized. To prove this, we can start with the function

$$f(p) = (s_{max}^t)^p / \sum_i (s_i^t)^p = 1 / \sum_i \left(\frac{s_i^t}{s_{max}^t} \right)^p .$$

Because all the terms in the parentheses are less than or equal to 1, the function $f(p)$ is monotonically increasing when $p \geq 1$. Therefore, we have $f(p) \geq f(1)$, that is

$$\frac{(s_{max}^t)^p}{\sum_i (s_i^t)^p} \geq \frac{s_{max}^t}{\sum_i s_i^t}.$$

Since $\sum_i s_i^t = 1$, we have

$$\hat{s}_{max}^{t+1} \geq s_{max}^t.$$

This shows, in the long run, the largest synchronized group tends to increase to the whole population, when all units are synchronized.

■

Theorem 3. *For a set of N units $i \in U$, $N \leq \theta_m$, each connecting to all other units with sufficiently large negative weights w_{ji}, when they are active ($a_i > 0$) and have small decay rates $\lambda_i < 1$, then all units will be stably desynchronized.*

Proof. Since $N \leq \theta_m$, it's possible that each unit has a distinct phase. Upon this condition, all the phases will remain unchanged and the system remains fully desynchronized. This is because for a unit i whose phase is unique, the only positive phase input $\tilde{x}_{i,\phi}$ has its own phase $\phi = \theta_i$, that is, $\tilde{x}_{i,\theta}^{t+1} > 0$ and $\tilde{x}_{i,\phi}^{t+1} \leq 0$ for $\phi \neq \theta$. Using the prenormalized probability $\tilde{p}_{i,\phi}^{t+1} = \alpha_i + [\tilde{x}_{i,\phi}^t]_+$, if the noise term $\alpha_i \ll \tilde{x}_{i,\theta}^{t+1}$ is ignorable, then the normalized phase distribution $p_{i,\phi}^{t+1} = \delta(\phi, \theta_i^t)$, which means the phase remains the same: $\theta_i^{t+1} = \theta_i^t$. Thus, after all units are first desynchronized, they will stay in the desynchronized state. This is an absorbing state of the system.

When there is at least a pair of units i and j sharing the same phase, it can be shown that the phases θ_i and θ_j will freely change, and the system will eventually

settle in a desynchronized absorbing state. Suppose $\theta_i^t = \theta_j^t = \theta$, when the negative weight w_{ij}^{phase} is sufficiently strong, the inhibition from unit j can totally cancel out the self-recurrent input of unit i so that the input with phase θ is negative: $\tilde{x}_{i,\theta}^{t+1} = (1-\gamma_i)\, a_i^t + \sum_{j \in S_i} \delta(\theta, \theta_j^t)\, w_{ij}^{phase}\, a_j^t \leq 0$.

Further, since all interconnections in the system are negative, $\tilde{x}_{i,\phi}^{t+1} \leq 0$ for all $\phi \in \Theta$. From (3.20), we know $\tilde{p}_{i,\phi}^{t+1} = \alpha_i + [\tilde{x}_{i,\phi}^{t+1}]_+ = \alpha_i$ for all ϕ. So $p_{i,\phi}^{t+1}$ is a uniform distribution over all phases $\phi \in \Theta$. Similarly, the phase distribution $p_{j,\phi}^{t+1}$ for unit j is also uniform. Thus, the phases of i, j can freely change. Because the system has absorbing states where all phases are distinct, it will eventually settle in one of these states where all units are desynchronized[2].

Therefore, networks of phase neurons indeed possess the desired basic synchronization properties: synchronization between excitatorily connected neurons, and desynchronization between inhibitorily connected neurons. Thus, the phase neurons may be used to model neural processes involving dynamic binding, synchronization and segmentation.

■

A.3 Neural assemblies that count to 4

The counter model in Section 5.3.2.1 shows that variance in biases can give a neural assembly a counting property. That is, different subgroups in the assembly can form stable memories for different instances of an item, and the number of such subgroups represents a count of the item. This is a simple but powerful mechanism. It can be used for counting, tokenization, position coding, movement repetition, and numerosity judgments. However, the basic model requires exact biases. To make it more biologically plausible, I will show that this condition can be relaxed, and a neural assembly with random biases can also possess the counting property. What's more interesting is that the model has a natural capacity of four, which coincides with an important capacity of animal and human non-linguistic numerical capability (J. Atkinson et al., 1976; Luck & Vogel, 1997; Carey, 1998; Cowan, 2001a).

It is worth noting that there are several interesting numerical capacities. Without language, numerical capability may rely on two neural mechanisms: analog and digital. In the analog model, the number is represented by a continuous quantity. The

[2] In practice, the system reaches full desynchronization in a few time steps.

capacity is likely to be around 9 in animal numerosity judgment tasks (E. M. Brannon & Terrace, 1998; E. Brannon & Terrace, 2000; Terrace, Son, & Brannon, 2003; Judge, Evans, & Vyas, 2005). In the digital mechanism, a.k.a. the object file model, number representations are limited to the number of objects in short-term memory, which may be 3 or 4 (Carey, 1998; Cowan, 2001b; Luck & Vogel, 1997). The latter capacity is close to the natural capacity of the neural assembly that I will discuss. However, there is an important distinction. The above capacity relates to the situation where all objects are *simultaneously* presented. When objects are *sequentially* presented, 3 or 4 appear to be a special capacity. For example, when phone numbers are subgrouped in 3 or 4 items, recall is usually the most accurate (Severin & Rigby, 1963). In the following, I will show the neural mechanism that can stably and accurately count up to four sequentially presented items.

A.3.1 Gaussian biases

Let's start with an example neural assembly with N units whose biases are Gaussian. Similar to the neural counter in Section 5.3.2.1, all units have a binary activation function

$$\sigma(x) = \begin{cases} 1, & x > 0 \\ 0, & x \leq 0, \end{cases} \tag{A.24}$$

and the lateral connection weights are uniformly w/N. The net input x_i for each unit $i = 1 \ldots N$ is thus

$$x_i(t+1) = \tfrac{w}{N} \sum_{j=1}^{N} \sigma(x_j(t)) - b_i + \alpha I(t), \tag{A.25}$$

where b_i is the bias of the unit i, $I(t)$ is the external input and α is the input weight.

The sum in (A.25) only depends on the proportion of active neurons because all active neurons output 1 with the same weight. Substituting the proportion of active neurons $a(t) = N^{-1} \sum_i \sigma(x_i(t))$ in the equation, we have

$$x_i(t+1) = wa(t) - b_i + \alpha I(t).$$

In order to find out the assembly's counting capacity, we would need to find out the proportions of active neurons that can be stably maintained by recurrent connections

when there's no external input, $I(t) = 0$. The active proportion for the next time step $t + 1$ is

$$\begin{aligned} a(t+1) &= N^{-1} \sum_{i=1}^{N} \sigma(x_i(t+1)) \\ &= N^{-1} \sum_{i=1}^{N} \begin{cases} 1, & wa(t) - b_i > 0 \\ 0, & \text{otherwise.} \end{cases} \end{aligned}$$

When $N \to \infty$, the average in the above equation is the probability $\Pr[wa(t) - b_i > 0]$. Let $g(x)$ be the probability density function (PDF) of the biases b_i, and $G(u)$ be the cumulative distribution function (CDF) of $g(x)$, so we have

$$\begin{aligned} \lim_{N \to \infty} a(t+1) &= \Pr[b_i < wa(t)] \\ &= G[wa(t)]. \end{aligned}$$

For the neural assembly to be stably active, the proportion of active neurons a must be a solution of the equation:

$$a = G(wa). \tag{A.26}$$

The roots of this equation can be geometrically observed. They are the intersections of the graph $y = G(x)$ and the line $y = x/w$ (substituting $x = wa$). As shown in Figure A.2, if the bias distribution $g(x)$ is Gaussian, Eq. (A.26) has at most 2 roots. That is, the counter has a capacity up to $K = 2$.

