Combinatorial Analysis of Sequential Firing Patterns Across Multiple Neurons
Applied to
Decoding Memory of Sequential Spatial Experience in Rat Hippocampus

by

Albert K. Lee

A.B. Chemistry and Physics
Harvard College, 1994

Submitted to the Department of Brain and Cognitive Sciences
in Partial Fulfillment of the Requirements for the Degree of
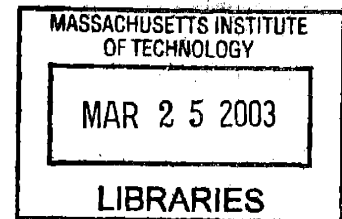
Doctor of Philosophy

at the

Massachusetts Institute of Technology

February 2003

Signature of Author: _____

Department of Brain and Cognitive Sciences
January 10, 2003

Certified by: _____

Matthew A. Wilson
Associate Professor of Neurobiology
Thesis Supervisor

Accepted by: _____

Earl K. Miller
Professor of Neuroscience
Chairman, Department Graduate Committee

Combinatorial Analysis of Sequential Firing Patterns Across Multiple Neurons
Applied to
Decoding Memory of Sequential Spatial Experience in Rat Hippocampus

by

Albert K. Lee

Submitted to the Department of Brain and Cognitive Sciences
on January 10, 2003
in Partial Fulfillment of the Requirements for the Degree of
Doctor of Philosophy

ABSTRACT

There is broad agreement that the hippocampus is crucially involved in the formation of richly-detailed, long term memories of events in humans. A key aspect of such memories is the temporal order and spatial context of the events experienced. Evidence from a wide variety of behavioral and electrophysiological experiments indicates that the rodent hippocampal spatial memory system is a model system for studying this type of memory in humans.
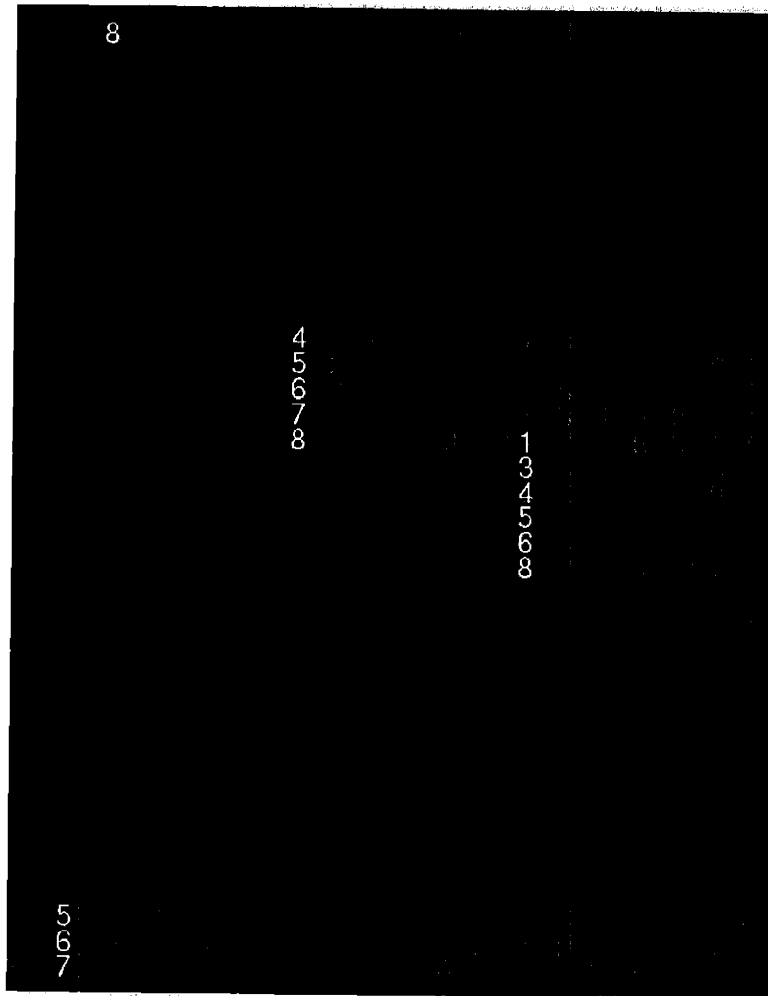
Here, we develop a new combinatorial method for analyzing sequential firing patterns involving an arbitrary number of neurons based on relative time order. We then apply this method to decode memories of sequential spatial experience in the rat hippocampus during slow wave sleep.

Specifically, rats are trained to repeatedly run through a sequence of spatial receptive fields ("place fields") of hippocampal CA1 "place cells" in a fixed temporal order. The spiking activity of many such individual cells is recorded before (PRE), during (RUN), and after (POST) this experience. By treating each place field traversed as an individual event, the rat's experience in RUN can be represented by the resulting sequence of place fields traversed, and therefore by the activity of the corresponding place cells. Then to characterize the extent to which the sequential nature of the RUN experience has been encoded into memory, we search for firing patterns related to the RUN sequence in POST. To do so, we develop a method that statistically quantifies the similarity between any desired "reference sequence" (here chosen to be the RUN sequence) and arbitrary temporal firing patterns. We find that the RUN sequence is repeatedly re-expressed during POST slow wave sleep in brief bursts involving four or more cells firing in order, but not so during PRE. This provides direct neural evidence of the rapid learning of extended spatial sequences experienced in RUN. The results may shed light on the encoding of memories of events in time ("episodic memories") in humans. Furthermore, the multiple spike train analysis method developed here is general and could be applied to many other neural systems in many different experimental conditions.

Thesis Supervisor: Matthew A. Wilson
Title: Associate Professor of Neurobiology

TABLE OF CONTENTS

This shows a continuous 200-second period of the activity of 8 pyramidal neurons from the CA1 region of a rat hippocampus. This activity is taken from the first slow-wave sleep period immediately after the rat performed a task in which it repeatedly ran through the place fields of cells 1 through 8 in that order. The record starts in the upper left-hand corner, and time proceeds downward in a vertical column. At the bottom of each column, time wraps around to the top of the next column to the right. Each number signifies that the corresponding neuron fired a burst of action potentials. Gaps represent short periods of no activity. Low probability matches to the behaviorally experienced sequence are highlighted. (Isolated activity from single cells has been dimmed.) A novel combinatorial analysis method reveals that such matches occurred far more often than expected by chance in the sleep after, but no more often than expected by chance in the sleep before, the experience. Thus this may represent a neural code of sequential memory.

CHAPTER 1
A BRIEF INTRODUCTION

We humans can effortlessly encode many aspects of our daily experience, such as the details and order of events (e.g. people, places, conversations) experienced during a walk through the park. These memories (often called "episodic" memories) are formed rapidly, do not require repetition, and can last a lifetime. What is the physical basis of this ability? Where does the encoding occur, when does it occur, how does it occur, and what does such a memory "look like" in the brain?

Fortunately, there are already many clues to help us answer these questions. People with damage to a specific brain area called the hippocampus can no longer recall what happened more than 15 minutes ago, even though they appear normal in every other respect, including perception, personality, motivation, attention, comprehension, and reasoning (Scoville and Milner, 1957; Milner, 1966, Zola-Morgan et al., 1986). Even the memory problem itself is very specific. Such a person can remember what just happened ("short term" memory), can clearly recall events from years before the damage, and can even learn and retain new motor skills (without remembering having the practice sessions!). Thus the hippocampus appears to be particularly crucial for forming new "long term" memories of just the type we would like to study. So we have a good start for the "where." And as for "when," further experiments suggest the hippocampus is necessary at the time of the original experience (Milner, 1966).

However, many details cannot currently be learned from humans. For instance, what are the individual neurons of the hippocampus doing during learning? How might these parts allow the hippocampus to do its job? To answer questions like these, we need an animal model. Again, there is much previous work we can build on. Rats too have a hippocampus. Furthermore, rats without a hippocampus can no longer learn where food is located within a maze, which is something rats are normally very good at (O'Keefe and Nadel, 1978; Morris et al., 1982). Consistent with this evidence of hippocampal involvement in rodent spatial learning, individual hippocampal neurons (called "place cells") exhibit clear spatially-related activity. A place cell fires action potentials (spikes) at a high rate when the rat is in a particular, relatively small area within a maze (called the cell's "place field"), and is almost silent when the rat is elsewhere (O'Keefe and Dostrovsky, 1971; for a review see Muller, 1996). Different neurons fire in different locations, and each cell's firing location is stable across time. We can see this in the

activity of 10 different hippocampal neurons recorded simultaneously from a single rat as it runs repeatedly from left to right along a narrow track (Figure 1.1).
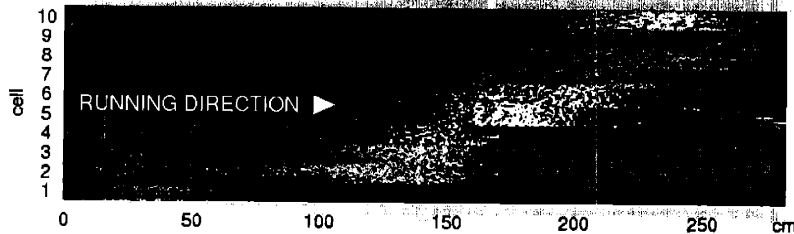

Figure 1.1 Location-dependent activity of 10 individual hippocampal neurons recorded from a rat as it runs repeatedly (here 30 times) from left to right along a narrow track. Each point represents an action potential fired by that cell. For each cell, the 30 laps are represented by 30 rows, with lap 1 at the bottom and lap 30 at the top. Each lap takes about 5 seconds.

Based on the hippocampal dependence of both, we make the leap that rodent spatial learning is an animal model of our ability to encode memories of events in time. The corollary: what happens to "place cells" in rats may tell us how human hippocampal neurons create our memories. In particular, the sequence of places where neurons 1 through 10 fire may be a good model for the sequence of events during our walk in the park. (One might imagine that different neurons in the human hippocampus fire for the different events in the sequence.) We then ask: Can we detect specific memory traces of this experience by observing the activity of these neurons after the experience? The specific form of any such traces, which is unknown a priori (e.g. firing in the same order, the reverse order, all at once, or something more complicated), could also tell us in what form our brain stores memories of sequential experience.

We start by looking for such traces during the rat's sleep immediately after it has run. Why sleep? Because in searching for traces of previous spatial experience, we do not want interference caused by any new spatial behavior, and there is clearly no spatial behavior during sleep. A sample 10-second period of "slow wave" sleep (Figure 1.2) reveals no obvious structure related to the 5-second pattern during running (Figure 1.1). How will we make sense of the hours of data like this?
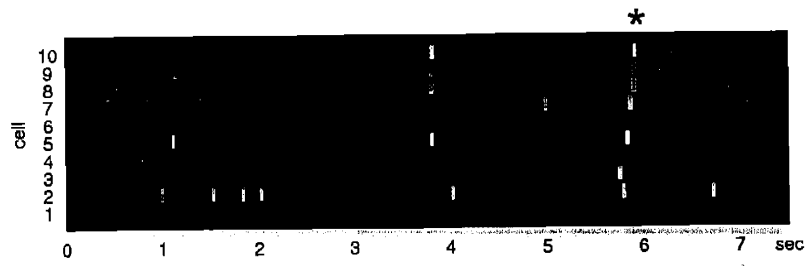
Figure 1.2 Example activity of the same 10 neurons from Figure 1.1, this time during the rat's slow wave sleep immediately following its running experience (shown in Figure 1.1). Each vertical tick represents an action potential fired by that cell.

However, a closer look reveals a pattern suggestive of a memory of the rat's sequential experience while running (Figure 1.3).

not before, the running experience). (A note on Chapters 3 and 4: each are intended to be self-contained pieces on their respective topics, thus a significant amount of repetition between them, and the other parts of this work, is unavoidable.)

And now for Chapter 2, in which we will make the case (in more detail) that the hippocampus is *the* area to look at for studying this kind of memory (our walk through the park), and that the rodent place field sequence is a good animal model for it. Then we will briefly describe our experiment to study memory of such place field sequences in the rat hippocampus. We will also describe the previous work that is related to our own.

CHAPTER 2
BACKGROUND

Overview

The overall goal of this chapter will be to make the case that the rodent hippocampal spatial memory system is a model system for studying memory of events in time ("episodic" memory) in humans. The following is an outline of this chapter. First, we will review the human literature on what episodic memory is and why it is worth studying. Second, we will review a debate on the role of the hippocampus and related structures in episodic and semantic memory. The conclusion from this will be that there is a broad consensus that the hippocampus itself is crucially involved in the formation of episodic memory (even though the same cannot be said for semantic memory). Third, we will describe relevant aspects of the rodent hippocampal literature, including the role of the hippocampus in spatial memory and its basic electrophysiology, focusing in particular on place cells. Fourth, we will argue that the temporal sequence of place fields traversed by a rat is a good model for an episode. Fifth, we will review the work to date concerning electrophysiological evidence of memory in the rodent hippocampus. This will indicate a trend towards higher-order analyses (i.e. analyses of the interaction between multiple individual neurons) in trying to understand exactly what kind of memory processing the hippocampus does. Sixth, we will review the literature on the higher-order analysis of spike trains in general, covering methods which have been developed for a variety of systems. Finally, we will describe the specific aims of this thesis: the development of a new higher-order method to analyze sequential firing patterns, and then an experiment to test for memory of sequential spatial experience in the rat hippocampus using this method.

H.M., the Hippocampus, and Episodic Memory

The case of patient HM is cited so often that it is sometimes easy to forget the magic of the discoveries flowing from it. The essential observation from HM and later work on other patients can be idealized as follows: remarkably specific damage to a particular brain area (the hippocampus) is associated with a remarkable and specific deficit: the total inability to form new long term memories of facts and events while perfectly preserving all other cognitive functions, in addition to preserving all such previously learned memories (Scoville and Milner, 1957; Milner, 1966; Milner, 1972; Corkin, 1984; Zola-Morgan et al., 1986). (Note that this deficit is

10

present only if the hippocampus is damaged *bilaterally*, i.e. in *both* hemispheres (Scoville and Milner, 1957; Milner, 1966; Milner, 1972).) The idealized conclusion to be drawn from this is that the hippocampus itself is the machine that specializes in forming long term memories of experiences which we can consciously recall, such as all the details about what happened at your birthday party last week, or during a walk through the park, as opposed to such things as motor learning (e.g. how to ride a bike). Memories of birthday parties, vacations, and conversations— these are examples of what we usually mean when we think about memory, and this kind of memory is named episodic memory (defined more formally later). Thus how the hippocampus forms episodic memories is a question one can ask, and one that is of interest to many.

To what extent is this conclusion (that the hippocampus itself is the machine that forms conscious memories) idealized? First of all, lesions are not always so specific. For instance, HM lost 2/3 of his hippocampus bilaterally, had unknown damage to the other 1/3, lost his amygdala bilaterally, and lost substantial portions of nearby areas from temporal cortex. Second, the deficit in new memory formation (called anterograde amnesia) is not always so complete. Some but not all facts and events may be remembered or partially remembered. In particular, patient RB, who had damage limited to the CA1 subregion of hippocampus bilaterally, was characterized as having a moderately severe deficit, that is, not as severe as HM's (Zola-Morgan et al., 1986). Third, there is sometimes an inability to remember some things learned right before the damage (called retrograde amnesia). On the other hand, in all cases the preservation of other cognitive functions (e.g. perception, personality, attention, motivation, short term memory, IQ, vocabulary, comprehension) is usually perfect. In addition, retrograde amnesia may reflect an integral feature of normal hippocampal memory function (Milner, 1966; Squire, 1992). In particular, the hippocampus may only be necessary for a limited amount of time (days to years) after the initial experience before a memory trace can be independently viable elsewhere, perhaps in neocortex. (This idea is called memory consolidation.) But even with all these caveats, there is strong evidence (e.g. RB) in favor of the idealized conclusion. In fact, in their original paper, Scoville and Milner had studied many patients in addition to HM and had concluded that the degree of bilateral hippocampal damage was the *best predictor* of impairment of new memory formation (Scoville and Milner, 1957).

Finally, before continuing, it is important to further clarify what exactly the hippocampus is and is not claimed to be doing. In particular, it should be stressed that all research to date supports two sharp distinctions. The first is between short and long term memory for facts and

events. Patients with hippocampal lesions have no deficit in short term memory. For instance, they can retell a short story right after hearing it, or can redraw a complex figure right after seeing it. Their deficit is apparent only after a delay of more than 15 minutes (Scoville and Milner, 1957; Milner, 1966; Milner, 1972; Zola-Morgan et al., 1986). In the case of HM, a distraction will cause instant and total loss of memory for immediately preceding events, and any delay of more than 30 seconds is enough to cause loss of memory for nonverbalizable items (e.g. clicks, colors) (Milner, 1966). Thus perception, attention, and short term (i.e. less than 15 minutes) memory are intact, while long term memory and the ability to bridge gaps in attention are impaired.

The second distinction is between long term memory for facts and events (often called declarative or explicit memory) and other kinds of long term memories (often grouped under the names nondeclarative or implicit memory). Milner made the original observation that HM was able to form some kinds of new long term memories. For example, HM learned at a normal rate in performing the motor skill task of tracing objects by looking off a mirror, even though he did not remember the actual practice sessions (Milner, 1962; Milner, 1966). The list of preserved long term memory abilities has since been expanded by other researchers to include priming, simple classical conditioning, habituation, and more. Such learning is generally gradual and unconscious, whereas fact and event learning can occur rapidly (often in a single trial) and is available for conscious recollection. The main point of these different memory categories is that memory is not unitary, that different brain areas are responsible for different kinds of learning, and that the hippocampus (and perhaps areas surrounding it) is specifically responsible for forming new long term memories of facts and events (Milner, 1966; Milner, 1972; Corkin, 1984; Squire 1992).

Episodic Versus Semantic Memory

So far we have described the deficit from hippocampal lesions as an inability to form new long term memories for facts and events. But researchers have further divided this into two kinds of memory: semantic (for facts) and episodic (for events) (Tulving, 1972; Tulving 1983). (This general distinction has been noted in the past by several others using different names.) We have already informally defined episodic memory, but more formally, episodic memory refers to the "capacity for recollecting happenings from the past, for remembering events that occurred in particular spatial and temporal contexts," while semantic memory refers to the "capacity for

12

recollecting facts and general knowledge about the world" without reference to when and where they were learned (Squire, 1998). Again, an example of an episodic memory is the sequence and details of events during a walk through the park, while an example of a semantic memory is knowledge of who was the first president of the United States.

The distinction between episodic and semantic memory is important because there is currently an unresolved debate about the role of the hippocampus in each. One side (the unitary theory) argues that the hippocampus and surrounding cortex of the medial temporal lobe (i.e. the entorhinal, perirhinal, and parahippocampal cortices) all work together such that damage to any part of it impairs episodic and semantic memory equally, and that the amount of impairment is proportional to the amount of damage (Squire, 1998). The other side (let's call it the non-unitary theory) argues that the entorhinal and perirhinal cortices are sufficient to form "context-free" semantic memories, but that forming "context-rich" episodic memories requires the additional processing of the hippocampus (Vargha-Khadem et al., 1997; Mishkin, 1998; Tulving, 1998).

The evidence cited in support of the non-unitary theory includes the following. First, several patients who suffered damage limited to the hippocampus early in life have highly impaired episodic memory yet seem to have acquired a large amount of semantic memory (Vargha-Khadem et al., 1997). Second, Mishkin states that in monkeys, lesions of the hippocampus alone do not result in deficits in recognition memory (which is somewhat like, but not technically equivalent to, semantic memory), but lesions including subhippocampal cortices result in near "total failure in recognition memory" (Mishkin, 1998). Finally, neuroanatomical evidence suggests a hierarchy in which the hippocampus is at the top, below which is the entorhinal cortex, followed by the perirhinal and parahippocampal cortices, and below that the higher sensory cortices. Thus, Vargha-Khadem and Mishkin argue that it makes sense that each higher level makes successively more complex associations, and therefore that the hippocampus makes the most complex associations, that is, those in "context-rich" episodic memories (Vargha-Khadem et al. 97; Mishkin, 1998). Against this, Squire argues that rigorous analysis may reveal that episodic and semantic memory are still equally impaired in the patients with hippocampal lesions early in life. In addition, Squire argues that studies which attempt to quantitatively compare episodic and semantic memory deficits in patients with hippocampal damage show them to be equally impaired (Squire, 1998).

My view of the debate is as follows. I agree with Squire that Vargha-Khadem et al. have not convincingly proven that semantic memory is truly spared relative to episodic memory in the early hippocampal damage patients. Furthermore the entire human lesion literature does not provide convincing evidence of this dissociation (semantic spared relative to episodic memory) in either young, or especially adult, patients. It would have been especially interesting if formation of new episodic and new semantic memories had been compared in RB (since he had damage limited to the hippocampus, in fact, to CA1 only), but that was not done. Nevertheless, I find Vargha-Khadem and Mishkin's hierarchical view of the hippocampus and related structures to be compelling because of convergent evidence from human patients, animal lesions, and neuroanatomy. But the most important conclusion relevant for my work is this: regardless of which side one takes in this debate, both sides agree that the hippocampus itself is crucial for at least episodic memory formation in humans. That is, the hippocampus plays a key role in forming new long term memories of a certain kind—those in which the spatial and temporal context are an essential part. The still uncertain role of the hippocampus in semantic memory is not relevant to my work.

The Rodent Hippocampus, Spatial Memory, and Place Cells

Work on the rodent hippocampus supports three conclusions of importance for our purposes. First, as in humans, the rodent hippocampus is crucial for forming certain kinds of memory, at least one of which is spatial memory. Second, the rodent hippocampus appears to contain many neural connections whose strengths can be altered in a rapid and persistent manner (i.e. long-term potentiation and long-term depression, LTP and LTD). Third, the hippocampus of a freely behaving rodent exhibits cells (called place cells) which fire in a highly spatially selective manner. Thus rodent spatial memory is a possible model for human hippocampal-dependent memory, and place cells are an electrophysiological (versus behavioral) way to observe this spatial memory in action.

A note: While the remainder of this chapter and all our experimental work (Chapter 4) deals with rats, it is likely that everything would also apply to mice. For instance, mice exhibit hippocampal place cells that appear to be much like those of rats (McHugh et al., 1996). This is relevant because the mouse is becoming an increasingly important system due to the ability to employ genetic engineering. Results from rats could then be applied to mice, with the added ability to genetically manipulate neural circuitry and related molecules.

The hippocampus is crucial for spatially-guided behavior in the rat (O'Keefe and Nadel, 1978). A particularly nice illustration of this is given by the water maze task. Unlike normal rats, rats with hippocampal lesions cannot learn the location of a hidden platform in a circular pool of water surrounded by distant visual cues (Morris et al., 1982). This deficit is not due to impaired motor skills, motivation, or visual perception, since the lesioned rats are unimpaired at reaching a visible platform. Thus in its specificity, the spatial learning deficit in rats is like the fact and event learning deficit in humans with hippocampal lesions (who had basically no other cognitive deficits). That is, in both humans and rats, hippocampal lesions devastate some functions (episodic memory in humans, spatial memory in rats) while leaving many other functions intact. While several researchers believe the rodent hippocampus is not just necessary for spatial behavior and memory (Squire, 1992; Eichenbaum et al., 1999), that issue is not of primary concern to us here. Everyone agrees that space is a prominent aspect of hippocampal function, and this is sufficient for our present purposes since we are only looking for at least one good model of hippocampal function, not all good models.

Furthermore, there is a large body of evidence that the neural connections within the rodent hippocampus are very plastic, and thus would be capable of supporting rapid learning and long term memory storage there. (One should note, however, that many of the plasticity-inducing protocols are not obviously physiological, and thus may not directly apply to the hippocampus of a freely behaving animal.) The best studied form of plasticity at the cellular level, LTP (Bliss and Collingridge, 1993), has an associative mechanism similar to that hypothesized by Hebb (Hebb, 1949). That is, if two neurons fire together (closely in time), the connection between them is strengthened, such that the next time one of them fires, there is a greater tendency for it to help fire the other one too. Such an associative mechanism is attractive because it seems to describe what is required for fact and event learning: the learning of random associations (e.g. name-face-conversation-room). Especially relevant for sequence learning (i.e. forming associations ordered across time) is the finding that plasticity rules can be temporally asymmetric in the following manner. If the presynaptic neuron fires just (within approximately 20 msec) before the postsynaptic neuron then the pre→post synapse is strengthened, but if the presynaptic neuron fires just (within approximately 20 msec) after the postsynaptic neuron then the pre→post synapse is weakened. This phenomenon is general, having been found in many systems, such as the rat hippocampus (in vivo and in vitro) (Levy and Steward, 1983; Gustafsson et al. 87; Bi and Poo. 98), rat neocortex (in vitro) (Markram et al., 1997), and others (Zhang et al., 1998). With

this learning rule, if the sequence of events A→B→C→... occurs, the particular trace A→B→C→... will be strengthened. Thus there is cellular evidence that the hippocampus could learn sequences of events (i.e. "episodes").

