Automation of Sleep Staging

by

Jessie Y. Maggard

A thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Master of Applied Science
in
Electrical and Computer Engineering

Waterloo, Ontario, Canada, 2009

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners. I understand that my thesis may be made electronically available to the public.

Jessie Y. Maggard

Abstract

This thesis primarily covers the automation problem for sleep versus awake detection, which is sometimes accomplished by differentiating the various sleep stages prior to clustering. This thesis documents various experimentation into areas where the performance can be improved, including classifier design and feature selection from EEG, EOG and Context.

In terms of classifiers, it was found that the neural network MLP outperforms the continuous Hidden Markov Model with an accuracy of 91.91%, and additional performance requires better feature sets and more training data. Improved EEG features based on time frequency representation were optimized to differentiate Awake with 93.52% sensitivity and 94.60% specificity, differentiate REM with 96.12% sensitivity and 93.63% specificity, differentiate Stages II and III with 96.81% sensitivity and 89.28% specificity, and differentiate Stages III and IV with 93.60% sensitivity and 90.43% specificity. Due to the limited data set, an example of applying contextual information using a One-Cycle-Duo-Direction model was built and shown to improve EEG features by up to 10%. This level of performance is comparable if not superior to the human scorer accuracy of 88% to 94%.

This thesis improved some aspects of sleep staging automation, but due to the limitations on resources, the full potential of these improvements could not be demonstrated. To further develop these improvements, additional data sets customized by sleep staging experts is crucial.

Acknowledgements

I would like to sincerely thank my supervisor, Prof. Magdy Salama, for his significant support and guidance during my studies. I thank Dr. Charles George and the staff of his sleep lab for their abundant assistance. I also thank Wendy Boles and other staff members for their patience and special consideration. Lastly, I offer my regard for all those, in particular my husband and my parents, who supported me in any respect during this thesis.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Proper sleep is very important to a person's health and "the quality of sleep will directly affect quality of life" [1]. Excessive sleep, sleep deprivation or disturbed sleep not only cause the person discomfort but its effects can be far reaching. Therefore, physicians needed a means of assessing the quality of a patient's sleep. One objective method is a polysomnogram (PSG), which detects the brain's electrophysiological activity through non-intrusive sensors. PSG is interpreted through sleep staging - a process well known for being time consuming and error prone. Medical experts and engineers believe that automating sleep staging will be a significant improvement. However, thus far no automatic stager has provided the kind of performance as to be widely accepted for clinical use [2].

In addition to the diagnostic needs to assess sleep quality, new treatment and intervention technology to prevent dangers of drowsiness and sleepiness need portable lightweight algorithms that can detect changes in alertness or wakefulness in real time. Such algorithms can be based on automatic stager but they are often subjected to more unpredictable conditions as well as able to only access limited physiological signals. Therefore, further improvements on these algorithms need special analysis.

1.1 Scope

This thesis primarily covers the automation problem for sleep versus awake detection, sometimes in real time settings. Due to the needs of this goal, the automation problem is also expanded to differentiate the various sleep stages

prior to clustering into the simpler binary results. While the ultimate goal is to identify an overall algorithm, the limited resources in this thesis only permitted a series of experimentation into potential areas of improvement. The potential improvements include feature selection from EEG and EOG, which are the two signals in PSG most frequently used in the definition of the sleep stages. In addition to these two obvious approaches to increase the amount of input information, contextual information are analyzed to provide a third not so clearly defined venue of providing addition inputs.

1.2 Layout

The next section discusses background on sleep and sleep staging both to provide terminology and an understanding of the biomedical problems associated with the sleep staging process. Then, a section is dedicated to a critical review of previous automation attempts. Chapter 3 contains experimentation with classifiers such as Artificial Neural Networks as well as Hidden Markov Model. Chapter 4 looks at feature extraction from electroencephalograms, both in terms of band activity as well as the characteristic waves. Chapter 5 studies features from electro-oculogram. Chapters 6 examines contextual information, in terms of selecting a model, implementing that model and identifying implications of its use. Chapter 7 is the conclusion and it contains a brief list of work that should be continued.

Chapter 2

Background on Sleep Staging

Sleep staging, also known as sleep scoring, is the process of extracting sleep cycle information from the electrophysiological signals. This process is currently carried out manually, but research for its automation are well underway. This chapter describes the manual sleep staging process and examines the trends in the research of automatic sleep staging.

2.1 Manual Sleep Staging

Sleep staging extracts features[1], otherwise known as characteristic waves, from segments of electrophysiological signals and classifies[2] the segments according to the feature values to sleep stages, which are major events in the sleep [2]. The sleep stages are chained together with respect to time to form a sleep architecture and to provide an assessment of sleep quality. This process is demonstrated in the upper half of the flowchart in Figure 2.1. Currently, the sleep medicine community uses the sleep staging standard defined by Rechtschaffen and Kales (R&K) in 1968 [3].

Strictly speaking, sleep staging only entails the mapping from electrophysiological signals to the sleep stages. However, due to the shortcomings of R&K standard, clinical practices engage the patient data to validate the staging results and to adjust certain staging parameters. These activities are shown in the lower half of the flowchart in Figure 2.1.

To fully elaborate on the sleep staging process, this section covers the

[1]In sleep analysis, feature extraction may often be referred to as waveform detection.

[2]In sleep analysis, classification may often be referred to as stage or state determination.

Figure 2.1: Flowchart of activity in sleep staging.

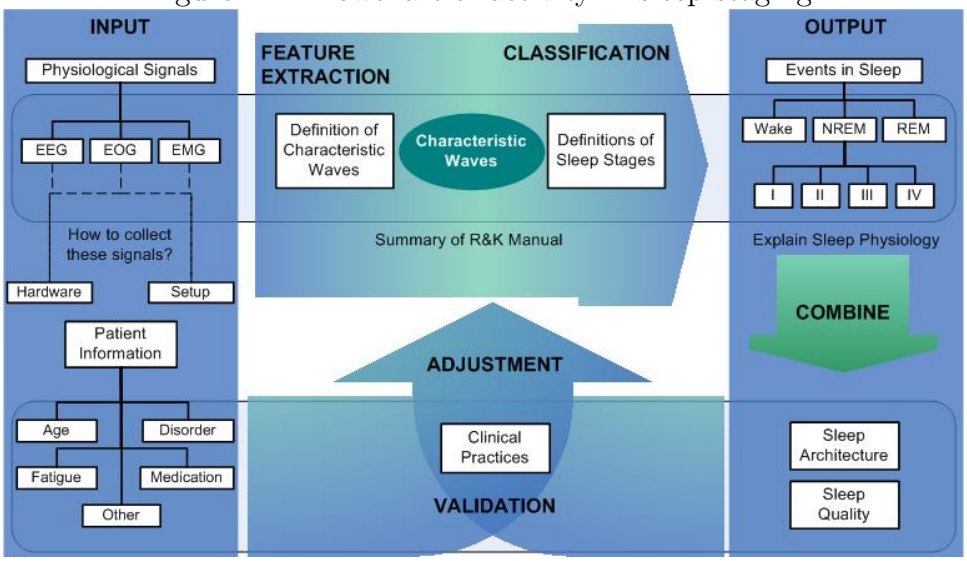
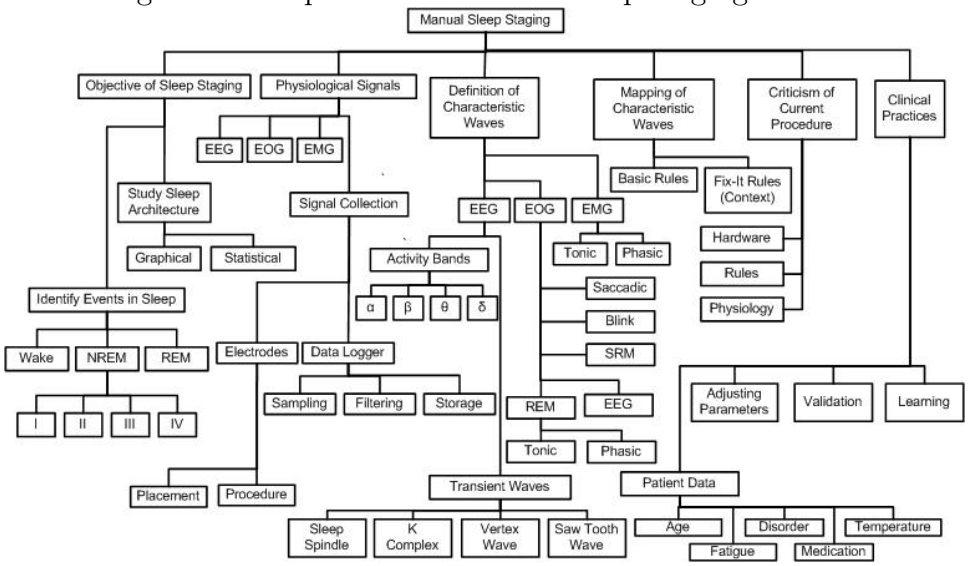


Figure 2.2: Topic tree for manual sleep staging section.



topics shown in Figure 2.2. First, it identifies the objectives of sleep staging by studying the outputs, which come in the form of sleep events and sleep architecture. The inputs - electrophysiological signals - are described next in terms of their function and their collection techniques. Then the section

defines the characteristic waves, in other words the feature extraction process, followed by the rules that map the features to the sleep stages. Finally, the section looks at the shortcomings of the current sleep staging process and the clinical practices that attempts to amend the problems.

2.1.1 Objective of Sleep Staging

As explained previously, the objectives of sleep staging are to identify the events in sleep and to study the sleep architecture.

Identifying Events in Sleep

Sleep is often described in everyday language by the terms: awake, light and deep sleep, and dreaming. Table 2.1 matches these common terminology to the sleep stage names (WAKE, NREM I to IV, REM, MT) and the classification encoding between 0 and 6 respectively. Note that the dream phase of sleep is named REM after its identifying feature - rapid eye movement. Light sleep and deep sleep do not exhibit this feature; therefore, they are called non-REM or NREM.

Table 2.1: Sleep stage sequence in NREM-REM cycle.

Common Description	Scientific Terminology	Computer Classification
Awake	Wake	0
Light Sleep	NREM I	1
	NREM II	2
Deep Sleep	NREM III	3
	NREM IV	4
	NREM III	3
	NREM II	2
Light Sleep	NREM I	1
Dream	REM	5
	MT	6

NREM is further divided into four stages. During sleep onset or NREM I, the brain activity slows down and become more organized. This stage is generally very short in duration, lasting between 1 to 7 minutes [3]. True sleep is counted at the beginning of NREM II. Its duration lies between 10

to 25 minutes [4]. NREM III and IV are deep sleep. NREM III lasts only a few minutes, but NREM IV can last 40 minutes.

The sleep stages occur in a sequence called the NREM-REM cycle, which is shown in Table 2.1. This sequence is expected to last between 90 to 120 minutes, and it may repeat 4 to 6 times per night. The bottom of the figure shows a classification Movement Time (MT), which represents the segments of time with significant body movements. It is not part of the NREM-REM cycle because it may occur at any time.

Studying Sleep Architecture

When the sleep stages are combined as a proper time sequence, the sleep architecture is formed. There are two methods of looking at the sleep architecture: graphical representation - hypnogram and statistical representation - sleep quality.

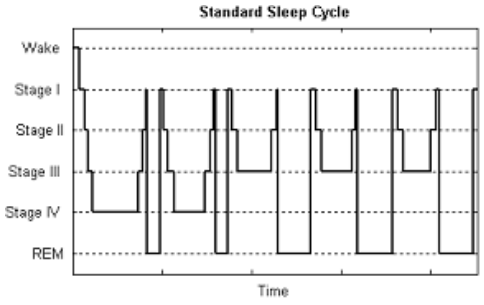


Figure 2.3: Hypnogram.

Hypnogram Healthy humans with a regular night's sleep will follow the sleep stages in the particular pattern shown in Figure 2.3. As discussed earlier, each cycle from light to deep to REM sleep will last approximately 90 minutes. The portion of the cycle spent in NREM decreases and REM time increases as morning approaches. While this pattern may deviate slightly from person to person, and from infancy to old age, the pattern repeats closely in the short term. Factors [4] that affect the sleep architecture include age, level of fatigue, disorder, medication, environment, etc. These factors and their specific affects will be discussed in Section 2.2.3.

Sleep Quality The R&K manual lists eight sleep quality parameters [3]. They measure the duration of REM, NREM, NREM I, and NREM II sleep, and their ratio to total night of sleep. Other factors [5][6] considered by physicians may be total sleep time, sleep onset latency, number of stage shifts, number of awakenings, amount of awake time, etc. Often physicians use patterns of abnormal parameters to assist with diagnoses. For instance, a

patient with narcolepsy may demonstrate a high ratio of sleep time in REM sleep.

2.1.2 Physiological Signals

The three electrophysiological signals required for sleep staging are electroencephalogram (EEG), electro-oculogram (EOG), and electromyogram (EMG). These signals are commonly collected in polysomnograms (PSG) - an overnight test used to identify sleep disorders. In today's practice, the potential is collected by electrodes and recorded by a data logger.

Electroencephalogram

Electroencephalogram is a recording of the electrical potential of the brain as it is detectable on the surface of the scalp. Electrodes are positioned according to the 10/20 System defined by the International Federation of Societies for EEG and Clinical Neurophysiology[3]. In this system, a channel of data is the difference between two electrodes: one on some part of the scalp and the other on the earlobe of the opposite hemisphere. Sleep staging typically use central channel, C3, referenced with A2 on the earlobe, or C4 referenced to A1.

Electro-oculogram

Electro-oculogram is the recording of voltage changes caused by eyeball movement. EOG is a key differentiator for REM sleep because the definition of REM is based on the occurrence of rapid eye movement. Furthermore, it is difficult to separate drowsiness from dreaming by EEG alone. The electrode is typically placed on both temples with a 1 cm offset from each other in lateral position. R&K recommends recording both channels to avoid misclassifying signals that only resemble eye movement. It is typically sampled at the same rate as EEG.

Electromyogram

Electromyogram is the recording of electrical activity in the muscles. In sleep staging, EMG studies the muscle activity on the chin. This position is often

[3]For details on the 10/20 system, please refer to Jasper's 1958 article [7]

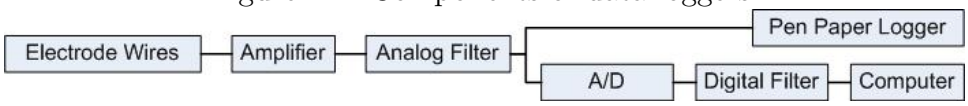
referred to as mental or submental EMG. EMG is often collected a much lower frequency.

Data Logger

Data logger is the device that detects the difference in potential on the electrodes and records this data. Figure 2.4 shows key components of a data logger. Traditionally, the logger consisted of analog circuitry for amplification and basic filtering. The data are recorded on moving paper where the pen's deflection is proportional to the potential. At that time, the temporal resolution is defined in centimeter of paper per second and the amplitude is scaled according to millimeter per microvolt.

Since incorporating computers, the signals are now recorded digitally. Their sampling rate is generally 100 or 128 Hz. The current loggers should be able to differentiate micro-volts and to resolve maximum voltage of 1000 mV. While it is still possible to use analog filters prior to the analog to digital conversion, it is generally preferred to collect the raw data and digitally filter them during analysis.

Figure 2.4: Components of data loggers.



The procedure defined by R&K sets up the hardware without computers. Details on modern hardware setup and computer procedures can be taken from Thomas Penzel's article [8].

2.1.3 Definition of Characteristic Waves

The description of characteristic waves is separated according to EEG, EOG, and EMG.

EEG Features

The first group of EEG features are its 8 characteristics waves relevant to sleep staging. They fall into two groups, frequency band activities and transient waveforms. The remaining features are either built upon the characteristic waves or based on simple measurements.

Frequency band activities are segments of EEG signal that demonstrate frequencies in specific ranges. Often their definition is augmented with amplitude requirements. The 4 frequency band activities are defined in Table 2.2[9].

Table 2.2: Description of EEG's band activities.

Band Activity	Frequency (Hz)	Amplitude (μV)	Present
Alpha (α)	8–13	20–60	Awake, NREM I, and REM
Beta (β)	13+	2–20	Awake
Theta (θ)	4–8	50–75	NREM I, II, III, and IV
Delta (δ)	0–4	75+	NREM III and IV

Sample segments of the band activities α and θ are shown in Figure 1. Band activity δ is shown in Figure 4.

Transient waves are relatively short segments of EEG signal displaying particular patterns. They often occur singularly in a stretch of EEG showing some frequency band activity or they may repeat consecutively. These transient waveforms are described in Table 2.3. The features can be a simple binary assignment where 1 represents the waveform was detected and 0 represents its absence. Alternatively, the feature may be a counter recording the number of epoch since the last occurrence of this transient waveform.

Sample sleep spindle and K complex are shown in Figure 2. A short train of vertex waves are shown in Figure 1. Note that vertex waves may have similar shape to K complex but they have much reduced amplitude.

Built on top of the characteristic waves is the feature, Mixed Band Activity (mba). It occurs whenever two or more types of activity occur together. For instance, α and β both occur in Wake, which constitutes mixed frequency activity. Note that these activities may display themselves consecutively or superimposed on one another.

EOG Features

EOG features consist of 5 characteristic waves, of which one has two forms. Therefore, the 6 waves are described in Table 2.4.

Slow rolling eye movement is shown in Figure 1. Figure 2 shows that the EOG signals are reflecting the background EEG signals. The two types of

Table 2.3: Description of EEG's transient waves.

Transient Wave	Description	Comments
Sleep Spindle (ss)	Low voltage activity at 12–14 Hz. Amplitude below 50 μV. Duration should exceed 0.5 seconds.	Present in NREM II and REM
K Complex (kc)	Delineated negative sharp wave immediately followed by a positive component. Duration should exceed 0.5 seconds. Amplitude exceed 75 μV.	Present in NREM II
Vertex Waves (vw)	Sharp potential, maximal at the vertex, negative relative other areas. Amplitude below 75 μV.	Present in Stage I
Saw-Tooth Waves (stw)	Special type of central theta activity that has a notched morphology resembling the blade of a saw	Present in REM

Table 2.4: Description of EOG's characteristic waves.

Characteristic Wave		Description
Saccadic Eye Movement (*sem*)		Movement as the eye follows some object. Present in Awake.
Rapid Eye Blinks (*reb*)		Natural eye blinks that occur at the frequency of. Present in Awake.
Slow Rolling Eye (*sre*)		Synchronous eye movement. Present in NREM I
Background EEG (*beeg*)		Present in NREM II, III, and IV
Rapid Eye Movement	Tonic (*remt*)	Present in REM
	Phasic (*remp*)	Sharp rapid "deflections". Present in REM

rapid eye movement, tonic and phasic, are shown in Figure 6 and Figure 7, respectively.

EMG Features

EMG features do not have easily recognizable characteristic waves. They are simply quantified by measurements of amplitude, frequency, change in amplitude, and change in frequency. In particular, flat EMG which is associated with REM sleep shown in Figure 6 can be contrasted to the elevated EMG in Figure 3.

2.1.4 Mapping Characteristic Waves to Sleep Stages

The R&K standard contains very specific rules to connect features extracted from 20- or 30-second epochs to the correct sleep stage. A much simplified version of the rules is presented in table 2.5. For details, refer to R&K's manual [3].

Table 2.5: Sleep stages as recognized by R&K.

Stage	Signals	Rules
Awake	EEG	Mixed α, β, θ, δ
	EOG	*sem* and *reb*
NREM I	EEG	α activity exceeds 50% of time AND no *ss* AND no *kc*
	EOG	*srm*
NREM II	EEG	α activity less than 50% of time AND θ activity the rest of the time AND *ss* may be present AND *kc* may be present
	EOG	Background EEG
NREM III	EEG	δ activity between 20% and 50%
	EOG	Background EEG
NREM IV	EEG	δ activity greater than 50%
	EOG	Background EEG
REM	EEG	Mixed α, β, θ, δ (with lower amplitude than Awake) AND *ss* may be present
	EOG	*rem*
	EMG	Lowest level
MT	EMG	Very high amplitude and very high

Fix-it Rules make up a significant part of the manual, because many special situations are missed by the basic rules. For example, NREM II is

scored in the presence of transients like *ss* and *kc*. If *ss* and *kc* do not occur in one epoch, the classification does not change until the 3 minutes have lapsed.

2.1.5 Criticism of Current Procedure

The issues with R&K standard are widely known. Aside from the comments the editors left in the manual itself, various articles have reported on its limitations. This section primarily reference from Kubicki's 1985 critical comments on R&K [10], Hasan's 1985 critical review [11] and Himanen's 2000 review on its limitations [12].

Many researchers find the enduring popularity of the R&K standard confounding. Since it was not designed to be a standard, but a reference, it lacks the level of validation [13] required for a "gold standard". Only a very few suggestions to improve the R&K standard have been adopted [2]. Many experts in the sleep medicine field are simply waiting for an improved standard based on modern technology [13][12].

This section will divide the criticism into 3 main categories: hardware setup, rule set, and neurophysiology.

1. *Hardware Setup* The hardware section of the R&K standard is now outdated. Clearly, specifications on paper recording can be converted to equivalent settings in a digital system. However, there are expanded capabilities that the R&K fails to exploit. The list below contains a few examples.

    (a) *Signal Processing* in the R&K standard is limited to analog filters pre-built into the system. The technician must select the filters before the signals are recorded on paper. However, such restriction does not exist in a digital system. The signal can be recorded in its raw form and filters can be selectively applied and modified during the analysis.

    (b) *Data Channels* were limited in the R&K standard because each additional channel increases the demand on recording paper. Current capabilities in processing and storage power mean that large quantities of data can now be stored and analyzed at minimal increase in cost. Many researchers indicate that one central EEG channel cannot pick up certain events [13][2]. For example, proper

detection of sleep spindles and K complexes would require a frontal or parietal channel. Now these channels can be recorded without significantly increasing cost.

(c) *Resolutions* both for time and amplitude can only be set to one value in the R&K standard, and originally the selection was based on a trade off of the differentiation needs of all the sleep stages. However, the viewer can now zoom in or out in both time and amplitude resolution as long as a higher resolution was initially sampled.

While criticism in terms of hardware is reported, many new systems simply incorporate the improvements.

2. *Rule Set* Regarding the R&K rules, the following issues are raised,

   (a) *Absolute Definition Boundaries* are misleading and sometimes incorrect. Many researchers feel that the ranges should not have hard boundaries, because human physiology is a continuous process [8]. In some instances, researchers found R&K's definitions arbitrary.

   (b) *Complexity of the Rules* is very high. The manual uses numerous fix-it rules to classify the many obscure sequences not covered by the basic rules. These fix-it rules significantly add to the overall complexity.

   (c) *Epoch Sizes* at 20- or 30-seconds was selected as a trade-off between temporal resolution and time efficiency [12]. Many researches feel that this temporal resolution is too low and that the following two problems will occur,

      i. Micro-structures, such as sleep spindles and K complexes, may be missed.

      ii. Division between stages is likely to occur within an epoch, which with a lower temporal resolution, a higher ratio of the epoch will be misclassified. This problem is particularly prevalent with brief stages like NREM I.

   (d) *Sleep Onset*[4] contains many steps, which R&K grouped into NREM

[4]Details on Sleep Onset is presented in Appendix 7.2.