Interestingly, if the biases follow a bimodal distribution, as shown in Figure A.3, the CDF $G(x)$ can have at most 4 intersections with the line $y = x/w$ (in the figure, $w = 1$). In this case, the counter capacity is at most $K = 4$.

A.3.2 Biological implementation

The above analysis shows a fully connected neural assembly with bimodal biases can count up to 4. Is the bimodal bias distribution biologically plausible? There is some evidence for this condition. It has been observed that cortical pyramidal cells have up and down states (C. Wilson, 2008), and their membrane potentials have a bimodal

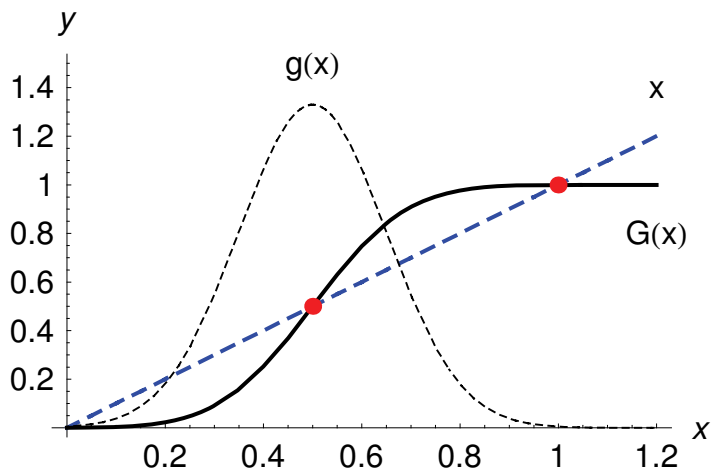


Figure A.2: Unimodal bias distribution

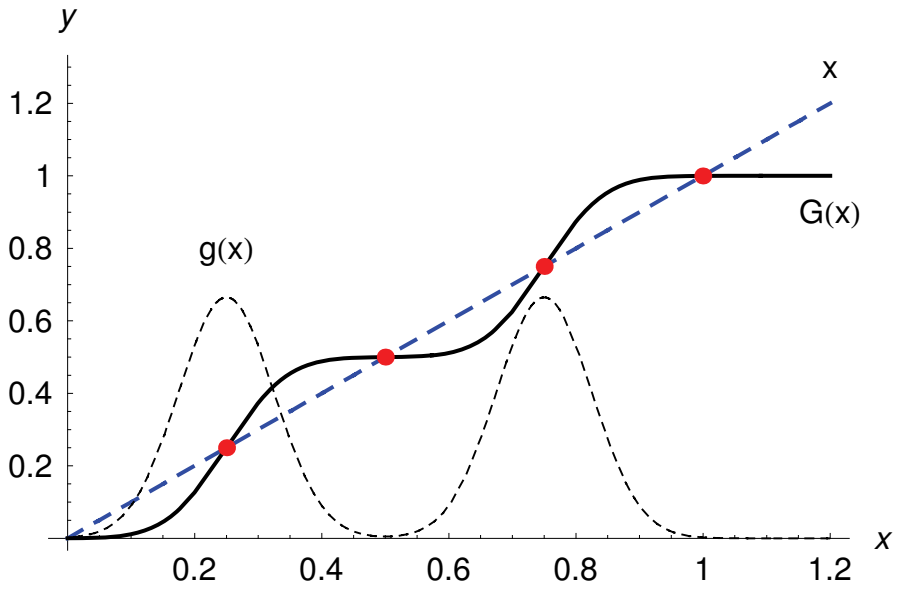


Figure A.3: Bimodal bias distribution

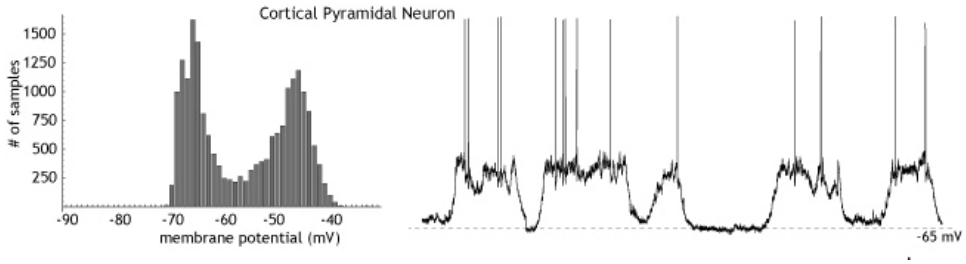


Figure A.4: Bimodal membrane potential distribution of cortical pyramidal cells (Source: C. Wilson, 2008)

distribution as shown in Figure A.4. This neural assembly can theoretically serve as a counter with a capacity of 4.

It is worth noting that the capacity of 4 is hard to increase, suggesting that it could be a "magic number" for the mechanism. Although a trimodal distribution could increase the capacity to 6, a trimodal distribution is qualitatively harder to achieve. Recall the fact that the sum of any two Gaussians is still a Gaussian, the achievement of even a bimodal distribution is nontrivial. For example, one can obtain a bimodal distribution by a two-step process. In Step 1, obtain a binary variable $m = 0$ or 1. In Step 2, sample from one of the two Gaussians depending on the value of m. The sampling from different Gaussians must be largely mutually exclusive. Analogously, to obtain a trimodal distribution, for each neuron, one has to exclusively sample one out of three unimodal distributions, and this would require a special three-way selection mechanism. It is qualitatively more difficult than generating a unimodal or bimodal distribution. If the theory is correct, it may provide an interesting explanation for animal and human prelinguistic numerical mechanisms and their capacities.