Most importantly, the striking phenomenon of place cells provides a direct link between the electrophysiology of a freely behaving rodent's hippocampus and its presumed spatial function. A "place cell" of the hippocampal CA1 region is a pyramidal cell which fires at a high rate (approximately 5 to 50 Hz) whenever the rat is in a given, restricted part of an environment (e.g. a segment along a linear track, a corner of a rectangular field), and is virtually silent when the rat is elsewhere in that environment (O'Keefe and Dostrovsky, 1971; for a review see Muller, 1996). (While there are place cells in hippocampal region CA3, and while place-like firing has been observed in other regions within and around the hippocampus, such as the entorhinal cortex, dentate gyrus, and subiculum (Muller, 1996), by far the best studied are the CA1 place cells. These are the place cells that we too will study.) The region in which a given place cell fires is called its "place field," and different place cells in the hippocampus generally have place fields in different locations in a given environment. In linear environments (e.g. narrow tracks that are straight or curved), a large fraction of place cells have a strong directional bias. That is, they fire strongly only when the rat runs through that location in a certain direction, and much less or not at all in the opposite direction. In this case the place field is not just a location, but a location and a direction.

Since each environment is tiled by the place fields (many of which are overlapping) of a set of place cells, this forms a "spatial map" of that environment which the rat could presumably use to perform those tasks requiring spatially-guided behavior. Furthermore, place fields, once formed, are generally stable across days and longer (Muller, 1996), thus this map can serve as a basis for spatial memory. For instance, a rat could use such a map to learn the location of the hidden platform in the water maze, use it to navigate there from any starting point, and do it day after day (Blum and Abbott, 1996). What has been learned about the space could be stored in the connections "between" the place cells in the map. (I say "between" because I do not want to imply that we are talking about direct connections between CA1 place cells, which is unlikely given the relatively low probability of connection. A better word would be "relating.") By analyzing the changes in the combined activity of a population of hippocampal place cells from a given map, one could try to infer the changes in connection strengths relating them. This is an

electrophysiological way to study rodent spatial learning and memory, and this in turn may be a good model for human hippocampal-dependent learning and memory.

A Rodent Model of Episodic Memory, i.e. Memory of Sequential Experience

Based on the relationship between the human hippocampus and episodic memory on the one hand, and the rat hippocampus and spatial learning on the other, we propose the following:

The spatiotemporal sequence of hippocampal place fields traversed as a rat goes from one place to another is a good and minimal model for a human episode, i.e. for the temporal order of events experienced by humans.

This sequence of places in time contains three essential elements of episodic memory: spatial context, the temporal dimension, and hippocampal dependence. Furthermore, since each place field corresponds to a place cell, we have a simple neural representation of this episode: the sequential activity of the corresponding place cells. Thus we can look for *memory* traces of this episode electrophysiologically by analyzing the activity of these place cells *after* the episode (i.e. after the rat has run through these place fields). This is what we are going to do. In particular, we will search for the presence of sequential firing patterns from these place cells that *match* the order in which the corresponding place fields had been traversed. (Note that a straightforward matching of order is only one of the possible patterns that might represent a memory trace of this episode.)

The key to this approach is the spatial specificity of place cells. If individual place fields were wide and/or complex in structure (e.g. had lots of peaks and troughs), then the episode could not be represented by a simple sequence of place cells. In that case, the memory patterns we would have to look for would be correspondingly complex, and thus our search would likely be much harder to do.

We should note that while we have chosen to use electrophysiology and place cells to study hippocampal dependent sequence memory in rats, other recent work has employed hippocampal lesions and behavioral odor recognition tasks to do the same (Fortin et al., 2002; Kesner et al., 2002).

17

Electrophysiological Studies of Memory in the Rodent Hippocampus

Now we will review previous work which has used hippocampal place cells in the rodent to study memory. A key aspect of these studies is analysis of hippocampal activity during sleep. It will also become apparent that there is a trend towards higher-order analyses of hippocampal activity, both as recording methods advance, and as researchers attempt to more fully describe the memory-related processing that the hippocampus does.

The attractiveness of place cell electrophysiology as a way to study hippocampal-dependent memory has resulted in several previous studies, only one of which has directly studied the episodic aspect of spatial memory (Louie and Wilson, 2001). While place cell properties (e.g. field size, distribution, and stability over time) have been studied for decades (since 1971, when place cells were first reported (O'Keefe and Dostrovsky, 1971)), the use of them to study memory is more recent (since 1989 (Pavlides and Winson, 1989)). Furthermore, while work on place cell properties generally involves observing hippocampal activity in the awake behaving rat (for obvious reasons), work on place cells and memory has generally also involved observing hippocampal activity during sleep before and after behavior. The reason is that during sleep there is no ongoing overt spatial behavior, thus the hippocampal activity observed then may more clearly reveal the effect of previous spatial behavior (due to reduced "interference" from any current spatial behavior). By comparing the pattern of activity before and after behavior, one can then attempt to infer what changes may have occurred to the neural connectivity due to learning during that behavior. In particular, researchers have tried to directly relate such changes to the activity of the place cells in behavior.

Before continuing, we will first describe more details of hippocampal electrophysiology during wake and sleep. Pyramidal cells in the CA1 pyramidal layer of the hippocampus (where the place cells that all the memory studies deal with come from) display "complex-spiking": bursts of 2-7 action potentials with inter-spike intervals (ISI's) of 1.5-6 msec, often occurring mostly in what is called slow wave sleep (SWS) (Ranck, 1973). Sleep in rodents, following the original classification in humans, can be broadly classified into two types: SWS and rapid eye movement (REM) sleep. During REM sleep, as well as during awake alert behavior (such as running and spatial exploration), the hippocampal EEG is characterized by relatively pure oscillations in the theta band (5-12 Hz). During SWS, as well as during periods when the rat is awake but still (or engaged in eating, drinking, or grooming), the hippocampal EEG is

characterized by large amplitude "irregular" activity exhibiting a broad spectrum of frequencies. A prominent feature of this irregular activity is the occurrence of large EEG spikes of 50-100 msec duration (often called "sharp-waves") which are often accompanied by a 120-250 Hz high frequency "ripple" of similar duration (O'Keefe and Nadel, 1978). These events occur every few seconds on average during SWS (e.g. at a rate of approximately 0.5-1 Hz in our data). During each of these events, the firing rate of a subset of pyramidal cells is highly elevated. Thus a typical EEG spike results in a burst of activity consisting of many different pyramidal cells each firing a complex spike of many action potentials, all occurring within a time window of 100-200 msec. Furthermore, the overall activity of pyramidal cells during SWS is generally very low (average firing rates on the order of 0.1-0.5 Hz), and it appears that most of the activity occurs during such EEG spike events. Therefore, most of the hippocampal place cell memory work we will describe (which mainly deals with SWS) has been based on analysis this type of activity. This is important, since the new method of analysis we develop includes a "parsing" step that was designed especially for dealing with this kind of low-rate, "bursty" neural activity that characterizes SWS in hippocampus.

The following chronological review of previous hippocampal place cell memory work will reveal a clear trend towards higher-order analysis.

Pavlides and Winson (Pavlides and Winson, 1989) showed a first-order (i.e. firing rate) effect of experience on activity in sleep. By letting rats run in only particular parts of a familiar environment, they were able to "drive" the subset of recorded place cells which had place fields there. (From now on, we will call such periods of active awake behavior RUN.) The individual driven place cells showed increased firing rates in the sleep after (POST) the experience (RUN), compared to the rates in the sleep before (PRE) the experience. This was true for SWS and REM separately. There was no such increase for the undriven place cells. They interpreted this result as circumstantial evidence of memory processing of the immediately preceding spatial experience in subsequent SWS and REM.

Wilson and McNaughton (Wilson and McNaughton, 1994) then showed a second-order effect of pairwise correlated activity during experience on subsequent activity in SWS. Rats ran back and forth in various environments (RUN), and pairs of CA1 place cells which had overlapping place fields (and thus were generally co-active in RUN) were compared with those which had non-overlapping place fields (and thus were generally not co-active in RUN). For the

overlapping pairs, the mean zero-time-lag correlation coefficient (binsize 200 msec) during SWS increased greatly from PRE to POST. This was not the case for the non-overlapping pairs. By interpreting the elevated correlations in SWS as revealing an increase in the strength of the intrinsic connectivity relating these co-active cells, this result can be taken as evidence of rapid associative memory formation in the hippocampus (Hebb, 1949). Furthermore, this study demonstrates the importance of higher-order analysis of population activity in trying to understand hippocampal function. The first-order analysis of Pavlides and Winson was unable to detect this second-order effect. All the place cells here (whether overlapping or not) would have been predicted to have increased firing rates. It is important to keep in mind that this second order study did not address the temporal *order* aspect of the spatial experience in RUN. It only showed that the structure of generally *coincident* activity in RUN is preserved in POST. (A later study (Kudrimoti et al., 1999) showed basically the same effect using a more complex analysis method involving the correlation of these same zero-time-lag correlation coefficients across pairs of time intervals.)

In contrast, Skaggs and McNaughton (Skaggs and McNaughton, 1996) explicitly showed a second-order temporal order effect of experience. In particular, they showed that the temporal bias of pairs of CA1 pyramidal cells (i.e. the mean tendency of the spikes of one cell to fire before or after the spikes of another cell) in RUN was preserved in POST SWS. They interpreted this result as evidence of the encoding of temporal order from experience by the hippocampus. However, their data also revealed some preservation of temporal bias between PRE SWS and RUN. This raises the possibility that the temporal ordering was not learned during RUN, but instead already existed before RUN (perhaps reflecting pre-existing biases in intrinsic connectivity). In addition, the magnitude of the POST effect found was small (on the order of 10% above the null hypothesis), and significantly smaller than the second-order non-temporal effect of Wilson and McNaughton above (which was on the order of 2-3 times the null hypothesis).

Nádasdy et al. (Nádasdy et al., 1999) did a third-order analysis of temporal firing patterns of CA1 hippocampal pyramidal cells. They searched all possible ordered spike triplets (three different cells firing in a fixed order separated by a specific pair of time delays), determined which triplets occurred significantly more than expected (with respect to the observed pairwise correlations) using the JPSTH method (Palm et al., 1988; Aertsen et al., 1989) (described in the section on higher-order analysis methods), then compared the resulting significant triplets across

PRE SWS, RUN, and POST SWS. They found many significant ordered triplets in each of PRE, RUN, and POST. Furthermore, they claimed that, with respect to these significant triplets, there was more in common between POST and RUN than between PRE and RUN, and interpreted it as evidence that ordered triplet sequences from RUN experience were learned then replayed in POST (and replayed at a faster timescale than the sequences in RUN). There are two caveats. First, unlike in the above experiments, here the rat's behavior in RUN was confined to a running wheel. Thus it is unclear what the relationship between the neural activity in RUN and the rat's experience was, since the pyramidal cells were no longer firing in their natural (place cell) mode. Second, their claim that RUN triplet activity resembles POST more than PRE is debatable, since alternative measures could support the opposite conclusion. In either case, the high degree of triplet similarity between PRE and RUN makes it hard to isolate the contribution of temporal experience in RUN itself to POST. Overall, this study demonstrated that there is a high degree of conserved ordered triplet activity across all of PRE, RUN, and POST. This may reflect pre-existing underlying circuitry that reveals itself at all times, rather than specific learning of sequences in RUN.

Recently, Louie and Wilson (Louie and Wilson, 2001) performed an "Nth-order" analysis (i.e. analysis of patterns involving an arbitrary number of neurons—in their study 8-13 neurons) of CA1 place cell activity in RUN and REM sleep. This is the only study that has addressed the episodic nature of spatial memory. They took the RUN activity of N place cells over a period of approximately 1 minute and created a template (i.e. the average firing rates of each cell binned every few seconds) of this episodic experience. There was significant overlap (essentially a correlation coefficient between two N-by-m matrices, where m = the number of time bins) between this template and the activity of these N place cells in REM sleep intervals, indicating the replaying of the temporal structure of the episodic experience from RUN in REM. The best overlap indicated that the activity in REM was approximately 1.4 times slower than in RUN. These results depended on detecting broad matching between two noisy spatiotemporal patterns (RUN and REM), thus most probably could not have been discovered using lower-order analysis methods. (Note that while there was significant overlap in PRE REM, control experiments showed that this represented learning from RUN on previous days. Such controls were not done in the above studies in which there was RUN structure in PRE SWS.)

Thus there is behavioral electrophysiology evidence for memory traces in the rodent hippocampus. The existence of such memory traces during sleep demonstrates that experience

from RUN was indeed encoded in some manner. But many researchers have further interpreted the recurring traces in sleep as possible evidence of memory consolidation in action (Pavlides and Winson, 1989; Wilson and McNaughton, 1994; Kudrimoti et al., 1999; Nádasdy et al., 1999; Louie and Wilson, 2001). The idea is that the hippocampus replays memory traces during sleep in order to transfer them to locations outside the hippocampus for permanent storage. Such a process would explain the preservation of previously learned episodic memories in patients with hippocampal damage. While this theory is attractive, it does not have to be true in this exact form, or even true in any form, for the hippocampal memory work to be of value. As stated above, the mere existence of detectable memory traces is important, because it demonstrates that at least those aspects of the experience in RUN have been, and thus can be, encoded. To the extent that the above work also shows evidence of RUN-related structure before RUN (Skaggs and McNaughton, 1996; Nádasdy et al., 1999), the evidence for encoding is weaker.

An alternative approach to analyzing sleep for evidence of spatial learning is analysis of changes in place cell properties during RUN itself. Mehta et al. (Mehta et al., 1997; Mehta et al., 2000) demonstrated systematic experience-dependent changes in the size, location, and shape of individual place fields. The results were explained by a model in which the repeated firing of place-specific CA3 inputs to a CA1 place cell led to temporally asymmetric LTP and LTD at these synapses. These changes were reset each day and recurred in the same RUN environment on the next day, suggesting that the hippocampus is involved in short term (less than one day) storage.

To summarize, the work of Pavlides and Winson was first-order spatial, Wilson and McNaughton second-order spatial, Skaggs and McNaughton second-order temporal, Nádasdy et al. third-order temporal, and Louie and Wilson Nth-order temporal. The work of Mehta et al. was first-order in terms of analyzing single cells, but higher-order in terms of analyzing higher-order properties of these place cells (e.g. skewness of field shape). The trend towards higher-order analyses is clear. In addition, there is a trend towards analyzing temporal firing patterns. While most of the work has focused on activity in SWS, the only direct evidence for episodic memory was in REM sleep (Louie and Wilson, 2001). Neither the temporal bias measure of Skaggs and McNaughton nor the triplet spike sequences of Nádasdy et al. seem to capture the episodic aspect of spatial experience. Our work will involve developing and using a novel Nth-order analysis method to search for direct evidence of episodic memory in SWS by analyzing sequential firing patterns of arbitrary length.

Higher-Order Analyses of Spike Trains

We have seen the necessity of higher-order analyses for understanding complex yet important processing done by populations of neurons. Recent (within the last 10-15 years) advances in technology have allowed the recording of large numbers of (on the order of 10 to 100) individual neurons simultaneously in various experimental systems (Meister et al., 1991; Abeles et al., 1993; Wilson and McNaughton, 1993; Wilson and McNaughton, 1994; Skaggs and McNaughton, 1996; Kudrimoti et al., 1999; Nádasdy et al., 1999; Wessberg et al., 2000; Louie and Wilson, 2001). Therefore the once-rare data needed to do higher-order analysis is becoming plentiful. In fact, it appears that there is now an explosion of neural population data, and a shortage of analysis methods to extract full value from them. (A similar explosion of data is occurring in many fields in science and engineering, such as genomics, as well as in economics, business, and other areas. This has led to the rising popularity of "data mining" techniques.) Thus a key problem for systems neuroscience, and one which will only grow more important in the future, is the development of good methods to analyze all of this neural data.

Here we will review the literature on the higher-order analysis of neural spike train data. While there are many general statistical methods for analyzing multivariable data, how to meaningfully apply them to spike train data is not obvious. In fact, there is no general statistical technique for analyzing the detailed relationship between N simultaneous discrete stochastic processes, which is the basic problem of analyzing the spike trains of N simultaneously recorded cells. We will thus only discuss the limited literature concerning higher-order methods specifically designed to analyze multiple spike trains.

The most popular method is the cross correlation. It is used to detect second-order interactions, that is, covariations in firing rate between two spike trains. This method is well-studied. But methods beyond second-order are few and generally not as well-studied. These are the methods we will focus on.

Perhaps the most rigorous method beyond second-order is the "normalized" Joint Peristimulus Time Histogram (JPSTH) (Palm et al., 1988; Aertsen et al., 1989). This third-order method can be used to determine whether a given ordered triplet of spikes 1-2-3 (with one spike coming from each of three spike trains: 1, 2, 3) with a given pair of time delays (i.e. inter-spike

intervals) between them occurs significantly more than expected based on the observed second-order cross correlations between trains 1 and 2, and between trains 1 and 3. Here we use the term "normalized" to refer to the use of observed lower-order interactions (here the observed second-order correlations) to isolate higher-order (here third-order) interactions that cannot be reduced to (i.e. predicted based on) those lower-order interactions. This method has been used in a number of studies, including that of Nádasdy et al. (Nádasdy et al., 1999) discussed above. Though mathematically rigorous in setting significance levels, unfortunately this method is not easily extended to more than three spike trains, and thus this has not been done. Gerstein and Perkel, who had originally suggested the normalized JPSTH, proposed the extension of the un-normalized JPSTH to N spike trains (Gerstein and Perkel, 1969). This method has not been used much, if at all, perhaps due to the difficulty of manipulating N-1 dimensional scatterplots. A suggested approach to normalizing Nth-order interactions restricted to coincident firing has yet to be completed (Martignon et al., 1995).

An early Nth-order method involved searching for repeating patterns of N adjacent inter-spike intervals within a single spike train (Dayhoff and Gerstein, 1983). Significance was determined via shuffling the spike train. Perhaps because it deals with patterns within only a single spike train, this method has been used in few studies.

A well-known Nth-order method developed by Abeles and colleagues (Abeles and Gerstein, 1988; Abeles, 1991; Abeles et al., 1993) involves searching for repeating temporal firing patterns consisting of N spikes from m distinct neurons (m $\leq$ N) with N-1 exact time delays (within a small jitter) between them. They developed an elegant algorithm that allows one to detect all such patterns of a given maximum duration that repeat at least once (i.e. at least two total occurrences). Significance of the number of distinct patterns of N spikes that repeat r times is determined via what they call the "ad hoc" method, which is based on an assumption of a uniform distribution of all possible inter-spike intervals (Abeles and Gerstein, 1988; Abeles et al., 1993). This approach is of considerable theoretical interest. Abeles developed a neural network-based theory which predicts the existence of such precisely repeating patterns consisting of multiple neurons firing with exact (within approximately 1 msec jitter) time delays. Specifically, Abeles argued that for neural activity to stably propagate across many successive layers of a network, the activity must result in firing patterns with precise (within 1 msec) time delays (Abeles, 1991). Application of this method to experimental data has generally resulted in the conclusion that there are many precise firing patterns which repeat many more times than

expected (based on their assumptions), but what these firing patterns might represent remains unknown (Abeles et al., 1993).

A completely different Nth-order method is based on the technique of gravitational clustering (Gerstein et al., 1985; Gerstein and Aertsen, 1985). Here each neuron is treated as a particle. Each spike fired by a neuron results in giving a "charge" to its particle that decays in time. Particles with charges attract each other. At the end of a period of activity, the group of N originally equidistant (in N-dimensional space) neurons should cluster into subgroups of neurons that have generally coincident firing. The advantage of this method over cross correlation is that subgroups of clustering neurons are not limited to just pairs of neurons. The advantage of this method over that of Abeles et al. is that the timing between neurons does not have to be nearly as precise. More complicated charge and attraction rules can be used to detect sequential, as opposed to just roughly coincident, firing among a group of neurons (Gerstein and Aertsen, 1985). While this method has the feature of being relatively free of assumptions (though several parameters such as for charge and dynamics must be set), a disadvantage is that there is no quantification of the significance of any clustering that is found.

Recently, as described in the preceding section, Louie and Wilson developed an Nth-order method based on matching the activity of N cells over a given period to a particular target template (Louie and Wilson, 2001). This method detects broad matching of modulations in firing rates across N cells. It is not so sensitive to extra or missing spikes, or to the exact timing of spikes, though the matching pattern must have the same overall structure of time intervals between the different parts of the activity as in the template. Overall timescale differences between the test activity and target template (e.g. 2 times faster, or 3 times slower) can be detected by trying different scaled versions of the template. In such cases the matching pattern must have the same overall *proportions* in its time interval structure. Significance of matching has been determined by comparing the matching actually found to the distribution of matching scores with respect to various shuffled versions of the template.

To summarize, both the third-order normalized JPSTH and Nth-order Abeles et al. methods deal with firing patterns that contain precise inter-spike intervals, while the Nth-order gravitational method is not so restricted. The Nth-order template method allows for deviations from precise time intervals, though matching patterns must have the same overall time interval structure. The JPSTH method has a rigorous quantification of significance, that of Abeles et al.

25

as well as the template method have reasonable quantifications of significance, and the gravitational method has none. All these methods detect sequential firing patterns (in the case of the gravitational method this requires using specific rules). While the JPSTH and Abeles et al. methods can be used to evaluate the significance of a particular temporal firing pattern (a particular sequence of neurons firing with particular time delays), both of these methods have been generally been used to search all possible patterns (up to a maximum duration) for ones which recur significantly more than expected (Abeles et al. 1993; Nádasdy et al., 1999). In particular, the efficient pattern searching algorithm of Abeles et al. (Abeles and Gerstein, 1988) was designed to search for all such patterns that repeat. In contrast, the gravitational method cannot quantify the significance of any subset of firing patterns, but can only search all possible patterns for candidates that may or may not be "significant."

Thus one key difference between these methods (especially that of Abeles et al. and gravitational clustering) and the one that we will develop is this: we will purposely single out a particular relevant pattern (i.e. the episodic place field sequence in RUN), then test to see if there are significant traces of it in a given period (i.e. POST SWS). The template method also singles out a particular relevant target pattern: the template itself. The other methods generally function as data mining tools used to search all possible patterns (of a certain type) for candidate patterns that recur significantly more than "expected." The functional significance of the usually large set of resulting candidate patterns is left for further analysis. Another key difference between our method and that of Abeles et al. is that ours will deal with the *relative* firing order of a group of neurons, while Abeles et al. is concerned with precise patterns with exact inter-spike intervals. The template method also basically deals with exact time intervals, although much more jitter is allowed. Other differences will be described in Chapter 3, where our method is explained in detail.

Specific Aims of this Thesis

There are two main aims of our research: (1) Develop a new method to analyze spike train data from large numbers of simultaneously recorded individual neurons, in particular to detect and quantify meaningful temporal firing patterns such as sequences, and (2) Apply this method to search for and quantify sequential firing patterns in the rat hippocampus during SWS as electrophysiological evidence of sequential memory encoding. To the extent that the hippocampal work is successful, the method we develop will be validated. In addition, we hope

to make a convincing case for using the sequence of place fields traversed as a good animal (rodent) model of human hippocampal-dependent episodic memory.

Experimental Design and Analysis Method

We design the following simple experiment to generate the place field sequence. An adult male Long Evans rat is trained to run repeatedly back and forth (for 25 or more round trips within 30 minutes) on linear tracks of various shapes for chocolate reward at each end. Before (PRE) and after (POST) running on a particular track (RUN), the rat is placed in a small, comfortable enclosure located away from the track, during which time the rat sleeps (for a total of at least 15 minutes in each of PRE and POST). During all three periods (PRE, RUN, and POST) we continuously record the rat's position and head direction, the spiking activity of large numbers of individual pyramidal cells from the dorsal CA1 region of the hippocampus, and the local field potentials (local EEG) around these cells. This is accomplished using multiple tetrode extracellular recording technology that allows one to simultaneously record large numbers of individual neurons in freely behaving rodents (Wilson and McNaughton, 1993). From the subset of cells which have reliable place fields in RUN, we determine the sequence of place fields repeatedly traversed by the rat. There will be two such sequences (one for each direction along the track). These represent the rat's sequential experience during RUN.