I. Therefore, current classifications do not provide enough information for an in depth study of sleep onset [13].

(e) *Performance* achieved by human scorers is generally around at 88% to 94% within a sleep lab. The agreement varies in terms of specific stages or characteristic waves, and it often depends on the scorer's level of expertise and the quality of data. Two measures, inter-observer reliability and intra-observer reliability, are often listed [14]. Inter-observer reliability is the level of agreement between two scorers. Intra-observer reliability measures the agreement between two trials by the same scorer. This issue will be revisited in the next chapter as automated sleep stager performances are compared against human scorers.

3. *Neurophysiology* R&K's rules are designed for healthy young adults[13], but it fails for subjects who do not fall into this pristine category. In both clinical and research settings, it is the uncommon cases that require sleep staging. The R&K fails in the clinical setting, because it cannot adequately account abnormal data sets [15]. For instance, subjects with disturbed sleep cannot be scored directly according to R&K[16]. In many instances, the technicians learn to make allowances for biological variability through experience.

2.1.6 Clinical Practices

In response to the problems of the R&K manual, various clinical practices have been developed. Figure 2.5 briefly entails the practices applied to each problem listed in the previous section. For instance, most concerns with outdated equipment are now resolved by equivalent standards based in digital technology [8]. The figurative representations of the three important practices highlighted in Figure 2.5 are shown in Figure 2.6. This section elaborates on these practices after a quick discussion on the required patient data and sleep physiology knowledge.

Patient Data

Patient data serves as the basis of many clinical practices in terms of algorithm adjustment. The most important is age, followed by factors such as

Figure 2.5: Clinical solutions to problems in R&K.

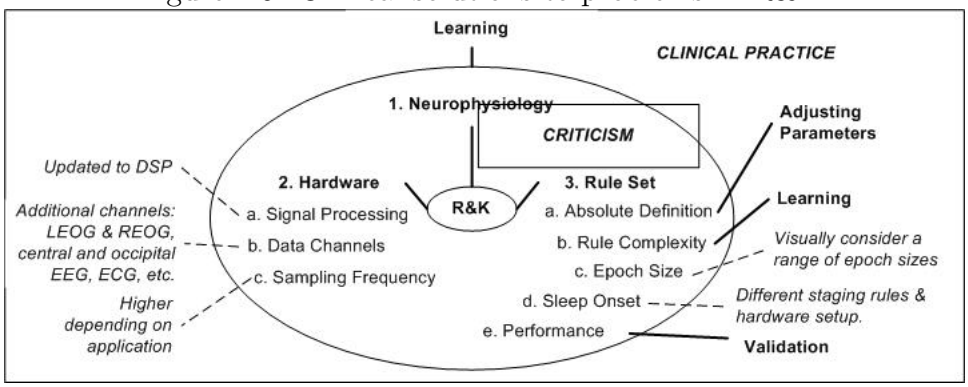
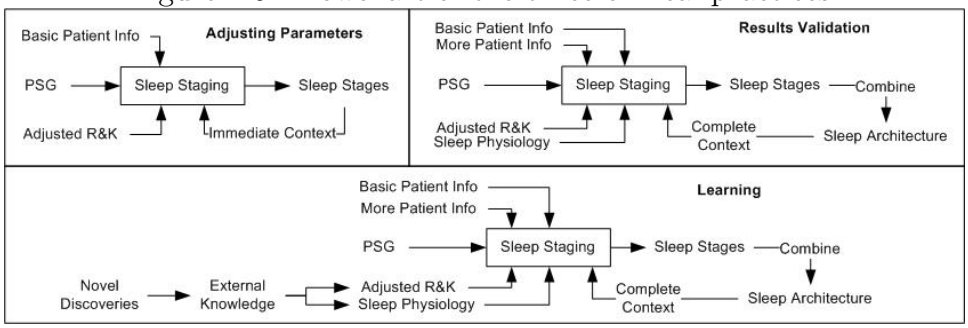


Figure 2.6: Flowchart of the three clinical practices.



disorder, medication, etc. Some factors affect the signals, as in adding artifacts or deviating from standard parameter ranges. Alternatively, they may change the composition of the sleep architecture.

Age Age[5] is by far the most important patient data factored into the sleep staging process. It influences sleep continuity and sleep stages [17][4]. There are three changes important for sleep staging,

1. As people age, the amplitude of their δ activity falls. The scoring method is modified to account for this change [18].

2. Deep sleep eventually disappear in the elderly.

[5]Sleep community recognizes normal sleep as patterns found in healthy adults between 18 and 29.

3. Sleep becomes increasingly fragmented and total sleep time decreases. [19][6].

The first change affects δ's feature extraction parameters. The latter two affects the overall sleep architecture.

Disorder Many sleep disorders affect the sleep pattern. Aside from sleep disorders, various medical and psychiatric disorders can impact sleep architecture [4]. For instance, depression and narcolepsy can cause shorter REM latency. Discomfort from chronic pain and sleep apnea can disrupt sleep continuity. Disorders like epilepsy will present strange waveforms in the physiological signals.

Medication Medication, such as anti-epileptic drug, psychotropic agents, and sedatives [20] and recreational drugs often affect sleep. Depending on the agent, the affect on sleep architecture may be sleep promotion, sleep consolidation, and certain stage suppression [4]. Some medication may also cause artifacts, such as alpha intrusion shown in Figure 5. Withdrawal from these agents often causes the affected sleep to rebound.

Temperature Brain temperature affect the power spectrum of EEG [21]. For instance, higher temperature means higher frequency content in alpha band. A relative body temperature drop normally indicates sleep onset. External temperature mostly affects sleep continuity instead of onset [4].

Adjusting Parameters

Figure 2.6 indicates that this is the simplest of the algorithmic clinical practices. It is carried out during the first scoring process. The basic patient data that is taken into consideration is age, which adjusts the parameters defined in R&K standard. The technicians make these adjustments in their mind and the degree of adjustment is based on their experience. For example, a scorer may lower the amplitude requirements for delta activity when the patient is more elderly. A scorer may adjust the expectation of the next stage based on the current stage. For instance, if the current sleep stage is NREM II, then the next stage has a high chance of being NREM II and a good chance of being NREM I, NREM III, or REM.

Results Validation

Clearly, results validation is applied on top of parameter adjustment. After the first scoring, the overall architecture is studied, and it is considered a complete context. If the overall architecture shows peculiarity, the technicians use the details in the patient data and their sleep physiology knowledge to justify these peculiarities.

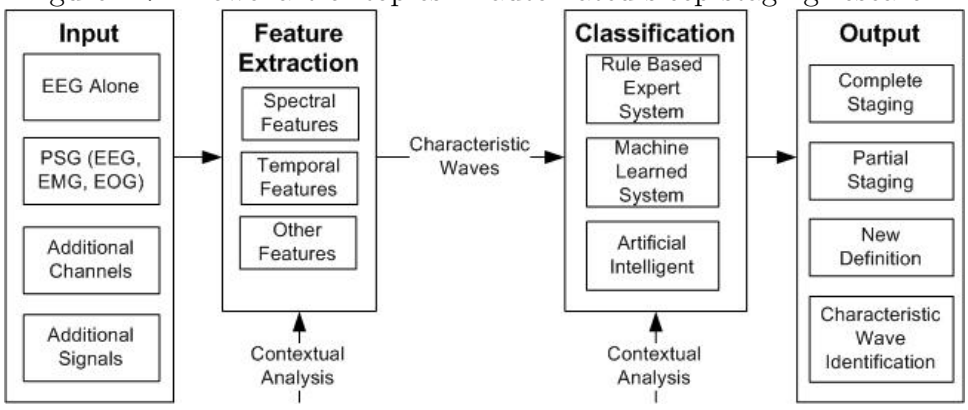
Learning

Given that validation fails, the technician would expect new knowledge to make sense of their staged results. Either by consulting with another sleep expert or researching new publications, the technician will absorb the new knowledge. The new knowledge may simply shift the fuzzy boundaries in the already adjusted R&K rules. Otherwise their understanding of human sleep physiology needs to be updated.

2.2 Automated Sleep Staging

Automated sleep staging follows the same format as manual staging. Automated sleep staging is also divided into feature extraction and classification. Figure 2.7 shows the major trends in automation research in each component of staging. After discussing the motivation for automation, this section will look at levels of input, feature sets, classification methods, output objective, and contextual analysis. The section concludes with a summary of issues in previous studies. The covered topics are organized in the tree in Figure 2.7. A detailed summary of the automation research referenced in this section is located in Appendix 7.2.

Figure 2.7: Flowchart of topics in automated sleep staging research.

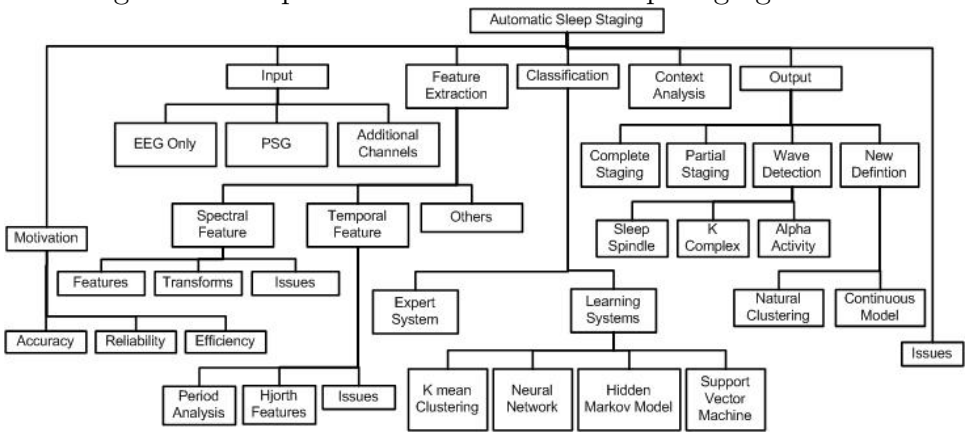


2.2.1 Need for Automation

For the interpretation of PSG data in overnight sleep studies, automation is called for to answer the following 4 points,

- Sleep studies generally run for 8 hours. Since the data is analyzed in 30-second epochs, sleep staging is estimated to take 2 to 5 hours of expert time [15]. Thus, this process is very time consuming.
- Humans are prone to make mistakes especially in a tedious and repetitive tasks.
- The inter-scorer accuracy is at 88% but there is still significant room for improvement.

Figure 2.8: Topic tree for automated sleep staging section.



- Many sleep staging results cannot be reproduced because scorers might not even agree with themselves in two trials.

These issues translate to cost increase either to the patient or to the health care system.

Automation is widely believed to be the solution. Automated sleep staging is aimed to reduce the workload for technician, as opposed to replacing them. Such a system would still require validation from human experts. It would improve accuracy, provide reliable and reproducible backup [16]. These improvements would ultimately result in a cost reduction for sleep disorder diagnosis, treatment, and research. Note that automation for the clinical setting has accuracy, adaptivity, and cost efficiency as primary goals.

Automating sleep staging is also required when trained experts are not available to produce a human scoring. For instance, a portable alertness detector or a long-term sleep data collector clearly cannot use a technician to score the sleep. They must be provided with internal automated sleep staging mechanisms. In these applications, the algorithms need to have lighter processing, energy efficiency, and reasonably good performance.

2.2.2 Input

Sleep staging automation research have studied three levels of input:

1. *EEG Only* While EEG, EOG, and EMG are used in the R&K standard, most researchers agree that the majority of sleeps staging information

can be derived from EEG. Therefore, significant portion of the automation research disregard the other two signals. Some researchers acknowledge the need for EOG and EMG, but choose to ignore them in their actual algorithms [15][22]. Others take pride in developing algorithms that uses EEG only [23]. Until the recent technological advances, EEG may have been previously analyzed alone due to lack of processing power.

2. *EEG, EOG, & EMG* The classical three physiological signals attached with sleep are used in most research aimed to develop a proper stager based on R&K manual.

3. *Additional Channels* Since the channels defined in R&K are often criticized for not containing sufficient information, some researchers have attempted to include more channels [24]. However, one study finds that additional electrodes provide no improvement in performance [16]. Additional channels may be applied in particular cases, such as sleep spindle detection [25].

4. *Additional Signals* Signals other than EEG, EOG, and EMG have been considered for sleep staging as well. They include electrocardiogram (ECG), respiratory signals, oxygen saturation, blood pressure, body temperature, body position and movement [16]. However, none of these signals can contribute more information towards sleep staging than EEG.

More inputs translate to more information that can be extracted, but it also means complexity in signal collection and additional processing.

2.2.3 Feature Extraction

Many features have been tested to automate sleep staging, and they fall into different sets depending on their calculation. This section looks at these feature sets, their applications, advantages, and problems. The equation to most of these features are listed in Appendix 7.2.

Spectral Features

Spectral analysis requires the signals in the time domain to be first transformed into frequency domain. Then the features are extracted from the

frequency domain.

Transforms The spectrum is commonly obtained from Discrete Fourier Transform (DFT), where the efficient adaptation is called Fast Fourier Transform (FFT). The main issue with FFT is the requirement that the signal is stationary, which can be relaxed to very slowly varying [26]. Since band activity tend to last longer, FFT is often applied. FFT is unsuitable for transient waves like sleep spindles and K complexes [2].

The popular alternative today is Wavelet Transform, because this decomposition method is well suited to non-stationary signals [25]. It was applied to extract band activity [27] and to analyze sleep spindles. In a study using spectral features derived from wavelet analysis and classified by a neural network, the tested accuracy was 77.6% [28].

Another alternative to FFT is autoregressive modeling (AR) which can use shorter signal segments [16][2]. AR expresses the signal as a linear combination of the M previous samples with a very small white-noise. The power spectrum is calculated from the M autoregressive coefficients, and M determines the resolution. One sleep stager, BioSleep chose the order $M = 10$ and replaced the regular coefficients with reflection coefficients in order to achieve a normalized spectrum [1]. Compared with human scorers, BioSleep's accuracy is 72.2%.

Adaptive segmentation is a method of identifying the best segment length for FFT. It calculates a spectral error measure for each segment considered. If the error of a segment is not sufficiently low, a shorter segment is considered.

Features Since the EEG band activities are divided primarily along frequency lines, frequency-based measures are the most popular features. The list below are common spectral features,

- The simplest features are the *spectral power* within in each activity band [29][30]. An alternative is the relative ratio of spectral power between bands [31][29], as defined in Equation 11. The feature, delta power, alone has been shown to have an overall classification accuracy of 70.7% [16].

- *Spectral entropy* [30] reflects whether EEG activities are concentrated in one frequency or spread out along the spectrum. Therefore, this

feature is very useful for the identification of mixed frequency activity seen in Wake and REM. Its function is defined in Equation 12.

- Harmonic parameters [32] consist of *center frequency*, *bandwidth*, and *power of the central frequency*, which are defined in Equations 19 to 21.

- Other spectral features can be *spectral edge* – frequency up to which 90% of total power, *cutoff frequency* at 45Hz, *spectral moment* – spectral power multiply with power of frequency [29], *fractional spectral radius* [33], *embedding space entropy* [33]. The latter two features measure complexity of EEG activity which should be superior to basic spectral measures by avoiding variability between individuals [34].

Comments Aside from the issue of EEG not being stationary, other issues lies in the lack of clear frequency boundary between characteristic waves and the computational requirements of transforms. In many instances, the frequency of sleep spindle is between α and β [2]. Compared to time domain analysis, spectral features are always going to require more computation [35]. However, frequency analysis has the advantage of showing all the frequency components whether or not the bands are superimposed [2].

Temporal Features

Features in time domain are popular because they respond to human scoring practices and because they require less computation capacity [2].

Period analysis, which studies zero-crossing and counting peak-to-peak, provides the features most closely resembling human scorer actions [2]. The most common feature are *amplitude*, *period*, *period duration*, etc. From zero-crossing, the problem is that often fast and slow signals are superimposed, which would cause this method to fail. However, it has several advantages,

1. Its fast and light computation and its ability to directly incorporate R&K's discrete thresholds support its popularity.

2. Without having to worry about window length, it can be used for transient wave detection.

3. It is also capable of identifying percentage of an epoch dominated by one band activity, which is generally lost in spectral features [11].

Hjorth [36] defined three time-domain properties,

1. *Activity* measures the squared standard deviation of the amplitude. Changes in *activity* reflect changes in amplitude, and they often signify transition into the next stage.

2. *Mobility* measures the standard deviation of the slope relative to *activity*. Changes in *mobility* shows change in frequency, which is another signal of stage transition.

3. *Complexity* quantifies the excessive details relative to the sine wave.

These features have been used to study background EEG [24]. In 2001, Van Hese[32] used k-mean clustering with the Hjorth parameters, Harmonic parameters, and ratio band energy; however, the author did not publish the accuracy of the unit. Aside from sleep staging, Hjorth features have been used in mental-task discrimination [37] and decorrelation of multichannel EEG [38].

For K complex identification, one research spent significant time visually inspecting K complexes [39]. He identified 14 amplitude and time features that characterizes K complexes despite their variability. These features had a 90% true positive rate and 8.1% false positive rate.

Other Features

Since EEG contains nonlinear dynamics [40], nonlinear features, such as *correlation dimension*, *largest Lyapunov exponent*, and *approximated Kolmogorof entropy* were applied to EEG analysis. These features were compared to spectral features without clear results. Studies have shown the combination of nonlinear features are superior [29] and vice versa [30]. Therefore, the value of these features is difficult to establish.

2.2.4 Classification

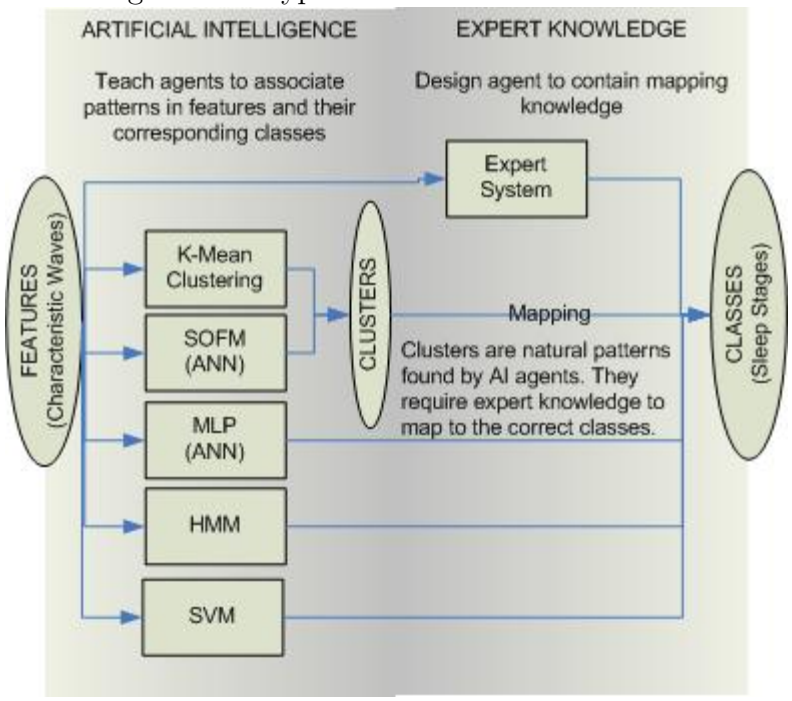
In sleep staging research, classification methods should be studied from two approaches,

1. Systems designed to contain the expert knowledge, in other words the classification know how. Therefore, these are commonly called expert systems or rule-based expert system.

2. Systems that contain the artificial intelligence to learn the patterns in between the features and the corresponding classes. The pattern recognition is established by interaction with a set of training data.

Figure 2.9 divides the major classification methods into these categories.

Figure 2.9: Types of classification methods.



Expert Systems

The rule-based expert systems were often designed to translate the R&K manual into computer logic in commercial systems [2]. These systems inherit all the flaws described in Section 2.1.5, where the most notable is the inflexibility in its boundary definitions. These systems are compared against results from clinical technician. Recall that clinical technician may adjust the rules to account for the biological variability [16]; thus, these expert systems cannot achieve very high accuracy as they are not following the human actions exactly.

The issues of soft boundaries used in clinical practices is overcome by fuzzy logic [2]. One study using fuzzy logic for pattern recognition on EEG

alone produced a 77% overall agreement [22]. Another example is the use of fuzzy logic for the identification of α activity which reached a true positive rate of 85% and a false positive rate of 13% [31].

Multi-Layer Perceptron (MLP)

MLP is a family of artificial neural network that are trained to associate input patterns with output classes. The most popular MLP training method is error backpropagation, which is the reason that MLP is sometimes called backpropagation network (BPN). MLP is quite popular because it can be quite easy to build and train. However, it is often stumped by inconsistencies in the training set or ambiguity in the sleep staging rules.

The performance of an MLP is best demonstrated in comparison with another classifier. In one study, MLP was compared with Fisher's linear discriminant, where at the same true positive rate, MLP has half the false positive rate [39]. This improvement in performance is due to the non-linearity of the MLP's decision surface.

The structure of the MLP meaning the number of neurons and their layout affect the system performance,

1. *Input Neurons* indicate the quantity of information from which MLP must identify patterns. Often without sufficient data reduction, the data simply overwhelms MLP. One study, where raw signal substitutes the extracted feature, maintains the same true positive rate at 90% and gives a false positive rate of 46.9% and 8.1%, respectively [39].

2. *Hidden Neurons* provide the non-linearity of MLP [13]. However, the system's complexity dictates a minimum number of neurons necessary for adequate representation. Additional hidden neuron simply increases computational complexity. The determination of this factor often relies on testing various setups for the best performance [39].

3. *Output Neurons* determine the number of classes and the representation of classes. For instance, sleep staging can be use several binary output neurons each corresponding to one class [41], or one neuron as an integer from 0 to 6 [39][28].

Hidden Markov Model

Hidden Markov Model[6] (HMM) evaluates the features with respect to the stage of the previous epoch, and uses the probabilities in sleep stage transitions to provide the stage of the current epoch. Though this method sounds quite practical for sleep staging, the sleep stage transitions are more random than HMM can allow. This randomness comes from the strict R&K rules. However some studies have used HMM to develop a new continuous sleep stager [43]. In correlation to the R&K standard, the continuous stager had a performance of approximately 80% for wake, deep sleep stages, and for REM only 26%, which was later improved to 68% [44].

Support Vector Machine

Support Vector Machine[7] (SVM) is a newer classification method. Its application to sleep staging shows that it can identify sleep spindles with 95.4% accuracy as opposed to the 88.7% by MLP [46].

Clustering

Clustering uses algorithms that detect natural patterns within data, and the data are grouped according to these patterns into clusters. The clustering process does not require *a priori* knowledge. However, these clusters need to be mapped to commonly known classes. For instance, K-mean clustering was applied to Hjorth parameters, Harmonic parameters, and ratio band energy [32]. Twenty spherical clusters were identified and they were mapped into 5 sleep stages, because some stages needed to be modeled by non-spherical clusters.

Another method of achieving clustering is the use of self-organizing feature map (SOFM). These neural networks are not provided with a reference classification, such that they modify their weight values purely based on their own pattern recognition abilities. One study trained an SOFM with features extracted from a Kalman filter [15]. It generated 8 natural clusters, from which the researchers were able to find 3 transition trajectories.

[6]For further information on HMM, such as definition and examples, please see [42].

[7]For further information on SVM, such as definition and examples, please see [45].

2.2.5 Context Analysis

Many studies believe that contextual analysis will improve the accuracy of a stager. However, contextual analysis have never been used to the extent of its role in the clinical practices. The simplest method is the use of HMM, which inherently builds the contextual information into the model.