A.3.3 Stability against noises

Though the above model can be biologically plausible, there is an additional issue that hasn't been addressed – the stability of the number representations in the model. It should be pointed out that the roots in the models, which represent counts, are unstable except the saturation points (the last roots in Figures A.2 and A.3). For instance, a small increase in x around the first root in Figure A.2 will lead to increases in y, and the system may slowly move away to the second root. If the brain uses such a neural assembly for counting, it will need to stabilize the fixed points. Here, I would show that inhibitory interneurons in the brain may stabilize the unstable fixed points in

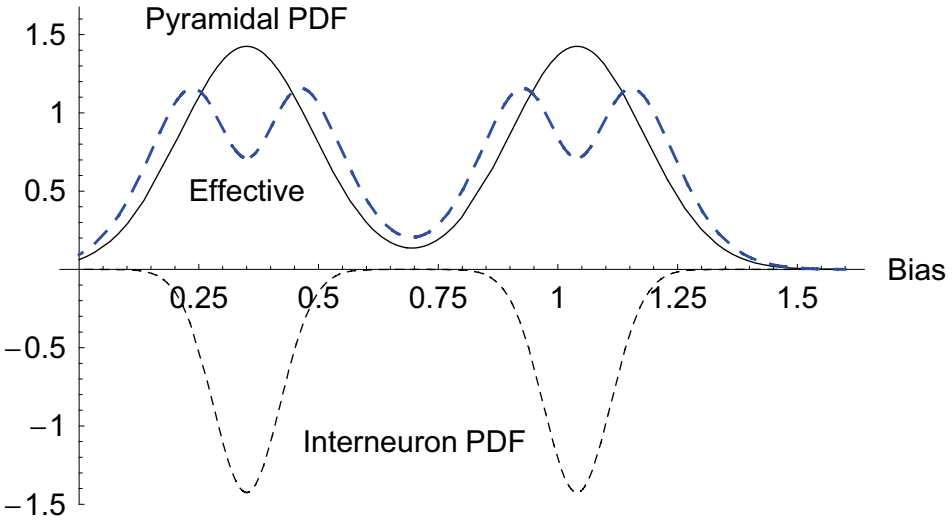


Figure A.5: Effective density of pyramidal cells after considering interneurons

the model.

The idea is the following. In the cortex, inhibitory interneurons are interconnected with excitatory pyramidal cells. Because the interneurons are activated by pyramidal cells, their firing thresholds (biases) can be seen as connected to those of the pyramidal cells, so they are centered at the same Gaussian means, as shown in Figure A.5 (the interneurons PDF is shown as negative for easy reading). With the introduction of interneurons, the "effective" density of pyramidal neurons, in terms of their excitatory output, is reduced by the density of the interneurons. The effective density has a quadmodal distribution. As we did before, the output-input relation of the neural assembly can be obtained by finding the cumulative distribution function (CDF) of this density, shown in Figure A.6. It can be seen that the fixed points representing numbers are much more stable than without interneurons (Figure A.3). This result has been confirmed by simulations.

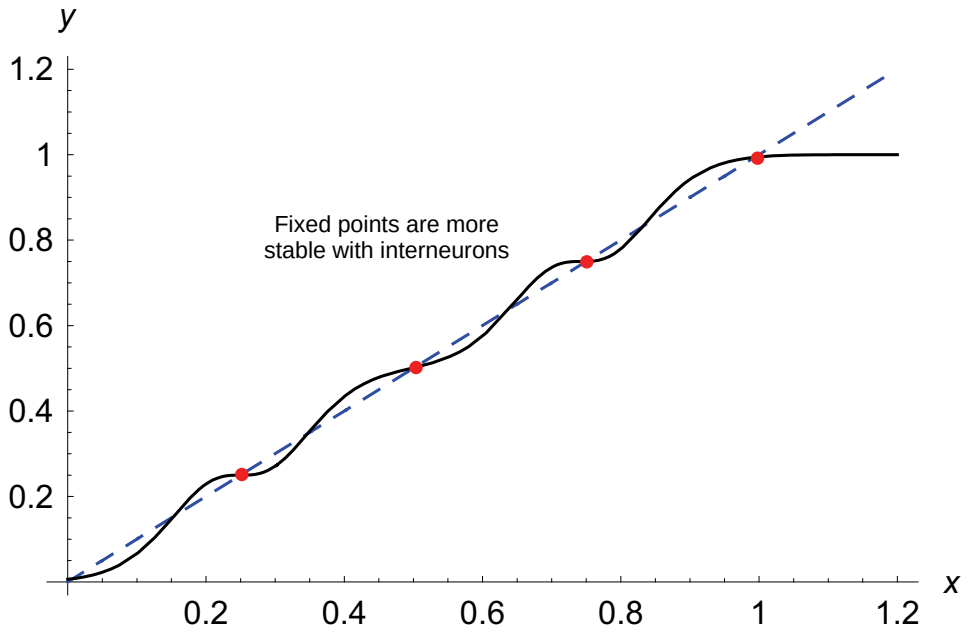


Figure A.6: Number representations (fix points) are stabilized by interneurons

Bibliography

Aaronson, D. (1968). Temporal course of perception in an immediate recall task. *Journal of Experimental Psychology*, *76*(1, Pt.1), 129-140.

Abarbanel, H. D. I., Gibb, L., Mindlin, G. B., Rabinovich, M. I., & Talathi, S. (2004). Spike timing and synaptic plasticity in the premotor pathway of birdsong. *Biological Cybernetics*, *91*(3), 159-167.

Abbott, L. F., & Nelson, S. B. (2000). Synaptic plasticity: taming the beast. *Nature Neuroscience*, *3*, 1178-1183.

Aldridge, J. W., & Berridge, K. C. (1998). Coding of serial order by neostriatal neurons: A "natural action" approach to movement sequence. *J. Neurosci.*, *18*(7), 2777-2787.

Aldridge, J. W., Berridge, K. C., & Rosen, A. R. (2004). Basal ganglia neural mechanisms of natural movement sequences. *Can. J. Physiol. Pharmacol.*, *82*, 732-739.

Alexander, G. E. (1987). Selective neuronal discharge in monkey putamen reflects intended direction of planned limb movements. *Experimental Brain Research*, *67*(3), 623-634.

Alexander, G. E., DeLong, M. R., & Strick, P. L. (1986). Parallel organization of functionally segregated circuits linking basal ganglia and cortex. *Annuual Review of Neuroscience*, *9*, 357-381.

Archibald, L. M. D., & Gathercole, S. E. (2007). Nonword repetition and serial recall: Equivalent measures of verbal short-term memory? *Applied Psycholinguistics*, *28*(4), 587-606.

Arciszewski, M., & Akins, K. (2000). The race for consciousness. *Trends in Neurosciences*, *23*(12), 648-649.

Atkinson, J., Campbell, F. W., & Francis, M. R. (1976). The magic number 4 ± 0: A new look at visual numerosity judgementshenson. *Perception*, *5*(3), 327-334.

Atkinson, R. C., & Shiffrin, R. M. (1968). Human memory: A proposed system and its control process. In K. Spence & J. Spence (Eds.), *The psychology of learning and motivation (vol. 2).* New York: Academic Press.

Atkinson, R. C., & Shiffrin, R. M. (1971). The control of short-term memory. *Scientific*

American, *Vol. 225*(2), 82-90.

Azouz, R., & Gray, C. M. (2000). Dynamic spike threshold reveals a mechanism for synaptic coincidence detection in cortical neurons in vivo. *Proceedings of the National Academy of Sciences of the United States of America*, *97*(14), 8110-8115.

Baars, B. J. (2005). Subjective experience is probably not limited to humans: The evidence from neurobiology and behavior. *Consciousness and Cognition*, *14*(1), 7-21.

Baddeley, A. D. (1968). How does acoustic similarity influence short-term memory? *The Quarterly Journal of Experimental Psychology*, *20(3)*, 249-264.

Baddeley, A. D. (1986). *Working memory.* Oxford: Clarendon Press.