We will then search for traces of these RUN sequences in both PRE and POST SWS. (We look for traces in sleep for the reasons explained above.) In particular, we will search for firing patterns in which the place cells fire in the same order as their place fields were traversed in RUN. A strong presence of RUN sequences in POST but not PRE would suggest that these sequences were encoded during RUN.

In order to search for such sequential firing patterns, our analysis approach is the following. We first parse the population activity of hippocampal CA1 pyramidal cells in a manner suited to the characteristic activity of these cells during SWS. This activity consists of short, intermittent bursts of spikes from several cells (which generally co-occur with the sharp-wave/ripple EEG events), with adjacent bursts separated by larger gaps with little firing. Thus we aim to parse this activity into the individual population bursts. Then in order to test whether the firing within each population burst matches a given RUN sequence, we consider only the *relative* order of firing within each burst and compare it to the *relative* order of place fields in the RUN

27

sequence. By considering only relative order, we allow ourselves to detect sequence matching without regard to specific inter-spike intervals, and we greatly reduce the sampling problem associated with searching for patterns consisting of precise inter-spike intervals. Our method of quantifying sequence matching using only relative order is based on combinatorics and probability. It is described fully in Chapter 3. Our experiment and the application of this method are described fully in Chapter 4. However, this method should be applicable to many other systems—in fact to any system in which the precise order of occurence of a large set of events (not just limited to spikes) is to be studied. Examples include trial-by-trial data of the effect of visual stimuli or motor movements on the activity of a population of responsive neurons or on the occurrence of EEG events from multiple locations.

We conclude this chapter with a short discussion concerning a few aspects of our experiment.

An implicit assumption of our experiment is that the spatial map exists prior to the sequential experience in RUN. That is, we assume that the locations of the place fields are already basically fixed, and that the rat may then learn a particular sequence of those place fields. It is the difference between forming the spatial map and learning with respect to that map. However, in reality these processes may be related. Whatever the reality is, empirically we observe that the locations of the place fields are largely stable during RUN (Figure 1.1). (The backwards shift of place fields observed by Mehta et al. (Mehta et al., 1997) does not appear to alter the average relative order of place fields traversed.) If it turns out that these processes are highly related, what we learn about the sequence encoding may help us understand spatial map formation.

A key part of our experiment is the behavioral repetition (i.e. 25 or more laps in each direction) which leads to repetition of each place field sequence. The idea is to expose the rat to essentially the same sequence many times, drive the hippocampus repeatedly, and thus increase the chances of detecting memory traces of this sequential experience. On the other hand, episodic memory in humans generally refers to memory of a single experience (e.g. a walk through the park on a particular day). However, it would probably be much harder to experimentally detect traces of a place field sequence that a rat experienced only once. It is presumably an issue of signal (i.e. the sequence) to "noise" (i.e. all other hippocampal activity—which could be random, related to other memories, or perhaps related to the sequence in more complex ways). Therefore,

while our experiment is inspired by human episodic memory, a more mechanistic way to view it is this: We are asking what kinds of complex sequential patterns can be learned by this machine called the hippocampus.

Summary

The hippocampus is crucially involved in the formation of long term episodic memories in humans and spatial memories in rodents. The sequence of place fields traversed by a rodent may be a good animal model for an episodic memory. In Chapter 3, we will develop a novel Nth-order analysis method to detect and quantify matching between a given sequence and temporal firing patterns involving an arbitrary number of neurons. In Chapter 4, we will use this method to test for direct electrophysiological evidence of sequential memory encoding in the rodent hippocampus. In particular, we will analyze the activity of a population of CA1 pyramidal cells during SWS before and after the rat has repeatedly run through a sequence of the place fields of those cells.

CHAPTER 3

A COMBINATORIAL METHOD FOR ANALYZING SEQUENTIAL FIRING PATTERNS INVOLVING AN ARBITRARY NUMBER OF NEURONS BASED ON RELATIVE TIME ORDER

Abstract

Finding meaningful patterns hidden within large amounts of data is now a key problem in many areas of science. In neuroscience, it is believed that information processing in the brain requires coordinated activity across many neurons. However, the code by which multiple neurons interact to perform their computations is largely unknown, and remains one of the fundamental problems in the field. With the recent development of methods to simultaneously record the spiking activity of large numbers of individual neurons, the search for complex firing patterns across many cells that could directly address this issue has become possible. But few studies exist describing ways to analyze firing patterns involving more than two simultaneous spike trains. Here we develop a new approach for analyzing sequential firing patterns involving an arbitrary number of neurons based on relative firing order. Specifically, we develop a combinatorial method for quantifying the degree of matching between a "reference sequence" of N distinct "letters" (representing a particular target order of firing by N cells) and an arbitrarily long "word" composed of those letters (representing the relative time order of spikes in an arbitrary firing pattern). The method involves computing the probability that a word would by chance contain a match to the reference sequence that is as good as or better than the best match (to that sequence) actually found in that word, assuming all permutations of the word's letters were equally likely. Lower probabilities thus indicate better matching. For very complex words, we develop formulae for computing upper and lower bounds of this probability. Using the probabilities from across a heterogeneous set of words (or even just a single word), the overall degree and statistical significance of matching can be computed without the use of Monte Carlo techniques. By using relative order, we can reduce the sampling problem associated with analyzing patterns based on precise inter-spike intervals (i.e. exact time delays between the constituent spikes). Words may contain repeated letters and need not contain each letter in the reference sequence. Thus this method can sample whether traces of a particular sequence exist in words that do not necessarily contain every letter, i.e. it naturally handles missing spikes. Variations of our approach emphasizing different aspects of matching are described, as well as variations that modify the equally likely permutation assumption. Our approach is general and can be applied to many

different types of neural data beyond multiple spike trains, such as EEG events from multiple locations, or signals from different regions in imaging data. We have recently applied this method to quantify memory traces of sequential experience in the rodent hippocampus during subsequent slow wave sleep.

Introduction

Finding meaningful patterns hidden within large amounts of data is now a key problem in many areas of science. In neuroscience, it is believed that information processing in the brain requires coordinated activity across many neurons. However, the code by which multiple neurons interact to perform their computations is largely unknown, and remains one of the fundamental problems in the field. With the recent development of methods to simultaneously record the spiking activity of large numbers of individual neurons, the search for complex firing patterns across many cells that could directly address this issue has become possible (Meister et al., 1991; Abeles et al., 1993; Wilson and McNaughton, 1993; Wilson and McNaughton, 1994; Wessberg et al., 2000; Louie and Wilson, 2001; Lee and Wilson, 2002).

To fully exploit these experimental advances, there is a corresponding need for new methods to analyze complex firing patterns across multiple simultaneous spike trains, i.e. N simultaneous continuous-time discrete stochastic processes. In practice, this may consist of 10-100 simultaneous spike trains (1 train per neuron) each sampled every 0.1 msec, with the sample value "1" (a spike) occurring in each train with average frequency 0.1-10 Hz (the remaining sample values being "0"), and recorded for several hours. The cross correlation function is a widely used method for detecting interactions between a pair of spike trains. A central peak indicates a tendency for coincident spikes, while an off-center peak indicates a tendency for one cell to fire after the other with a particular delay. But only a few studies exist which describe ways to analyze interactions among the spike trains of more than two neurons (Gerstein and Perkel, 1969; Gerstein et al., 1985; Gerstein and Aertsen, 1985; Abeles and Gerstein, 1988; Palm et al., 1988; Aertsen et al., 1989; Abeles, 1991; Abeles et al., 1993; Martignon et al., 1995, Louie and Wilson, 2001; Lee and Wilson, 2002). Furthermore, most of these methods analyze firing patterns in terms of a set of precise inter-spike intervals, i.e. exact time delays between the successive spikes.

31

Here we present our contribution to this effort. We develop a new approach to quantifying the degree and statistical significance of sequential firing patterns involving an arbitrary number of neurons based on relative firing order. Sequences are very important for neural processing. Examples include memories of a sequence of experienced events, a sequence of actions in an overall movement, or sequential recruitment of different brain areas during a task. Relative time order of (as opposed to exact time intervals between) events within a sequence is important for dealing with possibilities such as time scaling, noise, and order-sensitive plasticity rules (Levy and Steward, 1983; Gustafsson et al., 1987; Markram and Sakmann, 1997; Bi and Poo, 1998). It can also reduce the sampling problem associated with analyzing patterns based on exact time intervals. We have recently applied our method to quantify memory traces of sequential experience in the rodent hippocampus during subsequent slow wave sleep (SWS) (Lee and Wilson, 2002). We found recurring bursts of activity involving up to 6 neurons firing in an order that is directly related to the rat's previous spatial experience.

Though we have initially motivated our method in terms of single spikes from individual neurons, our approach is general and can be applied to other types of neural data, such as multi-spike bursts from multiple cells, EEG events from multiple locations, or suprathreshold signals across multiple pixels of an optical or fMRI image. That is, what we will often refer to as a single spike from a single cell could just as well refer to a burst of spikes from a single cell, a particular pattern from a subset of cells, an EEG event from a given location, or any other type of signal. Therefore we will describe our approach in abstract terms. Each type of event (e.g. a spike) will be captured in the abstract term "letter", and a sequence of such events from multiple sources will be represented by a "word", i.e. a string of letters (with different sources represented by different letters).

The Basic Problem

Consider the simultaneous spike trains from N neurons over a given time period. How could one detect and quantify meaningful firing patterns that may occur in this data? One type of meaningful pattern is a sequence of spikes, each one from a different cell. We would like to test whether an arbitrary pattern of spikes from multiple cells contains significant matching to a particular sequence of this type.

First, we cast the problem in its most general form. Since (as mentioned above) what we call a single spike from a single cell could represent any of a number of different types of events, we denote each event from a particular source (e.g. a spike fired by cell 1) by a "letter" (e.g. "1"). Since we are concerned with the relative time order of events from multiple sources, the timing of events can be reduced to an ordered string of letters (with different letters representing different sources). Then the basic problem we address can be stated as follows. We start by defining a "reference sequence" as an ordered list of an arbitrary number (N) of distinct (i.e. non-repeating) "letters." Then we ask: how can one quantify the degree of matching between a particular reference sequence and an arbitrary string consisting of only those letters? This arbitrary string, which we call a "word", may contain repeated letters (e.g. a cell may fire more than one spike) and also need not contain all the letters in the reference sequence (e.g. a particular cell may not fire a spike at all), but every letter in a word must appear in the reference sequence. For example, the word 4271 could represent an event in which cells 4, 2, 7, and 1 each fired a spike in that order.

Since the letters are just labels, and since we restrict ourselves to reference sequences containing only one event from each source, a reference sequence of length N can be represented as 1234...N. We will assume this form for all reference sequences. An example of an invalid reference sequence is 1234156789, since no repeats are allowed.

Our Basic Approach

Our approach to the above sequence matching problem is based on combinatorics and probability. First we describe the idea with simple cases. For instance, given a 2-letter word in which both letters are distinct (i.e. the 2 letters are different), there are 2 possible ways in which the letters could be ordered, and thus a 1/2 chance that the letters are in the same order as in the reference sequence, assuming each ordering is equally likely. Given a 3-letter word in which all 3 letters are distinct, there are $3! = 6$ ways in which the letters could be ordered, and thus a 1/6 chance that the letters are in the same order as in the reference sequence, assuming each of the possible orderings are equally likely.

Now we extend this process to words of arbitrary length and letter composition. Because longer words could contain imperfect matches to the reference sequence that are still highly improbable based on chance, we will include imperfect matches in our analysis. For instance,

while it is highly unlikely to find the word 123456789 by chance, it is also highly unlikely to find a word such as 124563789 by chance. Just as in the simpler cases, chance is defined as the probability of getting a match as good as the best one actually found in that word or better, assuming all n! permutations of that length n word are equally likely. The lower this probability, the more unlikely, and thus the higher the degree of matching between the word and the reference sequence. Note that the probability is conditional on the specific letters that were actually observed in the word. Later we will show that this is a reasonable approach.

The reason for computing the probability that a word would contain a match as good as *or better than* the best match found in that word, assuming all possible permutations of its constituent letters were equally likely, is that we are now dealing with imperfect matches. It is exactly analogous to the statistical procedure of determining a p value, such as in computing the probability that one would observe 80 heads in 100 tosses of a coin, assuming the coin is fair. To evaluate how unlikely this is by chance, one computes the probability of getting 80 *or more* heads, assuming the null hypothesis of a fair coin. Here "80 heads" corresponds to the best match found in a word, "more than 80 heads" corresponds to the better matches, and "fair coin" corresponds to the permutations being equally likely.

Thus we need (1) a precise definition of matching that covers perfect as well as various imperfect matches, and (2) a way to order matches from best to worst so that the phrases "best match found" and "better match" make sense.

Match Definition

A word W (which may have repeated letters) contains a (x,y) match to a given reference sequence S (which cannot have repeated letters) if there exist x+y consecutive letters in W of which at least x letters are in *strictly* increasing order in S (but not necessarily consecutive in S). According to this definition, if S = 123456789, the word 11377 does not contain a (5,0) match, but does contain a (3,0) match (as well as (3,1), (3,2), (2,0), and other matches), and the word 13436892 contains a (6,1) match (as well as (6,2), (5,1), (5,2), (5,3), (4,1), and other matches). Essentially, our notation means that x is the minimum number of letters in order, and y is the maximum number of interruptions in between those matching letters.

Ordered Match Lists and the Rationale for Different Lists

Given a word W of length n and containing k distinct letters (n ≥ k), the best possible match to the reference sequence S by optimally rearranging the letters of W is a (k,0) match, i.e. k consecutive letters in W in the same order as in S and with no interruptions. In our approach, in the case that n > k, the ordering of the n-k remaining letters does not matter. (Likewise, given any (x,y) match, the order of the n-x remaining letters does not matter.) Thus there could be several different permutations of W containing the best possible match. For instance, if S = 123456789 and W = 42717, both 12477 and 71247 contain the best possible match. One reason we are not concerned about the remaining n-k letters is that, as discussed later, we presume (but do not require) a pre-processing (i.e. "parsing") stage in which the letters and words are extracted from the raw data such that n is usually not much greater than k.

Now how do we order the other possible matches? Given a word W of length n and containing k distinct letters, the list of all possible matches is shown in Figure 3.1.

Figure 3.1

(k,0)	(k,1)	(k,2)	....	(k,n-k)			
(k-1,0)	(k-1,1)	(k-1,2)	....	....	(k-1,n-k+1)		
(k-2,0)	(k-2,1)	(k-2,2)	....	....	....	(k-2,n-k+2)	
....	....	....	....	....	....	....	....
(1,0)	....	....	....	....	....	....	.... (1,n-1)

To begin with, we will not consider matches (1,y) for any y or (2,y) for y > 0. This is because every word contains a (1,y) match, and any permutation containing a (2,y) match with y > 0 also contains a (2,0) match.

In terms of ordering matches, it makes intuitive sense that (I) (x,0) > (x,1) > (x,2) > ... for any x, and also that (II) (x,y) > (x-1,y) > (x-2,y) > ... for any x and y, where ">" means "better than". For instance, if x = 6, (I) says a (6,0) match is better than a (6,1) match, i.e. getting 6 in a row in order is better than having 6 in order with one interruption. If x = 6 and y = 0, (II) says a (6,0) match is better than a (5,0) match, i.e. getting 6 in a row in order is better than having 5 in a row in order. Thus (I) and (II) are constraints that any ordering of matches should obey.

But these constraints do not address how to order (x_1, y_1) and (x_2, y_2) in the case that $x_1 > x_2$ but $y_1 > y_2$. This is partially addressed by the following. The intuition behind constraints (I) and (II) can be stated as the single constraint (*): the ordering of matches must be such that a word which contains a worse match must not automatically also contain a better match. For instance, every word which contains a (x,y) match automatically contains a $(x-1,y)$ match, thus a (x,y) match cannot be considered worse than a $(x-1,y)$ match. But (*) also includes cases not covered by (I) and (II). For instance, according to (*), $(6,1) > (4,0)$, because the 1 interruption in the $(6,1)$ match will always still leave a $(4,0)$ match somewhere. If the interruption is placed in the middle, say in 123Z456, no matter what Z is, either 123Z or Z456 will be a $(4,0)$ match. Similarly, $(6,3) > (4,1)$.

Still, these constraints do not force a unique ordering of all the possible matches listed above. That is, several different orderings obey the intuitive constraint (*). Two such orderings are:

(1) The "Horizontal (H) Ordering": Given any 2 possible matches (where an (x,y) match is possible if and only if $x \leq k$ and $x+y \leq n$), assign $(x_1, y_1) > (x_2, y_2)$ if either (a) $x_1 > x_2$, or (b) $x_1 = x_2$ and $y_1 < y_2$. That is, the better matches are the ones with more letters in strictly increasing order regardless of interruptions, or the ones with the least interruptions given an equal number of letters in strictly increasing order. Thus the best possible match is the $(k,0)$ match, followed by $(k,1)$, then $(k,2)$, and so on until $(k,n-k)$, then $(k-1,0)$, followed by $(k-1,1)$, $(k-1,2)$, et cetera, then ending with $(2,0)$. This is called "Horizontal" because matches are ordered following left to right horizontal lines, starting with the top row, in Figure 3.1.

(2) The "Diagonal (D) Ordering": Given any 2 possible matches, assign $(x_1, y_1) > (x_2, y_2)$ if either (a) $x_1 - y_1 > x_2 - y_2$, or (b) $x_1 - y_1 = x_2 - y_2$ and $x_1 > x_2$. Furthermore, only consider matches (x,y) such that $x-y \geq 2$. That is, the better matches are the ones with the greater difference between the number of letters in strictly increasing order and the number of interruptions, or the ones with the greater number of letters in strictly increasing order if the differences are equal. Thus the best possible match is the $(k,0)$ match, followed by (x,y) such that $x-y = k-1$, ordered by decreasing x, i.e. $(k,1)$ (assuming $k+1 \leq n$, if not, skip it) then $(k-1,0)$. Then next are (x,y) such that $x-y = k-2$, i.e. $(k,2)$ (assuming $k+2 \leq n$, if not, skip it), then $(k-1,1)$, then $(k-2,0)$. Continue this as long as $x-y \geq 2$. Thus the last

(worst) match we consider is (2,0). This is called "Diagonal" because matches are ordered following upper-right to lower-left diagonals, starting with the leftmost such diagonal (i.e. that containing just (k,0)), in Figure 3.1.

We prove that both these orderings obey the constraint (*) in Appendix A1.

To help us evaluate the above ordering schemes, as well as other possible schemes, consider the probabilities of getting each type of match given different values for n (word length) and k (number of distinct letters). The probability for a particular match is computed as the fraction of the n! permutations of the letters which contain that match. Our intuition is that less probable matches should be considered better. We consider a few cases to illustrate some key points (Table 3.1).

Table 3.1

(A) For n = 7, k = 7, the probabilities for the following matches:

(7,0)

(6,0)   (6,1)

(5,0)   (5,1)   (5,2)

(4,0)   (4,1)   (4,2)   (4,3)

(3,0)   (3,1)   (3,2)   (3,3)   (3,4)

(2,0)

are:

0.00020

0.0026 0.0073

0.0222 0.0591 0.0909

0.1417 0.2942 0.4016 0.4522

0.5998 0.8248 0.8921 0.9119 0.9149

0.99980

Table 3.1 (continued)

(B) For n = 8, k = 8, the probabilities for the following matches:

(8,0)

(7,0) (7,1)

(6,0) (6,1) (6,2)

(5,0) (5,1) (5,2) (5,3)

(4,0) (4,1) (4,2) (4,3) (4,4)

(3,0) (3,1) (3,2) (3,3) (3,4) (3,5)

(2,0)

are:

0.000025

0.00037 0.0012

0.0038 0.0124 0.0221

0.0292 0.0811 0.1360 0.1735

0.1733 0.3641 0.4983 0.5776 0.6090

0.6687 0.8856 0.9421 0.9598 0.9640 0.9645

0.999975

(C) For n = 9, k = 9, the probabilities for the following matches:

(9,0)

(8,0) (8,1)

(7,0) (7,1) (7,2)

(6,0) (6,1) (6,2) (6,3)

(5,0) (5,1) (5,2) (5,3) (5,4)

(4,0) (4,1) (4,2) (4,3) (4,4) (4,5)

(3,0) (3,1) (3,2) (3,3) (3,4) (3,5) (3,6)

(2,0)

are:

0.0000028

0.000047 0.00018

0.00055 0.0021 0.0043

0.0050 0.0174 0.0347 0.0497

0.0361 0.1030 0.1776 0.2411 0.2785

0.2039 0.4285 0.5828 0.6751 0.7234 0.7400

0.7261 0.9250 0.9698 0.9822 0.9858 0.9865 0.9866

0.9999972

We see that (at least in these few cases) both the H and D ordering schemes roughly order matches by increasing probability (i.e. better matches having lower probability), though this breaks down somewhat for the D ordering scheme for worse matches. We can also see examples of the potential shortcomings of ordering based strictly on probability. For instance, in the $n = 9$, $k = 9$ case, $P(7,2) = 0.0043 < 0.0050 = P(6,0)$. Even though $(7,2)$ is less probable than $(6,0)$, intuitively one might consider $(6,0)$ to be a better match since the $(7,2)$ match has one additional letter in order at the price of 2 more interruptions. There are also shortcomings with the H ordering scheme (which also places $(7,2)$ as better than $(6,0)$). For an extreme example of this shortcoming, if $n = 100$ and $k = 100$, the H ordering scheme places $(51,20)$ as better than $(50,0)$. On the other hand, the D ordering scheme balances rewarding larger x with penalizing larger y (where x and y refer to matches (x,y)). We have used primarily the D ordering scheme in our previous work, but we also showed that the H ordering gave similar results (Lee and Wilson, 2002). The H ordering scheme is not without merit. For instance, in the $n = 100$, $k = 100$ case, one might want to rank $(60,30)$ as better than $(31,0)$. It is a matter of which aspects of matching one wants to emphasize.

There is, however, a major difficulty in using ordering based on strict probability. Since the exact probabilities of matches are sensitive to the length and specific letter composition of each word, one would have to compute the exact probability of getting each type match from scratch for each word encountered. This would be needed for determining the ordering of matches so that the best match contained in the word could be identified. For even moderately long words (e.g. $n > 10$) this would be very computationally expensive. On the other hand, the H and D ordering schemes give the ordering of matches before anything is calculated. Thus one only needs to calculate the probability of the best match found or better, and this can be estimated with formulae we show later.

Match Probability Calculation

Regardless of which ordering scheme is chosen, our combinatorial method provides a unified method for computing probabilities of sequence matching in words of arbitrary length and letter composition (including words with repeated letters and words with matches that have interruptions), and for reference sequences of arbitrary length. The method can be described simply as follows. Given a particular reference sequence and an arbitrary word of length n, find

the best match in the ordered match list which is actually found in the word. Then compute how rare such a match is by chance (i.e. assuming all n! permutations of the word's letters were equally likely) by determining the fraction of the n! permutations which contain a match that is as good as that match or better according to the ordered match list. We illustrate this procedure with examples.

Given reference sequence S = 123456789, say we observe the word 3256789. For this word n = 7, k = 7. Using the D ordering scheme, the ordered match list (best to worst) is: (7,0) > (6,0) > (6,1) > (5,0) > (5,1) > (4,0) > (5,2) > ... The best match from this list actually found in the word is a (6,0) match, i.e. 256789. Out of the 7! = 5040 possible permutations of the 7 letters, there are 12 permutations whose best match is a (6,0) match (not necessarily 256789), and 1 permutation with a better match, the (7,0) match 2356789. Thus the probability of getting a (6,0) match or better in a word of this letter composition by chance (assuming all permutations were equally likely) is very small: (12+1)/7! = 0.0026 (Figure 3.2).

Figure 3.2

Sequence match analysis of word 3256789

"D" ordered match list (best to worst): (7,0) > (6,0) > (6,1) > (5,0) > (5,1) > (4,0) > (5,2) > ...

Best match actually found: (6,0) = 3256789

Permutations whose best match is:

(7,0)	2356789			
(6,0)	2356798	2356897	2357896	2367895
	2567893	3567892	3256789	5236789
	6235789	7235689	8235679	9235678

P = (# permutations with (6,0) match or better)/(total # of permutations) = 13/7! = 0.0026

Instead, say we observe the word 51469784. For this word n = 8, k = 7. Again using the D ordering scheme, the ordered match list (best to worst) is: (7,0) > (7,1) > (6,0) > (6,1) > (5,0) > (6,2) > (5,1) > (4,0) > (5,2) ... The best match from this list actually found in the word is a (5,1) match, i.e. 146Z78 where Z = 9. Out of the 8! = 40320 possible permutations of the 8 letters, there are 2338 permutations which contain a (5,1) match or better, according to the ordered match list. Thus the probability of getting a (5,1) match or better in a word of this letter composition by chance (assuming all permutations were equally likely) is: 2338/8! = 0.0580.