Another method is to use artificial intelligent agents to identify the contextual rules. In one study, after the features are extracted from the raw EEG waveform by the Sleep EEG Recognition Neural Network (SRNN) and the features are classified by the Sleep Staging Diagnosis Neural Network (SSNN), the classifications enter the Contextual Diagnosis Neural Network (CDNN) for contextual analysis. This type of contextual post-processing using MLP can improve the accuracy from 65% to 82% [41].

As a follow up to this study, two types of ANN that have build in contextual analysis were examined. Time Delayed Neural Network (TDNN) uses the classification of previous epochs as one of the inputs. All Connecting Neural Network (ACNN) uses feedback connections to provide combine the effect of previous and current classifications. The original system performed better than the new networks [47], because the original system offered more control.

Another stager used the stage of the previous epoch and the current epoch's classification to provide contextual correction [24]. For instance, if the previous epoch was awake and the new one is deep sleep, then it corrects the current stage to NREM II.

2.2.6 Output

Aside from differentiating all the sleep stages as does the commercial system from Oxford Medical [48], some research aims only to contribute to a specific part of the process. For instance, K complexes are difficult to identify from the central electrode position, so much research have been devoted to its identification. This section looks at the different classification goals set by previous researchers.

Full and Partial Sleep Staging

Full staging means that the automation algorithm classifies the epochs as wake, NREM I to IV, REM, and MT. One system presented in 1983 used

filtering principle, bandpass filters, period counters, amplitude level detectors, pattern recognizers and all 3 signals [11]. Its performance was not sufficient for common clinical use. The sleep stager developed by Oxford Medical is quite well documented. It was evaluated by an external expert in 1989. Its performance difference with respect to human references is 26.9% [48]. Some commercial systems have successfully developed full sleep staging algorithms. However, the internal mechanisms of these proprietary systems are not published and they cannot be compared with published systems.

Often developing algorithms that classified all 7 stages can be quite difficult. Many studies opt to skip the epochs classified under MT [28][32]. A step further combines NREM I with REM due to the difficulty in separating them [24]. Similarly, NREM III and IV are combined as deep sleep or slow wave activity [29], which is characterized by δ band activity. These algorithms are partial sleep staging systems.

Characteristic Waves Detection

Aside from sleep staging studies, characteristic wave detection is also an important area of research. Wave forms like sleep spindle and K complex are vital to identifying NREM II or the beginning of true sleep. Another wave α have been studied for its uses in identifying NREM I [31].

Sleep Spindle Some studies attempted to identify the optimal EEG channel for sleep spindle detection [25][9]. Another one contrasted the pattern recognition capabilities of ANN and SVM for this application [46]. Matched filters were also applied for spindle at one time [16].

K Complex Like sleep spindles, K complexes are a differentiating feature for NREM II. Its identification receives significant research because the pattern for K complexes is very difficult to describe. One study [39] spent over two years to identify the amplitude and duration features that are applied in visual detection of K complexes. These features were supplied to a neural network with a sensitivity of 90% and an 8% false positive rate. A comparable K-complex detector developed in 1995 [49] uses a neuro-fuzzy agent and its accuracy is approximately 96%. Another study [50], using synthesized data, studied K-complexes with respect to wavelet analysis, orthonormal filters, and matched filters.

New Definition

Some researchers choose to develop new sleep stage definitions, because they believe that sleep stages listed in the R&K are not representative of actual mental states. One study uses self-organizing feature map (SOFM) to identify 8 halting states as being "more closely related to the bulk cortical action during the sleep" [15]. Another group of researchers wanted to build an automatic continuous sleep stager based on the probabilistic principles which overcomes the known drawbacks of traditional R&K sleep staging [43]. However, algorithms using new definitions are difficult to evaluate because their output cannot be referenced to the existing standards.

2.2.7 Issues

Despite the excellent application of technology to automate sleep staging, there are some shortcomings in these previous attempts. Aside from the challenges to automate sleep staging according to R&K, there are also issues within the studies. Below is a list of issues,

1. *Objective* Aside from a few studies [48][1], most researchers did not focus on automating sleep staging according to R&K. Therefore, their efforts are to develop new ones or to prove some features and classifiers.

2. *Feature & Classifiers* It is noted that some combinations of features and classifiers are more apt at identifying certain characteristic waves or differentiating between some two stages. Therefore, the use of only one classifier or feature set as is the general practice in these studies will not be enough.

3. *Fix-it Rules* In the case of manual sleep scoring, many of the special cases are identified using "fix-it" rules, which take an epoch's context to solve the ambiguity. For instance, sleep spindle and K-complex have disappeared for 3 minutes, and then the epoch is classified as NREM I despite being in NREM II previously. The perplexity of these rules complicates the translation of the rule set into the digital world of computer. Since these rules stretch the discontinuity in the rule set, they are not intuitive to AI algorithms. It often means most agents will have difficulties learning these rules.

4. *Context* Only a few studies [41][24] looked at the contextual information, such as patient data, sleep architecture, etc. Some of this knowledge is not explicitly written in R&K standard, but the trained technicians are likely to bear this information in mind.

5. *Epoch Size* Some scorers do not view the data at 30-second resolution right away. They look first at lower resolution, giving sections of epochs a stage assignment and refine the classification at the borders.

6. *Performance* The output is compared against the stages classified by trained medical experts, but the inter-observer agreement introduces question with regard to the actual performance of a system. Therefore, the AI agent's reliability may be rejected based on the possibly faulty classification provided by human scorers. Unfortunately, proving that the automated scoring unit is in fact more accurate is nearly impossible, in particular for agents that are not open-boxed [51].

2.3 Summary

This section has discussed much of the background on sleep staging as relevant to automating the process. While the main focus was to automate sleep staging according to R&K rules, this goal does require extensive research and significant medical and engineering resources. For instance, many data sets from a variety of patients, collected under different conditions, and scored by multiple experts would be necessary to train an artificial intelligent system to incorporate the various aspects of sleep staging.

Based on the resources available for this particular thesis, the goal, as stated previously, is limited to study and improve certain aspects of sleep staging automation. More specifically, the aspects being studied include comparisons between certain feature sets and classifier, optimizing certain feature sets, and expanding on the contextual information analysis.

Chapter 3

Experimenting with Classifier

In order to have overview of which aspects to improve, the first part of the study was to experiment with some classifiers. As discussed in Section 2.2.4, various classifiers, such as K-Mean Clustering, Artificial Neural Network (ANN), and Hidden Markov Model (HMM) have been used previous sleep staging research. Since this is early experimentation, the two classifiers requiring the least amount of specialized design and most likely compatible with various feature sets are selected. They are ANN and HMM. Specifically, Multi-Layer Perceptron (MLP) is chosen as the ANN that can be easily built and trained to recognize complex patterns. HMM is selected because its inherent use of contextual information meaning that new inputs are analyzed with reference to the system's current state.

3.1 Artificial Neural Network

In this section, MLP was used to differentiate sleep and awake. This algorithm was developed in Matlab v6.5 and the ANN uses Matlab Neural Network Toolbox v4.0.

Source Data

This section used EEG as the only input. The data set comes from the Physiobank [1] The data set used had the patient id sc4002e0. The data

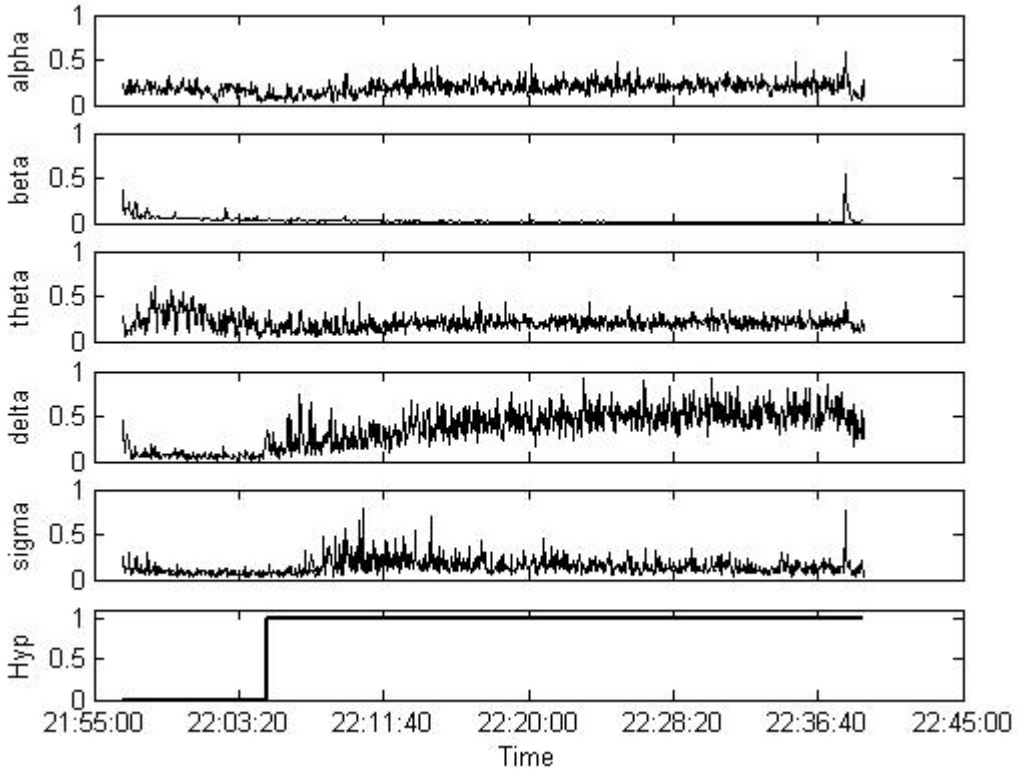
[1]The data set is available at `http://www.physionet.org/physiobank/database/sleep-edf/`.

is extracted from European Data Format (EDF). Channel 1 (central) and Channel 2 (occipital), sampled at $100Hz$, are used.

Feature Extraction

Spectral features consists of average power (MEAN), relative power (RATIO), standard deviation (STD), and power range (RANGE). The features were calculated for each activity band, specifically α in 9 to $12Hz$, β in 12.5 to $40Hz$, θ in 5 to $8Hz$, and δ in 1.5 to $4Hz$. Figure 3.1 shows a sample of these band activities relative to sleep and awake state. In the figure, the band activities were normalized. In the hypnogram, 0 denotes awake and 1 is asleep.

Figure 3.1: Normalized band activity relative to sleep states, Asleep (1) and Awake (0).



The spectral features were calculated using Fast Fourier Transform (FFT). The windowing method is Hamming. The window size is 2.56 seconds. In order to double the number of data sets that can be used for training and testing, windows were overlapped by 50%. This approach resulted in 33165 epochs of data.

Building the MLP

The MLP were structured to have 4 inputs, one for each activity band. In order for MLP to be trained quickly, the inputs were normalized. The output is a binary classification where 0 is Awake and 1 is all other stages, generally called Asleep. Of the 33165 epochs, there are 22090 epochs of awake state and 11075 epochs of asleep states. Various topology of MLP, meaning number of layers, neurons, and transfer functions, were examined.

During the training phase, the data set is split into two portions, 75% for training and 25% for testing. Therefore, the number of epoch for training and testing are 24873 and 8292, respectively. The neural networks were trained until the error rate is less than or equal to 1%. Various training algorithm and learning rates were examined.

3.1.1 Results

This section reports the various aspects studied to optimize the performance of MLP. It looks at the effect of feature set and the topology of the neural network. Accuracy is defined as the ratio between the number of epochs correctly classified by the MLP to the total number of epochs. Accuracy was only calculated for the test data set.

Feature Sets

Since the features as the inputs to the neural network will limit the amount of information that can be deduced, the first experiment was to choose the best feature. This experiment uses an MLP with 3 hidden neurons with a linear transfer function. The MLP was trained by default training algorithm and parameters in the Matlab ẗrainf̈unction. Table 3.1 contains the accuracies of MLPs built using the 4 spectral features calculated on each EEG channel.

Based on Table 3.1, the best feature is MEAN on either channel. Table 3.2 shows the breakdown of the training and testing performance of the feature

Table 3.1: Accuracy by feature and channel.

Feature	MEAN	STD	RANGE	RATIO
Channel 1	96.85%	96.07%	94.98%	81.00%
Channel 2	97.20%	95.53%	93.61%	75.88%

MEAN on Channel 1 in terms of sensitivity and specificity.

Table 3.2: Performance details of MEAN feature.

	Accuracy	Sensitivity	Specificity
Retrospective	96.91%	95.52%	97.60%
Prospective	96.85%	95.70%	95.70%

NN Topology

The layout of the neurons also affect the performance of the neural network. Since the system is clearly not linear, the network must at least have 1 hidden layer, which would model a continuous boundary. This experiment looked at 1 to 9 fully connected neurons in the hidden layer. For this experiment, the MLP uses the feature MEAN as the inputs. Table 3.3 shows the accuracy as well as storage requirement for the various numbers of hidden neurons. The storage is the size of the neural network data object in Matlab. A similar experiment was conducted for linear, tangential, and logarithmic transfer functions. The results showed no noticeable difference in performance.

Table 3.3: Performance of basic ANN with different size hidden layer.

Hidden Neurons	Accuracy	Sensitivity	Specificity	Storage (bytes)
1	96.82%	95.74%	97.36%	33181
3	96.61%	95.18%	97.34%	33533
5	96.94%	95.43%	97.69%	33885
7	96.80%	95.33%	97.54%	34237
9	96.88%	95.44%	97.06%	34589

NN Training

Different training algorithms[2] and learning rates are also studied as potential factors to influence the performance of the MLP. This experiment uses the MLP with 5 hidden neurons, logarithmic transfer function, and the feature MEAN. The train algorithms provided by Matlab were studied and their performances after 250 training epochs are listed in Table 3.4. The number of training epochs was selected because experimentation shows that the neural networks consistently converged after 250 epochs.

Table 3.4: Performance of basic ANN with different training algorithm.

Algorithms	Error	Accuracy	Algorithms	Error	Accuracy
traingd	0.226	0.668	traingdm	0.1673	0.7476
traingda	0.0592	0.9468	trainoss	0.0275	0.9652
traincgf	0.0264	0.9653	traincgb	0.0258	0.9655
trainscg	0.277	0.966	trainbfg	0.0264	0.9661
trainlm	0.0241	0.9674	traincgp	0.02622	0.9679
trainrp	0.0258	0.9683			

Other training parameters studied are maximum number of epochs and learning rate (LR). The tradeoff on training epochs is that more epochs can improve the retrospective performance but it may also over-fit to the training data. Using the resilient backpropagation training algorithm on the same MLP setup as the previous experiment, the results are listed in Table 3.5.

Table 3.5: Performance of different maximum epochs and learning rate.

Epoch	Error	Accuracy	LR	Epoch	Error	Accuracy
100	0.0259	96.66%	0.005	250	0.0255	96.90%
250	0.0258	96.83%	0.010	250	0.0255	96.91%
500	0.0253	96.58%	0.050	22	0.0483	96.23%
1000	0.0250	97.05%	0.100	20	0.0892	94.01%
2500	0.0250	97.01%	0.150	8	0.1396	84.36%

[2]The training algorithms used in these experiments are provided under Matlab's Neural Network Toolbox. Due to the limited scope of this thesis, the definition and the characteristics of these training algorithms will not be covered. For definitions, see Matlab User Manual.

3.1.2 Discussion

This section analyzes the results reported above.

Feature Set

As mentioned above, Table 3.1 shows that the best spectral feature was MEAN, which is frequency value averaged across a certain range. Referring back to Figure 3.1, while alpha and beta waves did not show significant change from awake to sleep, theta significantly decreased and delta significantly increased. Therefore, the performance of this feature is likely to be resulting from information encoded in the theta and delta MEAN.

The performance for the features RANGE (maximum - minimum) and STD were fairly close to that of MEAN. Both sets of information correlate closely to the characteristics of MEAN; therefore, they may be redundant if MEAN was already used as an input.

The feature RATIO is defined as the ratio between the MEAN and the RANGE. Unlike the other features, this feature did not perform as well because both the MEAN and the RANGE trend similarly in the transition from Awake to Asleep and these changes offset each other.

The detailed analysis on the feature MEAN shows that the retrospective and prospective performance are similar, which means the neural network was not over fitted to the training data. Furthermore, the data set is likely to be fairly uniform, which makes it well suited for training and testing.

NN Topology

Table 3.3 shows that the number of neurons in the hidden layer do not contribute significantly to performance nor to storage. Therefore, in this particular instance, as long as the feature remains the same, additional nodes in the hidden layer cannot encode more information. The likely explanation for this observation is that the binary output relies on a very simple relationship with a limited number of the inputs. While additional neurons add to the storage, the single hidden layer means that the number of transfer functions is very limited. It should be noted that since sleep staging in general is not a continuous boundary, more hidden layers may be necessary and then the complexity of additional neurons will present significant processing and storage challenges. It is arbitrarily chosen that 5 hidden neurons would be used.

Similarly, when the various transfer functions were compared, minimal differences could be found. Considering the output is targeting 0 and 1, the logarithmic transfer function seems most appropriate.

NN Training

In Table 3.4, several training algorithms were shown to have similar performances, and they include traingda, trainoss, traincgf, traincgb, trainscg, trainbfg, trainlm, traincgp, and trainrp. Therefore, for simplicity sake, the resilient backpropagation (trainrp) will be used to train the MLP.

Table 3.5 shows that the number of training epoch will not improve the performance after a certain minimum level. Since the number of training epoch directly increases the training time, the minimum requirement of 100 is selected.

Table 3.5 also showed that the learning rate affects the training process. As the learning rate increases, initially the time or the number of epochs it takes to train the neural network decreases. However, after a certain peak, the learning rate is so great that the MLP cannot converge closer thereby adversely affecting the performance. To tradeoff the speed versus performance, the learning rate 0.01 was selected.

3.1.3 Post-Filtering

Upon analyzing the neural network, it was found that the network is more sensitive than physician scores. The network studies the alert state every 2.56-second interval[3], whereas the physician scores the system every 30 seconds. Therefore, the network produces singularities in the classification that either the human cannot visually identify. To closely approximate a physician's scores, post-filtering is applied to remove these singularities. The left graph in Figure 3.2 shows that filtering across 20 data points will raise the accuracy to 98.75%. In real-time, this filter translates to a 50 second delay in decision, meaning a state must hold for more than more than 25 seconds in a 50 second interval for the state to latch.

The final network was designed based on the network's performance in state classification; however, the system's performance in detecting awake-sleep transition is used to evaluate the system. The right graph in Figure 3.2

[3]This window size correspond to the 2-second windows supported by some researchers that believe 30 seconds is too low a temporal resolution [16].

shows that the final design can detect transitions in 45 seconds with a 95.5% accuracy and in 2 minutes 10 seconds with an 100% accuracy.

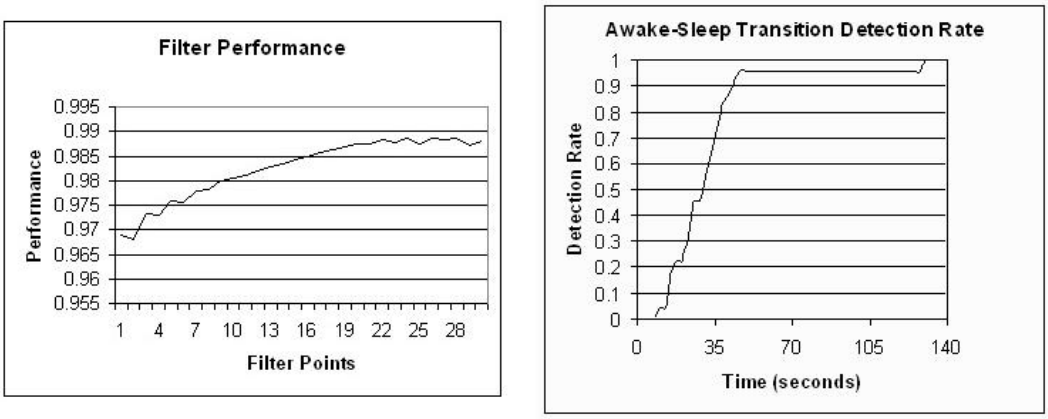


Figure 3.2: Left: Performance improvement with post-filtering. Right: Performance of overall ANN.

From the analysis above, a network using a topology of 4x5x1 would correctly identify sleep states approximately 96% of the time. The network is based on the "mean" feature. The hidden layer uses "logsig" transfer function. The training algorithm, the number of epoch, and the learning rate are "trainrp", 100, and 0.01, respectively. The output of the network is filtered across 50 seconds.

3.2 Hidden Markov Model

This section will document the construction of a HMM-based classifier to differentiate awake and asleep. In this instance, the goal is to adjust the parameters of an HMM $\lambda = (A, B, \pi)$ to maximize $P(O|\lambda)$ such that the model recognizes the observation sequence resembling sleep pattern.

HMM Toolkit

Several toolkits were examined. One package for discrete HMM was UMDHMM v. 1.02 by Tapas Kanungo at University of Maryland. This package can be downloaded at `http://www.cfar.umd.edu/~kanungo`. This package was in C. Another package was HTK Toolkit distributed by University of Cambridge

at http://htk.eng.cam.ac.uk/. This power package designed for HMMs used for speech processing was more complicated than the needs of this experiment. For continuous HMMs, the package examined was from Kevin Murphy and it is based in Matlab. This toolkit can be downloaded from http://www.ai.mit.edu/~murphyk/Software/HMM/hmm.html. The rest of the system was built in Matlab v6.5.

Source Data

Like the ANN, this system was designed based on the Physiobank data file sc4002e0. The system was tested on the remaining patient data files.

Feature Extraction

While spectral features were considered, other features were considered as well.

One feature was from the chaotic feature set called fractal dimension. This feature was found to be a relatively weak indicator of sleep versus awake, in that its performance fluctuates between the signal channels and the patient data sets. It was also unable to provide better resolution in classification, meaning that it cannot discriminate between the different sleep stages.

Another set of features considered, Hjorth parameters, are features that did not require transforming into the frequency domain. Their definitions are in Appendix 7.2. When Hjorth parameters, activity, mobility and complexity, were calculated for 30 second segments of data and mapped to a 3-dimensional vector space, it was found that the vectors from sleep state are distinctly different from awake state. Therefore, they are selected to produce the observation sequences for the HMMs.

State Transitions

The state transitions during sleep is shown in Figure 3.3[4]. Assuming that the model is designed with Awake as the starting state, the sleep cycle normally has the transition sequence 1-2-3-4-3-2-REM. Other transition sequences exist but only following the directed arrows. Therefore, the HMM will contain these states and the allowed transitions.

[4]Note that this diagram assumes a health adult patient and it does not consider cases such as parasomnia where abrupt transitions from Stage 4 to wakefulness can occur.