Baddeley, A. D. (2000). The episodic buffer: a new component of working memory? *Trends in Cognitive Sciences*, *4*(11), 417-423.

Baddeley, A. D. (2003). Working memory: Looking back and looking forward. *Nature Reviews*, *4*, 829-839.

Baddeley, A. D., & Hitch, G. J. (1974). Working memory. In G. H. Bower (Ed.), *The psychology of learning and motivation* (Vol. 8, p. 47-90).

Baddeley, A. D., Lewis, V., & Vallar, G. (1984). Exploring the articulatory loop. *Quarterly Journal of Experimental Psychology: Human Experimental Psychology*, *Vol 36A*(2), 233-252.

Baddeley, A. D., Thomson, N., & Buchanan, M. (1975). Word length and the structure of short-term memory. *Journal of Verbal Learning & Verbal Behavior*, *Vol 14*(6), 575-589.

Barto, A. (1995). Adaptive critics and the basal ganglia. In H. e. al. (Ed.), *Models of information processing in the basal ganglia* (p. 215-232). Cambridge: MIT Press.

Beaman, C. P., & Morton, J. (2000). The effects of rime on auditory recency and the suffix effect. *European Journal of Cognitive Psychology*, *12*(2), 223-242.

Beiser, D. G., & Houk, J. C. (1998). Model of cortical-basal ganglionic processing: Encoding the serial order of sensory events. *Journal of Neurophysiology*, *79*, 3168-3188.

Bergman, H., Feingold, A., Nini, A., Raz, A., Slovin, H., Abeles, M., et al. (1998). Physiological aspects of information processing in the basal ganglia of normal and parkinsonian primates. *Trends in Neurosciences*, *21*(1), 32-38.

Berns, G. S., & Sejnowski, T. J. (1998). A computational model of how the basal ganglia produce sequences. *Journal of Cognitive Neuroscience*, *10*, 108-121.

Bi, G. Q., & Poo, M. M. (1998). Synaptic modifications in cultured hippocampal neurons: Dependence on spike timing, synaptic strength, and postsynaptic cell type. *Journal of Neuroscience*, *18*(24), 10464-10472.

Bienenstock, E. L., Cooper, L. N., & Munro, P. W. (1982). Theory for the development of neuron selectivity: orientation specificity and binocular interaction in visual cortex. *Journal of Neuroscience*, *2*, 32-48.

Bjork, E. L., & Healy, A. F. (1974). Short-term order and item retention. *Journal of Verbal Learning & Verbal Behavior*, 80-97.

Bofill-i-Petit, A., & Murray, A. F. (2004). Synchrony detection by analogue vlsi neurons with bimodal stdp synapses. In S. Thrun, L. Saul, & B. Scholköpf (Eds.), *Advances in neural information processing systems 16.* Cambridge, MA: MIT Press.

Borgers, C., & Kopell, N. (2003). Synchronization in networks of excitatory and inhibitory neurons with sparse, random connectivity. *Neural Computation*, *15*, 509-538.

Botvinick, M., & Plaut, D. C. (2006). Short-term memory for serial order: A recurrent neural network model. *Psychological Review*.

Bradski, G., Carpenter, G. A., & Grossberg, S. (1994). Store working memory networks for storage and recall of arbitrary temporal sequences. *Biological Cybernetics*, 469-480.

Brannon, E., & Terrace, H. (2000). Representation of the numerosities 1-9 by rhesus macaques (macaca mulatta). *Journal of Experimental Psychology: Animal Behavior Processes*, *26*(1), 31-49.

Brannon, E. M., & Terrace, H. S. (1998). Ordering of the numerosities 1 to 9 by monkeys. *Science*, *282*(5389), 746-749.

Bressloff, P. C., & Coombes, S. (2000). Dynamics of strongly coupled spiking neurons. *Neural Computation*, *12*(1), 91-129.

Brown, J. (1958). Some tests of the decay theory of immediate memory. *Quarterly Journal of Experimental Psychology*, *10*, 12-21.

Brown, L. L., Schneider, J. S., & Lidsky, T. I. (1997). Sensory and cognitive functions of the basal ganglia. *Current Opinion in Neurobiology*, *7*(2), 157-163.

Buchsbaum, B. R., & D'Esposito, M. (2008). The search for the phonological store: From loop to convolution. *Journal of Cognitive Neuroscience*, *20*(5), 762-778.

Bullock, D. (2004). Adaptive neural models of queuing and timing in fluent action. *Trends in Cognitive Sciences*, *8*(9), 426-433.

Burgess, N., & Hitch, G. (2005). Computational models of working memory: putting long-term memory into context. , *9*(11), 535.

Burgess, N., & Hitch, G. J. (1992). Toward a network model of the articulatory loop. *Journal of Memory and Language*, *31*, 429-460.

Burgess, N., & Hitch, G. J. (1999). Memory for serial order: A network model of the phonological loop and its timing. *Psychological Review*, *106 (3)*, 551-581.

Callicott, J. H., Mattay, V. S., Bertolino, A., Finn, K., Coppola, R., Frank, J. A., et al. (1999). Physiological characteristics of capacity constraints in working memory as revealed by functional mri. *Cerebral Cortex*, *9*(1), 20-26.

Campbell, S., & Wang, D. (1996). Synchronization and desynchronization in a network of locally coupled wilson-cowan oscillators. *IEEE Transactions on Neural Networks*, *7 (3)*, 541-552.

Carey, S. (1998). Knowledge of number: its evolution and ontogeny. *Science*, *282*(5389), 641-2.

Castelo-Branco, M., Neuenschwander, S., & Singer, W. (1998). Synchronization of visual responses between the cortex, lateral geniculate nucleus, and retina in the anesthetized cat. *Journal of Neuroscience*, *18*(16), 6395-6410.

Chafee, M. V., & Goldman-Rakic, P. S. (1998). Matching patterns of activity in primate prefrontal area 8a and parietal area 7ip neurons during a spatial working memorytask. *J Neurophysiol*, *79*(6), 2919-2940.

Chalmers, D. J. (1996). *The conscious mind: In search of a fundamental theory*. Oxford University Press.

Cheung, S. W. (2005). Frequency map variations in squirrel monkey primary auditory cortex. *Laryngoscope*, *115*(7), 1136-44.

Chincotta, D., & Hoosain, R. (1995). Reading rate, articulatory suppression and bilingual digit span. *European Journal of Cognitive Psychology*, *7*(2), 201-211.

Chomsky, N. (1957). *Syntactic structures*. The Hague: Mouton.

Cleeremans, A. (1993). Explicit sequence learning. In *Mechanisms of implicit learning: Connectionist models of sequence processing* (p. 169-187). Cambridge, Mass: MIT Press.

Cohen, J. D., Perlstein, W. M., Braver, T. S., Nystrom, L. E., Noll, D. C., Jonides, J., et al. (1997). Temporal dynamics of brain activation during a working memory task. *Nature*, *386*(6625), 604-608.

Colle, H. A., & Welsh, A. (1976). Acoustic masking in primary memory. *Journal of Verbal Learning & Verbal Behavior*, *Vol 15*(1), 17-31.

Conrad, R. (1960). Serial order intrusions in immediate memory. *British Journal of Psychology*, 45-48.