As these examples show, our method allows us to quantify matching in words that do not necessarily contain every letter in the reference sequence, that may have repeated letters, that may have matches with interruptions, and that may have additional letters outside the matching segment. All of these situations are handled in the same way with permutations and ordered match lists.

To reiterate the null hypothesis, we assume that, given a word of particular length and letter composition, all possible orderings (i.e. permutations) of those letters (including any repeated letters) were equally likely. That is, we were assuming no particular ordering. We then compute the probability that such random ordering of these letters would by chance produce a match to a specific reference sequence that is at least as good as the best match observed in the actual ordering of letters in the word.

As mentioned above, this is a conditional probability, i.e. conditional on the letters actually observed in the word. But this approach gives us the same answers as other reasonable null hypotheses.

For instance, consider the following null hypothesis. Given a reference sequence of length N, say we observe a word of length n containing $k = n$ distinct letters ($n \leq N$). Find the best match in that word (according to the chosen match ordering scheme). Compute the probability of getting a match as good as or better than this in any word of the same length and with as many distinct letters. Assuming all $n!$ permutations of each of the $N!/(n!*(N-n)!)$ possible words were equally likely, the fraction containing such a match is the same as the conditional probability computed above. On the other hand, if $k < n$ (there are repeated letters), then the form of this alternate null hypothesis is less clear, while our conditional approach need not be altered.

Now consider a different null hypothesis. In this case, assume each of the N letters in the reference sequence occurs exactly once in a given word, but that we only observe $n < N$ of them. Further assume that the relative position of the unobserved letters could be anywhere. Consider the probability that we would observe a match as good as or better than the best match found among any n observed letters, assuming all $N!$ underlying permutations of the N letters were equally likely. Again this reduces to the conditional probability computed above, and again it is not clear how this alternate null hypothesis should handle repeated letters. In essence, our conditional probability approach quantifies the degree of matching through sampling the activity,

i.e. by analyzing the letters we actually observe. Furthermore, our approach handles repeated letters in a natural way.

Essentially, our conditional probability approach separates the issue of generating a word from that of the order within a word. One could imagine a null model which did not separate these issues and instead specified how each word was to be produced from among all possible words. This model might include the probability of producing a word of a given length, of getting a particular set of letters, of having certain letters repeat, of having these letters in a certain order, and all this perhaps as a function of previous words. For instance, consider a first-order Markov process with N+1 states (one state for each letter plus a termination state), with the initial letter drawn from some probability distribution, and with transition probabilities for moving from each state to any other state (including itself) specified. This model would implicitly specify a null hypothesis of what letter orderings to expect (most likely not equally probable). However, our conditional probability approach sidesteps the complicated issue of how words are generated by taking the letters of each observed word as given. Instead we focus exclusively on the issue of order by assuming that every possible ordering of those observed letters was equally likely. This also means that the ordering of letters within each word is assumed to be independent of the ordering in every other word.

Is it a good idea to not assume any particular model of word generation? A modified version of the coin flip analogy can again be useful here. Say we observe 100 consecutive trials of an unknown experiment with two possible outcomes (called "heads" and "tails"). Unlike with a real coin, here we do not know whether there is a fixed probability of getting a head on any given trial, whether trials are independent, or whether any aspect of the system is constant in time. How would one analyze such a potentially complicated system? One of the first things to measure would be the observed fraction of heads. Say it is 0.6 (i.e. 60 heads). To get a sense of whether this fraction is unusually large, we could compare it to the fraction expected from the simplest possible model: 100 independent trials each with fixed probability 0.5 of getting heads. Based on this model, the probability of observing a fraction ≥ 0.6 is 0.03, i.e. low. What can we conclude from this? We cannot conclude that the trials are independent each with fixed probability 0.6 of getting heads. In fact, based on the fraction alone, we cannot conclude that any particular model of how outcomes are generated is likely be correct. For instance, a deterministic sequence of "HHHTT" repeating forever would also produce the fraction 0.6. We can only say that it is unlikely for the simplest model—100 independent flips of a fair coin—to be correct. But

this knowledge is of value. This is exactly what we have done with respect to sequences. We have developed a way to *measure* the amount of sequence structure in an arbitrary word (and, as we shall show later, in a set of words, i.e. trials) by comparing it to the amount of sequence structure expected from the simplest possible model—i.e. the random model of ordering. It is analogous to measuring the fraction of heads. This measure says nothing about how the observed amount of sequence structure comes about, i.e. how the words are generated. But as we have seen with the coin flip analogy, no such measure can prove a particular model is correct. Even with a set of many different measures, in the absence of constraints (such as those based on possible physical mechanisms) the space of possible models is too large to conclude that any particular model is likely to be correct. Thus here we avoid the complex issue of generative models.

More Efficient Ways to Compute Exact, Upper Bound, and Lower Bound Match Probabilities

Though our combinatorial approach applies to words of arbitrary length and letter composition, this does not mean that it is always easy to calculate the corresponding probabilities, in particular for longer words. Consider a word of length n with k distinct letters. Let m be a N dimensional vector representing the multiplicity of each of the N possible letters in this word. Thus k elements are nonzero, and $\sum_i m_i = n$. A brute force probability calculation would require testing each of the $n!/(\prod_i (m_i!))$ distinct permutations to see whether they contained the best match found or better. Fortunately, we have developed more efficient ways to calculate the exact probability for some important cases, and upper and lower bounds of the probability for other cases. In particular, we have a formula for computing the exact probability of getting a (k,0) match (i.e. the best possible match) in any word (i.e. given any m). Furthermore, we have a formula for computing the upper bound probability for getting any (x,y) match, and a lower bound probability for getting any (z,0) match where z < k. (See Appendix A2 for the formulae and derivations.)

With these formulae we can compute both upper and lower bound probabilities of getting a given match or better in any word with respect to any ordering scheme. For example, consider again the reference sequence 123456789 and the word 51469784. For this word m = [1 0 0 2 1 1 1 1 1]. Using the D ordering scheme, the ordered match list (best to worst) is: (7,0) > (7,1) > (6,0) > (6,1) > (5,0) > (6,2) > (5,1) > (4,0) > (5,2) ..., and (5,1) is the best match found. We can

use formula A2-2 to calculate the upper probability of getting a (5,1) match or better according to this ordering scheme as follows. The matches we must count are (7,0), (7,1), (6,0), (6,1), (5,0), (6,2), and (5,1). Since formula A2-2 gives the probability of getting any (u,v) match with $u \geq x$ and $v \leq y$, we use it twice—once with x = 5 and y = 1, and once with x = 6 and y = 2. The first gives an upper bound probability of getting a match inside A, the second an upper bound probability of getting a match inside B (Figure 3.3). Adding these two upper bounds together gives us the desired upper bound (though not necessarily the tightest). Here we get 0.1038 compared to the exact probability 0.0580.

Figure 3.3

To compute a lower bound probability of getting a (5,1) match or better, find the worst (z,0) match as good as or better than (5,1), here (5,0), and compute the lower bound probability of getting this match using formula A2-3. This is the desired lower bound (though, again, not necessarily the tightest), since it is a lower bound of getting a match that is as good as or better than (5,1). Here we get 0.0195 compared to the exact probability 0.0580. An analogous approach will work for any ordering scheme.

Modified Null Hypotheses

In computing probabilities, we have so far used the null hypothesis that all permutations are equally likely, but our approach also handles other null hypotheses in which permutations have unequal weightings. All that is necessary is a way to weight the permutations in a manner appropriate to test what is desired.

For instance, suppose it is known that pairs of letters tend to occur in the same order as in the reference sequence S with a probability B, where B is called a bias. If B > 1/2 then we would expect more sequence matching to S than based on equally likely permutations. To quantify the degree of sequence matching beyond that expected based on the bias, we weight permutations

with more pairs of letters in order more. The exact weighting is a matter of choice. We could weight each n letter permutation by $B^f*(1-B)^r/H$ where f = the number of adjacent letter pairs in the permutation that are in the same order as in S, r = the number in the opposite order, and H is a normalization factor to make the weights of all permutations sum to 1. (Note repeated adjacent letters would be ignored, thus $f + r \leq n-1$.) If B > 1/2, this would tend to increase the probability of getting a given match or better. A more complicated version of this would involve using bias values that are specific to each letter pair, e.g. B(3,5) = the assumed probability of observing the pair 35 given that one observes a 35 or 53. Alternatively, we could weight each permutation by $B^f*(1-B)^r/H$ where f = the number of all $n*(n-1)/2$ letter pairs in the permutation that are in the same order as in S, r = the number in the opposite order, and H is again for normalization. This treats each of the $n*(n-1)/2$ letter pairs as independent, but since they cannot all be independent, this gives an upper bound of the effect of a bias B > 1/2. We have used this latter weighting in our work on hippocampal sequences in SWS (Lee and Wilson, 2002). Other weighting schemes are possible, including those that account for other pair effects, as well as for triplet effects, and many other types of interactions. However, the general problem of how to "subtract out" the effects of lower-order interactions is a topic for further study.

Another example of an interaction we could control for is shown by the following case. In many areas of the brain, during certain states (e.g. SWS) individual neurons (e.g. hippocampal CA1 pyramidal neurons) tend to fire bursts spikes separated by short (e.g. 5 msec) inter-spike intervals (Ranck, 1973). If we let each such burst be represented by a single letter (which is given the time of the first spike in the burst), then words created from letters representing bursts from different cells will have the property that it is harder for repeated letters to appear close together than for distinct letters. Thus not all permutations would be equally likely—those with repeated letters closer together than in the observed word would have been less likely to occur than the other permutations. One way to account for this is to ignore (i.e. weight = 0) all permutations in which repeated letters occur closer together than in the observed word, and assume all other permutations were equally likely (i.e. weight = 1). For instance, the probability of the (4,0) match in the word 12341 would increase from 0.0333 to 0.1667 since our constraint would prohibit permutations such as 11234. Again, we have used this in our work (Lee and Wilson, 2002).

Significance of Matching over Single and Many Words

So far we have been dealing with single words. As mentioned above, the probability we calculate (whether with the equally likely—or any other—assumption for the permutations) is identical to a p value, thus giving us the significance of sequence matching for that word.

More importantly, an overall significance of sequence matching in a set of words can be determined in the following manner. First we define a "$P \leq P'$ trial" as a word for which the best possible sequence match produced by optimally rearranging its letters (generally a (k,0) match) has probability $\leq P'$ of occurring, i.e. a word which *could* have contained a match of probability P' or rarer based on its constituent letters. We define a "$P \leq P'$ match" as a word for which the probability of getting the best match *actually* found in the word or better is $\leq P'$. Thus the expected $P \leq P'$ match/trial ratio for a set of words is P', assuming the ordering of letters within each word is independent of the ordering within all the other words. Given M matches out of T trials, the significance of the observed ratio M/T can be quantified with a Z score determined from the normal approximation to the binomial distribution:

$$Z = (M - T*P')/sqrt(T*P'*(1-P'))$$

The exact p value is given by the binomial distribution:

$$p \text{ (observing} \geq M \text{ matches)} = \sum_{i=M}^{T} C(T,i)*(P')^{i}*(1-P')^{T-i}$$

where $C(a,b) = a!/(b!*(a-b)!)$. This becomes difficult to compute for large T. For moderate sized P' (e.g. 1/2 or 1/6), moderate sized Z, and sufficiently large T, the normal approximation is good and the Z score can be directly translated into a p value using the normal distribution. For small P' (e.g. 1/24) and/or large Z, the normal approximation is not good, and one should use other approximations or bounds of the p value. For example, the binomial distribution can be approximated by a Poisson distribution for small P', and thus an approximate p value is given by:

$$p \text{ (observing} \geq M \text{ matches)} = \sum_{i=M}^{T} e^{-(T*P')}*(T*P')^{i}/i!$$

One should note the difference between the ratio (matches/trials) and the significance (Z score and p value). The comparison between the observed and expected match/trial ratios reveals what the effect size is likely to be, while the Z score (and p value) gives an idea of the confidence we should have of that effect size based on the amount of data we have. To prevent an underestimate of the degree of matching, one should check that most of the $P \leq P'$ matches do not have probabilities that are far lower than P'.

This approach is valid for any chosen P'. Since we are generally concerned with finding at least moderately long matching sequences, we set P' to values $< 1/2$, such as 1/6, 1/24, or 0.01. Since $1/3! = 1/6$ is the probability of getting a (3,0) match in a $n = 3$, $k = 3$ word, setting $P' = 1/6$ means that we are focusing on matching sequences that generally have at least 3 letters in order (i.e. "3rd-order interactions"). Likewise, setting $P' = 1/4! = 1/24$ focuses on matching sequences with generally 4 or more letters in order (i.e. "4th-order interactions"), and $P' = 0.01$ 5 or more letters. Thus by setting P' we can focus on sequence matches of any desired length (i.e. "nth-order interactions") within the same framework.

To quantify matching over a set of words using the upper and lower bound probabilities described above, there are two approaches with different goals. First, to compute a lower bound of the match/trial ratio, count a word as a $P \leq P'$ trial unless shown otherwise (i.e. unless some lower bound probability of the best possible match is $> P'$), and do not count a word as a $P \leq P'$ match unless shown otherwise (i.e. unless some upper bound probability of the best match found in the word is $\leq P'$). Alternatively, to compute an upper bound of the match/trial ratio, count a word as a $P \leq P'$ match unless shown otherwise (i.e. unless some lower bound probability of the best match found in the word is $> P'$), and do not count a word as a $P \leq P'$ trial unless shown otherwise (i.e. unless some upper bound probability of the best possible match is $\leq P'$, or unless it has already been counted as a $P \leq P'$ match). In the case of the standard equally likely permutation assumption, formula A2-1 for the exact probability of a (k,0) match allows an unambiguous determination of whether a word is a $P \leq P'$ trial for both the upper and lower bound ratios. The lower bound ratio should be used as a conservative test to show that significant matching exists, and the upper bound ratio as a conservative test to show that significant matching is not present. In addition, in trying to show the latter, one should (as mentioned above) verify that most of the $P \leq P'$ matches do not have probabilities that are far lower than P'. In our hippocampal sequence learning work, we computed the lower bound ratio (and Z score) to demonstrate significant matching between the experienced sequence and the set of words from

the period subsequent to that experience. We computed the upper bound ratio (and Z score) to demonstrate a lack of matching between the experienced sequence and the set of words from the preceding period. We also verified that most of the $P \leq P'$ matches from that preceding period did not have probabilities that were far lower than P' (Lee and Wilson, 2002).

By considering matching over a set of words, our method allows one to detect traces of a particular reference sequence in a heterogeneous set of words that do not contain every letter and which may have few letters in common with each other. For instance, consider the following set of 4 words made from 7 letters: 125, 2367, 146, and 3455. Each word contains no more than 4 out of the 7 possible letters, and no pair of words has more than one letter in common. However, together these 4 words indicate the strongest traces of 3 out of the 7! = 5040 possible reference sequences: 1234567, 1234657, and 1234675.

Application to Specific Types of Neural Data

We have described our approach in terms of abstract symbols (letters and words) to focus only on the essential elements of relative order and sequence matching. The results are general and can be applied to a broad range of multiple source neural data. To apply our results to a specific type of neural data one simply needs an appropriate (ideally natural) "parsing" method to convert the data into letters and words.

For instance, in an experiment composed of separate trials, the activity during each trial could give a different word. Within each trial, the order of letters (e.g. with each letter representing a particular neuron or local EEG site) could be determined by the time of the first event (e.g. spike or particular EEG event) from each source after a given reference time, such as the presentation of the stimulus, the beginning of the delay period, the beginning of a motor movement, et cetera. The reference sequence would be a presumed order of activity that one wanted quantitative evidence of in a set of trials.

One could also parse a continuous stretch of data into a set of words. In a recent study (mentioned several times above), we have parsed a continuous stretch of spike activity from many rat hippocampal CA1 pyramidal cells during slow wave sleep (SWS) (Lee and Wilson, 2002). The characteristic activity of such cells during SWS consists of intermittent population bursts of nearly coincident (within 100 msec) activity from a subset of cells, separated by

relatively long periods (from 200 msec to seconds) with little activity. Furthermore, each time a cell fires in such a population burst, it tends to produce not a single spike, but a burst of spikes with small (5 msec) inter-spike intervals (Ranck, 1973). Thus we parsed the data in the following manner. Each cell was identified with its own letter. Each time a cell fired a burst of spikes, the activity was represented by the corresponding letter occurring at the time of the first spike. The letters from the subset of cells firing within a population burst were then ordered by time to form a single word. Thus each word represented the relative order of activity within a population burst. These words (representing a set of population bursts) were then tested for matching to a reference sequence representing the order of spatial activation of these same cells during running behavior.

Finally, as mentioned above, while our approach works for words of arbitrary length and letter composition, it gives the most useful answers if the length of each word is not significantly larger than the number of distinct letters in that word (i.e. if n is not much greater than k). Thus one criterion for a good parsing method is that it produces a set of words with this property.

Discussion

To summarize, we have developed a new approach for analyzing spike train data from an arbitrary number of neurons. It can also be applied to many other types of multiple source neural data. In particular, we analyze sequential activity based on relative time order. We quantify the degree of matching between a reference sequence and a segment of activity (represented by a word) by computing the probability (which is equivalent to a p value) of getting a match as good as or better than the best match actually observed in that word. This requires choosing what we mean by a match and a way to rank the quality of various possible matches. Here we chose to define matches in terms of the number of letters in order (x) and the number of interruptions (y). We describe criteria that any ranking of these matches must satisfy, and discuss some possible ranking schemes (e.g. horizontal and diagonal). Computing the probability also requires a null hypothesis. We generally choose to assume that all permutations of the observed word were equally likely. Thus the probability we compute is actually a conditional probability, i.e. conditional on the letters actually observed in each word. The degree (match/trial ratio) and significance (Z score and p value) of matching over a set of words can be computed from the corresponding set of probabilities. Furthermore, we derive formulae that make these match probability calculations for longer words more manageable.

Our method does not cover all possible meaningful patterns, for instance, a sequence of spikes some of which come from the same cell (i.e. repeats in the reference sequence), (nearly) coincident spikes from a subset of cells regardless of relative order, and patterns with exact time delays. But the type of sequences our method does cover represents a meaningful category of pattern, i.e. a sequence of events (e.g. spikes, bursts of spikes, EEG events) with each one from a different source (e.g. cell, EEG location). For example, we have recently studied rodent hippocampal activity and shown long, recurring sequences in sleep that match the sequence of places repeatedly traversed by a rat in immediately preceding behavior. This, combined with an absence of such sequences in sleep before the behavior as well as other analyses, provides evidence of the rapid and precise encoding of sequential spatial experience (Lee and Wilson, 2002).

In general, any method of quantifying pattern matching (regardless of whether the patterns are based on relative time order, exact time delays, or any other characteristics) must have the following two features: (1) a way to rank (i.e. score) the different possible matches, and (2) a null hypothesis describing the expected distribution of matching scores. Given these, the significance of a particular matching score is determined as the probability of getting that score or better with respect to the expected distribution of scores. Here we chose to look for patterns based on relative time order and to restrict reference patterns to sequences without repeats. Then the following choices needed to be made: a definition of matching, a way to rank matches, and a null hypothesis. We chose to define matches in terms of (x,y). Since this had two parameters (x and y), the way to rank different matches was not obvious, so we described different reasonable ways to rank them. (With a one parameter matching score, the ranking would be implied—better matches corresponding to higher scores. An example of such a one parameter matching score is the Spearman rank correlation coefficient (Lehmann, 1975), which could be adapted to measure the similarity between a word and a reference sequence.) For the null hypothesis we took the letters observed in each word as given and assumed that each permutation of these letters was equally likely. However other choices could have been made. For instance, with our definition of matches, the words 12477 and 71247 both contained the best possible match with respect to the reference sequence S = 123456789. But a different definition of matches could have ranked 12477 as better than 71247. Or we could have penalized interruptions more, or different interruptions differently (e.g. 124356 as better than 126345). (A measure like the Spearman rank correlation coefficient would have ranked 12477 as better than 71247, and 124356 as better than

126345, but it also would have ranked 321654 as better than 612345.) Such alternate definitions could have still used the same null hypothesis, or some other one. For instance, we could have chosen a generative model which specified how the words were produced from scratch. Such a model would implicitly specify a distribution of matching scores. Above we discussed reasons why we did not choose a generative model and instead chose to compute conditional probabilities. Nevertheless, one should keep in mind the variety of choices and the reasons for particular decisions. Ultimately, choices are informed by the nature of the data being analyzed and by the types of patterns one is interested in looking for. One might even try several different ways of quantifying matching, then apply a Bonferroni correction to the lowest p value.

Previous methods of analyzing firing patterns across multiple cells include correlation methods, such as the cross correlation function between a pair of spike trains, the joint peristimulus time histogram (JPSTH) which can be applied to 3 spike trains (Palm et al., 1988; Aertsen et al., 1989), and the suggested extension of the JPSTH to an arbitrary number of spike trains (Gerstein and Perkel, 1969; Martignon et al., 1995). Other methods of analyzing the spike trains of an arbitrary number of neurons include an elegant algorithm to detect all repeating firing patterns consisting of spikes separated by exact time delays (within some jitter) (Abeles and Gerstein, 1988; Abeles, 1991; Abeles et al., 1993), gravity methods (Gerstein et al., 1985; Gerstein and Aertsen, 1985), and template methods (Louie and Wilson, 2001; Lee and Wilson, 2002).

With the exception of gravity and template methods, these previous methods have classified patterns by the exact time delays (within binning) between the spikes. While patterns with exact time delays that repeat may be of special significance for some types of neural processing (Abeles, 1991), this can result in a sampling problem, especially with more complex patterns. That is, with larger n (i.e. more spikes in each pattern), it becomes harder to accurately estimate probability densities in the (n-1)-dimensional space of exact (binned) delays. Furthermore, when firing rates are high (and thus the sampling problem is reduced), such methods will tend to find patterns that are embedded within much non-pattern activity. Such embedded patterns are presumably less important physiologically than unembedded ones, at least in terms of their effect on downstream (i.e. postsynaptic) targets. In contrast, by classifying patterns based on relative firing order, we can greatly reduce the sampling problem, even in data with low firing rates (e.g. hippocampal pyramidal cells in SWS (Lee and Wilson, 2002)). In addition, by using certain parsing schemes, our method will ignore patterns embedded in the

51

midst of other activity. Since every letter counts, embedded matching patterns are detected only to the extent that the non-matching activity explicitly does not increase by too much the probabilities of getting such a pattern.

Let's investigate this sampling issue in more detail. Say we have N neurons and are concerned with patterns that occur within J msec. First consider patterns with exact time delays. Say we divide J into T time bins and for each neuron consider only the bin with peak firing. Then there are

$$\sum_{i=1}^{N} C(N,i)*(T-1)^{N-i}$$

possible patterns, with the exact delays quantized by J/T msec. (To see this, consider the i = 2 term. This term represents the number of ways to choose 2 particular neurons to occupy the first time bin, times the number of ways to place each of the remaining N-2 neurons in any of the remaining T-1 time bins. The sum is over the different numbers of neurons that can be chosen to occupy the first time bin.) In contrast, if we take the peaks of the smoothed (but unbinned) firing rates and consider only relative firing order, then there are N! possible patterns. Suppose N = 5 neurons, J = 500 msec, and T = 50 bins, i.e. 10 msec bins. Then there are approximately 3.0×10^7 patterns with exact time delays and 120 patterns based on relative firing order. If N = 10, then there are approximately 1.8×10^{16} patterns with exact time delays and 3.6×10^6 patterns based on relative firing order. But for large N, the number of patterns with exact time delays is approximately $N*(T-1)^{N-1}$, which is less than the number of patterns based on relative time order. However, this is the case only if T is fixed. On the other hand, if T α N then the comparison is N^N versus N!, i.e. there are more patterns with exact time delays. What are the conditions under which T would be fixed versus α N? Large N and fixed T means that the patterns with exact time delays will be forced to contain many neurons that fire coincidentally (i.e. fire in the same time bin). Large N and T α N means that these patterns can be truly sequential (i.e. fire in different time bins). Thus these are some considerations that may go into determining which method of analysis to use. As mentioned above, other important considerations include such things the particular nature of the data being analyzed.