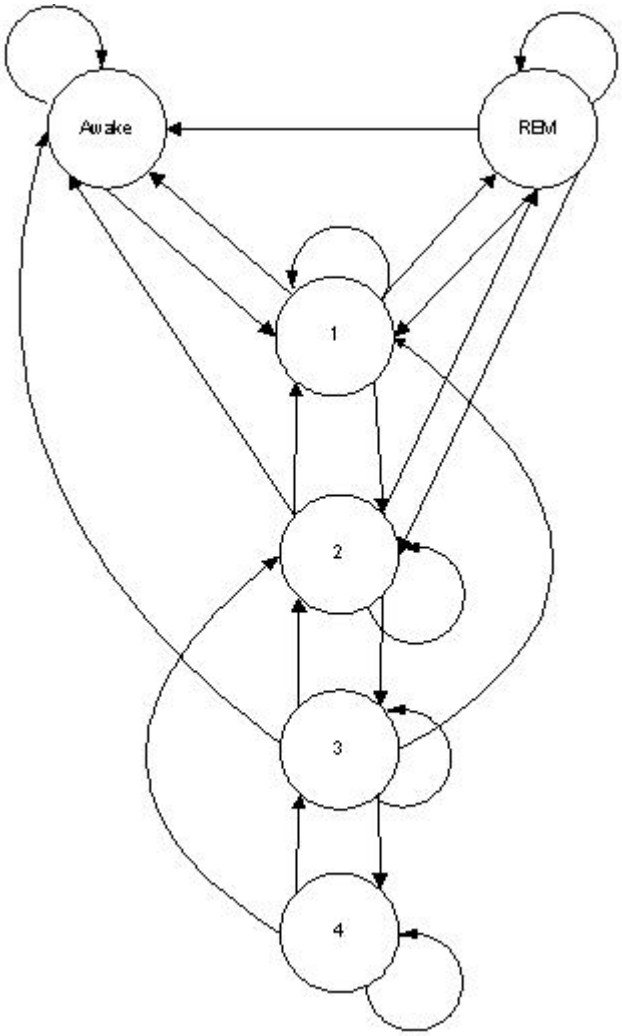


Figure 3.3: Sleep state transition diagram.

Upon experimentation, the complexity of individual states was such that they needed to be modeled with their own HMMs. Within these HMMs, there are states that do not directly correlate to any biological state. Instead they are true hidden states in the sense that the model as a whole represents a specific sleep stage, while the states within the model represent underlying order in the EEG signal. The number of hidden states in each model will affect the performance of the model in terms of ability to approximate the

sequence. At the same time, the number of hidden state also poses storage and processing challenges. Based on a quick experiment to balance between system accuracy and computational cost, the number of hidden state within each model is determined to be 6.

Building the HMMs

Initially, the design used discrete HMMs, because the sleep stages seemed like a set of distinct states that can be individually identified. For each sleep stage, an HMM is built by training it with the Hjorth feature set calculated on the 30 second data epochs of that sleep stage. Currently, the training algorithm used is Baum-Welch parameter re-estimation [52]. According to Rabiner, there is no known way to analytically solve for the model which maximizes the probability of the observation sequence [42]. Therefore, several iteration of the Baum-Welch method was used to find the local maximum. At the end of this process, the patient's sleep process is characterized by the 6 HMMs, representing Awake, NREM I, II, III, IV, and REM. Each HMMs contain 6 hidden states.

When test data is put into this set of models, the model yielding the highest probability will be selected as the "current state" for this frame. During experiments, it was found that this output can sometimes be wrong but the misclassification can be corrected after validating against a few ensuing epochs. Therefore, by using a smoothing FIR filter, the false classifications can be eliminated.

While working with discrete HMMs, the model used a rudimentary codebook to store the feature values for use as a discrete observation. However, it was found that discrete HMMs have following drawbacks:

- Discrete HMMs require the use of a codebook to store the observation symbols. This approach introduces error because all observations in the observation vector space are rounded to the nearest observation vector in the codebook.

- Discrete HMMs cannot handle observations that were not part of their training. Since this model computes the probability of getting an observation in a given state directly from the raw data, the corresponding probability entry of receiving the symbol is 0. Therefore, the system has the tradeoff between the size of the code book, the accuracy of the

Model, and the length of training data required. Computational cost also became a limiting factor for large codebook sizes.

Therefore, continuous HMMs were considered. Since continuous HMMs' observations are treated as Gaussian random variables with means and variances, they can handle more effectively observation symbols that have not been seen during the training phase. Although more computationally intensive, this approach performed better as it eliminates the need for a code book and the observations can now take on any real values.

Again, a smoothing filter was necessary for the output signal to correct the spurious false alarms. While the performance of the system is improve, it also introduced a delay.

3.3 Comparison between ANN and HMM

Having design 2 classifiers, ANN and HMM, this section compares their performance. In this step, both classifiers were tested on data sets identified as sc4002e0, sc4012e0, and sc4102e0. Note that sc4002e0 is the retrospective results and the other 2 are prospective results.

3.3.1 Detection Accuracy

The first point of comparison is the accuracy of the classifiers. This measure calculates the ratio of true positives and true negative to the total number of cases. Table 3.6 contains the results comparing that of the ANN to that of HMM.

Table 3.6: Detection accuracy comparison between ANN and HMM.

Patient ID	sc4002e0	sc4012e0	sc4102e0	Average
ANN	93.09%	93.16%	89.18%	91.91%
HMM	86.61%	81.43%	64.03%	77.36%

Note that not only is ANN performing better retrospectively, it is also performing well prospectively. ANN held the accuracy fairly steady around 90%. However, the HMM started in the mid 80% and the prospective falls to 64%.

3.3.2 Detection Time

Another point of comparison is the time lag before detecting a transition from awake to asleep. This criterion is important for a classifier used in devices that are designed to sound an alarm when the user falls asleep at an undesirable time. The detection time using both types of classifier are in Table 3.7.

Table 3.7: Detection time comparison between ANN and HMM.

Patient ID	sc4002e0	sc4012e0	sc4102e0	Average
ANN	0 min	0.9 min	0.3 min	0.4 min
HMM	0 min	7 min	3.5 min	3.5 min

The detection time for ANN is under 1 minute while for HMM it goes up to as much as 7 minutes. One of the cause of this difference is that the ANN uses 2.56 seconds of data while HMM uses 30 seconds. Since both uses a smoothing filter to correct false alarms, and the epoch length difference means that ANN would have many more opportunity to correct the error.

3.3.3 False Positive Rates

The last measure considered was false positive rates. While not as critical as accuracy or detection speed, high false positive rate can annoy the user. Table 3.8 shows that both classifiers had similar false positive rates.

Table 3.8: False positive rate of ANN and HMM.

Patient ID	sc4002e0	sc4012e0	sc4102e0	Average
ANN	0%	9.33%	22.73%	10.87%
HMM	0%	6.22%	23.01%	10.03%

3.4 Summary

In this section, it was found that the neural network MLP can be easily configured to differentiate sleep and awake. The best spectral feature is the average power in each of the 4 bands with a prospective sensitivity of 95.7% and specificity of 95.7%. The topology selected to balance between

storage, computation, and accuracy was 4x5x1. The MLP was trained by resilient backpropagation training method with a learning rate of 0.05. When validated across 3 patients, the MLP has approximately 90% accuracy and a false positive rate of approximately 10%.

Continuous HMMs, built with with 6 hidden states each, are designed for the sleep stages. The HMMs were trained with Hjorth parameters as input. The outputs are probabilities that a certain input set is part of that sleep stage. Therefore, the system uses a maximum probability selector to identify the most likely stage. HMM has an approximate accuracy of 77.36% and a false positive rate of 10.03%. The performance of HMM can most likely be improved if their model was better tuned to represent the biological states.

MLP not only outperforms HMM in terms of accuracy, it was also more easy to build and optimize its configuration. Nevertheless, it is expected that the difference in feature set also plays a significant role. While the MLP can recognize its training set fairly well, its performance with new test data can be improved.

To bridge some of the remaining performance gap, new features should be designed to better imitate the analytical process by the human scorers. In the next section, improved EEG features are identified. Aside from EEG information, EOG and context information are studied in ensuing sections.

Chapter 4

EEG Feature Extraction

After the use of spectral features and the Hjorth (time-domain) features in the previous chapter, it is necessary to look at other potential features that can be extracted from EEG to offer additional information. Based on the sleep staging rules, EEG contains two types of information, band activities and characteristic waves.

4.1 Band Features

When the EEG bands are considered by the human scorers, they look at frequency, amplitude, and duration. This process is analogous to mentally producing the time-frequency representation(TFR) of the signal. Two common and effective methods of TFR, scalograms and spectrograms, are examined for features to represent band power, band time, and mixed band activity.

This section first describes the training data and sets the evaluation bases. Then it documents the process of selecting features and determining feature settings.

4.1.1 Data

The features are defined based on one set of 8 hour sleep study data provided by the Physiobank (physiologic signal archives for biomedical research). This set was selected at random from the 8 studies stored in the Physiobank. The Physiobank stores the sleep study data in the European Data Format (EDF).

The features are designed for the EEG channel Fpz-Cz[1], as this signal was one of the two available from the Physiobank data set. The EEG is segmented into 30 second epochs to match the human scorer' classification. As a result, there are 1884 segments of Awake epochs, 59 Stage I epochs, 373 Stage II epochs, 94 Stage III epochs, 203 Stage IV epochs, 215 REM epochs, and 1 movement time segment. Each set of epochs are stored in Matlab data files.

4.1.2 Evaluation Basis

Since accuracy determines the features' utility value, it will be the first evaluation criterion. TFR are known for their computation complexity, and have been ignored in the past due to efficiency requirements. However, current processing capabilities have made it possible for TFR to be applied more widely, but their efficiency should still be considered as a second evaluator.

Accuracy

The band features are evaluated based on their ability to differentiate the relevant sleep stages. The relation between the bands and the sleep stages are indicated in Table 4.1. Some bands like the δ band is used as the primary defining feature for NREM II, III, and IV. Other bands like α are only part of stage definitions for NREM I and II. Finally, the third type of relation exhibited in β band is tertiary in that it is included in the description but does not act as a defining feature. The expectation of feature performance in each band depends on the level of relation. This expectation will later be reflected as a weight in the final system.

Efficiency

The time efficiency in the calculation of features will determine whether the feature is feasible for sleep staging. The calculation of spectrogram and scalogram both depend heavily on the setting of their parameters. Scalograms can be affected by the number of levels and the scaling factors. Spectrograms are affected by the window size and the amount of overlap. The trend these

[1]The specific parameters of these features will be specific to this channel, but the same methodology can be reapplied to other channels.

Table 4.1: Level of relation between bands and sleep stages (*** primary, ** secondary, * tertiary definition).

Band	Awake	NREM I	NREM II	NREM III	NREM IV	REM
α	*	**	**			*
β	*	*				*
θ	*		*	*		*
δ	*		***	***	***	*
MBA	***					**

.

settings have on the calculation time requirement can assist in later analysis, such that a good balance between accuracy and complexity can be selected.

The time analysis is conducted on a Pentium M 1.5GHz computer with 480MB of RAM running Windows XP. During the test, only system processes and the test script are running.

Scalogram The test script times the function *cwt* in Matlab 7. The continuous wavelet transform is affect by levels - the number of scales and scale factor - the level of wavelet compression. The scales are formed by $[1:1:lvl]/SF$. The time analysis for scalogram considers 5 levels (lvl) and 6 scale factors (SF) as listed in Table 4.2. For each scale, 10 trials are conducted. Table 4.2 shows the average time, the standard deviation in time, and the total time[2].

In general, the time required to calculate a scalogram increases almost directly proportional to an increase in the number of levels and to a decrease in the magnitude of scaling factor. This trend is demonstrated in Figure 4.1. Note that the figure shows exponential increase because the axes for levels and for scaling factors are both set to exponential. The lower range of the recorded times shows that the values approach 0.0094 seconds per scalogram. Despite the reduced number of calculations required, the processing overhead remains constant.

The average and the total times are more consistent when the number of levels are higher and when the scaling factors are lower. At the opposite corner, the average values do not always follow the trend noted in the previous paragraph. However, the total time better preserves this trend, possibly due to fewer clocking operations. The standard deviation is relatively stable

[2]Total time avoids the repeated timing function calls.

Table 4.2: Time analysis results for generating continuous scalograms (10 Trials - in seconds - SF scale factors).

	Average Time				
Levels	8	16	32	64	128
4	0.0188	0.0140	0.0235	0.0593	0.2063
2	0.0094	0.0140	0.0328	0.0953	0.3375
1	0.0094	0.0203	0.0485	0.1640	0.6016
0.5	0.0125	0.0296	0.0860	0.2984	1.1188
0.25	0.0172	0.0453	0.1500	0.5515	2.2516
0.125	0.0265	0.0797	0.2859	1.1188	4.2453
SF	Standard Deviation				
4	0.0070	0.0088	0.0083	0.0065	0.0067
2	0.0081	0.0049	0.0050	0.0052	0.0110
1	0.0081	0.0074	0.0047	0.0112	0.0132
0.5	0.0066	0.0051	0.0080	0.0048	0.0080
0.25	0.0052	0.0047	0.0084	0.0075	0.2963
0.125	0.0075	0.0050	0.0077	0.0432	0.0504
SF	Total Time				
4	0.094	0.125	0.250	0.609	2.047
2	0.094	0.141	0.312	0.938	3.343
1	0.109	0.188	0.500	1.594	5.953
0.5	0.125	0.265	0.828	2.922	11.172
0.25	0.171	0.453	1.500	5.546	21.485
0.125	0.266	0.797	2.812	10.719	42.250

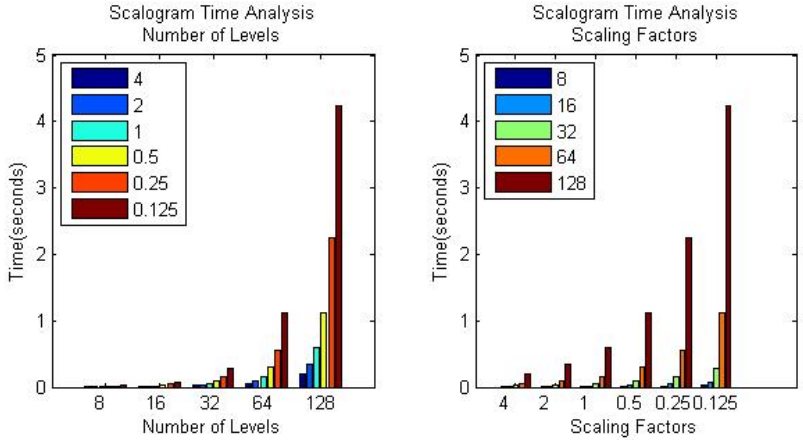
across the settings studied.

Therefore, it is more time efficient to use fewer levels and larger scaling factors in the middle of the ranges studied.

Spectrogram This test looks at the processing time of the Matlab function *specgram*. For spectrogram, efficiency is tested according to the window size and the amount of overlap. Window sizes tested are 30, 100, 150, 300, 500, and 1000. The overlaps are 16.7%, 33.3%, 50%, 66.7%, 83.3%, and $\sim$ 100% which in calculation uses window size minus one. Again, 10 trials are conducted for each settings. The results are recorded in Table 4.3.

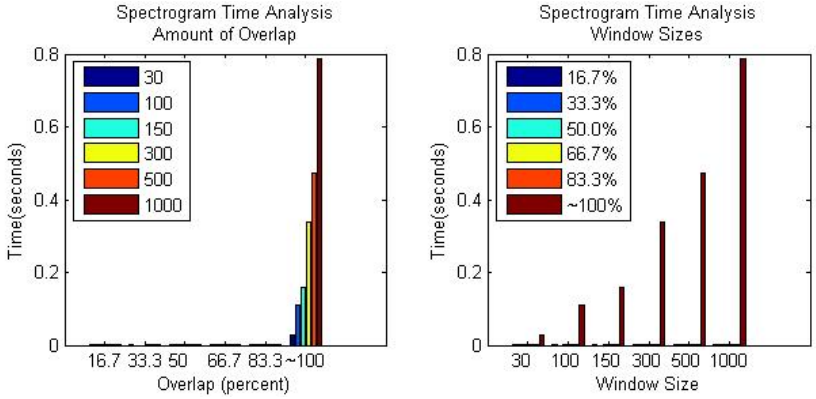
The similarities in the calculation time for window sizes 30 to 500 indicate

Figure 4.1: Comparison of calculation times for scalograms.



that the overhead of the function dominate the time requirement. However, for window size 1000, the calculation time increases significantly from that of 500. This window size also shows that the calculation time increases proportional to the increase in the amount of overlap. Figure 4.2 emphasizes the difference in time requirements. Note that in general the calculation time for scalograms are 10 times longer than the spectrograms.

Figure 4.2: Comparison of calculation times for spectrograms.



Therefore, it is more time efficient to select window sizes less than 1000 with any amount of overlapping.

Table 4.3: Time analysis results for generating continuous scalograms (10 Trials - in seconds - WS window size - OL overlap).

	Average Time					
WS	30	100	150	300	500	1000
16.7%	0.0016	0.0015	0.0016	0.0015	0.0032	0.0297
33.3%	0.0015	0.0000	0.0016	0.0015	0.0016	0.1094
50.0%	0.0015	0.0000	0.0016	0.0016	0.0031	0.1594
66.7%	0.0015	0.0016	0.0016	0.0015	0.0031	0.3375
83.3%	0.0016	0.0016	0.0015	0.0016	0.0016	0.4718
$\sim$ 100.0%	0.0016	0.0016	0.0015	0.0031	0.0016	0.7875
OL	Standard Deviation					
16.7%	0.0051	0.0047	0.0051	0.0047	0.0067	0.0048
33.3%	0.0047	0.0000	0.0051	0.0047	0.0051	0.0005
50.0%	0.0047	0.0000	0.0051	0.0051	0.0065	0.0067
66.7%	0.0047	0.0051	0.0051	0.0047	0.0065	0.0080
83.3%	0.0051	0.0051	0.0047	0.0051	0.0051	0.0070
$\sim$ 100.0%	0.0051	0.0051	0.0047	0.0065	0.0051	0.0080
OL	Total Time					
16.7%	0.000	0.031	0.016	0.015	0.032	0.297
33.3%	0.015	0.000	0.016	0.015	0.016	1.109
50.0%	0.000	0.016	0.016	0.015	0.032	1.609
66.7%	0.000	0.016	0.015	0.016	0.031	3.391
83.3%	0.015	0.016	0.016	0.015	0.047	4.719
$\sim$ 100.0%	0.031	0.016	0.047	0.031	0.015	7.829

4.1.3 Band Power

The *band power* of the four bands are grouped together because their derivation will be very similar. Since *band power* requires significant resolution in particular frequency bands, features derived from scalograms are investigated. Scalogram is defined in Equation 9. The various formulations are defined discretely as they would be applied in Matlab. They are

1. *Total Power* calculates the total power covered by the scales that convert to frequencies within a band definition. Its equation is

$$tpwr(x) = \sum_{t} \sum_{f=L}^{f=U} |SCAL(t, f)| \tag{4.1}$$

where x is the band in question, L is the lower frequency bound and H is the upper frequency bound of band x. This feature is sensitive to the width of the band. For instance, β covers far more frequency ranges than the other bands.

Another issue arises from a change in epoch size. Epoch size will affect the width of the scalogram which in turn affects the value of the total power. Consequently, changes in epoch size will require changing the thresholds and the membership functions.

2. *Average Power* calculates the average power covered by the scales corresponding to a band. It is defined as

$$avgpwr(x) = mean_t(mean_{f=L}^{f=U}|SCAL(t,f)|) \tag{4.2}$$

The average power avoids the problem where certain bands are wider than others. It is also independent of epoch size changes.

One concern is that depending on the equipment the signal strength may vary, again making defining thresholds and membership functions difficult. Aside from signal strength, noise level may also cause changes in power.

3. *Relative Power* finds the ratio of the average power in a band to the sum of the average powers all four bands. It is defined as

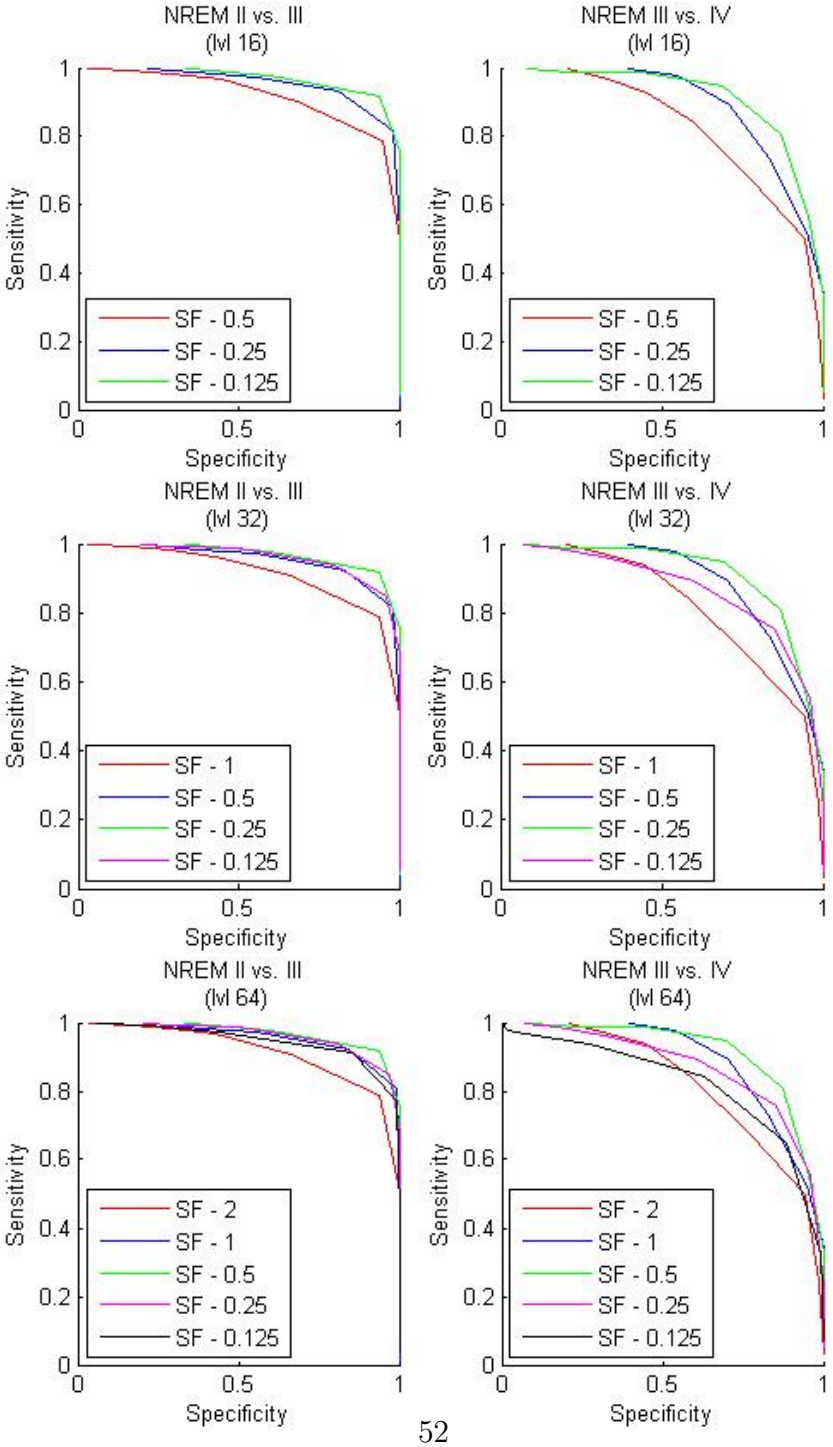
$$relpwr(x) = \frac{avgpwr(x)}{avgpwr(\alpha) + avgpwr(\beta) + avgpwr(\theta) + avgpwr(\delta)} \tag{4.3}$$

This feature is normalized, therefore less likely to be affected by different equipment.