Conrad, R. (1964). Acoustic confusions in immediate memory. *British Journal of Psychology*, *55*, 75-84.

Conrad, R. (1965). Order error in immediate recall of sequences. *Journal of Verbal Learning and Verbal Behavior*, *4*, 161-169.

Conrad, R., & Hull, A. J. (1964). Information, acoustic confusion and memory span. *British Journal of Psychology*, *55*(4), 429-432.

Courtney, S. M., Ungerleider, B. G., Keil, K., & Haxby, J. V. (1997). Transient and sustained activity in a distributed neural system for human working memory. *Nature*, *386*(6625), 608-611.

Cowan, N. (2001a). The magical number 4 in short-term memory: A reconsideration of mental storage capacity. , *24*, 87-114. (TY - JOUR RP - NOT IN FILE U1 - TH04.105)

Cowan, N. (2001b). The magical number 4 in short-term memory: A reconsideration of mental storage capacity. , *24*, 87-114.

Crannell, C. W., & Parrish, J. M. (1957). A comparison of immediate memory span for digits, letters, and words. *Journal of Psychology*, *44*, 319-327.

Cromwell, H. C., & Berridge, K. C. (1996). Implementation of action sequences by a neostriatal site: A lesion mapping study of grooming syntax. *Journal of Neuroscience*, *16*(10), 3444-3458.

Crowder, R. G. (1972). Visual and auditory memory. In J. F. Kavanagh & I. G. Mattingly (Eds.), *Language by ear and by eye* (p. 251-276). Cambridge MA: MIT Press.

Crowder, R. G., Harvey, N., & Routh, D. A. (1983). The purity of auditory memory. *Philosophical Transactions of the Royal Society of London B*, *302*(1110), 251-265.

Crowder, R. G., & Morton, J. (1969). Precategorical acoustic storage (pas). *Perception & Psychophysics*, *5*(6), 365-373.

Damasio, A. R., & Tranel, D. (1993). Nouns and verbs are retrieved with differently distributed neural systems. *Proceedings of the National Academy of Sciences of the United States of America*, *90*(11), 4957-4960.

Daniele, A., Giustolisi, L., Silveri, M. C., Colosimo, C., & Gainotti, G. (1994). Evidence for a possible neuroanatomical basis for lexical processing of nouns and verbs. *Neuropsychologia*, *32*(11), 1325-+.

Desmond, J. E., & Moore, J. W. (1988). Adaptive timing in neural networks - the conditioned-response. *Biological Cybernetics*, *58*(6), 405-415.

De Zeeuw, C. I., Simpson, J. I., Hoogenraad, C. C., Galjart, N., Koekkoek, S. K. E., & Ruigrok, T. J. H. (1998). Microcircuitry and function of the inferior olive. *Trends in Neurosciences*, *21*(9), 391-400.

Di Chiara, G., Morelli, M., & Consolo, S. (1994). Modulatory functions of neurotransmitters in the striatum: Ach/dopamine/nmda interactions. *Trends Neurosci*, *17*(6), 228-33. (Journal Article Review)

Duncan, M., & Lewandowsky, S. (2005). The time course of response suppression: No evidence for a gradual release from inhibition. *Memory*, *13*, 236-246.

Ebbinghaus, H. (1885). *Uber das gedachtnis*. Leipzig: Dunker.

Elman, J. L. (1990). Finding structure in time. *Cognitive Science*, *14(2)*, 179-211.

Elman, J. L., Bates, E., Johnson, M. S., Karmiloff-Smith, A., Parisi, D., & Plunkett, K. (1996). *Rethinking innateness: a connectionist perspective on development.* Cambridge, MA: The MIT Press.

Estes, W. K. (1972). An associative basis for coding and organisation in memory. In A. W. Melton & E. Martin (Eds.), *Coding processes in human memory* (p. 161-190). Washington, D.C.: Winston & Sons.

Farrell, S., & Lewandowsky, S. (2002). An endogenous distributed model of ordering in serial recall. *Psychon Bull Rev*, *9*(1), 59-79. (1069-9384 Journal Article)

Ferezou, I., Haiss, F., Gentet, L. J., Aronoff, R., Weber, B., & Petersen, C. C. H. (2007). Spatiotemporal dynamics of cortical sensorimotor integration in behaving mice. *Neuron*, *56*(5), 907-923.

Ferguson, R. P., & Bray, N. W. (1976). Component processes of an overt rehearsal strategy in young children. *Journal of Experimental Child Psychology*, *21*(3), 490-506.

Fletcher, P. C., & Henson, R. N. A. (2001). Frontal lobes and human memory: Insights from functional neuroimaging. *Brain*, *124*(5), 849-881.

Frank, M. J., Loughry, B., & O'Reilly, R. C. (2001). Interactions between frontal cortex and basal ganglia in working memory: A computational model. *Cognitive, Affective and Behavioral Neuroscience*, *1*(2), 137-160.

Frankish, C. (1985). Modality-specific grouping effects in short-term memory. *Journal of Memory and Language*(24), 200-209.

Fuster, J. M. (1997). Network memory. *Trends in Neurosciences*, *20 (10)*, 451-459.

Fuster, J. M., Bauer, R. H., & Jervey, J. P. (1982). Cellular discharge in the dorsolateral prefrontal cortex of the monkey in cognitive tasks. *Experimental Neurology*, *77*(3), 679-694.

Fuster, J. M., Bauer, R. H., & Jervey, J. P. (1985). Functional interactions between inferotemporal and prefrontal cortex in a cognitive task. *Brain Res*, *330*(2), 299-307.

Fuster, J. M., & Jervey, J. P. (1982). Neuronal firing in the inferotemporal cortex of the monkey in a visual memory task. *Journal of Neuroscience*, *2*, 361-375.

Galaburda, A. M., & Pandya, D. N. (1983). The intrinsic architectonic and connectional organization of the superior temporal region of the rhesus monkey. *The Journal of Comparative Neurology*, *221*(2), 169-184.

Gayler, R. W. (2003). Vector symbolic architectures answer jackendoff's challenges for cognitive neuroscience. In P. Slezak (Ed.), *Iccs/ascs international conference on cognitive science* (p. 133-138). Sydney, Australia: CogPrints.

Gerstner, W., & Kistler, W. M. (2002). *Spiking neuron models: Single neurons, popula-*

tions, plasticity. Cambridge University Press.

Gillund, G., & Shiffrin, R. M. (1984). A retrieval model for both recognition and recall. *Psychological Review*, *Vol 91*(1), 1-67.

Glanzer, M., & Cunitz, A. R. (1966). Two storage mechanisms in free recall. *Journal of Verbal Learning & Verbal Behavior*, *5*(4), 351-360.

Goel, P., & Ermentrout, B. (2002). Synchrony, stability, and firing patterns in pulse-coupled oscillators. , *163*(3-4), 191-216.

Goodglass, H., Klein, B., Carey, P., & Jones, K. (1966). Specific semantic word categories in aphasia. *Cortex*, *2(1)*, 74-89.

Gray, C. (1999). The temporal correlation hypothesis of visual feature integration: still alive and well. *Neuron*, *24*(1), 31-25.