Much important neural processing probably occurs in discrete events. However, cross correlation methods provide time-averaged values of interactions, and thus cannot analyze the

significance of individual events (words). Our method allows a determination of the significance of (matching within) such individual events through the calculated match probability. For instance, each recurring, low probability sequence we found in hippocampal SWS activity may represent a discrete burst of learned information that is being broadcast to target structures elsewhere in the brain in a process of memory consolidation (Lee and Wilson, 2002). Other methods (Abeles and Gerstein, 1988; Abeles, 1991; Abeles et al., 1993) also allow an identification of important individual events.

Our approach allows a calculation of the degree and significance of high-order (i.e. long) sequence matching based on first principles (i.e. the equally likely permutation assumption and treating a set of words as independent Bernoulli trials) without the use of Monte Carlo techniques. We have verified both the validity of the null hypothesis and the accuracy of the calculated Z score in real data by computing the distribution of significance (Z scores) of matching for a set of words with respect to a large set of shuffled reference sequences (Lee and Wilson, 2002). In contrast, methods to estimate the significance of patterns found with gravity techniques have not been presented (Gerstein et al., 1985; Aertsen and Gerstein, 1985).

One goal of methods that analyze higher-order interactions (e.g. interactions involving more than two neurons) is to subtract out the influence of lower-order interactions. For instance, one may want to know whether a larger than expected number of longer sequences can simply be explained by the random chaining together of pairs of neurons that tend to fire in order. If so, this would not necessarily mean that the longer sequences were unimportant, but it would at least suggest a certain type of underlying mechanism. Our approach can account for pairwise biases by weighting permutations unequally instead of equally. The analogous correction for pairwise effects can be done in the 3-neuron JPSTH analysis (Palm et al., 1988; Aertsen et al., 1989). However, a general method of correction for all possible lower-order interactions has not been described for any previous higher-order method, including our own. It is even hard to specify which effects one would want to adjust for. This is a topic of future research for all higher-order methods. Whichever lower-order interactions one decides to account for, our approach would handle it by simply weighting the permutations appropriately.

Finally, while all the other methods described above attempt to discover previously unknown patterns which may be significant, our method and the template method (Louie and Wilson, 2001) specifically test for evidence of a particular reference pattern (sequence) within the

data. That is, unlike the other methods, our approach does not automatically search all possible patterns for interesting candidate patterns. One could simply try all possible reference sequences, but this would be prohibitively time consuming for longer reference sequences. Thus our method generally requires educated guesses of good candidate reference sequences. But once one has chosen a reference sequence, our method allows evidence for that sequence to be detected even with missing spikes (due to sampling via conditional probabilities) and errors (interruptions).

In conclusion, a full understanding of brain computations will most likely require knowledge of complex interactions among large numbers of neurons, and across multiple brain areas. However, the number of possible firing patterns grows exponentially with the number of neurons (or other sources) involved. The current ability to record from over 100 neurons simultaneously (Meister et al., 1991; Wilson and McNaughton, 1993; Wessberg et al., 2000) already makes a search of all possible patterns virtually impossible. Therefore, intelligent guesses of what important patterns may look like in each experiment will be necessary for any analysis. Intelligently designed experiments can help by making it more likely that important patterns are revealed. Given a manageable set of candidate patterns, methods such as ours can then be used to test them quantitatively for significance.

CHAPTER 4

MEMORY OF SEQUENTIAL EXPERIENCE IN THE HIPPOCAMPUS DURING SLOW WAVE SLEEP[1]

Summary

Rats repeatedly ran through a sequence of spatial receptive fields of hippocampal CA1 "place" cells in a fixed temporal order. A novel combinatorial decoding method reveals that these neurons repeatedly fired in precisely this order in long sequences involving 4 or more cells during slow wave sleep (SWS) immediately following, but not preceding, the experience. The SWS sequences occurred intermittently in brief (~100-millisecond) bursts, each compressing the behavioral sequence in time by approximately 20-fold. This rapid encoding of sequential experience is consistent with evidence that the hippocampus is crucial for spatial learning in rodents and the formation of long term memories of events in time in humans.

Introduction

Information processing in the brain is believed to require coordinated activity across many neurons. However, the exact nature of this code is still largely unknown, and remains a fundamental problem in neuroscience. With the recent development of methods to simultaneously record the spiking activity of large numbers of individual neurons, the search for complex firing patterns across multiple cells that could directly address this issue has become possible (Abeles and Gerstein, 1988; Aertsen et al., 1989; Abeles, 1991; Meister et al., 1991; Abeles et al., 1993; Wilson and McNaughton, 1993; Wilson and McNaughton, 1994; Skaggs and McNaughton, 1996; Kudrimoti et al., 1999; Nádasdy et al., 1999; Wessberg et al., 2000; Louie and Wilson, 2001). Here we present a new method for analyzing sequential firing patterns involving an arbitrary number of neurons, and use it to decode the activity of a population of rat hippocampal neurons during slow wave sleep (SWS).

The hippocampus has been a target of particular interest due to its involvement in the formation of long term memories of events in time in humans (Scoville and Milner, 1957; Milner,

---

1966; Zola-Morgan et al., 1986) and spatial memory (O'Keefe and Nadel, 1978; Morris et al., 1982), as well as more general sequence memory (Kesner et al., 2002; Fortin et al., 2002), in rodents. It sits atop a neuroanatomical hierarchy, receiving highly processed information from many areas of the brain, and is thus in an ideal position to link events across space and time (Mishkin et al., 1998). A rodent hippocampal CA1 pyramidal cell ("place cell") fires selectively when the animal is in a particular location (the cell's "place field") of an environment (O'Keefe and Dostrovsky, 1971). Since different place cells fire selectively in different locations, a rat's experience in moving from one location to another can be represented by the resulting sequence of place fields traversed. Such a sequence may also be a good model for the temporal order of events experienced by humans. We tested whether repetitive experience of a particular spatial sequence resulted in detectable memory traces of that sequence during SWS immediately afterwards.

A major reason for analyzing neural activity during sleep is that, because of the clear absence of ongoing spatial behavior, the effect of previous spatial experience may be more easily identified. Specifically, by comparing the activity in sleep before and after a task, one may identify experience-dependent changes that could correspond to learning. Here we find recurring sequences during SWS that involve 4 or more cells and match the rat's immediately preceding spatial experience. In contrast, there is no indication of any matching sequential structure in SWS immediately before the experience. Together with additional analyses, this provides direct neural evidence of the rapid encoding of extended spatial sequences.

Results

Adult rats (n = 3) ran back and forth (23-35 round trips in 20-25 minutes) along linear tracks (180 cm straight or 450 cm U-shaped) for chocolate reward at each end. The behavior consisted of running in one direction (POS, ~5 sec), followed by eating at one track end (~10 sec), then running in the other direction (NEG, ~5 sec), followed by eating at the other track end (~10 sec), then back to a POS lap, and so on. Both immediately before (PRE) and after (POST) this behavior (RUN), the rats slept in a small enclosure located away from the track. During all three periods (PRE, RUN, POST) we continuously recorded each rat's position and head direction, the spiking activity of many individual hippocampal CA1 pyramidal cells (Figure 4.1), and the local field potentials (local EEG) in the region around these cells. (See Experimental Procedures for details and Table 4.1 for information on each rat.)

Each rat's spatial experience during RUN was divided into two separate place field sequences (POS, NEG), one for each direction along the track. These sequences (each lasting about 5 seconds) were determined by ordering the peaks of the cells' smoothed firing rate fields (Figures 4.2A-B,G). In linear environments, a large fraction of place cells fire in only one direction; thus the two sequences generally consist of different sets of cells. Some cells fired in both directions and thus participated in both sequences. Place fields of such cells were determined separately for each direction. The average number of cells per sequence was 8.5 +/- 2.1. (See Experimental Procedures for details.)

Figure 4.3A illustrates the characteristic activity of a population of CA1 pyramidal neurons during SWS (in this case from POST): short bursts of spikes from several cells, with adjacent bursts separated by larger gaps with little firing. Closer inspection of one burst (Figure 4.2C) reveals that 6 cells (2,5,7,8,9,10) fired in the same order as during RUN (Figure 4.2B), and were highly compressed in time relative to RUN (2.4 sec in behavior becomes 120 msec in SWS). Note that the order matches only if we count the first spike from each cell, with the later spikes representing a continuing burst of activity related (but not limited) to complex-spiking of CA1 pyramidal cells (Ranck, 1973). POST SWS contains many such examples of short bursts which closely match the order of their place field peaks in RUN (Figures 4.2C-F match Figure 4.2B, Figures 4.2H-J match Figure 4.2G).

To quantify these sequential patterns, we developed a two step decoding procedure. First, each SWS population burst of cells from a given RUN sequence was parsed into a single "word" representing only the relative firing order of the cells within that burst. Then we computed the probability that the relative firing order within each word would by chance alone match the RUN sequence as well as it actually did. We computed the overall degree of matching between the RUN sequence and (PRE or POST) SWS activity by considering the probabilities from all the words. The details of this procedure are as follows.

To extract the individual population bursts, our parsing procedure consisted of 2 parameters applied in order to the activity of a set of cells in SWS. First, for each cell, each set of consecutive spikes with inter-spike intervals less than max_isi (msec) was grouped into a single event (represented by a single "letter" identifying that cell) occurring at the time of the first spike. Each isolated spike (i.e. a spike separated by at least max_isi from any other spike of that cell) also produced a letter. Then the letters from all the cells were merged to form a single time-

ordered string. This string was then broken between every pair of adjacent letters separated by more than max_gap (msec) into a set of shorter strings ("words"). Each word thus represents the relative order of cell activity within a SWS population burst. We parsed the SWS activity of the cells from the POS and NEG place field sequences separately—the SWS activity of the cells active in the POS behavioral sequence gave one set of words (to be tested for matches to the POS experience), and those from NEG another set (to be tested for matches to the NEG experience). (See illustration on page 5 which shows a 200-second stretch of words parsed from the POST SWS activity of the cells from the POS place field sequence of RAT2.)

To quantify the degree of matching between (PRE or POST) SWS activity and a given RUN sequence, we classified the words parsed from SWS into 3 different groups based on their complexity—"Pairs": 2-letter words with 2 distinct letters, "Triplets": 3-letter words with 3 distinct letters, and "Low Probability trials": defined precisely below, but generally consisting of words with 4 or more distinct letters as well as any additional repeated letters. (Appendix A3 contains a summary of all the words parsed from SWS (Table A3.1).) Sequence matching within each group was evaluated separately. For Pairs, we computed the ratio of the number of Pairs in which the 2 letters matched their order in the RUN sequence (Pair matches) to the total number of Pairs regardless of letter order (Pair trials). The ratio's significance is the number of standard errors above the expected ratio (1/2), assuming either ordering of letters was equally likely, i.e. a Z score. The same was done for Triplets, where the expected ratio of 3-letter matches (all 3 letters in order) to all Triplets is 1/6, assuming each of the $3! = 6$ possible orderings of 3 letters was equally likely.

The most important group was the Low Probability trials, since it potentially contained long, very low probability matches (4 or more letters in order) to the RUN sequence. To analyze this group, we developed a novel combinatorial method that allowed us to quantify the degree of matching between the RUN sequence and the relative order of letters in words of arbitrary length and letter composition. The method involves identifying the best match within a word to the given RUN sequence, then computing the probability that this word would contain such a match or better, assuming all possible permutations of its constituent letters were equally likely. Thus the probability indicates how likely it is by chance that the order of letters in a word matches the RUN sequence as well as it does. For example, given the RUN sequence 123456789A (where "A" represents cell 10) (Figure 4.2B), the best match found in the 7-letter word 325789A (Figure 4.2C) is a (6,0) match (6 letters in order with 0 interruptions: here 25789A). Out of the $7! = 5040$

possible permutations of the 7 letters, there are 12 permutations whose best match is a (6,0) match (not necessarily 25789A), and 1 permutation with a better match, the perfect (7,0) match 235789A. Thus the probability of getting a (6,0) match or better by chance is very small: $(12+1)/7! = 0.0026$ (Figure 4.3B). This method provides a unified way of computing the probabilities of different matches in words with differing lengths and letter compositions, including matches with interruptions (e.g. the (6,1) match in 246579A, Figure 4.2E) and matches in words with repeated letters (e.g. the (5,0) match in 22569A8, Figure 4.2D). (See Experimental Procedures for more details.)

The precise definition of a Low Probability trial is a word for which the best possible sequence match produced by optimally rearranging its letters (in general, a (K,0) match for a word with K distinct letters) has probability $\leq P$ of occurring (assuming all permutations are equally likely), i.e. a word which *could* have contained a match of probability P or rarer based on its constituent letters. A word is a Low Probability match if the probability of getting the best match *actually* found in that word or better is $\leq P$. The expected match/trial ratio is P, with significance determined as before. We chose $P = 1/4! = 1/24$, the probability of a perfect 4-letter match. Thus Pairs cover 2-letter matches, Triplets 3-letter matches, and Low Probability trials n-letter matches for $n \geq 4$. However P could be any low value (e.g. $P = 0.01$ focuses on n-letter matches for $n \geq 5$). Therefore our match probability method can analyze sequential firing patterns involving any number of neurons. It is general and can be applied to other types of neural data. It differs from Abeles' method by analyzing the relative ordering of activity as opposed to patterns defined by a set of specific inter-spike intervals (Abeles and Gerstein, 1988; Abeles, 1991; Abeles et al., 1993).

The final results consist of a match/trial ratio and Z score for Pairs, Triplets, and Low Probability words in PRE and POST SWS. This analysis was repeated for a range of max_isi and max_gap values (with the constraint that max_isi $\leq$ max_gap, since max_isi > max_gap would be more likely to allow a single letter to represent activity that extended into the following word) (Figure 4.4). (Note that Z < 0 (Z > 0) corresponds to fewer (more) matches than expected based on chance.) PRE SWS activity shows no significant similarity to RUN sequences for all parameter values, while POST SWS activity shows significant similarity for a wide range of values. This suggests that the sequential spatial experience was encoded during RUN. Furthermore, the most significant matching in POST occurs for longer, lower probability matches. (The exception is max_isi = 0. It results in very little POST Low Probability matching

because if multi-spike bursts from individual cells are treated as multiple letters, they tend to interrupt sequence matches.) The peak Low Probability match/trial ratio for POST ($35/270 = 0.13$, i.e. 35 matches out of 270 words that could have had a Low Probability match, compared to $270/24 = 11$ expected matches, $Z \geq 7.2$, $p < 4\text{E-}9$, Figure 4.4B) occurs around max_isi = 50 msec and max_gap = 100 msec, suggesting that these may be the best values for decoding CA1 pyramidal cell activity in SWS. These values parse the activity into words such that the resulting mean inter-word interval is 804 msec, a number consistent with the hippocampal EEG sharp-wave/ripple occurrence rate of approximately 0.5-1 Hz during SWS. Significant POST Low Probability matching, as well as the other trends of Figure 4.4, are present in the individual rats (Table 4.1). All results that follow are pooled over all 6 RUN sequences of the 3 rats (2 sequences per rat: POS, NEG), using max_isi = 50 msec and max_gap = 100 msec.

Figure 4.5A shows the match/trial ratios for Pairs, Triplets, and Low Probability trials. The most dramatic deviation from expected occurs for POST Low Probability trials: the actual ratio of 0.13 is triple the expected ratio $1/24$ ($= 0.042$) ($Z \geq 7.2$, $p < 4\text{E-}9$), while the actual ratio for Pairs is only 0.52 compared to the expected ratio 0.5 ($Z = 1.6$, $p > 0.05$). (See Appendix A3 for results based on an expansion of the class of Triplets to include all words with exactly 3 distinct letters (Figure A3.1).) Histograms of all words containing matches of probability ≤ 0.1 (Figure 4.5B) show that POST has many events of probability far less than $1/24$ (and thus the lower bound Z of 7.2 is an even greater underestimate), while PRE shows no evidence of RUN sequence structure. Considering even lower probability events ($P = 0.01$ instead of $1/24$ ($= 0.042$)) yields an even greater deviation from the expected match/trial ratio in POST (11 matches out of 140 trials, $140*0.01 = 1.4$ expected, $p < 3\text{E-}7$).

We performed the following control analyses to investigate what might be responsible for the high occurrence of long sequences in POST that match the place field sequences from RUN.

First, we computed the significance for Low Probability trials with respect to randomly shuffled versions of the actual RUN sequences. Figure 4.5C shows that POST specifically matches the actual RUN sequences, and not other random sequences. If general non-RUN-sequence effects were responsible for significant matching between POST and the RUN sequences, significant matching should also have occurred for shuffled sequences, but it did not. Thus the RUN sequence is a special sequence for POST SWS. This shuffle analysis also allows us to test the validity of our match probability method. It shows that our match probability

method produces a statistic (i.e. the Low Probability Z score) that accurately assesses the significance of sequence structure within real population activity. In particular, the POST Low Probability Z score of 7.2 produced by our method agrees with the Z score of 7.9 determined from the POST shuffle distribution (POST shuffle distribution mean Z score = -0.38, standard deviation 0.96, thus the POST Z score of 7.2 is (7.2 - (-0.38))/0.96 = 7.9 standard deviations above the shuffle mean), and likewise for PRE (PRE shuffle distribution mean Z score = -0.31, standard deviation 0.84, thus the PRE Z score of -0.5 is 0.2 standard deviations below the shuffle mean). Furthermore, the PRE and POST shuffle distributions closely match the distributions predicted by a model in which each Low Probability trial is an independent trial with equally likely permutations (Figure 4.5C). (Note that these predicted distributions are not centered exactly around Z = 0, but rather at slightly negative Z scores. This is because not every Low Probability trial can contain a match of probability exactly equal to 1/24 due to its particular letter composition. Thus for some fraction of Low Probability trials, the probability of getting a Low Probability match is strictly less than 1/24, assuming equally likely permutations.) A similar analysis of Triplets gives the same result. Thus the real data are consistent with the equally likely permutation model on which our match probability (and resulting match significance) calculations are based. A non-parametric comparison which uses only match rankings and does not deal with permutations or match probabilities confirms that POST matches RUN sequences more than PRE does ($p < 0.01$, K-S test).

Second, we checked that the long sequence (Low Probability) matching effect in POST was not simply due to a tendency for cells with nearby place fields to fire together regardless of order (Wilson and McNaughton, 1994). If this were the case, then we would also expect significant matching in POST to the *reverse* RUN place field sequences (e.g. A987654321 from Figure 4.2B), but this was not so ($Z \le 1.1$, $p > 0.16$).

Third, we recomputed the probabilities for POST Low Probability trials, this time assuming that all permutations were not equally likely, but rather were weighted by the actual Pair ratio in POST (which, at 0.52, was slightly biased in favor of the RUN sequence) (see Experimental Procedures). The idea was to test whether the observed tendency for pairs of letters in POST to occur in the same order as in the RUN sequence could alone explain the unexpectedly large number of Low Probability matches. The significance was reduced (as expected), but still highly significant (actual match/trial ratio = 0.11, expected ratio ≤ 0.054, $Z \ge 4.2$, $p < 2E-4$).

Thus pair biases cannot account for the high occurrence of long, low probability sequence matches.

Fourth, since the parsing method makes it harder for repeated (versus distinct) letters to appear close together in a given word, we recomputed the match probabilities to account for this. In particular, we did not count permutations in which repeated letters occurred closer together than in the observed word, while the remaining permutations were assumed equally likely. Results were unchanged (Low Probability trials: PRE $Z \leq -0.07$, $p > 0.6$, POST $Z \geq 7.5$, $p < 2E-9$).

Fifth, to specifically control for artifactual sequence effects that could result from imperfect clustering (Quirk and Wilson, 1999), we re-parsed and re-computed the results after each of the following manipulations. We first eliminated a given fraction of spikes from either the outer border or the low amplitude tail of each cell's cluster, then either stopped there or further eliminated all remaining spikes which potentially could still be part of a single complex-spike incorrectly split across multiple cells. For every manipulation, the remaining spikes still contained highly significant Low Probability matching in POST (Figure 4.5D).

Thus none of these potential causes can account for the high occurrence of long RUN sequence matches in POST.

Furthermore, we computed the significance for Low Probability trials with respect to RUN sequences bridging the 10-second break between POS and NEG direction laps. These "wraparound" sequences (2 per rat) were constructed by adding the first half of the NEG sequence to the end of the last half of the POS sequence (giving one sequence), and by adding the first half of the POS sequence to the end of the last half of the NEG sequence (giving the other sequence). The lack of matching in POST ($Z \leq -1.2$, $p > 0.9$) supports the idea that the POS and NEG sequences represent distinct bursts of experience that are encoded separately. Cells which participate in both POS and NEG sequences do in fact appear in the appropriate positions in both POS- and NEG-like SWS sequences (e.g. RAT1 POS cells 5, 9, 10 in Figures 4.2C-F correspond to RAT1 NEG cells 7, 4, 5, respectively, in Figure 4.6C).

For one rat (RAT2), RUN constituted the first exposure to that environment, while the other two rats had some previous training in their respective RUN environments (though all three

rats had been trained before on various linear tracks in other rooms). In both cases (Novel, Familiar), PRE SWS has no sequence structure related to RUN, while POST does (Low Probability trials: Novel: PRE $Z \leq -0.1$, $p > 0.6$, POST $Z \geq 4.8$, $p < 6E-5$; Familiar: PRE $Z \leq -0.9$, $p > 0.9$, POST $Z \geq 5.5$, $p < 2E-5$; also see Table 4.1). This suggests that long sequences can be encoded in a single RUN session, and that the mechanisms leading to expression of RUN sequences in SWS (possibly including modified synaptic connections) are reset sometime after POST but before the next exposure (which occurred at least a day later; see Experimental Procedures). Similar resetting of experience-dependent changes in the hippocampus has been observed before (Mehta et al., 1997; Mehta et al., 2000).

Figure 4.6A shows the distribution of durations (time between first and last letters) of all PRE and POST Low Probability trials (mean 154 +/- 84 msec), as well as all POST Low Probability matches (time between first and last letters in match) (mean 106 +/- 38 msec). The population bursts that constitute the Low Probability trials tend to occur in periods with heightened high frequency "ripple" (100-250 Hz, 25-100 msec duration events) activity (Figures 4.6B,C), a prominent feature of hippocampal EEG during SWS. This is not surprising, since it is known that overall CA1 pyramidal cell activity in SWS is greatly increased during such ripple events.

As seen in Figure 4.2, POST SWS sequence matches are compressed in time relative to the experienced RUN sequence. We estimated the approximate compression factor (CF) by comparing the times of the letters in each POST Low Probability match to the times of their place field peaks in the RUN sequence (e.g. Figures 4.2B,G) (see Experimental Procedures). The median of 19.7 (Figure 4.7A) indicates sequences in POST SWS occurred approximately 20 times faster than the original experience. To verify this using a different method, we computed the average overlap between the POST Low Probability matches and various time-compressed templates of the overall place field activity in RUN (e.g. the smoothed firing rates in Figures 4.2B,G as a function of time) (see Experimental Procedures). The template overlap method detects broadly matching temporal activity patterns at any selected time scale (CF) despite numerous extra spikes and imperfect ordering. The peak occurs around CF = 16 (Figure 4.7B), in agreement with the first method.

Finally, we looked for evidence of any additional RUN sequence structure, at any time scale (CF), in POST SWS besides that detected using the parsing and match probability method.

We again used the template method because of its ability to detect broadly matching activity at any time scale. For each POS and NEG RUN template we computed the overlap as a function of time for many different CF's (Figure 4.7C). Figure 4.7D shows the overlap averaged over all of PRE SWS, all of POST SWS, and all of POST SWS except for those times immediately around sequence matches detected using our parsing and match probability method (see Experimental Procedures). We find no evidence of any sequence structure in PRE SWS. POST SWS has sequence structure with a single peak around $CF = 20$, and this peak disappears when the small fraction of overlap scores around matches of probability $\leq 1/6$ (i.e. Triplet matches or longer) is eliminated. Thus the sequences found with our decoding method (using max_isi = 50 msec and max_gap = 100 msec) likely represent all the RUN sequence structure present in POST SWS (though there could be additional non-RUN sequence structure in PRE and/or POST SWS).