Delta Power

Average power in the delta band can be used to differentiate II from III and III from IV. The ROCs of this feature at various settings are shown in Figure 4.3. The ROCs show that the best performance for both cases occurs at the settings (LVL=16, SF=0.125), (LVL=32, SF=0.25), and (LVL=64, SF=0.5). In fact, the ROCs are the same for all pairs of LVL and SF where $LVL \cdot SF$ remains constant. This phenomenon is due to the same range of

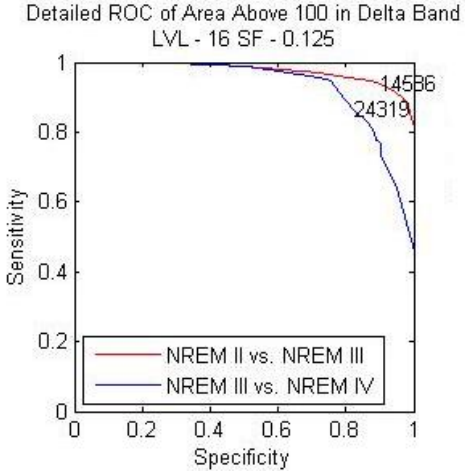
Figure 4.3: ROC of average power in delta band to differentiate NREM II, III, and IV using 16, 32, and 64 levels.



scales are covered when $LVL \cdot SF$ are held constant[3]. Since performance is the same, the setting of 16 levels, which implies simpler computations, is selected for further investigation. For (LVL=16, SF=0.125), the frequencies in delta band consist of 14 elements ranging from 0.5208Hz to 2.7778Hz, which provides a good coverage of the delta band.

The best thresholds are identified through iteratively increasing the resolution in the ROC of (LVL=16, SF=0.5). The detailed ROCs are shown in Figure 4.4 on the left. They show that the best threshold to differentiate NREM II from NREM III is approximately 76.8 with a sensitivity of 91.7% and a specificity of 93.6%. The best threshold to differentiate NREM III from NREM IV is 116.1 with a sensitivity of 80.9% and a specificity of 87.2%.

Figure 4.4: ROC of average power in delta band (LVL=16 SF=0.125).



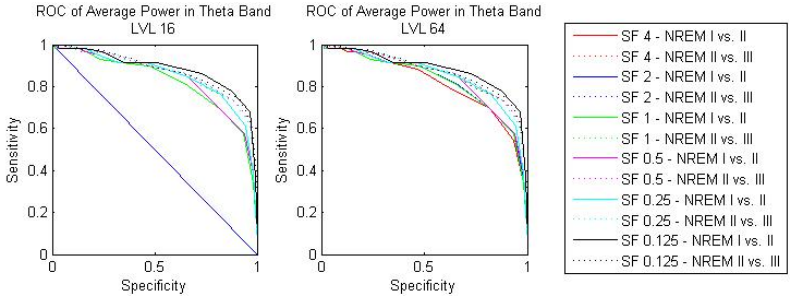
Theta Power

Average power in the theta band can assist in the differentiation of NREM I from II and NREM II from III. The ROC using the setting (LVL=64) is shown in the Figure 4.5. By using 64 levels, the most number of behavior types can be observed. The figure shows that all the settings work reasonably well, with the scaling factor 0.125 being the best in both scenarios. Again,

[3]The range of $LVL \cdot SF$ is $[2^3, 2^7]$, and these 5 types of behavior can all be observed using 64 levels. Therefore, in later analysis, only LVL=64 settings need to be generated to study all the behaviors

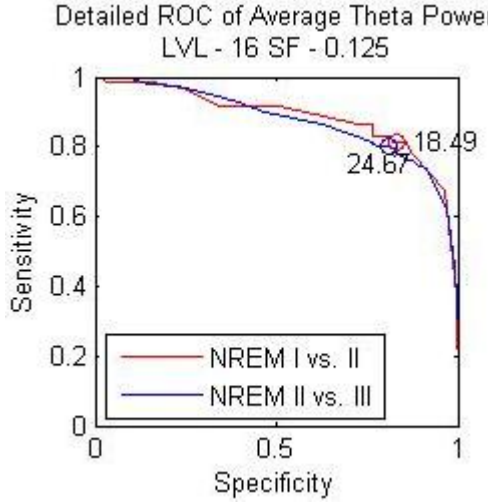
based on behavior consistency in relation to $LVL \cdot SF$, the least number of levels required is 16 and the scaling factor of 0.125.

Figure 4.5: ROC of average power in theta band (64 levels).



Again, thresholds are determined for the discrete rule base by iteratively refining the resolution. The results are shown in the left plot of Figure 4.6. To differentiate NREM I from II, the threshold is 18.5 with a sensitivity of 81.2% and a specificity of 83.1%. To differentiate NREM II from III, the threshold is 24.7 with a sensitivity of 79.9% and a specificity of 80.9%.

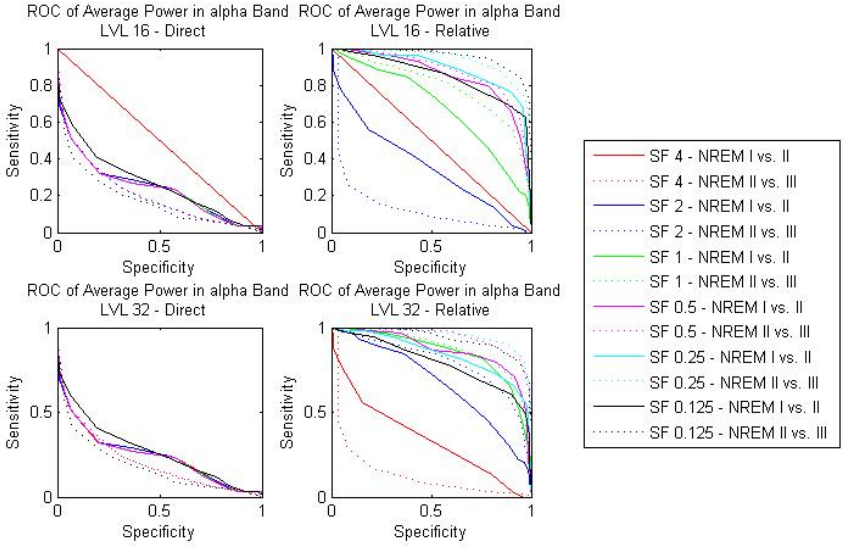
Figure 4.6: Thresholds of average theta power(LVL=16 SF=0.125).



Alpha Power

NREM I and NREM II are differentiated in part by the amount of activity in alpha band. NREM I is supposed to contain more alpha activity. However, the data set suggest statistically that the average alpha power is higher in NREM II than NREM I. The plots on the left side of Figure 4.7 are constructed with the direct power calculations and the expectation of NREM I having more alpha activity. The ROCs are below curving downwards indicate that statistically the alpha power in NREM II is greater than that of NREM I. This phenomenon is caused primarily by a higher power spectrum across all bands in NREM II. In order to achieve consistency with the R&K manual, the feature must be adjusted by using the relative power. Figure 4.7 demonstrates the use of relative power in the right side plots.

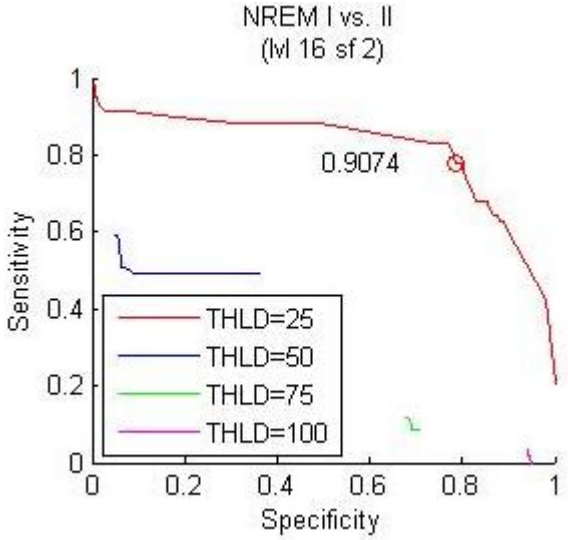
Figure 4.7: ROC of alpha power using direct power and relative power (LVL=16) and (LVL=32).



The settings (LVL=16, SF=0.125), (LVL=16, SF=0.25), (LVL=16, SF=0.5), and (LVL=32, SF=0.5) have comparable performance for separating NREM I from II. The settings for separating NREM II and III are (LVL=16, SF=0.125) and (LVL=32, SF=0.25). Setting (LVL=16, SF=0.125) is selected for both because the scalogram with this setting is already used for delta and theta bands. The thresholds are shown in Figure 4.8. The threshold separating NREM I and II is 0.138 with a sensitivity of 76.3% and a specificity of 76.9%.

The threshold separating NREM II and III is 0.107 with a sensitivity of 90.1% and a specificity of 89.4%.

Figure 4.8: Thresholds for alpha power (LVL=16, SF=0.125).



Beta Power

Beta power is used to differentiate NREM I, Awake, and REM. Figure 4.9 contains the ROCs of average power(top row) and relative power(bottom row) in beta band with (LVL=16) and various scaling factors. Note that the ROC of SF=4, which is in red, is covered by the ROC of SF=0.125 in black. While beta power cannot differentiate REM from NREM I, its ability to differentiate Awake from NREM I and Awake from REM is clear. Using relative power, the best scaling factor is 2 for both cases. Direct power shows that SF other than 0.125, the performances are similar. Since another feature will be using the setting (LVL=16,SF=2), this setting is selected for further investigation.

The thresholds are determined on the ROCs in Figure 4.10. The threshold separating REM and Awake is 5.098 with a sensitivity of 97.6% and a specificity of 96.3%. The threshold separating NREM I and Awake is 5.682 with a sensitivity of 94.5% and a specificity of 94.9%.

Figure 4.9: ROC of average and relative beta power using (LVL=16) to differentiate NREM I, REM, and Awake.

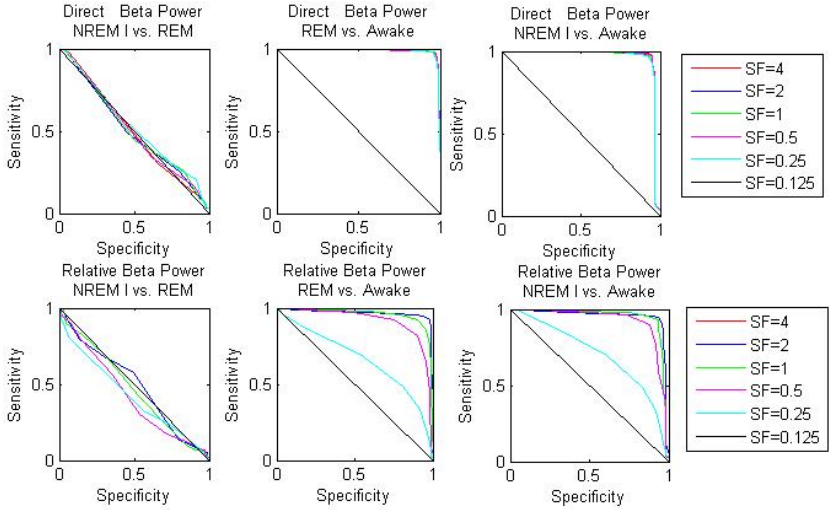
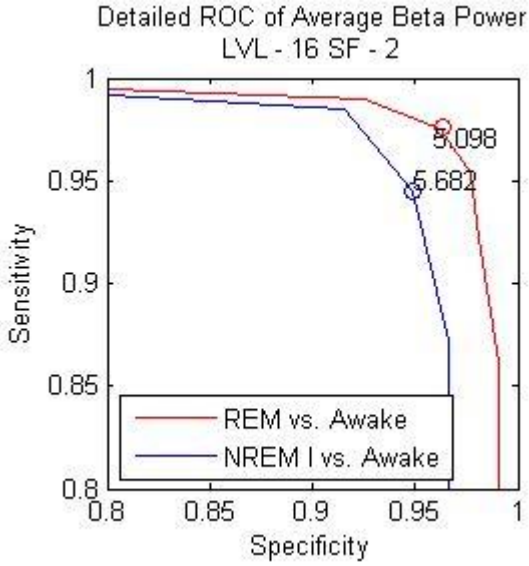


Figure 4.10: ROCs of beta power (LVL=16, SF=2)



4.1.4 Band Time

Unlike power, *band time* does not have any obvious formulations. The key is to identify the concept of band activity in terms of frequency and amplitude

and measure that activity along the time axis. One way of characterizing band activity is to identify regions in the scalogram exceeding some threshold. The following features can be considered,

1. *Area* of Region Above Threshold is defined as

$$area(x, thld) = \sum_{t} \sum_{f=L}^{f=U} (|SCAL(t, f)| > thld) \tag{4.4}$$

   where *thld* is the threshold.

2. *Relative Area* is the normalized version of the *Area* feature. It is defined as

$$rel_area(x, thld) = \frac{area(x, thld)}{area(\alpha, thld) + area(\beta, thld) + area(\theta, thld) + area(\delta, thld)} \tag{4.5}$$

3. *Width* of Region Above Threshold is defined as

$$width(x, thld) = \sum_{t} \bigcup_{f=L}^{f=U} (|SCAL(t, f)| > thld) \tag{4.6}$$

Delta Time

The area above a threshold in the delta band is analogous to the duration of delta activity, which can also differentiate NREM II from III and NREM III from IV. Aside from selecting good settings for the number of levels and the scaling factor, the threshold must also be considered.

First, the number of levels and the scaling factor are selected. From the analysis of average power in delta band, it is noted that the full range of behavior can be uncovered using 64 levels. Figure 4.11 shows the results. The ROCs of the same scaling factor despite different thresholds are shown in the same color. The set of ROCs in green, meaning SF=0.5, have the best performance. Again to reduce computation complexity, the lowest number of level required to achieve this type of behavior, which is 16, is selected.

Using (LVL=16,SF=0.125), the thresholds 25, 50, 75, 100, 150, 200, and 300 are considered. The ROCs are plotted in Figure 4.12. Since the thresholds 25, 50, 75, 100, and 150 have similar performance, Figure 4.12 zooms

Figure 4.11: ROC of area above thresholds in delta band to differentiate NREM II, III, and IV using 64 levels.

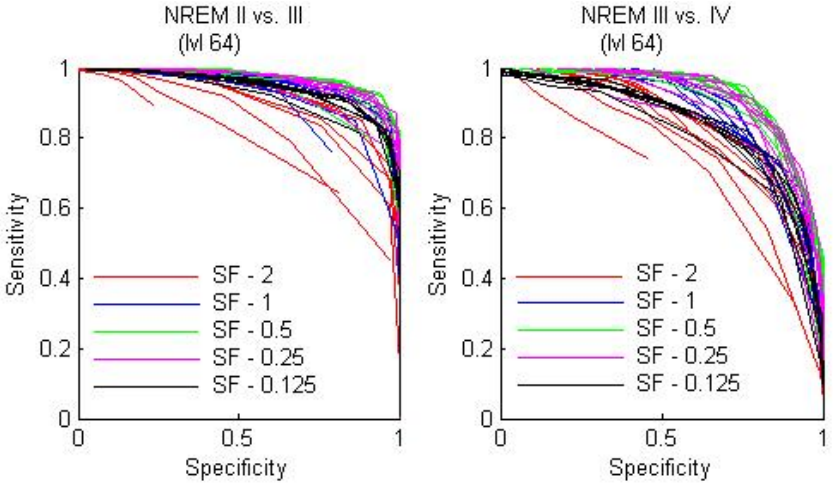
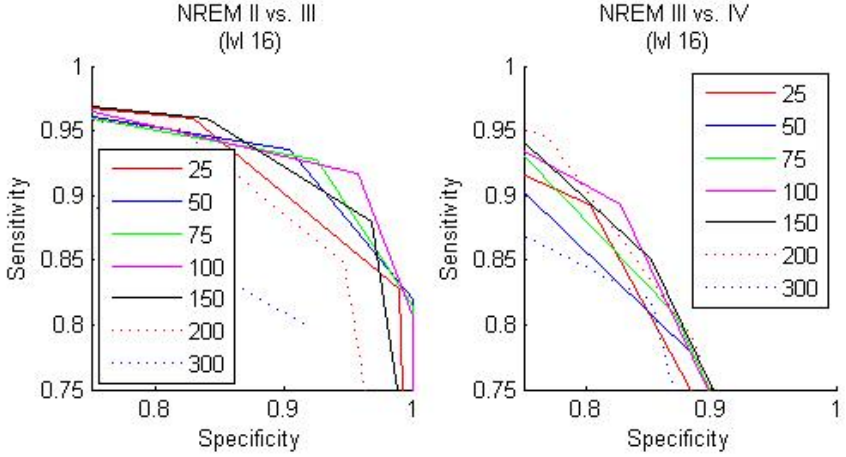


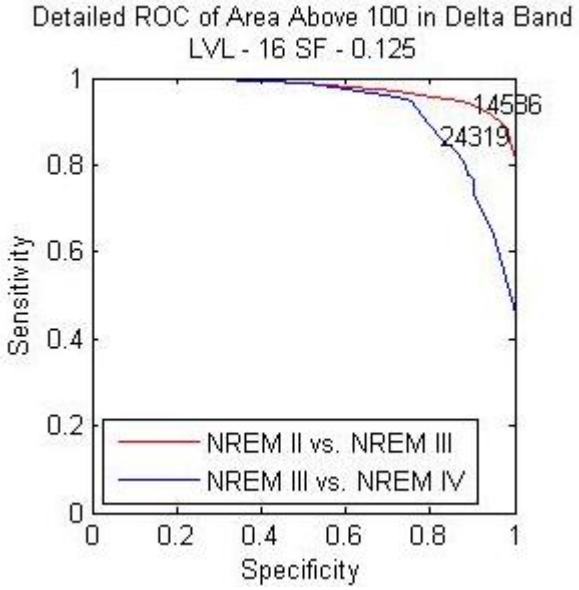
Figure 4.12: ROC of area above thresholds in delta band with the setting (LVL=16,SF=0.125).



into the 0.75 to 1 range on both axes. The ROC for a threshold of 100 demonstrates the best performance.

Using the setting (LVL=16,SF=0.125,THLD=100), the ROC in Figure 4.13 is generated. Using three iterations, the actual thresholds were set to be 14500 and 24300 respective to differentiate NREM II-III and differentiate NREM III-IV.

Figure 4.13: Thresholds of area above threshold in delta band (LVL=16, SC=0.125, THLD=100).



Theta Time

Adopting LVL=16 for consistency with the feature average theta power, the ROCs of various scaling factors are generated and shown in Figure 4.14. The yellow ROCs representing SF=0.125 have the best performance for differentiating NREM I from II and NREM II from III. Figure 4.15 zooms in on the ROCs for (LVL=16, SF=0.125) to select a good threshold. The red ROC indicates that the height threshold of 25 presents the best performance.

Using iteration and the setting (LVL=16, SF=0.125, THLD=25), the theta time threshold to differentiate NREM I and II is set at 762 with a sensitivity of 83.1% and a specificity of 79.9%. The threshold to separate NREM II and III is set at 1169 demonstrating a sensitivity 83.4% and a specificity 83.0%. These thresholds and their corresponding refined ROC are shown in Figure 4.16.

Alpha Time

The formulation *area* is tested first. The ROCs using LVL=16 are plotted in Figure 4.17. By the downward curve of the ROCs, this formulation does

Figure 4.14: ROCs of theta time (LVL=16) to select SF.

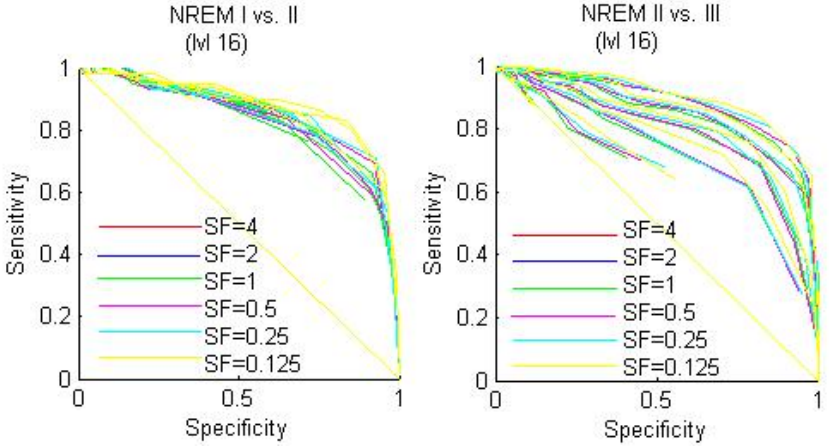
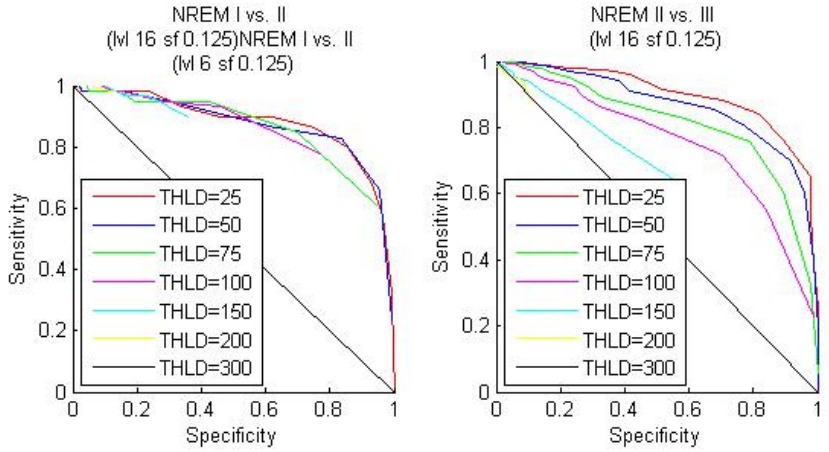


Figure 4.15: ROC of theta time (LVL=16, SF=0.125) to select THLD.



not have good predictive value as the alpha time feature. Like alpha power, the normalized or relative version is investigated instead.

The relative formulation is defined in Equation 4.5. The ROCs with (LVL=16) and at various SF and thresholds are plotted in Figure 4.18. The ROCs are colored according to SF. The plot shows that one threshold exists for (LVL=16, SF=2) that provides reasonable performance in separating NREM I from II. However, there is no setting of appropriate performance to separate NREM II from III. Therefore, only the performance of separating NREM I from II is investigated in the rest of this section.

Figure 4.16: Thresholds for theta time (LVL=16, SF=0.125, THLD=25).

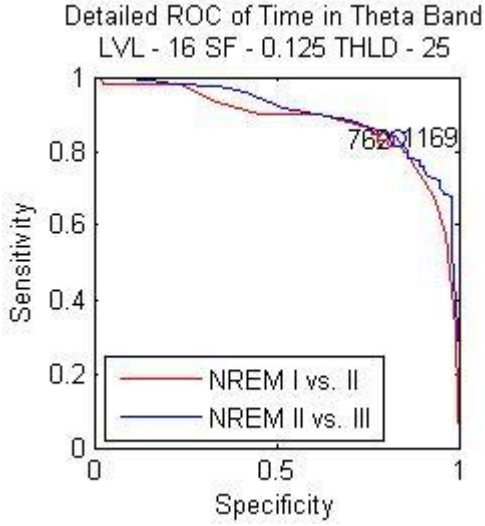


Figure 4.17: ROCs of alpha time (LVL=16) using the formulation of *area*.