Graybiel, A. M., Aosaki, T., Flaherty, A. W., & Kimura, M. (1994). The basal ganglia and adaptive motor control. *Science*, *265*(5180), 1826-31.

Grossberg, S. (1978). Behavioral contrast in short term memory: Serial binary memory models or parallel continuous memory models? *Journal of Mathematical Psychology*, *Vol 17*(3), 199-219.

Grossberg, S. (1988). Nonlinear neural networks: Principles, mechanisms, and architectures. *Neural Networks*, *1*, 17-61.

Gruber, A. J., Dayan, P., Gutkin, B. S., & Solla, S. A. (2006). Dopamine modulation in the basal ganglia locks the gate to working memory. *Journal of Computational Neuroscience*, *20*(2), 153-166.

Gupta, P., Lipinski, J., Abbs, B., & Lin, P.-H. (2005). Serial position effects in nonword repetition. *Journal of Memory and Language*, *53*(1), 141-162.

Hartley, T., & Houghton, G. (1996). A linguistically constrained model of short-term memory for nonwords. *Journal of Memory and Language*, *35*(1), 1-31.

Haugland, R. P. (2005). *The handbook – a guide to fluorescent probes and labeling technologies* (10th edition (January 2005) ed., Vol. 2008). Molecular Probes.

Haykin, S. (1998). *Neural networks: A comprehensive foundation*. Prentice Hall.

Healy, A. F. (1974). Separating item from order information in short-term memory. *Journal of Verbal Learning & Verbal Behavior*, *13*(6), 644-655.

Hebb, D. O. (1949). *The organization of behavior: A neuropsychological theory*. New York: Wiley.

Henson, R. N. (1998). Item repetition in short-term memory: Ranschburg repeated. *J Exp Psychol Learn Mem Cogn*, *24*(5), 1162-81.

Henson, R. N. A. (1986). *Short-term memory for serial order*. Unpublished doctoral dissertation, University of Cambridge.

Henson, R. N. A. (1998). Short-term memory for serial order: The start-end model. *Cognitive Psychology*, *36*, 73-137.

Henson, R. N. A. (2001). Serial order in short-term memory. *Psychologist*, *14*(2), 70-73.

Hitch, G. J. (1975). The role of attention in visual and auditory suffix effects. *Memory & Cognition*, *3*(5), 501-505.

Hodgkin, A. L., & Huxley, A. F. (1952). A quantitative description of membrane current and application to conduction and excitation in nerve. *Journal of Physiolology (London)*, *117*, 500-544.

Hopfield, J. J. (1984). Neurons with graded response have collective computational properties like those of 2-state neurons. *Proceedings of the National Academy of Sciences of the United States of America-Biological Sciences*, *81*(10), 3088-3092.

Houghton, G. (1990). The problem of serial order: A neural network model of sequence learning and recall. In R. Dale, C. Mellish, & M. Zock (Eds.), *Current research in natural language generation.* London: Academic Press.

Houghton, G. (1994). Inhibitory control of neurodynamics: Opponent mechanisms in sequencing and selective attention. In M. Oaksford & G. D. A. Brown (Eds.), *Neurodynamics and psychology.* London: Academic Press.

Houghton, G., & Hartley, T. (1996). Parallel models of serial behaviour: Lashley revisited. *Psyche: An Interdisciplinary Journal of Research on Consciousness*, *2*(25).

Howard, M. W., Rizzutoy, D. S., Caplany, J. B., Madseny, J. R., Lismany, J., Aschenbrenner-Scheibe, R., et al. (2003). Gamma oscillations increase with working memory load in humans. *Cerebral Cortex*, *13*, 1369-1374.

Huber, D. E., Shiffrin, R. M., Lyle, K. B., & Ruys, K. I. (2001). Perception and preference in short-term word priming. *Psychological Review*, *108*(1), 149-182.

Ivry, R. B. (1996). The representation of temporal information in perception and motor control. *Current Opinion in Neurobiology*, *6*(6), 851-857.

Ivry, R. B., Keele, S. W., & Diener, H. C. (1988). Dissociation of the lateral and medial cerebellum in movement timing and movement execution. *Experimental Brain Research*, *73*(1), 167-180.

Izhikevich, E. M. (2000). Neural excitability, spiking and bursting. *International Journal of Bifurcation and Chaos*, *10 (6)*, 1171-1266.

Izhikevich, E. M. (2003). Simple model of spiking neurons. *IEEE Transactions on Neural Networks*, *14*(6), 15691572.

Izhikevich, E. M. (2004). Which model to use for cortical spiking neurons? *IEEE Transactions on Neural Networks*, *15*, 1063-1070.

Jackendoff, R. (2002). *Foundations of language: Brain, meaning, grammar, evolution.* Oxford, UK: Oxford University Press.

Jaeger, D., Kita, H., & Wilson, C. J. (1994). Surround inhibition among projection neurons is weak or nonexistant in the rat neostriatum. *Journal of Neurophysiology*, *72*(5), 2555-2558.

Jahnke, J. C., & Bower, R. E. (1986). Are there two ranschburg effects? *The American Journal of Psychology*, *99*(2), 275-288.

Jensen, O., & Lisman, J. E. (1998). An oscillatory short-term memory buffer model can account for data on the sternberg task. *The Journal of Neuroscience*, *18 (24)*, 10688-10699.

Joel, D., & Weiner, I. (1994). The organization of the basal ganglia-thalamocortical circuits - open interconnected rather than closed segregated. *Neuroscience*, *63*(2), 363-379.

Jones, D. M., Hughes, R. W., & Macken, W. J. (2007). The phonological store abandoned. *The Quarterly Journal of Experimental Psychology*, *60*(4), 505-511.

Jordan, M. I. (1986). *Serial order: A parallel distributed approach* (ICI report No. 8604). Institute for Cognitive Science, University of California, San Diego.

Judge, P. G., Evans, T. A., & Vyas, D. K. (2005). Ordinal representation of numeric quantities by brown capuchin monkeys (cebus apella). *J Exp Psychol Anim Behav Process*, *31*(1), 79-94.

Kahana, M. J., & Jacobs, J. (2000). Interresponse times in serial recall: Effects of intraserial repetition. *Journal of Experimental Psychology-Learning Memory and Cognition*, *26*(5), 1188-1197.

Kanamaru, T., & Sekine, M. (2003). Array-enhanced coherence resonance in the diffusively coupled active rotators and its analysis with the nonlinear fokker-planck equation. *IEICE Transactions on Fundamentals of Electronics Communications and Computer Sciences*, *E86a*(9), 2197-2202.

Karlsen, P. J., Imenes, A. G., Johannessen, K., Endestad, T., & Lian, A. (2007). Why does the phonological similarity effect reverse with nonwords? *Psychological Research/Psychologische Forschung*, *71*(4), 448-457.

Kawaguchi, Y. (1997). Neostriatal cell subtypes and their functional roles. *Neuroscience Research*, *27*, 1-8.

Keating, J. G., & Thach, W. T. (1997). No clock signal in the discharge of neurons in the deep cerebellar nuclei. *Journal of Neurophysiology*, *77*(4), 2232-2234.

Kistler, W. M., & De Zeeuw, C. I. (2002). Dynamical working memory and timed responses: The role of reverberating loops in the olivo-cerebellar system. *Neural Computation*, *14*(11), 2597-2626.