Discussion

In general, not all differences between PRE and POST can be attributed to learning. RUN may result in many changes (e.g. increases in firing rate) that are not related to the particular spatial sequences experienced. However, we demonstrate that random sequences do not result in POST matching (Figure 4.5C). Thus the observed matching between POST and the actual RUN sequences cannot simply be explained by phenomena such as general changes in firing rates (which would have affected random sequences similarly). Furthermore, we demonstrate that reverse RUN sequences do not result in POST matching. Thus the matching between POST and the RUN sequences cannot be explained by non-sequence-specific increases in the coactivity of cells with nearby place fields. Therefore, RUN sequence structure is specifically present in POST. Moreover, we demonstrate that the longer sequences in POST cannot be accounted for by the occasional, random chaining together of shorter Pair sequences. In contrast, PRE shows no evidence of RUN sequence structure of any length (from Pairs to longer sequences) or at any time scale beyond that expected by chance (whether chance is based on the assumption of equally likely permutations, or on the distribution of matching of random sequences to PRE) (Figures 4.4, 4.5A-C, 4.7D). Therefore, we conclude that long spatial sequences experienced during behavior (i.e. the separate POS and NEG place field sequences) are indeed encoded precisely (i.e. preserve relative firing order) and rapidly (i.e. within a single RUN session). These memories can be decoded by applying our 2-parameter parsing and match probability method to the population activity of CA1 pyramidal neurons during SWS. Proper decoding requires that the activity of multi-spike bursts from individual neurons be treated as

single events. (Thus our results suggest that these first spikes, determined with max_isi's as large as 50 msec, might be particularly important for indicating information about temporal order.) The intermittent, time-compressed sequences found with this method likely represent all the RUN sequence structure present in POST SWS (Figure 4.7D).

Our combinatorial match probability method for analyzing relative firing order among an arbitrary number of neurons is general and can be applied to other types of neural data (and it is not just limited to spike data, e.g. it can be applied to EEG events from multiple locations). All that is necessary is a preliminary parsing step appropriate for each type of data. Here, since hippocampal CA1 pyramidal cell activity in SWS consists of brief, intermittent population bursts, each burst naturally provided a word. In the case of an experiment which consists of a series of trials, each trial might provide a word.

Previous work has shown traces of hippocampal activity from RUN in sleep using various methods (Pavlides and Winson, 1989; Wilson and McNaughton, 1994; Skaggs and McNaughton, 1996; Kudrimoti et al., 1999; Nádasdy et al., 1999; Louie and Wilson, 2001). Skaggs and McNaughton, whose analysis was limited to pair interactions, showed a tendency for the average firing order between pairs of cells to be conserved from RUN to POST and also (but less so) from PRE to RUN, although they did not discuss possible timescale differences between RUN and sleep (Skaggs and McNaughton, 1996). Here we find that the POST pairwise sequence effect is small (0.52 versus an expected 0.5, which is similar in magnitude to that found using their method), and that it cannot explain the much stronger effect (> 3 times expected) of the high occurrence of longer sequences. Using a non-spatial running wheel task, Nádasdy et al. showed that many triplet spike sequences were common to PRE, RUN, and POST (Nádasdy et al., 1999). Unlike our place field sequences, their RUN spike sequences were not clearly related to any specific behavioral events. Furthermore, while they noted that their sleep sequences generally occurred at a faster timescale than the RUN sequences, no compression factor was computed. Our work differs from both these studies by analyzing longer sequences (i.e. lengths of 4 and more, versus pairs and triplets). In addition, we demonstrate that the specific sequences experienced in RUN are strongly present in POST, but completely absent (i.e. match/trial ratios below expected based on chance; i.e. Z scores < 0, Figure 4.5A) in PRE, thus implying the RUN sequences were rapidly encoded during that RUN itself. In contrast, the other studies reported either a small POST effect along with a small amount of RUN structure in PRE (Skaggs and McNaughton, 1996), or a larger POST effect along with a larger amount of RUN structure in PRE

(Nádasdy et al., 1999). Strong, specific matching in POST combined with no matching in PRE makes it unlikely that we are merely observing intrinsic features of the network conserved across all of PRE-RUN-POST and unrelated to learning in RUN.

What could be the mechanisms by which these RUN sequences are encoded? Known plasticity rules require precisely ordered activity occurring within short time windows (Levy and Steward, 1983; Gustafsson et al., 1987; Markram et al., 1997; Bi and Poo, 1998). Presynaptic activity must precede postsynaptic activity by less than approximately 20 msec to yield strengthening of connections, while the reverse order yields weakening. To encode spatial sequences experienced at the behavioral timescale (1 sec) using plasticity mechanisms that operate at this shorter timescale (10 msec), hippocampal phase precession has been proposed as a possible bridging mechanism (O'Keefe and Recce, 1993; Blum and Abbott, 1996; Skaggs et al., 1996; Mehta et al., 2002). Connections may be modified within the hippocampal CA3 network (Blum and Abbott, 1996; Wallenstein and Hasselmo, 1997), with encoded sequences being replayed by triggering a cascade of activity (August and Levy, 1999). Alternatively, the same changes in the CA3-CA1 feedforward network which produce asymmetric place fields (Mehta et al., 2000) may be involved. Asymmetric drive translated into latency could both encode (Mehta et al., 2002) and replay RUN sequences. There is evidence that the sharp-wave/ripple events which generally accompany SWS population bursts (Figures 4.6B,C) are generated by the hippocampus itself (Buzsáki et al., 1983). If so, the SWS sequences we observe might predominantly reflect the intrinsic connectivity of the hippocampus, thus suggesting the sequential experience was indeed encoded within the hippocampus (versus in external structures only).

Recently, long stretches (up to 2 min) of RUN sequential structure have been observed during rapid eye movement (REM) sleep at timescales similar to the original experience (CF = 0.7) (Louie and Wilson, 2001). While the highly-compressed, 100-millisecond-timescale SWS sequences could be replayed via intrinsic hippocampal mechanisms, the extended replay in REM likely involves activation of a broader network which can sustain regular activity for several minutes. Unlike the SWS sequences, REM replay was prominent in PRE, representing spatial experience encoded at least a day before. The SWS sequences we find immediately after only one RUN session may represent rapid learning of discrete episodes (POS, NEG) by the hippocampus, while REM replay could represent a later stage of memory processing involving longer stretches of experience and additional brain areas.

Sleep may play an important role in learning (Buzsáki, 1989; Pavlides and Winson, 1989; Wilson and McNaughton, 1994; Skaggs and McNaughton, 1996; Kudrimoti et al., 1999; Nádasdy et al., 1999; Dave and Margoliash, 2000; Stickgold et al., 2000; Frank et al., 2001; Louie and Wilson, 2001). In addition to revealing asymmetric modifications to the hippocampal synaptic weight matrix, replay of sequences learned in behavior may indicate a process of memory consolidation. The occurrence of low probability matches cannot be accounted for by pairwise biases in firing order, indicating that hippocampal replay consists of brief bursts of long sequences that occur intermittently, rather than a more continuous replay of pairwise biases. These intermittent, time-compressed sequences may represent discrete packets of learned information being broadcast by the hippocampus to modify target structures elsewhere in the brain (Buzsáki, 1989; Pavlides and Winson, 1989; Wilson and McNaughton, 1994; Skaggs and McNaughton, 1996; Siapas and Wilson, 1998; Kudrimoti et al., 1999; Nádasdy et al., 1999), with the precise ordering and timescale (10 msec) required by plasticity mechanisms (Levy and Steward, 1983; Gustafsson et al., 1987; Markram et al., 1997; Bi and Poo, 1998).

Experimental Procedures

Electrophysiology

Spikes from many individual CA1 pyramidal neurons (RAT1: 57 cells, RAT2: 49 cells, RAT3: 26 cells) were recorded simultaneously from each rat (adult male Long Evans) using a microdrive array containing 4-8 independently-adjustable tetrodes and placed above the right dorsal hippocampus (3.6 mm posterior, 2.4 mm lateral with respect to bregma). Tetrodes were first lowered over 1-2 weeks to the CA1 pyramidal cell layer. Then fine adjustments to maximize the number of isolatable pyramidal cells on each tetrode were made, with net movement limited to 20-120 μm per day. For stability, recordings were made no less than 12 hours after the last adjustment. Individual cells were isolated from each tetrode (Figure 4.1) via spike waveform clustering using XCLUST (M.A.W.) (Wilson and McNaughton, 1993). Spike times were recorded with precision 0.1 msec. Rat head position (resolution 0.8 cm) and head direction were sampled at 30 Hz. Hippocampal CA1 pyramidal cell layer EEG (filtered between 1-475 Hz and sampled at 1.5 kHz) was recorded from each tetrode. All procedures were performed in accordance with institutional guidelines.

Behavioral details

For several days before recording, rats were trained on various linear tracks to run back and forth (without stopping in the middle of the track) for chocolate reward at each end. The last training session occurred no less than 1 day before recording. For the "Novel" rat (RAT2), RUN constituted the first

exposure to that enviroment. Of the 2 "Familiar" rats, for RAT1 the immediately preceding exposure to the RUN environment occurred 5 days before (with no intervening training in any environment), while for RAT3 it occurred 1 day before (along with an intervening training session in a different environment on that same preceding day). For PRE and POST, rats were left in a 50(l) x 50(w) x 75(h) (cm) black box, open at the top and lighted from above. Rats were placed on a 20 cm diameter padded circular dish centered and raised 40 cm above the box floor so that movement was restricted and the side walls could not be reached. There the rats' behavior consisted of sleeping, grooming, or remaining still while awake. After PRE, the rat was immediately moved to the track for RUN, then immediately returned to the box for POST.

Determination of RUN place field sequences

Only a fraction of CA1 pyramidal cells have place fields in any given environment (e.g. a track). To select the cells used to determine the POS and NEG direction place field sequences, we applied a set of criteria to each cell twice, once for each direction. Thus each cell could be in 0, 1, or both sequences. Specifically, to determine which cells would be included in a given direction's sequence, we first eliminated all spikes fired at the track ends (because of variable behavior and behavioral state there) or when the rat stopped moving in the middle of the track (which was very infrequent). With the remaining spikes, we took all pyramidal cells firing ≥ 1 spike/lap (mean) in that direction, then eliminated those cells that either exhibited double-peaked place fields within that direction (thus no unique position in sequence) or which stopped firing in that direction for the middle or last 1/3 of the RUN session (indicating unreliable firing). We then computed the smoothed 1-dimensional directional firing rate field of the remaining spikes of each remaining cell. The order of peaks was the same whether smoothing with a Gaussian of $\sigma = 5$, 10, or 15 cm. For the 6 directions in the 3 rats, a total of 64 cells (46 of them distinct) fired ≥ 1 spike/lap, of which 4 had double-peaked fields and 9 stopped firing for the middle and/or last 1/3 of the session, leaving 51 cells (40 distinct). Table 4.1 shows the number of these cells in the POS and NEG sequences of each rat. The POS and NEG sequences of a rat were generally not reversed versions of each other because many cells had place fields in only one direction, and because those cells with place fields in both directions did not necessarily have both fields in the same location.

Identification of SWS

Sleep, identified based on visual assessment (through videotape) of rat posture during PRE and POST, was partitioned into SWS and REM as follows. REM was identified as periods with an elevated ratio (averaged every 1 sec) of hippocampal EEG power in the theta band (5-12 Hz) to overall power (1-475 Hz). The remainder was classified as SWS. All SWS from PRE and POST was used in our analyses.

Match probability analysis

A word W (which may have repeated letters) contains a (x,y) match to a given sequence S (which cannot have repeated letters) if there exist x+y consecutive letters in W of which at least x letters are in

68

strictly increasing order in S (but not necessarily consecutive in S). For example, if S = 123456789, the word 11377 does not contain a (5,0) match but does contain a (3,0) match, and the word 13436892 contains a (6,1) match. We construct an ordered list L of matches to S (best to worst) as follows. The list starts with (K,0) as the best match, where K = the length of S. Next are (x,y) such that x-y = K-1, ordered by decreasing x. Thus (K,1) is the 2nd best, and (K-1,0) the 3rd best match. Next are (x,y) such that x-y = K-2, ordered again by decreasing x. This continues for x-y ≥ 2. Thus (2,0) is the last match in the list. Matches with x-y < 2 are not considered. This ordering obeys the obvious constraints that (x,y) > (x-1,y) > (x-2,y) > ... and (x,y) > (x,y+1) > (x,y+2) > ... (where ">" means "better than"), and, more generally, obeys the constraint that a word which contains a given match does not necessarily contain a match better than it. This ordering also balances rewarding larger x (longer matches) with penalties for larger y (more interruptions). The best match to S found in a given word W is the best match in L that is found in W. For a given word W of length n letters (not necessarily all distinct), the probability of getting a given match (x,y) from L or better is the fraction of the n! permutations of the letters of W which contain a match (x,y) or better according to L. This provides a unified way of computing the probabilities of different matches in words with differing lengths and letter compositions, including matches with interruptions and matches in words with repeated letters. (A different L in which for any 2 matches $(x_1,y_1) > (x_2,y_2)$ if (a) $x_1 > x_2$, or if (b) $x_1 = x_2$ and $y_1 < y_2$, gives similar results, e.g. for max_isi = 50 msec and max_gap = 100 msec: Low Probability PRE $Z \leq -0.5$, $p > 0.4$; POST $Z \geq 7.8$, $p < 3E-10$. This suggests that our results do not depend on the minute details of how matches are ordered.)

The reason for computing the probability that a word would contain a match as good as *or better than* the best match found in that word, assuming all possible permutations of its constituent letters were equally likely, is this: it is exactly analogous to the statistical procedure of determining a p value, such as in computing the probability that one would observe 80 heads in 100 tosses of a coin, assuming the coin is fair. To evaluate how unlikely this is by chance, one computes the probability of getting 80 *or more* heads assuming the null hypothesis of a fair coin. Here "80 heads" corresponds to the best match found in a word, while "more than 80 heads" corresponds to the better matches.

Determination of significance

Given M Pair matches out of N Pair trials, the normal approximation of the binomial distribution gives Z score = (M - N*P)/sqrt(N*P*(1-P)) where P = 1/2. The same formula applies to Triplets (with P = 1/6), and Low Probability trials (with P = 1/24). For Low Probability trials, we computed a lower bound Z for POST and upper bound Z for PRE, i.e. we were most conservative in testing for matching in POST and non-matching in PRE. To calculate p values, we used the normal approximation for Pairs (since the deviations from expected values were small) and the exact binomial distribution for Low Probability trials (since the deviations were often very large).

Control for pair bias

To correct for pair bias, we recomputed match probabilities as follows. Instead of weighting each permutation equally, we weighted each n-letter permutation of a word by treating each of its $n*(n-1)/2$ letter pairs as an independent trial with bias $B = 0.52 =$ the observed POST Pair match/trial ratio. That is, we weighted each permutation of a word by $B^f*(1-B)^r/H$, where $f =$ the total number of letter pairs in the permutation that were in the same order as in RUN, $r =$ the total number of letter pairs in the permutation that were in the opposite order as in RUN, and $H =$ normalization to make the weights of all the permutations sum to 1. This gives an upper bound of the effect of the pair bias since not all these pairs are independent. As an example, the probability of a quadruplet match (4-letter word with 4 distinct letters, all in order) increased from $1/24$ ($= 0.042$) to 0.054 with this pair bias.

Identification of ripples

Hippocampal EEG was filtered in the ripple band (100-400 Hz), the mean and standard deviation of the absolute value of this filtered trace for all of SWS was computed, and a threshold was set to 5 standard deviations above the mean. Threshold-crossings (times when the absolute value of the filtered trace was above threshold) with inter-event interval ≤ 20 msec were grouped together to form a single interval. Intervals < 20 msec in duration were rejected, and the rest were classified as ripples. Threshold was verified by visual inspection of correspondence between these classified ripples and the original (1-475 Hz) EEG trace. Results in Figure 4.6B were unchanged for several different values of threshold.

CF analysis of Figure 4.7A

Given a Low Probability match (x,y), we took the x letters in the match, then determined the pairwise CF for each of the $x*(x-1)/2$ pairs of those letters by dividing the time interval between the pair of letters in that match into the time interval between the corresponding pair of place field peaks in the RUN sequence (e.g. Figures 4.2B,G in terms of time). The CF for the match was taken to be the median of those $x*(x-1)/2$ pairwise CF values.

Template analysis of CF's

Computing template overlap is described with the following example. Figure 4.2B as a function of time gives a 5.5 second template of RAT1 RUN POS activity. The smoothed firing rate of each of the 10 cells was binned into 55 100 msec bins. To compute the overlap with SWS activity at a certain CF, a window of duration (5.5 sec)/CF from SWS was binned into 10 cells x 55 (100 msec)/CF bins. For each cell the number of spikes in each of the 55 bins was smoothed with a Gaussian of $\sigma = 1$ bin. Then the correlation coefficient between the RUN template and SWS window for each of the 10 cells was computed, summed, then divided by the number of cells (here 10). (This process is similar to that of Louie and Wilson (Louie and Wilson, 2001)).

To compute the overlap around each Low Probability match, the overlap was computed with all windows within +/- 1 window duration (which depended on the CF) of the match's center of mass time, at a resolution of 1/10 of a window duration (i.e. 11 windows total). This accomodated inexact alignment. The overlap score for a given match was the Z score of the maximum of the 11 overlaps with respect to the distribution of the maxima of the 11 overlaps with cell-shuffled versions of the template (i.e. the original template except with the identities of the cells randomly exchanged). The average over all Low Probability matches is shown in Figure 4.7B.

In computing the average overlap over a long segment of time (Figure 4.7D), we took care to do it in a manner that treated each CF equivalently so that the overlap scores could be compared across different CF's. First, to compute the overlap as a function of time during PRE (POST) SWS for a given CF (Figure 4.7C), a sliding window (duration = (RUN template duration)/CF) was moved across PRE (POST) SWS activity in increments of 1/10 of a window duration. At every 10th window, the last 10 windows were grouped together. If at least one of the 10 windows in a group contained a minimum level of activity (here set to 4 cells), that group was considered valid. The Z score (with respect to cell-shuffled templates) of the maximum overlap for a valid group of windows was the overlap score at that time. The overlap averaged over time for all of PRE (POST) SWS for that CF was the mean Z score of all valid groups in PRE (POST) SWS. The overlap for all of POST SWS (for a given CF) except for the times around a set of matches M was the mean Z score of all valid groups in POST SWS except for a small fraction of groups: if the window with maximum overlap for a valid group covered the center of mass time of letters of a match from M, then that group's Z score was excluded.
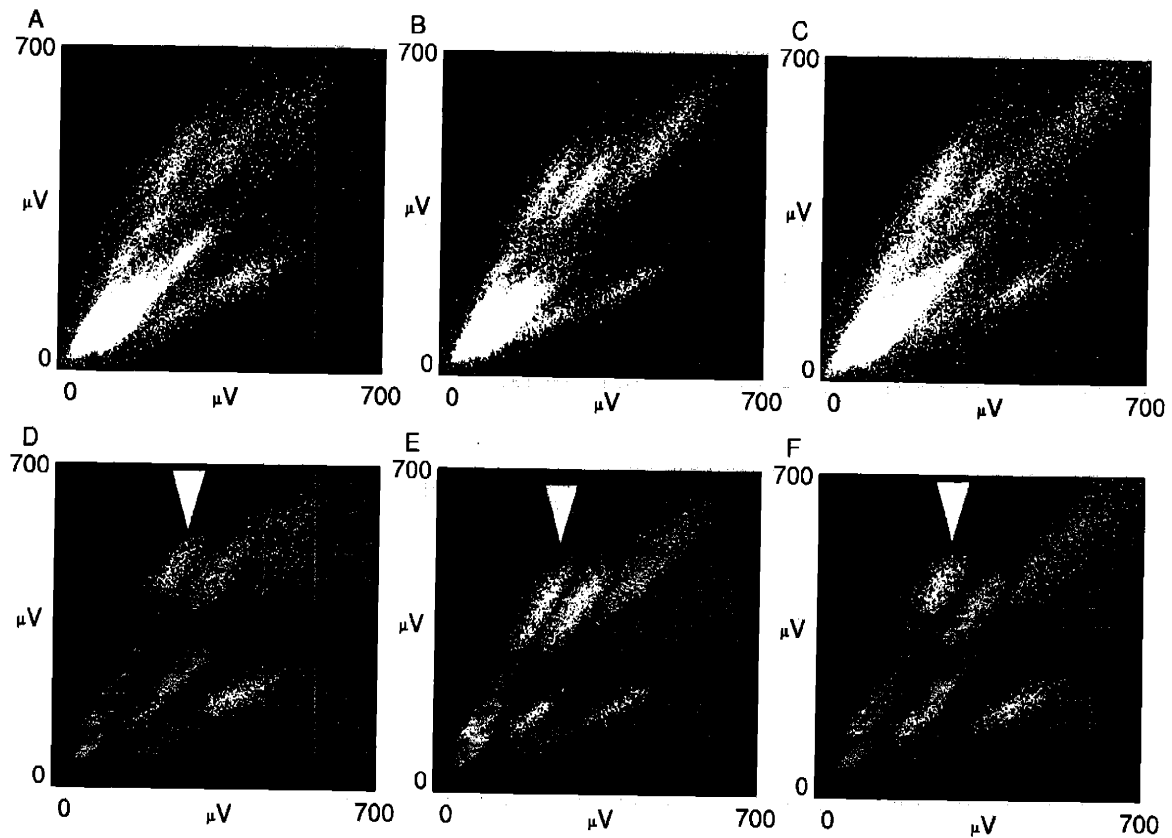
Figure 4.1 Recording of spike data from individual CA1 neurons. **(A)-(C)** Example of raw data from one tetrode (of RAT1) from which the spikes of several individual CA1 neurons are then clustered. Each point represents the peak amplitude (in microvolts) of a triggered waveform on 2 of the 4 tetrode channels (each tetrode consists of 4 electrodes bundled closely together). All triggered points shown; no noise has been filtered out yet. **(D)-(F)** Clusters from **(A)-(C)**, respectively. Clustering is done using waveforms from all 4 channels. Clarity and stability of clusters across the nearly 4 hours of PRE **(A,D,** 78 min), RUN **(B,E,** 23 min), and POST **(C,F,** 112 min.) allow unambiguous identification of the spikes of individual neurons across these 3 periods. Arrow points to cell 6 in (Figures 4.2A-F).

73

smoothed fields. Smoothed firing rate (Hz) at these peaks shown to the right. Non-uniform time axis below shows time within average lap when above positions were passed. **(C)** A population burst from RAT1 POST SWS, showing 6 cells in a row firing in the same order as the POS sequence from RUN **(B)**. Note difference in timescale. **(D)-(F)** More examples of RAT1 POST SWS population bursts that match the RUN POS sequence. **(G)** Same as **(B)**, except for RAT2 POS (rat running in direction of increasing position values). **(H)-(J)** RAT2 POST SWS population bursts that match the RUN POS sequence **(G)**. Words extracted from activity in **(C)-(F)**, **(H)-(J)** using max_isi = 50 msec and max_gap = 100 msec in upper left corner of each panel (with cell 10 represented in words by the letter "A"). Bar = 50 msec.