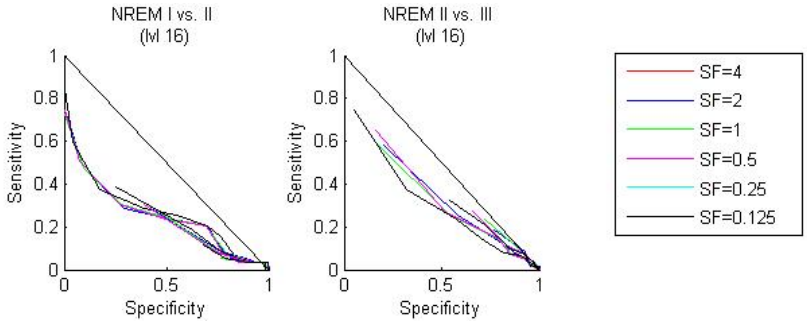
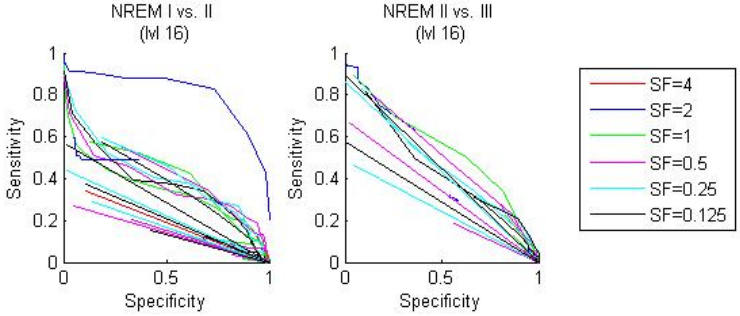
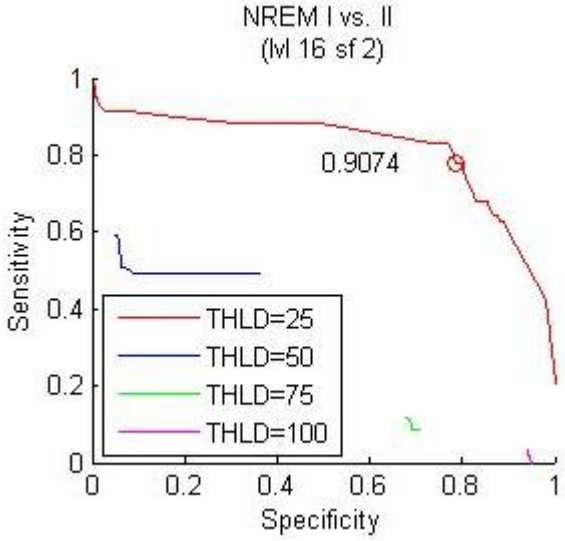


Figure 4.18: ROCs of alpha time (LVL=16) to select SF.



Using the setting (LVL=16, SF=2), ROCs of various thresholds are generated and plotted in Figure 4.19. It identifies the useful height threshold is 25 with the ROC in red. The threshold to differentiate NREM I and II is 0.9074 with a sensitivity of 78.0% and a specificity of 78.6%.

Figure 4.19: ROC and threshold of alpha time (LVL=16, SF=2).



Beta Time

Using LVL=16, the ROCs of various scaling factors and height thresholds are calculated and plotted in Figure 4.20. The ROCs are colored according to the scaling factors. Most scaling factors demonstrated that at some height threshold their performance will be significant. Since other features often use the scaling factor 2, this setting will be investigated further.

The ROCs using the setting (LVL=16, SF=2) are plotted in Figure 4.21. They indicate that the best height threshold is 25. The respective thresholds are also plotted in Figure 4.21. The threshold separating NREM I and Awake is 312 with a sensitivity of 92.9% and a specificity of 93.2%. The threshold between REM and Awake is 256 with a sensitivity of 94.4% and a specificity of 94.4%.

Figure 4.20: ROC of beta time (LVL=16).

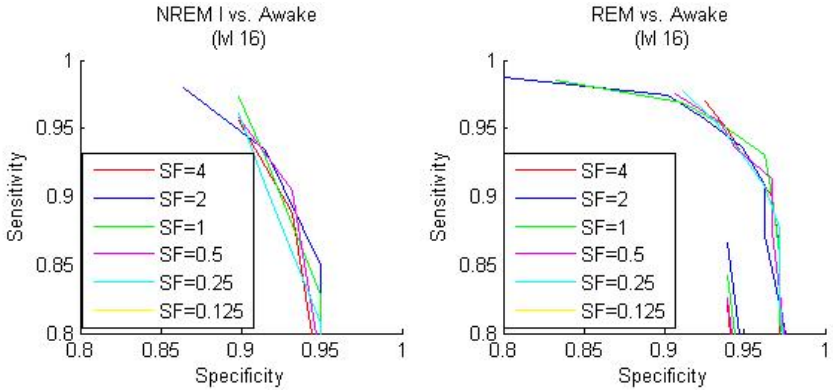
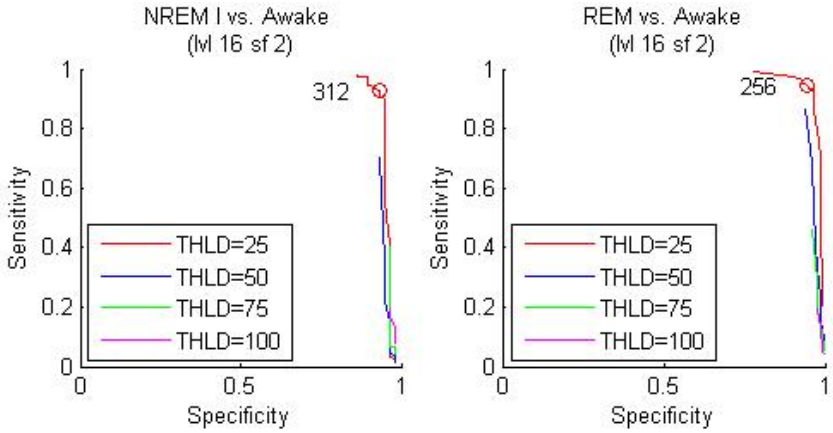


Figure 4.21: ROCs and threshold of beta time (LVL=16, SF=2).

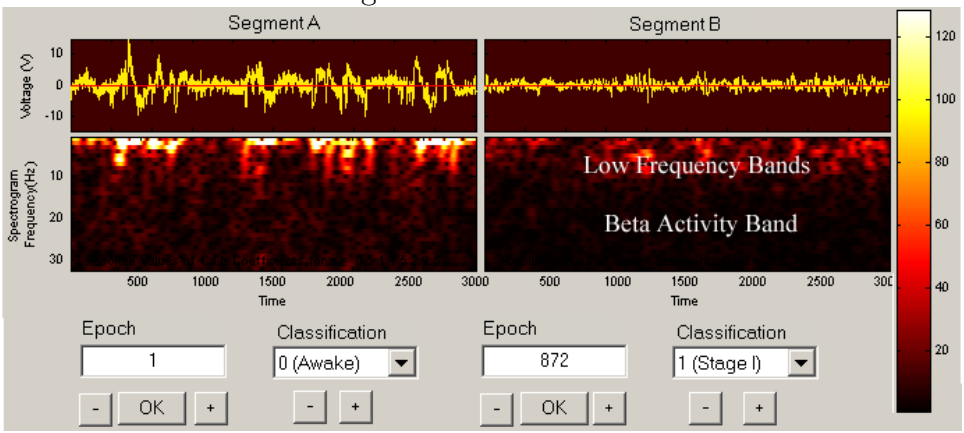


4.1.5 Mixed Band Activity

Since mixed band activity (MBA) does not have any obvious formulations, the features are first formulated through visual analysis. For convenience, visual analysis tools were developed and documented in Appendix **??**.

Awake and REM stages both contain mixed frequency activity, which means that to some degree $\alpha, \beta, \theta, \delta$ waves are all present. Figure 4.22 shows the manifestation of MBA in the spectrograms of an Awake epoch compared against a Stage I epoch. Note that there is a concentration of activities at very low frequencies and then a relatively higher level of activity through all the bands compared with Stage I.

Figure 4.22: GUI to compare two epochs of EEG data with their spectrogram, continuous and discrete scalograms.



Mixed frequency activity has two characteristics, intense activity spread across low frequency ranges, and notable activity in the beta band appearing as slightly brighter spots. The following four features were considered,

1. *Standard Deviation* across the entire frequency range is defined as

$$std_1 = std(SPEC(t, f)). \tag{4.7}$$

Its applicability stems from higher ripples in the spectrogram of the Awake stage. Since no particular time event need to be located, it is safe to average across time. In order to reduce the effect of averaging across positive and negative valleys, a second feature is calculated as the absolute value of the spectrogram, which is defined as

$$std_2 = std(|SPEC(t, f)|) \tag{4.8}$$

2. *Energy Ratio* between beta band and other lower frequency bands is determined as

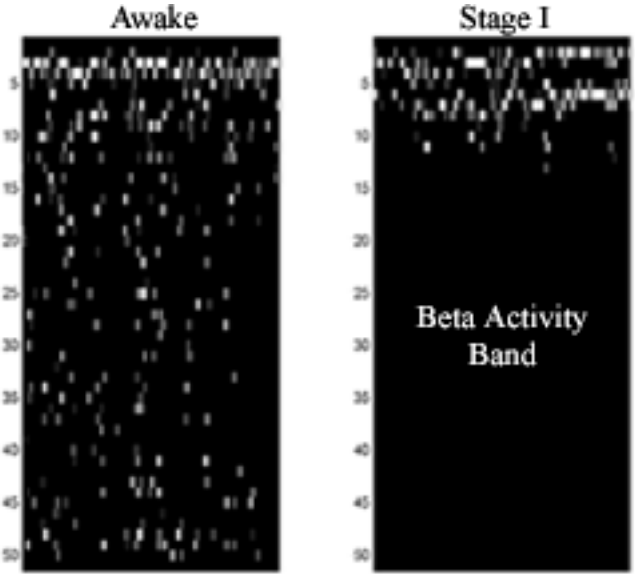
$$ratio = \frac{\sum_{f=L_\beta}^{f=U_\beta} SPEC(t, f)}{\sum_{f=L_{\alpha,\theta,\delta}}^{U_{\alpha,\theta,\delta}} SPEC(t, f)} \tag{4.9}$$

This feature draws from the fact that the energy under the lower frequencies is high regardless of mixed frequency activities. However, the

energy under beta band varies greatly depending the amount of mixed frequency activity.

3. *Edge* detection in the beta band uses edge detection function in Matlab to identify the stronger ripples occurring during mixed frequency activity. Through experimenting with the derivative estimators, the Laplacian of Gaussian method provided the best results. A sample image is provided in Figure 4.23.

Figure 4.23: A sample of edge detection in beta band between Awake and Stage I.



Awake

A simple way of comparing these four features is to use them individually differentiate stages with mixed frequency activity. Figure 4.24 shows the distributions of the feature values for each feature applied to the differentiation of Awake to Stage I. The low overlap in distribution in all four cases shows enough promise that these features are tested on differentiating Awake from all other stages. The results are presented as ROCs in Figure 4.25.

From the receiver operating characteristic curves, it can be noted that std_2 and the band activity ratio both perform very well in all cases. The

Figure 4.24: Distribution of feature values for differentiating Awake from Stage I.

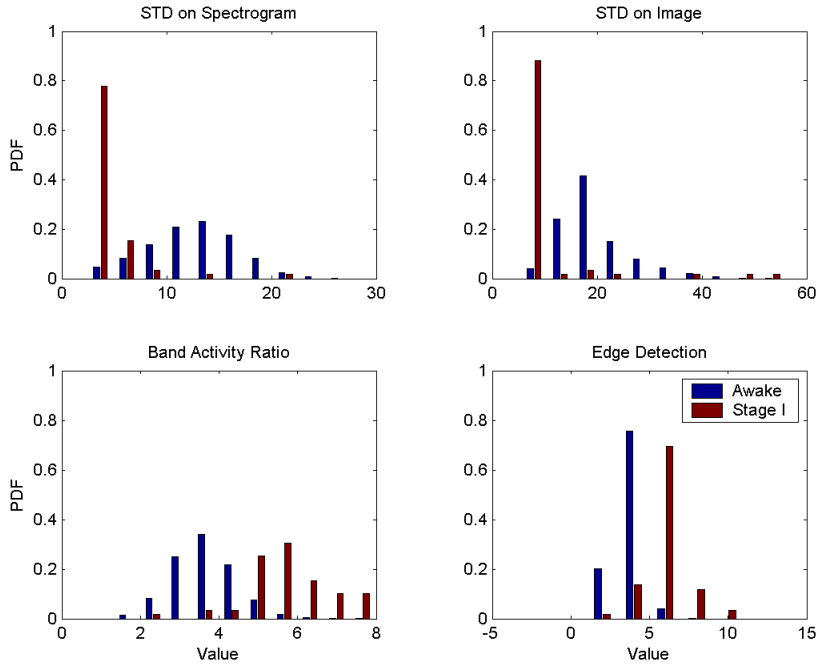


Table 4.4: Performance table for features in mix frequency activity detection.

Feature	Sensitivity	Specificity	Feature	Sensitivity	Specificity
std_1	77.81%	63.70%	Ratio	93.52%	94.60%
std_2	92.36%	95.45%	Edge	86.57%	44.44%

.

differentiation ability of Awake from all other stages using these features are listed in the Table 4.4. These results show that the feature std_2 and band activity ratio have the most consistent and accurate classification of the Awake stages based on mixed frequency activity.

REM

Since REM also demonstrate mixed frequency activity, these features are also applied to REM. The results are presented as receiver operating characteristic curves in Figure 4.26.

Figure 4.25: ROCs of MBA differentiating Awake from other stages.

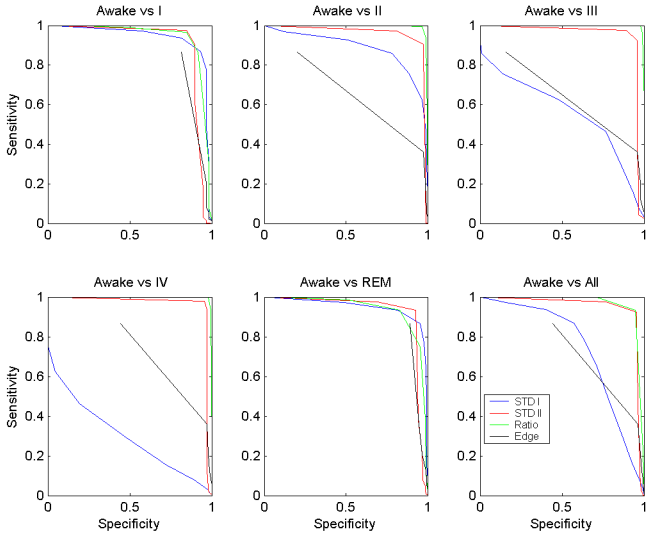
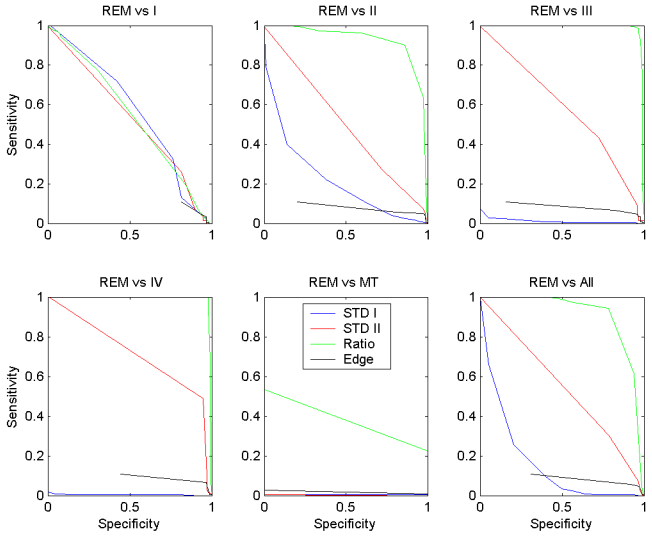


Figure 4.26: ROCs of differentiation of REM from other stages using mixed frequency activity detection.



These features demonstrate that they cannot differentiate REM as well

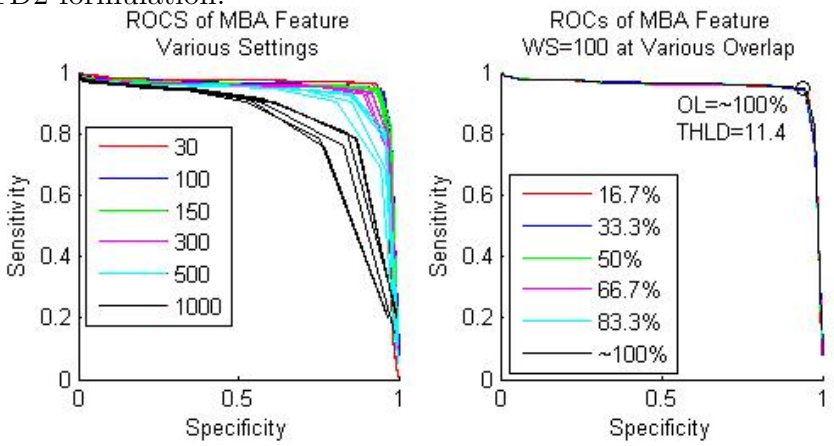
as Awake. The best feature is band activity ratio, which has an overall sensitivity of 94.42% and specificity of 78.49%.

This feature can successfully differentiate Stage II(90.23%, 86.06%), III(98.60%, 96.81%), and IV(99.53%, 98.03%). Its failure to differentiate Stage I from REM is consistent with the nature of Stage I sleep. Stage I is a transitory stage between wake and true sleep, which is generally accepted as the beginning of Stage II. Therefore, Stage I may have remnants of mixed frequency activity from Awake. As these activities are falling away, the behavior may be confused with the consistently low level mixed frequency activity present in REM. This issue, however, is not crippling, because contextual information can correct these misclassifications. Entrance to REM sleep generally occur from Stage II sleep, so the need for differentiating Stage I and REM is low. Furthermore, REM has characteristic behavior in other PSG signals that can further distinguish itself.

MBA Feature Setting for Awake

The window sizes and the amounts of overlap listed in Section 4.1.2 are investigated. The ROCs of differentiating Awake from all other sleep stages are plotted in left graph of Figure 4.27. The ROCs with various levels of overlap are colored according to their window size. The plot shows that the window sizes (WS) 30, 100 and 150 demonstrate similar level of good performance. Window size 100 is selected for further investigation.

Figure 4.27: ROCs of MBA to select settings for differentiating Awake using the STD2 formulation.



The right plot in Figure 4.27 contains the ROCs with WS=100 set to various amounts of overlap. It shows that overlap have little effect on the performance. Since at WS=100 there is little difference in time efficiency, the highest level of overlap $\sim 100\%$ is selected. The threshold is determined to be 11.4 with a sensitivity of 94.8% and a specificity of 94.7%. This threshold is also shown in Figure 4.27.

MBA Feature Setting for REM

Since the previous section demonstrates a good method of identifying Awake, the performance in this section only reflects the differentiation of REM from the non-Awake sleep stages. Using the formulation of energy ratio, the ROCs of various window sizes and overlap are shown in the left plot of Figure 4.28. By far WS=300 provides the best performance.

Figure 4.28: ROCs of MBA to select settings for differentiating REM using the Energy Ratio formulation.

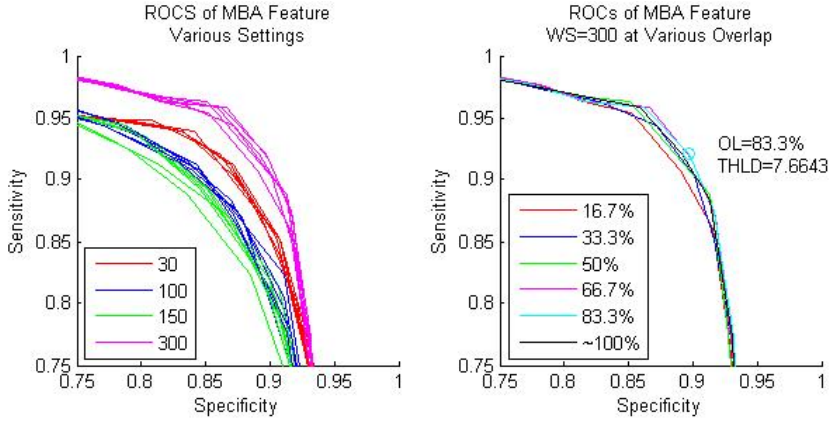


Figure 4.28 shows the ROCs of WS=300 and various levels of overlap. While the performances are similar for various levels of overlap, OL=83.3% is shown to be slightly better. The threshold, which is plotted in Figure 4.28, is found to be 7.6643 with a sensitivity of 92.1% and a specificity of 89.7%.

4.2 Characteristic Wave

In the previous section, EEG features based on time-frequency analysis were examined to better represent band activity. To complement those features, this section looks at characteristic waveforms, i.e. K complex, vertex waves, and sawtooth waves, that are also used to define specific sleep stages. Characteristic waves can be detected using AI methods like ANN. This section records the process of building the detectors for each of these wave types.

4.2.1 Data

The data used in this section is provided by Dr. C. George's sleep lab. The subject was a healthy young male. The EEG data are collected at 80 Hz and scored on 30-second epochs. Based on the scoring provided, EEG of the appropriate stages are examined visually for samples of each wave type. See Table 2.3 for the stage requirement of each wave type. The selected segments are uniformly cropped to 1.5 seconds[4]. The K complexes, vertex waves, and sawtooth waves are shown in Figure 4.29, 4.30, and 4.31. Aside from segments containing the waveforms, segments clearly without the waveforms are also selected. Segments containing the waveform is classed as 1 otherwise 0.

Table 4.5: Performance of ANN in 10 trials

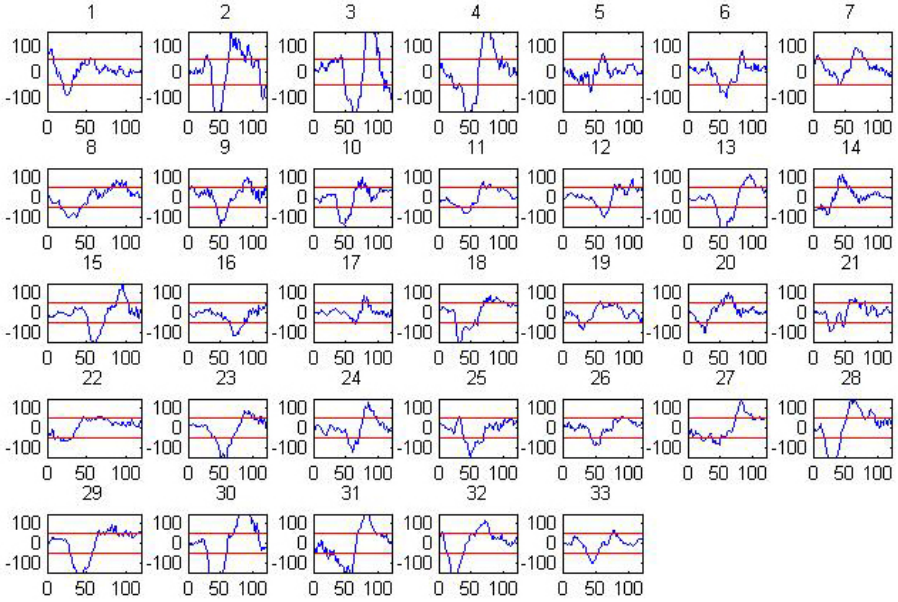
Wave Type	Train Positive	Train Negative	Test Positive	Test Negative
K Complex	20	20	13	15
Vertex Waves	20	20	21	15
Sawtooth	20	20	20	24

4.2.2 ANN Structure

A multi-layer perceptron (MLP) was trained by error backpropagation, using the gradient descent algorithm. For 1.5 seconds, there are 120 data points. These data points are each directed to an input neuron, making the input layer 120 neurons. To reduce the dimensions, ten neurons are chosen for the hidden layer and one neuron is used for the output layer. The first

[4]The minimal duration requirement for K complex is 0.5 seconds long.