Kleinfeld, D., & Waters, J. (2007). Wilder penfield in the age of youtube: Visualizing the sequential activation of sensorimotor areas across neocortex. *Neuron*, *56*(5), 760-762.

Knuth, D. E. (1997). *The art of computer programming* (3rd ed.). Reading, Mass.: Addison-Wesley.

Lai, C. S. L., Gerrelli, D., Monaco, A. P., Fisher, S. E., & Copp, A. J. (2003). Foxp2 expression during brain development coincides with adult sites of pathology in a severe speech and language disorder. *Brain*, *126*, 2455-2462.

Lian, A., Karlsen, P. J., & Eriksen, T. B. (2004). Opposing effects of phonological similarity on item and order memory of words and nonwords in the serial recall task. *Memory*, *12*(3), 314-37.

Lian, A., Karlsen, P. J., & Winsvold, B. (2001). A re-evaluation of the phonological similarity effect in adults short-term memory of words and nonwords. , *9*, 281.

Lieberman, P. (2000). *Human language and our reptilian brain: The subcortical bases of speech, syntax, and thought.* Cambridge: Mass: Harvard University Press.

Logie, R. H., Della Sala, S., Wynn, V., & Baddeley, A. D. (2000). Visual similarity effects in immediate verbal serial recall. *The Quarterly Journal of Experimental Psychology A: Human Experimental Psychology*, *53A*(3), 626-646.

Luck, S. J., & Vogel, E. K. (1997). The capacity of visual working memory for features and conjunctions. *Nature*, *390*, 279-281.

Malsburg, C. von der. (1981). The correlation theory of brain function. In *Internal report 81-2.* MPI Biophysical Chemistry.

Malsburg, C. von der. (1985). Nervous structures with dynamical links. *Berichte Der Bunsen-Gesellschaft-Physical Chemistry Chemical Physics*, *89*(6), 703-710.

Malsburg, C. von der. (1995). Binding in models of perception and brain function. *Curr Opin Neurobiol*, *5*(4), 520-6.

Markram, H., Lubke, J., Frotscher, M., & Sakmann, B. (1997). Regulation of synaptic efficacy by coincidence of postsynaptic aps and epsps. *Science*, *275*(5297), 213-215.

Marsden, C. D. (1980). The enigma of the basal ganglia and movement. *Trends in Neurosciences*, *3*(11), 284-287.

McCulloch, W. S., & Pitts, W. H. (1943). A logical calculus of the ideas immanent in nervous activity. *Bulletin of Mathematical Biophysics*, *5*, 115-133.

McNeil, A. M., & Johnston, R. S. (2004). Word length, phonemic, and visual similarity effects in poor and normal readers. , *32*, 687.

Melamed, O., Gerstner, W., Maass, W., Tsodyks, M., & Markram, H. (2004). Coding and learning of behavioral sequences. *Trends in Neurosciences*, *27*(1), 11-14.

Middleton, F. A., & Strick, P. L. (2000). Basal ganglia output and cognition: Evidence from anatomical, behavioral, and clinical studies. *Brain and Cognition*, *42*(2), 183-200.

Miikkulainen, R., Bednar, J. A., Choe, Y., & Sirosh, J. (2005). *Computational maps in the visual cortex*. New York: Springer.

Miller, E. K. (2000). The prefrontal cortex: No simple matter. *Neuroimage*, *11*(5), 447-450.

Mirollo, R. E., & Strogatz, S. H. (1990). Synchronization of pulse-coupled biological oscillators. , *50*(6), 1645-1662.

Molavi, D. W. (1997). *The w.u.s.m. neuroscience tutorial.* (Retrieved 6/20/2008, from `http://thalamus.wustl.edu/course/`)

Moore, J. W., Desmond, J. E., & Berthier, N. E. (1989). Adaptively timed conditioned-responses and the cerebellum - a neural network approach. *Biological Cybernetics*, *62*(1), 17-28.

Morton, J., & Holloway, C. M. (1970). Absence of a cross-modal "suffix effect" in short-term memory. *The Quarterly Journal of Experimental Psychology*, 167-176.

Mountcastle, V. B. (1957). Modality and topographic properties of single neurons of cat's somatic sensory cortex. *Journal of Neurophysiology J Neurophysiol*, *20*(4), 408-434.

Murdock, J., Bennet B. (1962). The serial position effect of free recall. *J Exp Psychol Learn Mem Cogn*, *64*(5), 482-488.

Murray, D. J. (1968). Articulation and acoustic confusability in short-term memory. *Journal of Experimental Psychology*, *78*(4, Pt. 1), 679-684.

Murray, S. S., Kersten, D., Olshausen, B. A., Schrater, P., & Woods, D. L. (2002). Shape perception reduces activity in human primary visual cortex. *PNAS*, *99 (23)*, 15164 -15169.

Newell, A., & Rosenbloom, P. S. (1981). Mechanisms of skill acquisition and the law of practice. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition.* Hillsdale, NJ: Erlbaum.

Nipher, F. (1878). On the distribution of errors in numbers written from memory. *Transactions of the Academy of Science of St. Louis*, *3*, CCX-CCXI.

Nishikawa, T., & Hoppensteadt, F. (2003). System of phase oscillators with diagonalizable interaction. *SIAM JOURNAL ON APPLIED MATHEMATICS*, *63*(5), 1615-1626.

Nottebohm, F. (2002). Birdsong's clockwork. *Nature Neuroscience*, *5*(10), 925-926.

Oja, E. (1982). A simplified neuron model as a principle component analyzer. *Journal of Mathematical Biology*, *15*(3), 267-273.

O'Neill, M. E., & Douglas, V. I. (1996). Rehearsal strategies and recall performance with boys with and without attention deficit hyperactivity disorder. *Journal of Pediatric Psychology*, *21*(1), 73-88.

O'Reilly, R. C., & Soto, R. (2001). A model of the phonological loop: Generalization and binding. *Advances in Neural Information processing Systems (NIPS)*, *14*.

Owen, A. M. (1997). Cognitive planning in humans: Neuropsychological, neuroanatomical and neuropharmacological perspectives. *Progress in Neurobiology*, *53*(4), 431-450.

Page, M., & Norris, D. (1998). The primacy model: A new model of immediate serial recall. *Psychological Review*, *105 (4)*, 761-781.

Paivio, A. (1986). *Mental representations: a dual coding approach.* Oxford, England: Oxford University Press.

Palmer, S. E., & Ornstein, P. A. (1971). Role of rehearsal strategy in serial probed recall. *Journal of Experimental Psychology*, *88*(1), 60-66.

Penney, C. G., & Blackwood, P. A. (1989). Recall mode and recency in immediate serial recall: Computer users bewarep. *Bulletin of the Psychonomic Society*, *Vol 27*(6), 545-547.

Peterson, L. R., & Peterson, M. J. (1959). Short-term retention of individual verbal items. *J Exp Psychol Learn Mem Cogn*, *58*, 193-198.

Placantonakis, D. G., Bukovsky, A. A., Zeng, X.-H., Kiem, H.-P., & Welsh, J. P. (2004). Fundamental role of inferior olive connexin 36 in muscle coherence during tremor. *PNAS*, *101*(18), 7164-7169.