A

895A   2   7   325789A   2

cell: 10 9 8 7 6 5 4 3 2 1

0   1   2   3   sec
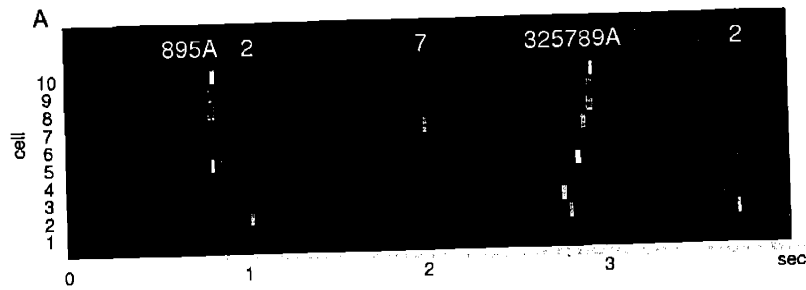
B   SEQUENCE MATCH ANALYSIS OF 325789A

BEST MATCH FOUND : (6,0) = 6-IN-A-ROW : 325789A
PERMUTATIONS WHOSE BEST MATCH IS :

(7,0)   235789A
(6,0)   23578A9   23579A8   23589A7   23789A5
        25789A3   35789A2   325789A   523789A
        723589A   823579A   923578A   A235789

$$P = \frac{\text{\# PERMUTATIONS WITH (6,0) MATCH OR BETTER}}{\text{TOTAL \# OF PERMUTATIONS}} = \frac{13}{7!} = 0.0026$$

Figure 4.3 Parsing and match probability method. **(A)** POST SWS activity of the 10 cells from RAT1 RUN POS sequence (Figure 4.2B). This illustrates the characteristic activity of CA1 pyramidal cells in SWS: short population bursts separated by larger gaps with little activity. Words extracted using max_isi = 50 msec and max_gap = 100 msec shown above each burst. The activity below "325789A" is shown at an expanded time scale in Figure 4.2C. **(B)** Example probability calculation for matching of word 325789A (Figure 4.2C) to RUN sequence 123456789A (Figure 4.2B). Letters in matches in red.
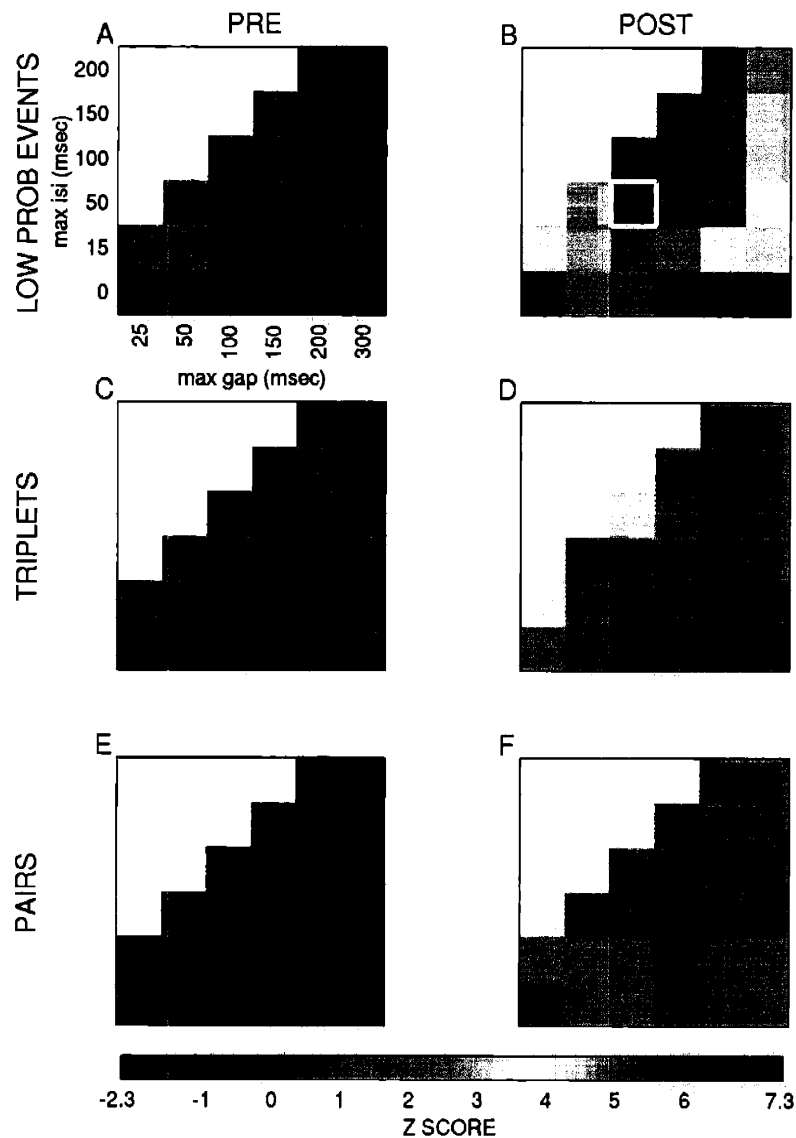
Figure 4.4 Significance (Z score) of matching of PRE and POST SWS activity to RUN spatial sequences as a function of the 2 parsing parameters max_isi and max_gap, pooled over all 3 rats. These results show that PRE SWS bursts have no similarity to the RUN sequences for all parsing parameter values, while POST SWS bursts exhibit significant similarity to the RUN sequences. This suggests that the sequential spatial experience was encoded during RUN. POST similarity is most significant in the longer bursts (i.e. Low Probability trials), and this is the case for a wide range of parameter values. Absence of sequence similarity in POST for max_isi = 0 shows that it is necessary to treat multi-spike bursts from a cell as single events in order to decode sequential activity in SWS. White box in (**B**) indicates maximum POST Low Probability match/trial ratio and Z score. This occurs at max_isi = 50 msec and max_gap = 100 msec.

All PRE and POST SWS matches to RUN sequence of probability ≤ 0.1. Log scale on abscissa. **(C)** Distributions of PRE (below, red) and POST (above, black) SWS Low Probability Z scores with respect to randomly shuffled RUN sequences. Thus high Z scores do not occur by chance. Vertical lines: PRE (red) and POST (black) Z scores for actual RUN sequences. Predicted distributions for PRE (below, blue) and POST (above, green) assuming each Low Probability trial is an independent trial with equally likely permutations. **(D)** Control for robustness of POST SWS Low Probability sequence matching effect with respect to imperfect clustering. Match/trial ratios +/- 1 standard error after systematic elimination of particular spikes (see text). Solid and dotted horizontal lines represent original and expected (1/24) POST Low Probability match/trial ratios, respectively. Sequence effect highly significant under every manipulation.

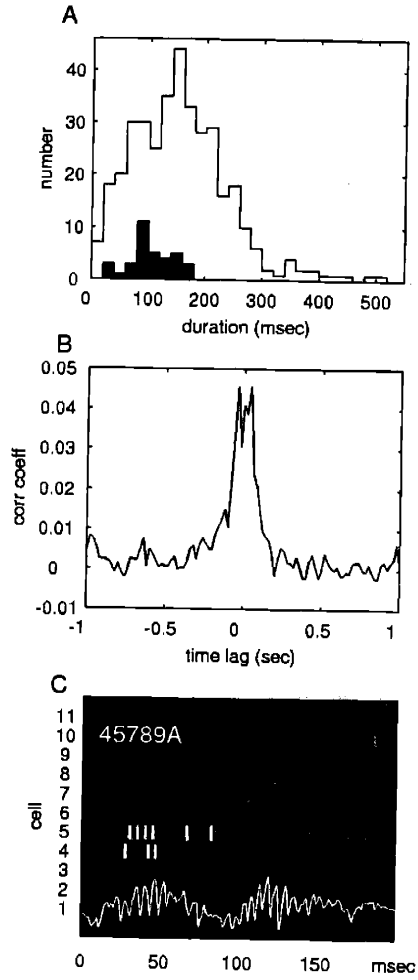**Figure 4.6** Low Probability trials and ripple events. **(A)** Durations of all Low Probability trials (line) in PRE and POST SWS, and matches (solid) in POST SWS. **(B)** Timing of ripples (midpoint) with respect to PRE and POST SWS Low Probability trials (center of mass time of letters) (for RAT1 and RAT2 only). Binsize = 20 msec. **(C)** Example showing timing of ripples about a POST SWS Low Probability match (from RAT1 NEG).

peak occurs around CF = 16. **(C)** Example showing overlap between POS sequence behavioral template (Figure 4.2G in terms of time) and a segment of POST SWS for RAT2 as a function of time. Unlike in **(B)**, the overlap here is computed continuously in time, not just around Low Probability matches. Red horizontal lines indicate significant matching which occurs intermittently. Note that **(D)** is essentially created by collapsing these results (plus all the other SWS data) vertically. **(D)** Overlap as function of CF averaged over all of PRE SWS, all of POST SWS, and POSTX (all of POST SWS except for those times immediately around matches of probability ≤ 1/6). This shows that the only significant RUN sequence structure found in PRE or POST SWS occurs in POST at CF's of around 20, and that this structure can be fully accounted for by the sequences found using our parsing and match probability method. All matches referred to in this figure extracted using max_isi = 50 msec and max_gap = 100 msec.

	RAT1	RAT2	RAT3
Recording duration (min)			
PRE	78	55	37
RUN	23	24	19
POST	112	51	31
Amount of sleep (min)			
PRE	16	45	24
POST	50	36	20
Amount of SWS (min)			
PRE	15	35	22
POST	47	33	19
Track length (cm)	450	180	180
Number of POS laps	30	23	35
Number of NEG laps	29	23	36
Number of cells in POS sequence	10	8	5
Number of cells in NEG sequence	11	9	8
PRE SWS PA+T+LP trials/min	16.1	32.4	15.6
POST SWS PA+T+LP trials/min	10.9	32.9	9.7
PRE SWS Z scores			
Pair	0.1	-1.5	-1.2
Triplet	-1.9	-1.0	-0.7
Low Probability	≤ -0.9	≤ -0.1	≤ -0.3
POST SWS Z scores			
Pair	0.0	1.3	1.4
Triplet	0.9	1.4	3.4
Low Probability	≥ 4.4	≥ 4.8	≥ 4.7
POST SWS Low Probability p value	$< 3E-4$	$< 6E-5$	$< 4E-3$

Table 4.1 Individual data for each of the 3 sessions (1 for each of 3 rats). All the Pair (PA), Triplet (T), and Low Probability (LP) trial results are for the case of max_isi = 50 msec and max_gap = 100 msec. Maximum interval between PRE, RUN, and POST was 3 min.

CHAPTER 5
A BRIEF CONCLUSION

We have developed a new method for analyzing complex temporal firing patterns across large numbers of neurons and shown that it can reveal meaningful structure in real data. In particular, we used the phenomenon of rodent place cells to electrophysiologically study the learning of sequences in the hippocampus. The high spatial selectivity of place cell firing allowed us to represent a rat's experience in moving from one place to another as a sequence of place fields. We then applied our analysis method to show evidence of the replay of extended spatial sequences from previous experience in the rat hippocampus during subsequent SWS. These replayed sequences represent direct neural evidence of the rapid encoding of sequential spatial experience. Our success in detecting significant traces of RUN sequences in POST SWS is a validation of our combinatorial method of analyzing sequential firing patterns based on relative time order.

The first step of our method involved parsing the activity of hippocampal CA1 pyramidal cells in SWS. We chose to parse the activity in a natural manner by dividing it into a series of population bursts that corresponds to the irregular, bursty structure of the hippocampal EEG during SWS. What do these population bursts represent? There is evidence that such bursts are generated intrinsically within the hippocampus itself (Buzsáki et al., 1983), and thus may reflect the underlying synaptic matrix of the hippocampus. Therefore, the strong presence of the RUN place field sequences in these population bursts immediately after, but not before, the experience may indicate that these RUN sequences were encoded via corresponding asymmetric changes to this synaptic weight matrix. Our parsing and sequence match analysis method—in particular using the parameter values max_isi = 50 msec and max_gap = 100 msec—might thus represent a way to decode the hippocampal synaptic weight matrix. This suggests that we could then apply our method (using these parameter values) to hippocampal SWS activity before and after *any* experience and try to determine *through neural activity alone* what has been learned by the hippocampus.

More generally, regardless of what mechanisms generate the words (e.g. a "passive read-out" of the synaptic weight matrix or, alternatively, a more "active" process), the words produced by parsing any SWS activity using max_isi = 50 msec and max_gap = 100 msec may contain important information. That is, since our method has decoded clearly meaningful patterns in at

least one case (i.e. words directly related to previous sequential experience), this same method may help us decode SWS activity in general.

We could begin by asking what the remaining POST SWS words represent. Though there is a strong presence of RUN sequence structure in more than 10% of the population bursts (a proportion that is far greater than expected based on chance), the content of the other 90% of population bursts remains un-decoded. The corresponding un-decoded words look like 539228, as opposed to 134568. Perhaps they just represent random noise in the network. Perhaps they represent more complex processing (e.g. "twisting and turning") of the RUN sequences. Or perhaps they represent other sequences learned during other experiences. Any of these, and more, are possible. Different experiments and different methods of analysis may be necessary. In any case, our ability to discover new, important, and potentially complex interactions involving large numbers of neurons will be limited by the analysis methods we develop.

APPENDIX A1

PROOFS THAT H AND D MATCH ORDERINGS OBEY CONSTRAINT (*)

(This appendix is for Chapter 3)

Recall from Chapter 3 the constraint (*) on ordered match lists: the ordering of matches must be such that a word which contains a worse match must not automatically also contain a better match. Also refer to Chapter 3 for the descriptions of the H and D orderings.

Given a word of length n with k distinct letters, we can assume without loss of generality that $N = k$, where N = the length of the reference sequence. Then the k distinct letters are 1, 2, 3, ..., k with corresponding multiplicities $m_1, m_2, m_3, ..., m_k$. That is, **m** is a length k vector whose entries give the number of occurrences of each of the distinct letters in the word, and $\sum_i m_i = n$.

H Ordering

To prove the H ordering obeys the constraint (*), we need to show that for any match in the H ordered match list, and for any multiplicity vector **m** (i.e. any arbitrary set of letters) which could be arranged to contain such a match as well as any set of matches better than it according to the H ordering scheme, there exists a permutation which contains that match but none of those better than it. For any (x,y) match in the H ordered match list except (2,0) (i.e. any (x,y) match with $x \geq 3$), construct a permutation as follows.

Start with the x letters k-x+1, k-x+2, ..., k and arrange them in order. They constitute the matching letters (and are shown below inside the [ ]'s). From the remaining n-x letters, let I = the y lowest letters (not necessarily distinct), i.e. 1...1 2...2 ... (For instance, if the number of 1's among the remaining n-x letters is $\geq$ y, then I consists of y 1's.) Divide the remaining n-x-y letters into G_1 = the letters $\geq$ k-x+1 and G_2 = the letters < k-x+1. Place the letters in G_1 in decreasing order in front of the matching letters, and the letters in G_2 in decreasing order behind the matching letters:

$$\text{A} \quad \text{B}$$
$$\downarrow \quad \downarrow$$

k...k k-1...k-1 ... k-x+1...k-x+1 [k-x+1 k-x+2 ... k-2 k-1 k] k-x...k-x ... 2...2 1...1

Then insert the letters of I between the matching letters as follows. Insert any k's where A is, and all the other letters in decreasing order where B is. This permutation contains an (x,y) match but no match better than it, since it contains no match (u,v) with u > x and no match (u,v) with u = x and v < y.

If (x,y) = (2,0), then simply construct the following permutation:

k...k k-1...k-1 ... 2...2 1...1 2

This permutation contains a (2,0) match but no match better than it, since it contains no match (u,v) with u ≥ 3.

D Ordering

To prove the D ordering obeys the constraint (*) is more complicated. As above, we need to show that for any match in the D ordered match list, and for any multiplicity vector **m** whose letters could be arranged to contain such a match as well as any set of matches better than it according to the D ordering scheme, there exists a permutation which contains that match but none of those better than it. For any (x,y) match in the D ordered match list (i.e. (x,y) with x-y ≥ 2), construct a permutation as follows.

Start exactly as in the H ordering proof:

k...k k-1...k-1 ... k-x+1...k-x+1 [k-x+1 k-x+2 ... k-2 k-1 k] k-x...k-x ... 2...2 1...1

The difference is where we insert the letters of I. There are x-1 gaps between adjacent matching letters. Since x-y ≥ 2, there is at least one more gap than letters in I. We will insert no more than one letter from I in each gap, and we will insert letters in such a way that each insertion is an interruption. Because we will add y interruptions between x matching letters with no more than one interruption per gap, this will leave no matches (u,v) such that u-v > x-y. Furthermore, we will insert the letters such as to leave no match (u,v) with u > x. Together, these will satisfy the constraint.

Here is the insertion scheme. First, take all the letters in I which are $\leq$ k-x+1 and place them in increasing order starting from the far right, one per gap. Say there are s such letters (labeled in increasing order $i_1,...,i_s$), then we have:

$$
\begin{array}{ccccccccc}
i_s & & i_{s-1} & & & & i_2 & i_1 \\
\downarrow & & \downarrow & & & & \downarrow & \downarrow \\
\text{k-x+1} & \text{k-x+2} & ... & \text{k-s-1} & \text{k-s} & \text{k-s+1} & \text{k-s+2} & ... & \text{k-2} & \text{k-1} & \text{k}
\end{array}
$$

With the remaining x-s matching letters, insert the remaining letters of I (relabeled $i_1,...,i_{y-s}$ in increasing order) in the following manner:

$$
\begin{array}{l}
\text{(3) insert } i_3 \text{ if } i_3 \leq \text{k-s-3} \\
\quad \text{(2) insert } i_2 \text{ if } i_2 \leq \text{k-s-2} \\
\qquad \text{(1) insert } i_1 \text{ if } i_1 \leq \text{k-s-1}
\end{array}
$$

$$
\begin{array}{cccccc}
\text{k-x+1} & \text{k-x+2} & ... & \text{k-s-3} & \text{k-s-2} & \text{k-s-1} & \text{k-s}
\end{array}
$$

Start with (1) (where i_1 is the smallest remaining letter), then (2), et cetera, until the required condition is not met. After the first time the condition is not met, insert the remaining letters in the following manner:

$$
\begin{array}{l}
\text{(1) insert } i_{y-s} \\
\quad \text{(2) insert } i_{y-s-1} \\
\qquad \text{(3) insert } i_{y-s-2}
\end{array}
$$

$$
\begin{array}{ccccc}
\text{k-x+1} & \text{k-x+2} & \text{k-x+3} & \text{k-x+4} & ... & \text{k-s}
\end{array}
$$

Start with (1) (where i_{y-s} is the largest remaining letter), then (2), et cetera, until there are no letters left in I. This is the end of the construction.

For example, say (x,y) = (8,6), we have [k-x+1 k-x+2 ... k-2 k-1 k] = [2 3 4 5 6 7 8 9], and I contains 1 1 3 5 8 9. The first insertion step gives [2 3 4 5 6 7 1 8 1 9]. With the remaining x-s letters [2 3 4 5 6 7], we insert the remaining letters 3, 5, 8, 9 of I to give [2 9 3 8 4 5 5 6 3 7].

The permutation so constructed contains an (x,y) match but no match better than it, since it contains no match (u,v) with u-v > x-y and no match (u,v) with u-v = x-y and u > x.

89

APPENDIX A2

METHODS FOR COMPUTING EXACT, UPPER BOUND, AND LOWER BOUND MATCH PROBABILITIES ASSUMING ALL PERMUTATIONS ARE EQUALLY LIKELY

(This appendix is for Chapter 3)

Recall from Chapter 3 the definition of match probability: given a particular reference sequence and an arbitrary word of length n, the probability of a (x,y) match—assuming all permutations are equally likely—is the fraction of the n! permutations of that word which contain a (x,y) match. Also refer to Chapter 3 for the definition of a (x,y) match.

Given a word of length n with k distinct letters, we can assume without loss of generality that N = k, where N = the length of the reference sequence. Then the k distinct letters are 1, 2, 3, ..., k with corresponding multiplicities $m_1, m_2, m_3, ..., m_k$. That is, **m** is a length k vector whose entries give the number of occurrences of each of the distinct letters in the word, and $\sum_i m_i = n$.

Formula A2-1: Exact Probability of a (k,0) Match

Given any multiplicity vector **m** of length k, where k = the number of distinct letters, the exact probability that a random permutation contains a (k,0) match, assuming all n! permutations of the n letters are equally likely, can be computed as follows.

Let M = the minimum entry in **m**, i.e. the multiplicity of the letter with smallest multiplicity. Thus each letter occurs at least M times. Because we have assumed **m** is of length k, M ≥ 1. Each permutation will contain exactly 0, 1, 2, ..., or M (k,0) matches. Out of the n! total possible permutations, let N(i) = the number of permutations simultaneously containing exactly i (k,0) matches. Therefore the exact probability we seek is given by (N(1) + N(2) + ... + N(M))/n!.
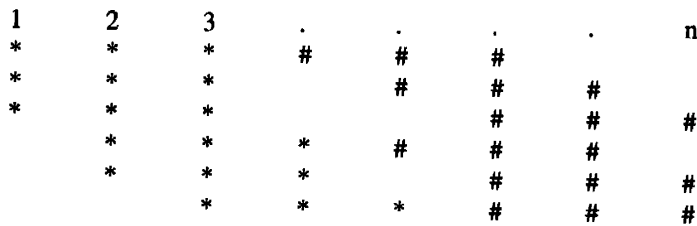
Consider trying to compute N(1). Out of the n letter positions for each permutation, there are n-k+1 unique sets of k consecutive positions in which to place a (k,0) match:

```
1    2    3    .    .    .    .    n
*    *    *    *
     *    *    *    *
     *    *    *    *
          .         .    .    .
          *    *    *    *
```
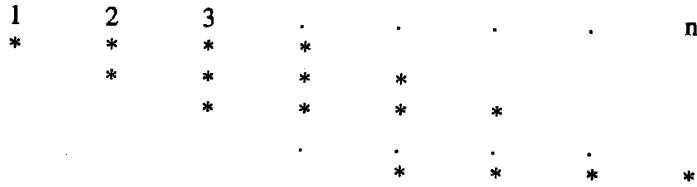
Since each distinct letter is used exactly once in each (k,0) match, there are $\prod_i m_i$ ways to choose which of the k letters (from among the repeats) to use in the (k,0) match. The remaining n-k letters can be arranged in any of (n-k)! ways. This gives a total of $(n-k+1)*(\prod_i m_i)*(n-k)!$ permutations with at least 1 (k,0) match. If M = 1, this gives N(1) and we are done. However, if M > 1, this overcounts N(1) because there will be permutations containing more than 1 (k,0) match, and this formula will count a permutation containing exactly R (k,0) matches exactly R times instead of once. Thus let $(n-k+1)*(\prod_i m_i)*(n-k)!$ be represented by N(1+), where N(1+) = N(1) if M = 1. Then:

$$N(1+) = 1*N(1) + 2*N(2) + 3*N(3) + ... + R*N(R)$$

Now how do we calculate N(R) for R > 1? For N(2), for example, we use an analogous procedure to that for N(1+). First we need to count the number of unique sets of (non-overlapping) pairs of k consecutive positions in which to place 2 (k,0) matches. For n = 8 and k = 3 there are 6 such pairs:

```
1    2    3    .    .    .    .    n
*    *    *    #    #    #
*    *    *         #    #    #
*    *    *              #    #    #
     *    *    *    #    #    #
     *    *    *         #    #    #
     *    *    *    #    #    #
```

Call the number of such sets Q(n,k,2). We must then multiply this by $(\prod_i m_i)*(\prod_i(m_i-1))$, which gives us the number of ways to select the 2*k letters (among the repeats) to place in the 2 (k,0) matches. Finally we must multiply this by (n-2*k)! for the number of ways to place the remaining n-2*k letters. Again, if M = 2, this gives us N(2) and we are done. However, if M > 2, this overcounts N(2) because there will be permutations containing more than 2 (k,0) matches, and our formula will count a permutation containing exactly R (k,0) matches exactly C(R,2) times instead of once, where C(a,b) = a!/(b!*(a-b)!). Thus let $Q(n,k,2)*(\prod_i m_i)*(\prod_i(m_i-1))*(n-2*k)!$ be represented by N(2+), where N(2+) = N(2) if M = 2. Then:

$$N(2+) = 1*N(2) + 3*N(3) + 6*N(4) + ... + C(R,2)*N(R)$$

Continuing in this fashion to compute N(3+), ..., N(M+) (i.e. N(q+) =
$Q(n,k,q)*(\prod_i m_i)*(\prod_i(m_i-1))*...*(\prod_i(m_i-q+1))*(n-q*k)!$, where $Q(n,k,q) = C(n-q*(k-1),q))$, we get:

$$N(1+) = 1*N(1) + 2*N(2) + 3*N(3) + ... + C(R,1)*N(R) + ... + C(M,1)*N(M)$$

$$N(2+) = \qquad \quad 1*N(2) + 3*N(3) + ... + C(R,2)*N(R) + ... + C(M,2)*N(M)$$

$$N(3+) = \qquad \qquad \qquad 1*N(3) + ... + C(R,3)*N(R) + ... + C(M,3)*N(M)$$

$$........ = \qquad \qquad ....$$

$$N(q+) = \qquad \qquad \qquad 1*N(q) + ... + C(R,q)*N(R) + ... + C(M,q)*N(M)$$

$$........ = \qquad \qquad \qquad ....$$

$$N(M+) = \qquad \qquad \qquad \qquad \qquad \qquad C(M,M)*N(M)$$

Observe that:

$$N(1+) - N(2+) + N(3+) - N(4+) + ... + (-1)^{M-1}*N(M+)) = N(1) + N(2) + N(3) + N(4) + ... + N(M)$$

therefore our exact probability answer is given by:

$$P = 1/n! * (N(1+) - N(2+) + N(3+) - N(4+) + ... + (-1)^{M-1}*N(M+)) \qquad (A2-1)$$

Formula A2-2: Upper Bound Probability of a (x,y) Match

Given any multiplicity vector **m** of length k, where k = the number of distinct letters, an upper bound of the probability that a random permutation contains a (x,y) match (where $2 \le x \le k$ and $x+y \le n$), assuming all n! permutations of the n letters are equally likely, can be computed as follows. In particular, this formula actually computes an upper bound of the probability that a permutation contains any (u,v) match where $u \ge x$ and $v \le y$.