Figure 4.29: Sample K complexes.



two layers use *tangent sigmoid* transfer function and the last layer uses *pure linear* transfer function. Since pure linear transfer function outputs a value on a continuous spectrum around 0, the results are converted to 0 and 1 depending on whether the value is greater than 0.5. The learning rate is set to 0.05. Training stops either when an error of less than 0.0001 or when a number of epochs greater than 3000 is reached, whichever occurs first.

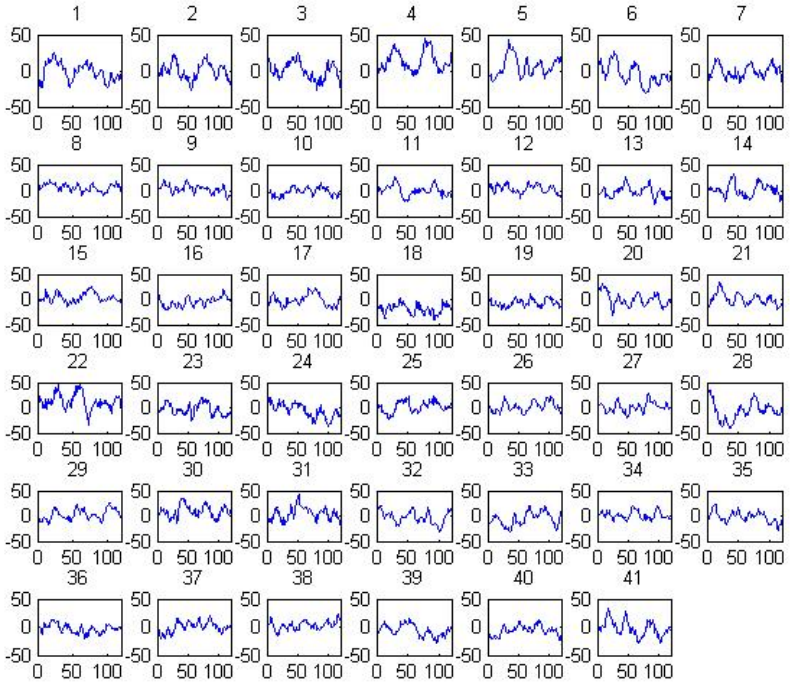
4.2.3 Results

To get a general idea of the performance of these detectors, ten trials are conducted for each wave type. In each trial, the data sets are randomly sorted and their performance tested. The three performance measures are overall accuracy, sensitivity and specificity. Table 4.6 documents their behaviors after retrospective performance has reached 100% on all measures.

4.2.4 Discussion

From the Results Section, it can be seen that the prospective performance of the ANN fluctuates significantly. These performance are not sufficiently

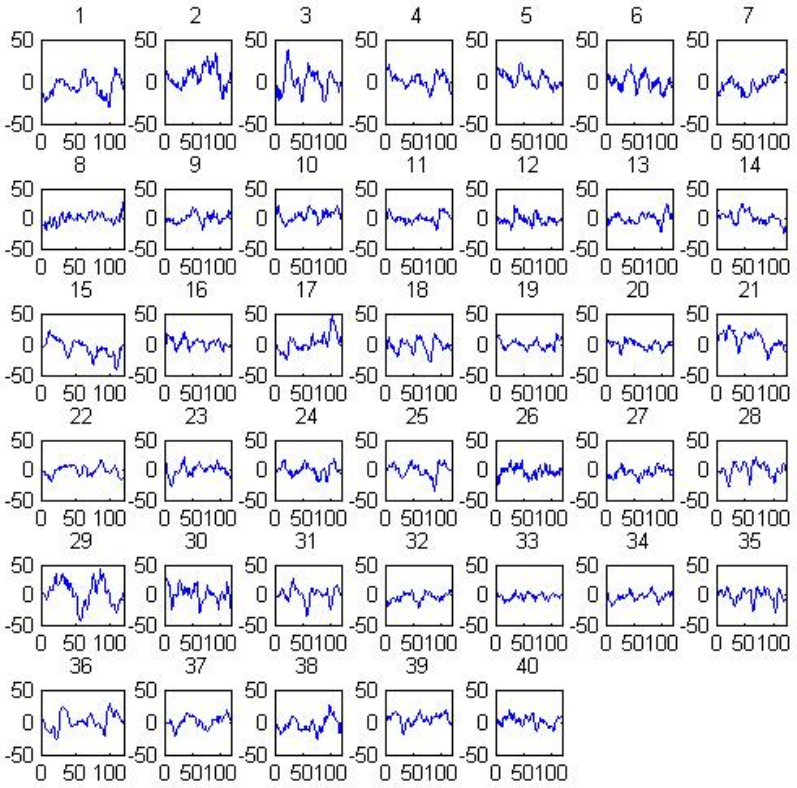
Figure 4.30: Sample vertex waves.



high to be significantly useful as additional information to the band activity. Further studies can be conducted to improve their individual performances. Below are some factors that contributed to the current performance rates,

- *Quality of training and testing data sets* limit the potential in an ANN. The following problems exist in the current training data sets,
  - *Conflict or ambiguity* exists in the training cases. This is evident from looking at the Figures 4.29, 4.30, and 4.31. The impact of this problem can be reduced or eliminated by working with a sleep specialist and building a better training data set.
  - *Real signals* often do not match the technical definition exactly. The human scorers allow some fuzziness around the shape described in the definition. They also qualify a characteristic wave identification as absolutely yes or somewhat or absolutely no.

Figure 4.31: Sample sawtooth waves.



Therefore, instead of classifying to 1 and 0, they use a pseudo-continuous spectrum as of quality value. Such a quality value, if established, should be evaluated by the specialists. Signals artificially generated from the description can overcome the problem of lacking ideal sample waveforms.

– *Case coverage* is another issue. If the training data do not cover all the possible patterns, in particular the extreme cases, the system will misclassify the boundary cases.

– *Data set size* should be bigger. More data will likely mean higher case coverage. Furthermore, it overcomes over fit effect of a complex ANN structure basically memorizing the training data. Also,

Table 4.6: Performance of pattern detecting neural networks.

K Complexes			Vertex Waves		
Accuracy	Sensitivity	Specificity	Accuracy	Sensitivity	Specificity
64.29%	92.31%	40.00%	44.44%	47.62%	40.00%
71.43%	100.00%	46.67%	50.00%	33.33%	73.33%
82.14%	100.00%	66.67%	63.89%	52.38%	80.00%
82.14%	100.00%	66.67%	50.00%	61.90%	33.33%
71.43%	100.00%	46.67%	47.22%	52.38%	40.00%
78.57%	100.00%	60.00%	36.11%	28.57%	46.67%
53.57%	92.31%	20.00%	41.67%	28.57%	60.00%
53.57%	76.92%	33.33%	41.67%	28.57%	60.00%
60.71%	100.00%	26.67%	61.11%	52.38%	73.33%
75.00%	92.31%	60.00%	55.56%	42.86%	73.33%
Sawtooth Waves					
56.82%	45.00%	66.67%			
56.82%	40.00%	70.83%			
36.36%	40.00%	33.33%			
59.09%	65.00%	54.17%			
56.82%	55.00%	58.33%			
65.91%	50.00%	79.17%			
61.36%	50.00%	70.83%			
52.27%	55.00%	50.00%			
54.55%	45.00%	62.50%			
68.18%	60.00%	75.00%			

single contradictions or ambiguities will be weighted far less.

- *Parameter adjustment* may be another simple way of improving the system performance. Parameters, such as target error rate, learning rate, transfer functions, will all effect the system's accuracy. However, these improvements are limited.

- *Appropriate topology* should be identified for each of these tasks. Topology could be the size of the neuron layers. For instance, if too much data reduction occurs in the hidden layer, insufficient data may be passed onto output. Also, the number of hidden layers should be determined based on the complexity or the degree of non-linearity in the

task. Finally, other types of ANN should be investigated. While MLP is the most popular, it may not be the most suited for this application.

- *Other PR methods* should be examined.

4.3 Summary

This section showed band activity features as well as characteristic waves. Band activity features which use time-frequency representation, namely spectrograms and scalograms, can differentiate certain sleep stages. In particular, spectrogram can differentiate Awake and REM by detecting mixed frequency activity. Similarly, scalogram can differentiate Stage II, III, and IV by measuring the quantity of delta activity.

The spectrogram feature, ratio between beta activity and lower frequency activity, can differentiate Awake from other stages with 93.52% sensitivity and 94.60% specificity. It can differentiate REM from Stages II, III, and IV, with an average performance of 96.12% sensitivity and 93.63% specificity. This feature corresponds to the mixed frequency activity described in the sleep staging manual.

The scalogram feature, area above threshold, can differentiate Stages II, III, with 96.81% sensitivity and 89.28% specificity. It can differentiate Stages III and IV with 93.60% sensitivity and 90.43% specificity. This feature maps to the quantity of delta activity used in the sleep staging manual.

Characteristic waves complement the band activity information in that they mark unique events of certain sleep stages but more importantly they often determine the points of transition from one stage to the next. Preliminary pattern recognition neural networks with the topology of 120x10x1 neurons were used model these waves. The accuracy for K-complex, vertex waves, and sawtooth waves topped off at 82.14%, 63.89%, and 68.18%, respectively. Their performance was not adequate to reliably add information to the band features. The performance of K-complex was significantly better than that of vertex waves and sawtooth waves because the pattern was more distinctive and the data set more easily built. Therefore, to improve the performance of these pattern recognition tools, better data sets where the waves are professionally classified is necessary. With the improved data sets, the topology can be optimized to further enhance the performance.

In addition to band activity and characteristic waves extracted from EEG

signals, sleep scorers use information encoded in EOG signals and deduced from the context. The next section looks at EOG features followed by a section looking at context information.

Chapter 5

EOG Feature Extraction

While EEG plays a very central role in sleep staging rules, EOG also contributes significantly. Below is a list of ambiguous cases where EOG is required.

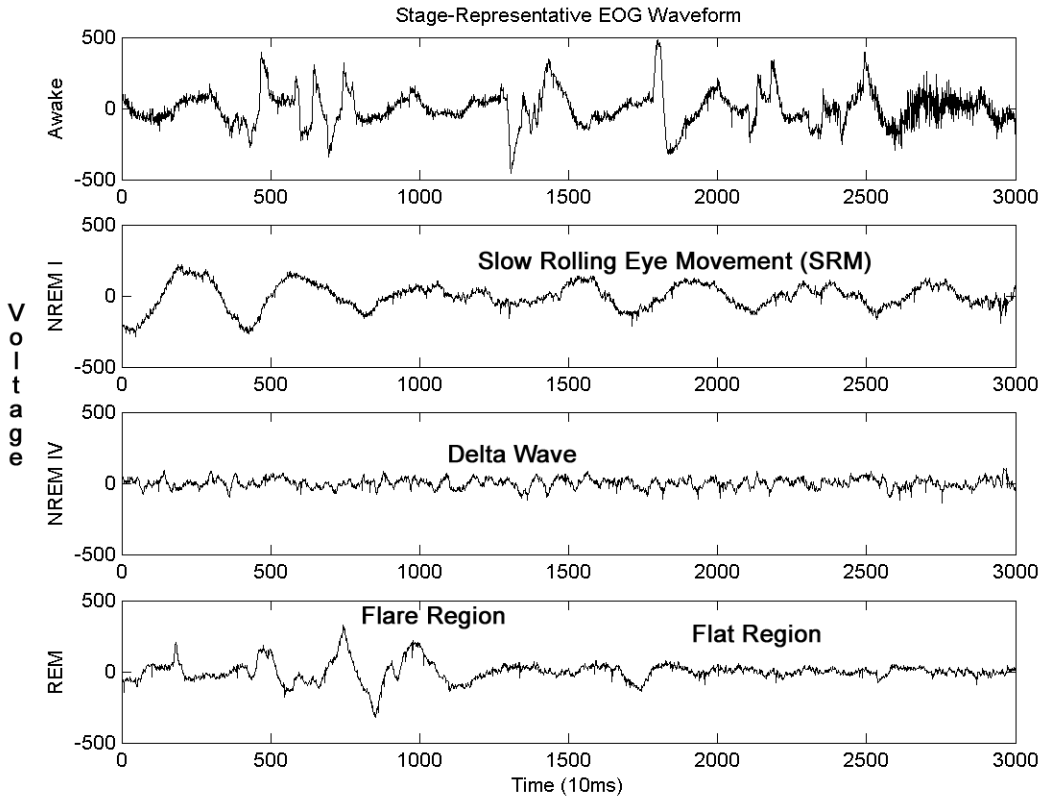
1. *The transition between Awake and NREM I* is a blurry border into a drowsy state. While EEG has distinctive patterns to characterize both Awake and NREM I, the two types of pattern overlap during the transition. EOG improves the ability to discretely identify this boundary by giving an early warning sign.

   When a person goes to sleep, they close their eyes before their brain activity slows down. As they close their eyes, the visual input is shut off, which also means the movement of their eyes changes. Open eyes move more frequently and more randomly. Figure 5.1 shows sample EOG segments for each type referred to in this section. Closed eyes eventually fall into a slow well-organized pattern, called *slow-rolling eye movement (SEM)*. Since the change in eye movement often precedes the change in brain activity, the occurrence of SEM may indicate to the technician to expect the beginning of NREM I.

2. *The transition between NREM II and NREM III* moves brain from theta activity to delta activity. Again, the two waves overlap during the transition. Unlike the previous case, EOG cannot provide an early indication as opposed to a confirmation. In NREM III and IV, EOG displays a slightly modified version of the delta wave. At this stage, eye movement basically disappears, so EOG is dominated by the delta waves.

3. *The differentiation of Awake and REM* is the most key function of EOG. EEG for these two stages are very similar, with possible amplitude reduction in REM. Their similarity is due to the fact that in both cases the brain is actively working. However, their differences in EOG are very apparent. Rapid eye movement mostly consists low amplitude segments, which we will call *Flat Region*[1], and short high amplitude high frequency segments, which we will call *Flare Region*[2].

Figure 5.1: Representative EOG waveform for various sleep stages



[1]The official terminology for Flat is tonic REM.
[2]The official terminology for Flare is phasic REM.

5.1 Observations about EOG

Similar to the EEG band activity analysis, human scorer also look at EOG from a frequency as well as time domain perspective. Since time frequency representation was successful at identifying useful features for EEG, the same approach is used to analyze EOG. This section looks at EOG waveforms in terms of scalograms and spectrograms. Sample segments and their TFR are shown in Figure 5.9. The following observations are made:

1. *Awake versus SEM*

   The spectrogram does not offer enough resolution at the lower frequencies to differentiate Awake from SEM. The scalograms show that the blotches in Awake are narrower and brighter than those of SEM. Therefore, possible features may look for higher peak values, higher count of bright blotches, etc.

2. *Isolating Delta*

   Delta's unique characteristics are most pronounced in the spectrogram in the frequencies 4 Hz to 12 Hz. In the other waves, the higher frequency band is nearly void. Therefore, testing the presences of higher frequency content can isolate the Delta[3] EOG waves. Another point that is not quite evident due to the quality of the images is that the blotches in Delta's scalogram are lighter in color. This phenomenon is due to the fact that Delta's signal range comparatively small. So another possible feature would look for low peak values.

3. *Dividing REM*

   As discussed previously, REM has two sections that we call Flare and Flat. These sections need to be identified separately, because their TFR expressions are different. In fact, the scalogram of Flare is very similar to that of Awake. Similarly, the scalogram of Flat resembles that of SEM, perhaps lighter.

   Frames of 5 seconds will likely contain only Flare or Flat. Therefore, we will divide each 30-second epochs into 11 5-second windows with 50% overlap. The same process will of course have to be carried out for

[3]The Delta waves discussed in this section is not equivalent to the Delta activity seen in EEG. To reduce confusion, it will be called Delta EOG waves.

the other waves, and a simple set of rules will then translate an array of segment classification to the epoch classification.

It is important to point out that removing a signal segment from its context will affect its TFR. Figure 5.3 demonstrates this property. The post-segment scalogram is derived from an isolated 500 data points. The pre-segment scalogram is extracted from a larger scalogram derived from the same 500 data points and 1750 neighboring data points on either side. Notice in particular that the scalograms and their contour maps show significant difference near either edge. Therefore, to make the scalograms true to the 30-second epoch, we will always use pre-segment method.

Based on these observation, we can now choose an approach to differentiate the different types of waves. First, we need to be able to classify each 5-second segment as Awake/Flare, SEM/Flat[4], or Delta. We, then, use a rule-based system to determine a set of 11 overlapping segments as being Awake, SEM, Delta, or REM. Section 5.2 and 5.3 design the classification of 5-second segments and 30-second epochs, respectively.

5.2 Segment Classification

This section identifies and evaluates features that differentiate each type of EOG waves.

5.2.1 Data

From the EOG data[5], 20 segments of each EOG segment type are selected as the training data set. To facilitate the algorithm design, the training data set primarily chooses segments that optimally demonstrate each segment type's characteristics.

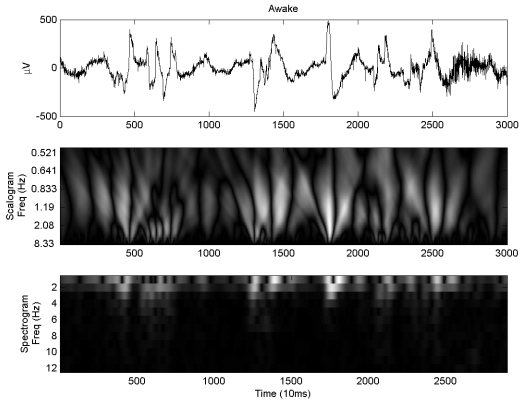
5.2.2 Feature Set

The following features were studied,

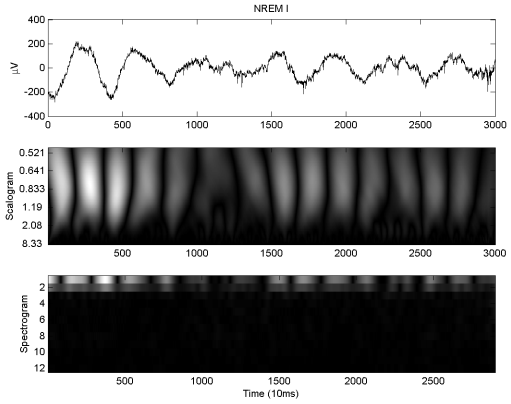
[4]SEM and Flat are not distinguished in this section because they are too similar to distinguish accurately.

[5]The EOG data is extracted from the public sleep study data set provided by Physiobank at `http://physionet.cps.unizar.es/physiobank/database/sleep-edf/`

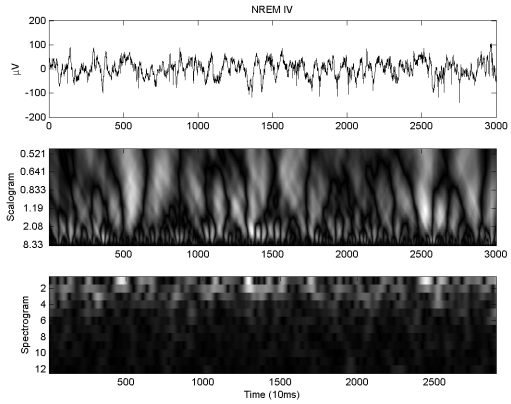
Figure 5.2: TFR of EOG waveforms (a) Awake showing EOG with high amplitude and low organizing. (b)NREM I showing SEM. (c)NREM IV showing Delta EOG waves. (d)REM showing Flare and Flat waves.



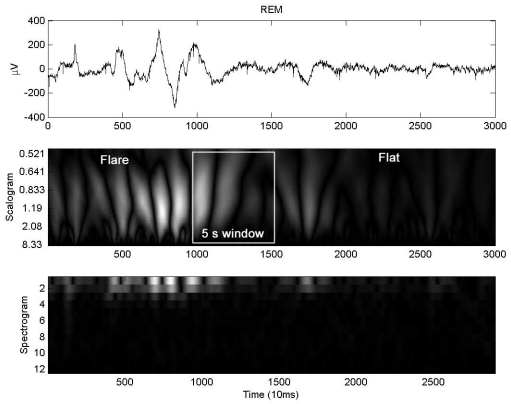
(a)



(b)

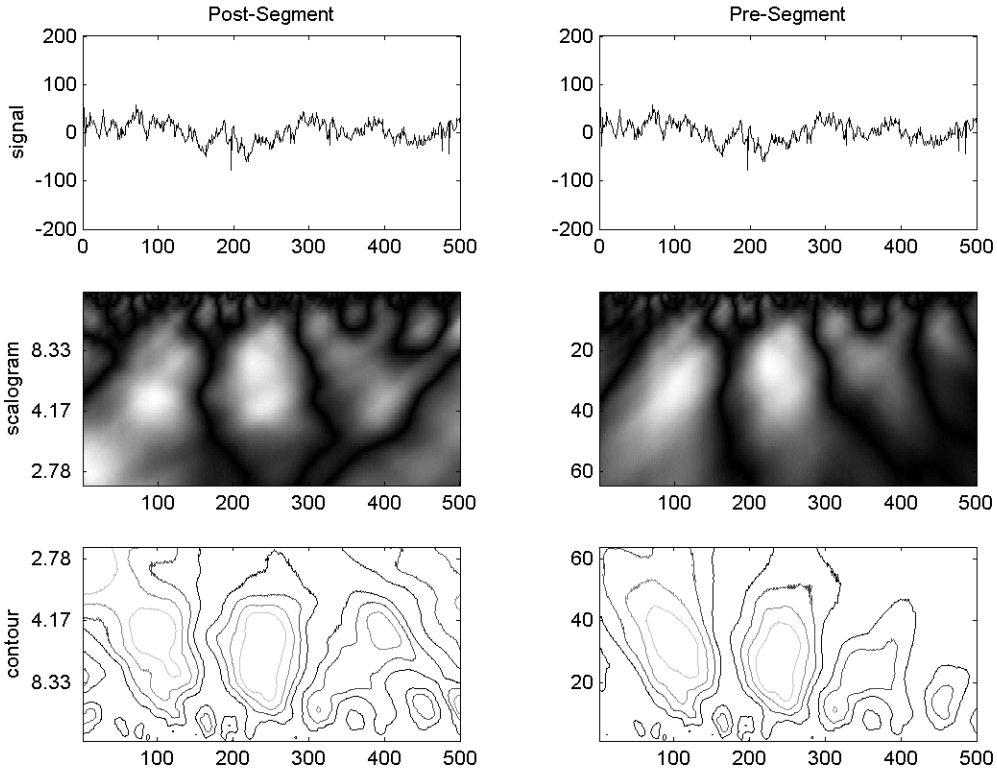


(c)



(d)

Figure 5.3: Contrast pre-segmentation and post-segmentation.



- *Upper frequency content* is one of two features derived from the spectrograms. It is defined as

$$upperFrequencyContent = \sum_{t} \sum_{f=4Hz}^{12Hz} SPEC(t, f). \qquad (5.1)$$

  Its definition is inspired by the upper frequency content in Delta EOG waves.