Postman, L., & Phillips, L. W. (1965). Short-term temporal changes in free recall. *Quarterly Journal of Experimental Psychology*, *17*(2), 132-138.

Raaijmakers, J. G. W., & Shiffrin, R. M. (1980). Sam: A theory of probabilistic search of associative memory. In G. H. Bower (Ed.), *The psychology of learning and motivation* (Vol. 14, p. 207-262). New York: Academic Press.

Ramnani, N., & Owen, A. M. (2004). Anterior prefrontal cortex: Insights into function from anatomy and neuroimaging. *Nature Reviews Neuroscience*, *5*(3), 184-194.

Rodriguez, P., & Elman, J. L. (1999). A recurrent neural network that learns to count. *Connection Science*, *11*, 5-40.

Rosenblatt, F. (1958). The perceptron - a probabilistic model for information-storage and organization in the brain. *Psychological Review*, *65*(6), 386-408.

Rubin, D. C., & Wenzel, A. E. (1996). One hundred years of forgetting: A quantitative description of retention. *Psychological Review*, *Vol 103*(4), 734-760.

Rumelhart, D. E., & McClelland, J. L. (1986). *Parallel distributed processing.* The MIT Press.

Rumelhart, D. E., & Norman, D. A. (1982). Simulating a skilled typist: A study of skilled cognitive-motor performance. *Cognitive Science*, *6*, 1-36.

Rundus, D. (1971). Analysis of rehearsal processes in free recall. *J Exp Psychol Learn Mem Cogn*, *Vol. 89*(1), 63-77.

Ryan, J. (1969). Grouping and short-term memory: Different means and patterns of grouping. *The Quarterly Journal of Experimental Psychology*, *21(2)*, 137-147.

Rypma, B., & D'Esposito, M. (2003). A subsequent-memory effect in dorsolateral prefrontal cortex. *Cognitive Brain Research*, *16*(2), 162-166.

Saint-Aubin, J., & Poirier, M. (1999). Semantic similarity and immediate serial recall: Is there a detrimental effect on order information? , *52*, 367.

Salame, P., & Baddeley, A. (1990). The effects of irrelevant speech on immediate free recall. *Bulletin of the Psychonomic Society*, *Vol 28*(6), 540-542.

Salame, P., & Baddeley, A. D. (1982). Disruption of short-term memory by unattended speech: Implications for the structure of working memory. *Journal of Verbal Learning & Verbal Behavior*, *Vol 21*(2), 150-164.

Schultz, W., Apicella, P., & Ljungberg, T. (1993). Responses of monkey dopamine neurons to reward and conditioned-stimuli during successive steps of learning a delayed-response task. *Journal of Neuroscience*, *13*(3), 900-913.

Schulz, R., & Reggia, J. (2004). Temporally asymmetric learning supports sequence processing in multi-winner self-organizing maps. *NEURAL COMPUTATION*, *16*(3), 535-561.

Seger, C. A. (2006). The basal ganglia in human learning. *Neuroscientist*, *12*(4), 285-90.

Severin, F. T., & Rigby, M. K. (1963). Influence of digit grouping on memory for telephone numbers. *Journal of Applied Psychology*, *47*(2), 117-119. (English)

Shastri, L. (1999). Advances in shruti - a neurally motivated model of relational knowledge representation and rapid inference using temporal synchrony. *Applied Intelligence*, *11*, 79-108.

Shieh, D. X., & Elman, J. L. (2006). The divergent-reconvergent model of serial order encoding and retrieval. In *Proceedings of the 29th annual meeting of the cognitive science society.*

Shieh, D. X., & Elman, J. L. (2007). How are repeated items encoded in working memory? In *Proceedings of the 30th annual meeting of the cognitive science society.*

Shiffrin, R. M. (2003). Modeling memory and perception. *Cognitive Science*, *27*(3), 341-378.

Shiffrin, R. M., & Steyvers, M. (1997). A model for recognition memory: Rem–retrieving effectively from memory. *Psychon Bull Rev*, *Vol 4*(2), 145-166.

Singer, W. (1999). Neuronal synchrony: a versatile code for the definition of relations? *Neuron*, *24*(1), 49-65, 111-25.

Smith, E. E., Jonides, J., Marshuetz, C., & Koeppe, R. A. (1998). Components of verbal working memory: Evidence from neuroimaging. *Proceedings of the National Academy of Sciences of the United States of America*, *95*(3), 876-882.

Song, S., Miller, K. D., & Abbott, L. F. (2000). Competitive hebbian learning through spike-timing-dependent synaptic plasticity. *Nat Neurosci*, *3*(9), 919-26.

Sternberg, S. (1966). High-speed scanning in human memory. *Science*, *153*, 652-654.

Strogatz, S. H., & Mirollo, R. E. (1988). Phase-locking and critical phenomena in lattices of coupled nonlinear oscillators with random intrinsic frequencies. *Physica D*, *31*(2), 143-168.

Talavage, T. M., Sereno, M. I., Melcher, J. R., Ledden, P. J., Rosen, B. R., & Dale, A. M. (2004). Tonotopic organization in human auditory cortex revealed by progressions of frequency sensitivity. *Journal of Neurophysiology*, *91*(3), 1282-1296.

Tepper, J. M., Koos, T., & Wilson, C. J. (2004). Gabaergic microcircuits in the neostriatum. *Trends in Neurosciences*, *27*(11), 662-669.

Terrace, H., Son, L., & Brannon, E. (2003). Serial expertise of rhesus macaques. *Psychological Science*, *14*(1), 66-73.

Voogd, J., & Bigare, F. (1980). Topographical distribution of olivary and corticonuclear fibres in the cerebellum. a review. In J. Courville, C. de Montigny, & Y. Lamarre (Eds.), *The inferior olivary nucleus, anatomy and physiology* (p. 207-235). Raven Press.

Wersing, H. (2000). *Spatial feature binding and learning in competitive neural layer architectures*. Unpublished doctoral dissertation, Faculty of Technology, University of Bielefeld.

Wersing, H., Steil, J. J., & Ritter, H. (2001). A competitive-layer model for feature binding and sensory segmentation. *Neural Computation*, *13*(2), 357-387.

Wickelgren, W. A. (1969). Context-sensitive coding, associative memory, and serial order in (speech) behavior. *Psychological Review*, *76*(1), 1-15.

Wilson, C. (2008). Up and down states. *Scholarpedia*, *3*(6), 1410-1410.

Wilson, C. J. (1995). The contribution of cortical neurons to the firing patterns of striatal spiny neurons. In J. C. Houk, J. L. Davis, & D. G. Beiser (Eds.), *Models of information processing in the basal ganglia* (p. 29-50). Cambridge: MIT Press.

Wixted, J. T., & Ebbesen, E. B. (1991). On the form of forgetting. *Psychological Science*, *Vol 2*(6), 409-415.

Yee, P. V., & Haykin, S. (2001). *Regularized radial basis function networks: Theory and applications*. John Wiley.

Zipser, D. (1986). A model of hippocampal learning during classical conditioning. *Behavioral Neuroscience*, *100*(5), 764-776.