The formula:

$$P \le 1/n! * X * Y * Z \qquad (A2-2)$$

is a product of 3 terms, where we will explain each term one at a time.

First, X = the number of unique sets of x positions spanning no more than x+y consecutive positions:

$$X = (n-(x+y)+1)*C(x+y-1,y) + \sum_{i=0}^{y-1} C(x+i-1,i), \qquad \text{if } y > 0$$

OR

$$X = (n-(x+y)+1)*C(x+y-1,y), \qquad \text{if } y = 0$$

This formula can be explained as follows. If $y = 0$, then the formula is just $n-x+1$, the number of unique sets of x consecutive positions (as in $N(1+)$ in A2-1 above). If $y > 0$, the first term, $(n-(x+y)+1) * C(x+y-1,y)$, consists of $n-(x+y)+1$ = the number of unique sets of x+y consecutive positions, and $C(x+y-1,y)$ = the number of ways to place the y interruptions in any of the x+y-1 possible positions excluding the first position. By fixing the first position (i.e. not allowing the first position to be occupied by an interruption), we assure that each set of x positions within x+y consecutive positions is counted only once. The second term accounts for the sets of x positions spanning less than x+y positions at the right end of the n positions.

Second, Y = the number of ways to pick x distinct letters (including accounting for the repeats) to put in order for the (x,y) match:

$$Y = \sum_{\substack{\text{all x-tuples } (i_1,i_2,...,i_x) \\ \text{where } 1 \leq i_1 < i_2 < ... < i_x \leq k}} \left\{ \prod_{j=i_1,i_2,...,i_x} m_j \right\}$$

Note that this sum has $C(k,x)$ terms, one for each of the ways to pick x distinct letters out of the k total distinct letters, while the product takes care of the repeats. Also note that if $x = k$ this expression reduces to:

$$Y = \prod_{j=1}^{k} m_j$$

Finally, $Z = (n-x)!$ = the number of ways to arrange the n-x remaining letters in any order in the remaining n-x positions. These n-x positions include any interrupting positions within the match.

The reason this is an upper bound is that it constructs all possible permutations containing a (x,y) match, but by letting the remaining positions be any letter, each permutation may be constructed (and thus counted) more than once. By letting the remaining positions be any letter, our construction also constructs (and counts) all permutations containing a (u,v) match where $u \geq x$ and $v \leq y$.

Formula A2-3: Lower Bound Probability of a (z,0) Match

Given any multiplicity vector **m** of length k, where k = the number of distinct letters, a lower bound of the probability that a random permutation contains a (z,0) match (where $2 \leq z < k$), assuming all n! permutations of the n letters are equally likely, can be computed as follows.

We start by constructing permutations containing at least one (z,0) match. There are n-z+1 unique sets of z consecutive positions,

$$Y = \sum_{\substack{\text{all z-tuples } (i_1,i_2,...,i_z) \\ \text{where } 1 \leq i_1 < i_2 < ... < i_z \leq k}} \left\{ \prod_{j=i_1,i_2,...,i_z} m_j \right\}$$

ways to pick z distinct letters (including accounting for the repeats) for the (z,0) match, and (n-z)! ways to arrange the remaining n-z letters. We call the product, (n-z+1)*Y*(n-z)!, of these 3 terms $N_{fix}(z)$, since it is constructed by fixing z consecutive positions and letting the remaining positions contain any letters. As with N(1+) of A2-1, this counts some permutations more than once.

To see exactly how $N_{fix}(z)$ overcounts, we introduce the following notation. Let $N(w_0;w_1,w_2,...,w_q)$, where $w_0 \leq w_1 \leq w_2 \leq ... \leq w_q$, be the number of permutations (out of the n! total) containing a $(w_1,0)$ match, a $(w_2,0)$ match, ..., a $(w_q,0)$ match, and no other (x,0) matches where $x \geq w_0$, where all these matches are non-overlapping (i.e. share no positions), and where each of these matches are as long as can be. Because of this last condition, for a fixed w_0, no permutation can belong to more than one such set. To see this, consider any permutation. Partition it into blocks of consecutive positions with strictly increasing letters. For instance, 31462322856791311145 is partitioned into 3, 146, 23, 2, 28, 5679, 13, 1, 145, revealing exactly 3 (1,0) matches, 3 (2,0) matches, 2 (3,0) matches, and 1 (4,0) match. If $w_0 = 3$, then this belongs to N(3;3,3,4).

$N_{fix}(z)$ overcounts by multiply counting any permutation that belongs to $N(z;w_1,w_2,...,w_q)$ except for $N(z;z)$. For instance, $N_{fix}(z)$ counts a given permutation in $N(z;z+y)$ exactly $y+1$ times, a permutation in $N(z;z,z,...,z)$ (q z's) exactly q times, and a permutation in $N(z;z+y_1,z+y_2,...,z+y_q)$ (where $0 \le y_1 \le y_2 \le ... \le y_q$) exactly $(y_1+1)+(y_2+1)+...+(y_q+1)$ times.

To correct for this overcounting, we introduce $N_{fix}(z,z)$, which constructs all permutations containing at least 2 non-overlapping $(z,0)$ matches. (Again, by non-overlapping we mean that the 2 $(z,0)$ matches cannot share any positions. For example, 1231134 is counted in $N_{fix}(3,3)$, but 1234 is not.) The number of unique sets of (non-overlapping) pairs of z consecutive positions in which to place 2 $(z,0)$ matches is $1/2*(n-2*z+1)*(n-2*z+2)$. The number of ways to select the $2*z$ letters (accounting for repeats) is:

$$Y' = \sum_{\substack{\text{all z-tuples } (i_1,i_2,...,i_z) \\ \text{where } 1 \le i_1 < i_2 < ... < i_z \le k}} \{\{ \prod_{j=i_1,i_2,...,i_z} m_j \} * \{ \sum_{\substack{\text{all remaining z-tuples} \\ (r_1,r_2,...,r_z) \\ \text{where } 1 \le r_1 < r_2 < ... < r_z \le k}} \{ \prod_{s=r_1,r_2,...,r_z} m'_s \} \} \}$$

which is a nested sum. The reduced multiplicity vector $\mathbf{m'}$ is constructed from the original multiplicity vector $\mathbf{m}$ by subtracting 1 from the multiplicity of each letter selected in the first (outer) z-tuple $(i_1,i_2,...,i_z)$. Then:

$$N_{fix}(z,z) = 1/2*(n-2*z+1)*(n-2*z+2)*Y'*(n-2*z)!$$

Generally, $N_{fix}(z,z)$ also counts some permutations more than once. Note that $N_{fix}(z,z)$ is 0 if no permutation can contain 2 (non-overlapping) $(z,0)$ matches.

We propose that $N_{fix}(z) - N_{fix}(z+1) - N_{fix}(z,z)$ will subtract off all the overcounting of $N_{fix}(z)$ and more, thus giving a lower bound. To see this, consider a given permutation in $N(z;z+y_1,z+y_2,...,z+y_q)$ (where $0 = y_1 = y_2 = ... = y_{q1} < y_{q1+1} \le y_{q1+2} \le ... \le y_{q1+q2}$ and $q_1+q_2 = q$, i.e. q_2 matches are longer than $(z,0)$), which $N_{fix}(z)$ counts exactly $q+y_1+y_2+...+y_q$ times. $N_{fix}(z+1)$ counts this permutation exactly $y_1+y_2+...+y_q$ times, and $N_{fix}(z,z)$ counts it at least $((y_1+y_2+...+y_q+q)^2 - \sum_i((y_i+1)^2))/2$ times. To understand this last expression, consider that each of the q $(z+y_i,0)$ matches, where $y_i \ge 0$, contains y_i+1 (overlapping) $(z,0)$ matches. $N_{fix}(z,z)$ counts this permutation as many times as there are ways to select 2 simultaneous $(z,0)$ matches out of these q $(z+y_i,0)$ matches. How many ways are there to do this? If we restrict ourselves to

selecting only 1 (z,0) match per (z+y_i,0) match, then this amounts to asking what is the sum of the product of all possible pairs of values y_1+1, y_2+1, ..., y_q+1. The square of the sum of these values, $((y_1+1) + (y_2+1) + ... + (y_q+1))^2$ gives us the sum we need, except that we must subtract all the $(y_i+1)^2$ terms, then divide by 2, since the remaining terms have coefficient 2. The reason this undercounts $N_{fix}(z,z)$ is because we have ignored the cases where $y_i \geq z$, in which case one can simultaneously select both (z,0) matches from such a (z+y_i,0) match. This expression has q*(q-1)/2 (i.e. C(q,2)) terms, each $\geq$ 1, thus $N_{fix}(z,z)$ counts such a permutation at least q*(q-1)/2 times.

Thus $N_{fix}(z)$ - $N_{fix}(z+1)$ - $N_{fix}(z,z)$ counts each permutation in N(z;z+y_1,z+y_2,...,z+y_q) no more than ((q+y_1+y_2+...+y_q) - (y_1+y_2+...+y_q) - (q*(q-1)/2)) times. This reduces to q*(3-q)/2, which is never > 1, for any q = 0, 1, 2, ..., i.e. it never overcounts. Therefore the lower bound probability is:

$$P \geq 1/n! * (N_{fix}(z) - N_{fix}(z+1) - N_{fix}(z,z))$$ (A2-3)

Note that by setting z = k (in which case $N_{fix}(z+1)$ = 0), one can see the relationship between this formula and A2-1.

Note

When the H ordering scheme is used and there are no repeated letters (i.e. k = n), one may be able to compute tighter upper and lower bounds of the probability of getting an (x,y) match or better using results from work on longest increasing (and decreasing) subsequences (Schensted, 1961).

APPENDIX A3

ADDITIONAL ANALYSIS OF PARSING RESULTS

(This appendix is for Chapter 4)

Recall from Chapter 4 the parsing of hippocampal CA1 pyramidal cell activity during slow wave sleep (SWS) into a set of words. The parsing procedure depended on 2 parameters: max_isi and max_gap. We then classified these words into 3 different groups: Pairs, Triplets, and Low Probability words. Sequence structure was analyzed in each of these groups separately. However, these 3 groups do not represent all the words from SWS. While these 3 groups constitute the vast majority of trials that could possibly contain sequence structure, there are some remaining words that could also do so. Here we look at the distribution of all the words. We also analyze the RUN sequence structure in the remaining words.

If we characterize each word by its length, n, and the number of distinct letters it contains, k, the parsing values max_isi = 50 msec and max_gap = 100 msec (used for the majority of the analyses in Chapter 4) result in the following distribution of words. Table A3.1 shows all the PRE and POST SWS words parsed from the 3 rats (2 RUN sequences per rat). For each value of n and k, the number of words are displayed (format: PRE, POST).

Table A3.1

		n				
		1	2	3	4	≥ 5
	1	8769,10067	492,552	43,60	4,9	1,0
	2		1371,1255	255,147	45,38	21,8
k	3			255,259	93,78	30,22
	4				43,77	28,53
	≥ 5					24,140

The Pair trials correspond to words with n = 2 and k = 2, Triplet trials to words with n = 3 and k = 3, and Low Probability trials to words with $n \geq 4$ and $k \geq 4$. (Note that the precise definition of a Low Probability trial is not in terms of n and k, but in our data the set of Low Probability trials and the set of words with $n \geq 4$ and $k \geq 4$ are the same.) Thus the sequence analysis of Chapter 4 ignored the following sets of words: words with k = 1, words with k = 2 and n > 2, and words with k = 3 and n > 3.

Now let us systematically go through each of these sets of words and consider the sequence structure they could possibly contain. Clearly, words with $k = 1$ cannot contain any sequence structure. Words with $k = 2$ and $n > 2$ do not contain much sequence information, because most permutations of the letters contain the best possible match, i.e. a $(2,0)$ match. For this reason, we restricted our sequence analysis of words with $k = 2$ to those with $n = 2$, i.e. Pairs. The situation is different for words with $k = 3$ and $n > 3$. Depending on the letter composition, some of these words may contain $(3,0)$ matches with probability $\leq 1/6$. In Chapter 4 we sampled the majority of $k = 3$ words by analyzing all those words with $n = 3$, i.e. Triplets. This was done for simplicity. An analysis of the RUN sequence structure in all $k = 3$ words (which includes all the Triplet trials) gives basically the same results, i.e. no evidence of RUN sequence structure in PRE and moderately significant RUN sequence structure in POST. These results are shown in Figure A3.1, which is identical to Figure 4.5A except for the replacement of Triplet results by those for all "$k = 3$" words. (Because of the heterogeneity of probabilities within the set of $k = 3$ words, for this set of words we computed the upper bound match/trial ratio and Z score for PRE and the lower bound match/trial ratio and Z score for POST.)

Figure A3.1

ACKNOWLEDGMENTS

REFERENCES

Abeles, M., and Gerstein, G.L. (1988). Detecting spatiotemporal firing patterns among simultaneously recorded single neurons. J. Neurophysiol. *60*, 909-924.

Abeles, M. (1991). Corticonics: Neural Circuits of the Cerebral Cortex (New York: Cambridge Univ. Press).

Abeles, M., Bergman, H., Margalit, E., and Vaadia, E. (1993). Spatiotemporal firing patterns in the frontal cortex of behaving monkeys. J. Neurophysiol. *70*, 1629-1638.

Aertsen, A.M.H.J., Gerstein, G.L., Habib, M.K., and Palm, G. (1989). Dynamics of neuronal firing correlation: modulation of "effective connectivity". J. Neurophysiol. *61*, 900-917.

August, D.A., and Levy, W.B. (1999). Temporal sequence compression by an integrate-and-fire model of hippocampal area CA3. J. Comput. Neurosci. *6*, 71-90.

Bi, G.Q., and Poo, M.M. (1998). Synaptic modifications in cultured hippocampal neurons: dependence on spike timing, synaptic strength, and postsynaptic cell type. J. Neurosci. *18*, 10464-10472.

Bliss, T.V., and Collingridge, G.L. (1993). A synaptic model of memory: long-term potentiation in the hippocampus. Nature *361*, 31-39.

Blum, K.I., and Abbott, L.F. (1996). A model of spatial map formation in the hippocampus of the rat. Neural Computation *8*, 85-93.

Buzsáki, G., Leung, L.S., and Vanderwolf, C.H. (1983). Cellular bases of hippocampal EEG in the behaving rat. Brain Res. *287*, 139-171.

Buzsáki, G. (1989). Two-stage model of memory trace formation: a role for "noisy" brain states. Neuroscience *31*, 551-570.

Corkin, S. (1984). Lasting consequences of bilateral medial temporal lobectomy: clinical course and experimental findings in H.M. Seminars in Neurology *4*, 249-259.

Dave, A.S., and Margoliash, D. (2000). Song replay during sleep and computational rules for sensorimotor vocal learning. Science *290*, 812-816.

Dayhoff, J.E., and Gerstein, G.L. (1983). Favored patterns in spike trains. I. Detection. J. Neurophysiol. *49*, 1334-1348.

Eichenbaum, H., Dudchenko, P., Wood, E., Shapiro, M., and Tanila, H. (1999). The hippocampus, memory, and place cells: is it spatial memory or a memory space? Neuron *23*, 209-226.

Fortin, N.J., Agster, K.L., and Eichenbaum, H.B. (2002). Critical role of the hippocampus in memory for sequences of events. Nat. Neurosci. *5*, 458-462.

Frank, M.G., Issa, N.P., and Stryker, M.P. (2001). Sleep enhances plasticity in the developing visual cortex. Neuron *30*, 275-287.

Gerstein, G.L., and Perkel, D.H. (1969). Simultaneous recorded trains of action potentials: analysis and functional interpretation. Science *164*, 828-830.

Gerstein, G.L., Perkel, D.H., Dayhoff, J.E. (1985). Cooperative firing activity in simultaneously recorded populations of neurons: detection and measurement. J. Neurosci. *5*, 881-889.

Gerstein, G.L., and Aertsen, A.M.H.J. (1985). Representation of cooperative firing activity among simultaneously recorded neurons. J. Neurophysiol. *54*, 1513-1528.

Gustafsson, B., Wigström, H., Abraham, W.C., and Huang, Y.Y. (1987). Long-term potentiation in the hippocampus using depolarizing current pulses as the conditioning stimulus to single volley synaptic potentials. J. Neurosci. *7*, 774-780.

Hebb, D.O. (1949). The Organization of Behavior (New York: Wiley).

Kesner, R.P., Gilbert, P.E., and Barua, L.A. (2002). The role of the hippocampus in memory for the temporal order of a sequence of odors. Behav. Neurosci. *116*, 286-290.

Kudrimoti, H.S., Barnes, C.A., and McNaughton, B.L. (1999). Reactivation of hippocampal cell assemblies: effects of behavioral state, experience, and EEG dynamics. J. Neurosci. *19*, 4090-4101.

Lee, A.K., and Wilson, M.A. (2002). Memory of sequential experience in the hippocampus during slow wave sleep. Neuron *36*, 1183-1194.

Lehmann, E.L. (1975). Nonparametrics: Statistical Methods Based on Ranks (San Francisco: Holden-Day, Inc.).

Levy, W.B., and Steward, O. (1983). Temporal contiguity requirements for long-term associative potentiation/depression in the hippocampus. Neuroscience *8*, 791-797.

Louie, K., and Wilson, M.A. (2001). Temporally structured replay of awake hippocampal ensemble activity during rapid eye movement sleep. Neuron *29*, 145-156.

Markram, H., Lübke, J., Frotscher, M., and Sakmann, B. (1997). Regulation of synaptic efficacy by coincidence of postsynaptic APs and EPSPs. Science *275*, 213-215.

Martignon, L., Von Hasseln, H., Grün, S., Aertsen, A., and Palm, G. (1995). Detecting higher-order interactions among spiking events in a group of neurons. Biol. Cybern. *73*, 69-81.

McHugh, T.J., Blum, K.I., Tsien, J.Z., Tonegawa, S., and Wilson, M.A. (1996). Impaired hippocampal representation of space in CA1-specific NMDAR1 knockout mice. Cell *87*, 1339-1349.

Mehta, M.R., Barnes, C.A., and McNaughton, B.L. (1997). Experience-dependent, asymmetric expansion of hippocampal place fields. Proc. Natl. Acad. Sci. USA *94*, 8918-8921.

Mehta, M.R., Quirk, M.C., Wilson, M.A. (2000). Experience-dependent asymmetric shape of hippocampal receptive fields. Neuron 25, 707-715.

Mehta, M.R., Lee, A.K., and Wilson, M.A. (2002). Role of experience and oscillations in transforming a rate code into a temporal code. Nature 417, 741-746.

Meister, M., Wong, R.O., Baylor, D.A., and Shatz, C.J. (1991). Synchronous bursts of action potentials in ganglion cells of the developing mammalian retina. Science 252, 939-943.

Milner, B. (1962). Les troubles de la mémoire accompagnant des lésions hippocampiques bilatérales [Memory impairment associated with bilateral hippocampal lesions]. In Physiologie de l'Hippocampe (Paris: Centre National de la Recherche Scientifique), pp. 257-272. (English translation (1965). In Cognitive Processes and the Brain, P.M. Milner and S. Glickman, eds. (Princeton: Van Nostrand Co., Inc.), pp. 97-111.)

Milner, B. (1966). Amnesia following operation on the temporal lobes. In Amnesia, C.W.M. Whitty and O.L. Zangwill, eds. (London: Butterworths), pp. 109-133.

Milner, B. (1972). Disorders of learning and memory after temporal lobe lesions in man. Clinical Neurosurgery 19, 421-446.

Mishkin, M., Vargha-Khadem, F., and Gadian, D.G. (1998). Commentary: Amnesia and the organization of the hippocampal system. Hippocampus 8, 212-216.

Morris, R.G., Garrud, P., Rawlins, J.N., and O'Keefe, J. (1982). Place navigation impaired in rats with hippocampal lesions. Nature 297, 681-683.

Muller, R. (1996). A quarter century of place cells. Neuron 17, 813-822.

Nádasdy, Z., Hirase, H., Czurkó, A., Csicsvari, J., and Buzsáki, G. (1999). Replay and time compression of recurring spike sequences in the hippocampus. J. Neurosci. 19, 9497-9507.

O'Keefe, J., and Dostrovsky, J. (1971). The hippocampus as a spatial map. Preliminary evidence from unit activity in the freely-moving rat. Brain Res. 34, 171-175.

O'Keefe, J., and Nadel, L. (1978). The Hippocampus as a Cognitive Map (Oxford: Clarendon Press).

O'Keefe, J., and Recce, M. (1993). Phase relationship between hippocampal place units and the EEG theta rhythm. Hippocampus 3, 317-330.

Palm, G., Aertsen, A.M.H.J., and Gerstein, G.L. (1988). On the significance of correlations among neuronal spike trains. Biol. Cybern. 59, 1-11.

Pavlides C., and Winson, J. (1989). Influences of hippocampal place cell firing in the awake state on the activity of these cells during subsequent sleep episodes. J. Neurosci. 9, 2907-2918.

Quirk, M.C., and Wilson, M.A. (1999). Interaction between spike waveform classification and temporal sequence detection. J. Neurosci. Methods 94, 41-52.

Ranck, J.B., Jr. (1973). Studies on single neurons in dorsal hippocampal formation and septum in unrestrained rats. I. Behavioral correlates and firing repetoires. Exp. Neurol. *41*, 461-531.

Schensted, C. (1961). Longest increasing and decreasing subsequences. Canadian Journal of Mathematics *13*, 179-91.

Scoville, W.B., and Milner, B. (1957). Loss of recent memory after bilateral hippocampal lesions. J. Neurol. Neurosurg. Psychiatr. *20*, 11-21.

Siapas, A.G., and Wilson, M.A. (1998). Coordinated interactions between hippocampal ripples and cortical spindles during slow-wave sleep. Neuron *21*, 1123-1128.

Skaggs, W.E., and McNaughton, B.L. (1996). Replay of neuronal firing sequences in rat hippocampus during sleep following spatial experience. Science *271*, 1870-1873.

Skaggs, W.E., McNaughton, B.L., Wilson, M.A., and Barnes, C.A. (1996). Theta phase precession in hippocampal neuronal populations and the compression of temporal sequences. Hippocampus *6*, 149-172.

Squire, L.R. (1992). Memory and the hippocampus: a synthesis from findings with rats, monkeys, and humans. Psychological Review *99*, 195-231.

Squire, L.R., and Zola, S.M. (1998). Commentary: Episodic memory, semantic memory, and amnesia. Hippocampus *8*, 205-211.

Stickgold, R., Whidbee, D., Schirmer, B., Patel, V., and Hobson, J.A. (2000). Visual discrimination task improvement: A multi-step process occurring during sleep. J. Cogn. Neurosci. *12*, 246-254.

Tulving, E. (1972). Episodic and semantic memory. In Organization of Memory, E. Tulving and W. Donaldson, eds. (New York: Academic Press), pp. 381-403.

Tulving, E. (1983). Elements of Episodic Memory (Oxford: Clarendon Press).

Tulving, E., and Markowitsch, H.J. (1998). Commentary: Episodic and declarative memory: role of the hippocampus. Hippocampus *8*, 198-204.

Vargha-Khadem, F., Gadian, D.G., Watkins, K.E., Connelly, A., Van Paesschen, W., and Mishkin, M. (1997). Differential effects of early hippocampal pathology on episodic and semantic memory. Science *277*, 376-380.

Wallenstein, G.V., and Hasselmo, M.E. (1997). GABAergic modulation of hippocampal population activity: sequence learning, place field development, and the phase precession effect. J. Neurophysiol. *78*, 393-408.

Wessberg, J., Stambaugh, C.R., Kralik, J.D., Beck, P.D., Laubach, M., Chapin, J.K., Kim, J., Biggs, S.J., Srinivasan, M.A., and Nicolelis, M.A. (2000). Real-time prediction of hand trajectory by ensembles of cortical neurons in primates. Nature *408*, 361-365.

Wilson, M.A., and McNaughton, B.L. (1993). Dynamics of the hippocampal ensemble code for space. Science *261*, 1055-1058.

Wilson, M.A., and McNaughton, B.L. (1994). Reactivation of hippocampal ensemble memories during sleep. Science *265*, 676-679.

Zhang, L.I., Tao, H.W., Holt, C.E., Harris, W.A., and Poo, M. (1998). A critical window for cooperation and competition among developing retinotectal synapses. Nature *395*, 37-44.

Zola-Morgan, S., Squire, L.R., and Amaral, D.G. (1986). Human amnesia and the medial temporal region: enduring memory impairment following a bilateral lesion limited to field CA1 of the hippocampus. J. Neurosci. *6*, 2950-2967.