- *Ratio of frequency content* is the other feature derived from the spectrograms. Its equation is

$$ratioFrequencyContent = \max_{t} \left(\frac{SPEC(t, nHz : 12Hz)}{SPEC(t, 0Hz : nHz)} \right), \qquad (5.2)$$

where n is a parameter to be determined. This feature is possibly useful to identify Flare and Flat that end up in the same segment.

- *Peak value* and *Peak frequency* are two basic features in the scalogram. Peak frequency is the frequency at which peak value is achieved in the scalogram.

- *Area above 50%* is a set of features that are based on the idea of the 50% contour line. The algorithm to determine these areas is

$$[maxValue, maxFreq, maxTime] = determinePeak(SCAL); [minValue, minFreq, minTim \tag{5.3}$$

  The features in this group include

  - *Region size*
  - *Content under the region*
  - *Maximum width of the region*
  - *Maximum length of the region*
  - *Time coverage*

    Note that time coverage and maximum width are different. See Figure 5.4 for clarification.
  - *Frequency coverage*

  The variations investigated are

  - *Neighbor definition*

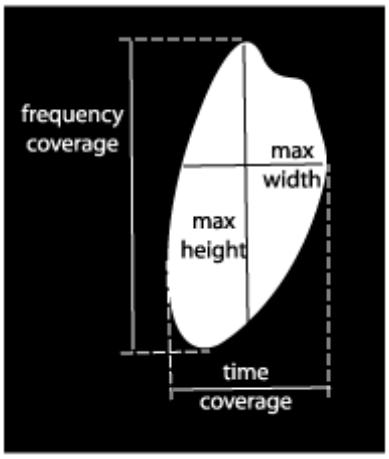
    The method *regionGrow* is derived from the one discussed in class. In this method, we can consider a point as having 4 neighbors or 8 neighbors.
  - *Multiple regions*

    Another aspect of variation skips the method *regionGrow* and just uses *find* to determine all the areas above the level. This variation may present multiple regions in the mask, and its count is calculated by the Matlab function *bwlabel*. We also looked at neighbor definition for *bwlabel*. Finally, we implemented a tolerance in the counting such that very small regions are avoided.

– *Level setting*

Level settings, meaning the cutoff feature values, were also studied. We looked at setting the level to 500, 1000, and 1500. These settings are often translatable to some signal amplitude feature, which should be used for computational efficiency.

Figure 5.4: Difference between Area features.



Differentiating Awake Segments

Awake can be differentiated by the features, *Area above 1000* and *Count of regions above 1000*. For each segment type, the feature values are calculated for the training segments. The results are presented in the first two graphs of Figure 5.5. The x-axis is the sample number and the y-axis is feature value. Clearly, the feature *Count above 1000* is the better feature, because the data points corresponding to Awake segments are almost entirely separated from those of the other types.

To determine the appropriate threshold, we find the ROC curve of each feature, which is the third graph in Figure 5.5. Since the ROC of *Count above 1000* passes closer to the (1,1) point, the ROC confirms that it is the better the feature. Its best performance occurs at a threshold of 5 with 100% sensitivity and 97.5% specificity.

In an attempt to improve the specificity, we combine the two features. The ROC of the combined system is shown in the last graph. It shows that the

best performance has a sensitivity of 100% and a specificity of 96.25%. This is due to the fact that the discrete logic rules actually forces an additional misclassification in the combined case.

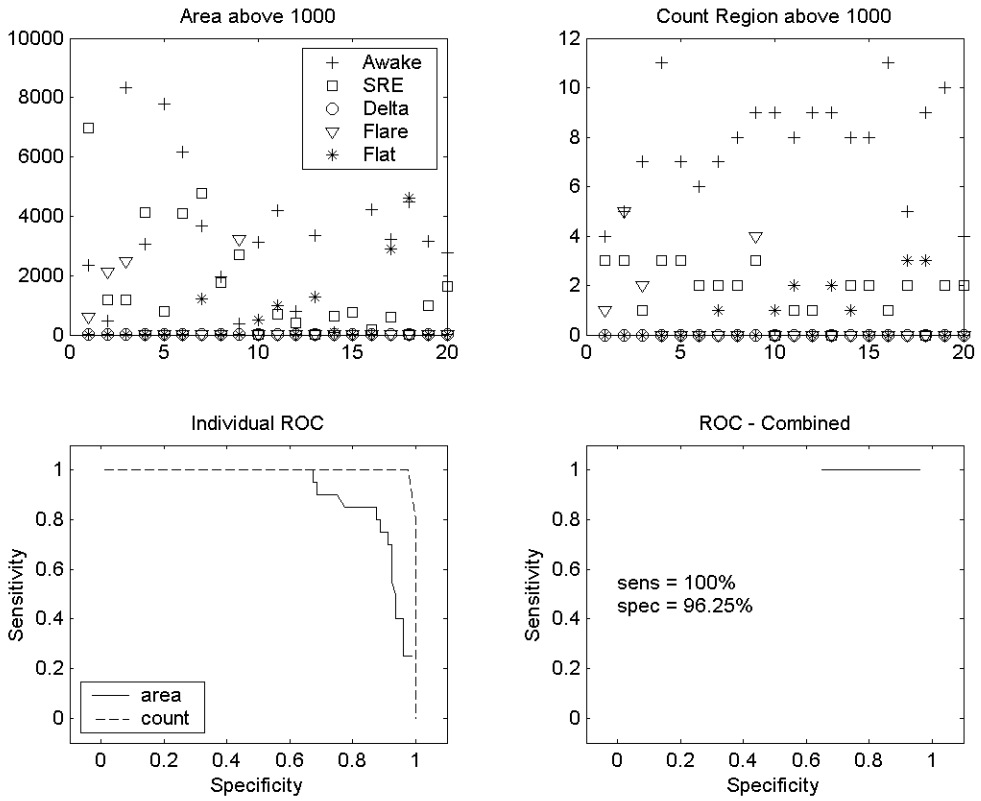


Figure 5.5: Features (Area above 1000 and Count Region above 1000) to differentiate Awake. ROC of individual features and combined.

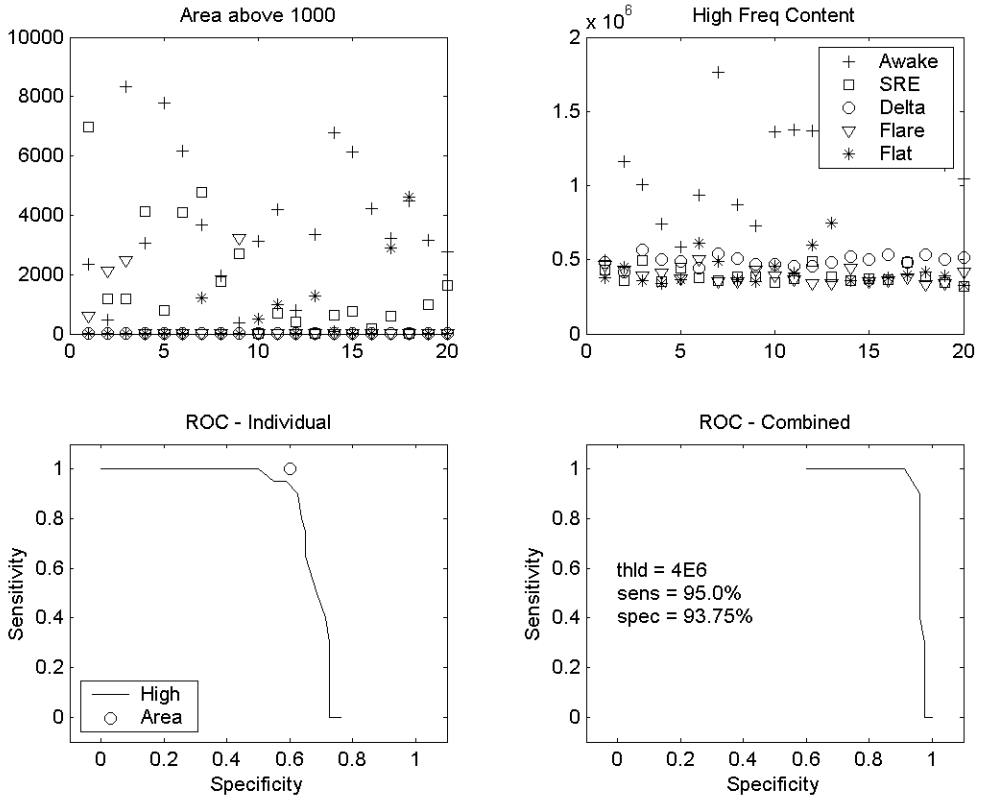
The feature *Count region above 1000* is selected to differentiate Awake. Aside from its performance, it is also faster than the other feature, because it does not use the recursive function *regionGrow*.

Differentiating Delta Segments

For Delta segments, two features, *Area above 1000* and *High frequency content*, were identified. Refer to the first two graphs in Figure 5.6 for these

two feature's behavior. Again, x-axis is sample number and y-axis is feature value. It is evident that these two features have a similar ability to separate Delta segments from the other waveforms. Therefore, sensitivity is ensured. The other segments also cover the same feature value space, which means the specificity will be low. This expectation is confirmed in the ROC curves in the third graph. Both features were able to provide reasonable sensitivity but the specificity lingers around 60%.

Figure 5.6: Features (Area above 1000 and High Frequency Content) to differentiate Delta. ROC of individual features and combined.

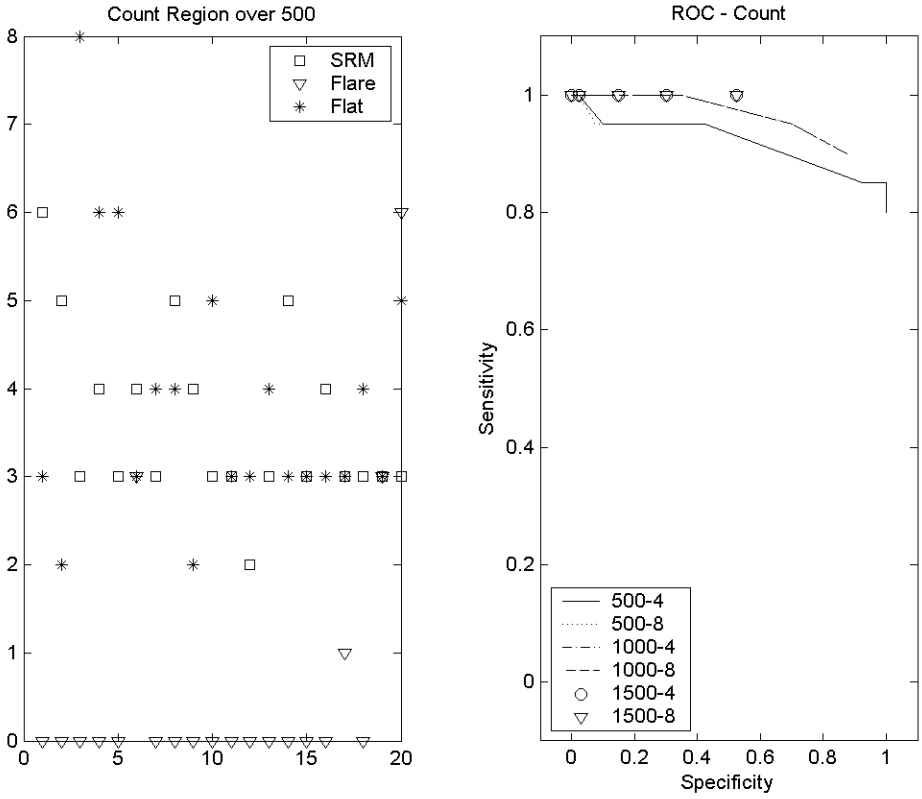


By combining the two features, the performance is dramatically improved. In the last graph of Figure 5.6, the combined ROC curve provides a performance of 95% sensitivity and 93.75% specificity. Therefore, we would use both features to determine Delta segments together.

Differentiating Flare Segments

When all the segment types are studied together, the characteristics of Awake and Delta dominate the graphs. In order to study the remaining three types of segment types, we remove Awake and Delta by assuming that the algorithms developed in the last two sections will be able to differentiate them already. Then for Flare, The feature *Count above 500* appeared very useful. Its behavior is shown on the left in Figure 5.7.

Figure 5.7: Features (Count Region above 500) to differentiate Flare with its ROC curve.



In order to determine the setting of the feature, we tried regions above 500, 1000, and 1500 with different neighbor definitions. At the same time, the performance is plotted for different threshold. The results are shown in the ROC curves in the second graph. The ROC curves show that the

neighbor definition makes no difference. However, the height level 1500 is much less useful. Level 500 with a threshold of 2 can be chosen to maximize specificity. The performance would be 85% sensitivity and 100% specificity. If more sensitivity is required, we can use a Level of 1000 with a threshold of 0. Its performance is 90% sensitivity and 87.5% specificity.

Differentiating Flat

Many features were looked at to differentiate these two features, but it was next to impossible to tell their difference in most feature values. Figure 5.8 shows the features *Area above 1000*, *Peak Value*, and *Content at 0.2Hz*. By eliminating the plot of the 3 waves determined above, the contrast between the remaining waves are more clear.

From the figure, it is clear that SEM and Flat will be easily mistaken for each other. Manipulating the data does not improve. Therefore, we derive the ROC curve for the three features. We find that the performance lulls between 70% and 85% for both indicators. We choose to use Peak value with a threshold of 1050.

5.2.3 Combining Differentiation Rules

Based on the segment type that has the best individual differentiation performance, the following rules are established to classify each segment.

```
if more than or equal to 4 region above 1000
segment is Awake
else if upper frequency content greater than 4.2e5
and area above 1000 less than 5
segment is Delta
else if less than 2 region above 1000
segment is Flare
else if peak value less than 1050
segment is Flat
else
segment is SEM
end if
```

We now test the entire sample set against this rule set and the results are presented in Table 5.1.

Figure 5.8: Feature investigation to differentiate SEM and Flat.

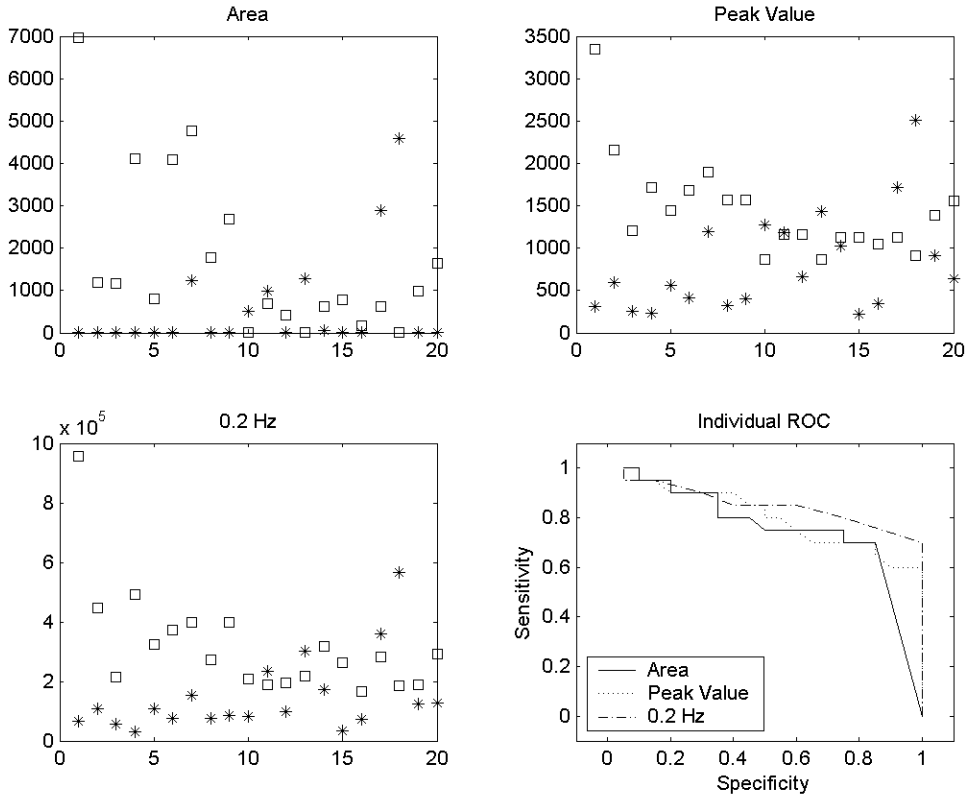


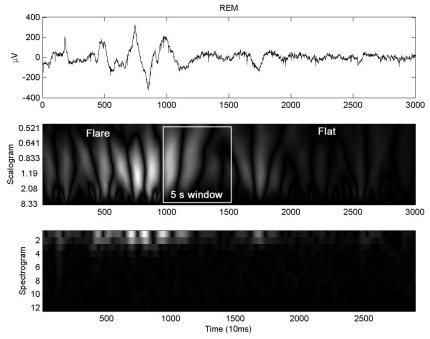
Table 5.1: Segment level accuracy.

Type / Class	Awake 20	SEM 20	Delta 20	Flare 20	Flat 20
Awake	17	0	0	0	4
SEM	3	9	0	1	12
Delta	0	1	20	12	1
Flare	0	10	0	7	3
Flat	0	0	0	0	0
Sensitivity	85%	45%	100%	35%	0%
Specificity	95%	80%	82.5%	83.75%	100%

As expected the highly distinguishable waves like Awake and Delta received far better performance than the other three. Fortunately, EOG is primarily used to identify Awake from REM, which these crude classification proves it can be done.

REM has two sections that we call Flare and Flat. These sections need to be identified separately, because their TFR expressions are different. Figure 5.9 shows the TFR of a REM segment. In fact, the scalogram of Flare is very similar to that of Awake[6]. Similarly, the scalogram of Flat resembles that of SEM, perhaps lighter.

Figure 5.9: TFR of REM showing Flare and Flat waves.



Frames of 5 seconds will likely contain only Flare or Flat. Therefore, we will divide each 30-second epochs into 11 5-second windows with 50% overlap. The same process will of course have to be carried out for the other waves, and a simple set of rules will then translate an array of segment classification to the epoch classification.

5.3 Epoch Classification

This section looks at the translation from arrays of segment classification into an epoch classification. First we design a rule based system to translate segment classification into an epoch classification. For each of the four types of EOG waves, a set of 20 epochs are selected as training data. The corresponding segment classifications are derived and applied to the rule base. Once the rule base is developed, the complete system should then be tested on a continuous stream of EOG data.

5.3.1 Rule Based System Design

There are several considerations in the design of this rule base. These considerations are

[6]Awake consists of blinking and saccadic eye movement.

1. *Reference Classification*

   In order to assess the goodness of the rule base, we must test it on actual EOG data. However, these data sets do not come with EOG wave classification. Instead, a sleep stage is assigned to each 30-second epoch. Therefore, we must translate the sleep stage into EOG wave classes. Roughly, we translate Awake stage to Awake EOG, NREM I and II to SEM, NREM III and IV to Delta, and REM stage to REM EOG. These translations are not exact, which means the reference classification introduces a level of error.

2. *Epoch Divisions*

   Sleep stage information is provided every 30 seconds. However, these epochs will not conveniently avoid containing a stage transition. In such cases, the rules must have the flexibility to deal with a transition. Since each epoch is divided into 11 overlapping segments, then we define rules that looks for dominant behavior as opposed to just any behavior. Furthermore, we incorporate this into our thinking as we review the test results.

3. *Segment Divisions*

   Segment divisions are similar to epoch divisions in that their placement may be very inconvenient. For instance, a segment may contain half of a characteristic wave. Since a segment is quite narrow, it is more difficult to identify the dominant behavior. The segment length is selected to accommodates one occurrence of flares. They are also overlapped such that the neighbor of a segment straddling a transition will be able to make a proper classification.

4. *Ambiguity between SEM, Flare, and Flat*

   From the previous section, we have determined that SEM, Flare, and Flat are difficult to distinguish because all three have similar amplitudes. However, the definition of Flare and Flat is very important. Their neighboring occurrence signifies a rapid eye movement. Therefore, in the rules, some effort must be made to cover cases where some or many of segments are misclassified.

With the above consideration, the following rule base is selected.

```
    if more than 9 segments are Delta
    epoch is Delta
    else if between 1 and 6 segments are Flare
    if the remaining segments are SEM or Flat
    epoch is REM
    else if all segments are SEM or Flat
    epoch is SEM
    else
    epoch is Awake
    end if
```

It was determined that 9 or more segments is considered a dominant behavior. Therefore, more than 9 Delta segments would mean the epoch is Delta. Flares are limited to half the segments such that they are contrasted to the Flats. When more segments are Flares, they are either using higher amplitude SEM or lower amplitude Awake. Sine SEM and Flat were very difficult to distinguish, they are used together. In other words, REM is considered as SEM with flares.

It should be noted that the rules proposed above are very basic and does not cover nearly the number of scenario that can be anticipated. These rules use discrete thresholds as decision criteria. Much better decision agents can be used to differentiate features that appear to be clumping together.

5.3.2 Results

The results from the 20 segment per wave type data set is in Table 5.2.

Table 5.2: Epoch level accuracy.

Type / Class	Awake 20	SEM 20	Delta 20	REM 20
Awake	18	16	0	10
SEM	0	0	0	0
Delta	0	4	20	1 I
REM	2	0	0	9
Sensitivity	90%	0%	100%	45%
Specificity	56.67%	100%	91.67%	96.67%

This epoch level system inherited the weaknesses of the segment level classifier. Its ability to classify Awake and Delta are clear, but for SEM

and REM the performance need to be improved. Possible improvements are discussed in the next section.

5.4 Discussion

Based on the results in this project, clearly there is room for improvement in terms of performance. This section discusses the possible cause of the poor performance and some of the solutions.

- *Rule Complexity*

  Due to the scope of this project, very simple rules were used to bring together the various features. In all cases they use simply discrete logic. However, this is a poor approach. Feature's ability to differentiate relaxes on both ends of its value space. This kind of properties cannot be captured by the boolean logic. Therefore, one improvement is to use fuzzy logic. It would make the system more complicated, but it would more closely imitate human visual processing. Furthermore, fuzzy logic allows us to weigh all the features whereas boolean logic would discount later rules after finding one satisfactory rule.

- *Loss of Context*

  In this project, we not only look at EOG epochs independently, we further divided them segments. The features and the rules are all applied to the segments or the epochs isolated from the context. In reviewing the poorest of the results, it becomes clear that we rely on the context to improve our classification. Therefore, another improvement is to bringing in contextual information. Again, the system would be much more complex due to the extra memory and processing requirements.

- *Efficiency*

  The algorithm designed in this project presents various efficiency problems. Due to the repeated need to transform to the time-frequency representation, these features are particularly slow. It is estimated to be making 200 epoch classifications per hour. Since each hour has 120 epochs of data, this speed is sufficient. However, it is slower than real time needs, where additional data points arrive at 100 Hz. Even without considering the speed, the computational needs of the TFR is well

known. Such systems would not be suitable for portable processing units.

Also, earlier in this report, the issue of segmenting the signal versus segmenting the TFR was raised. Segmenting the TFR would technically present a more accurate representation, but it also means more computation during the transformation into frequency domain.

- *Alternatives*

  There are more efficient alternatives that can replace some of the cumbersome features used in this project. For instance, the feature *range* can differentiate the Awake segments just as well if not better. This feature is simple and time-domain based, which would significantly reduce the computation power.

Aside from the problems and their solutions, the following is a list of tasks that could not be completed in the scope of this project. Their completion can significantly contribute to the performance of this system.

- *Better Differentiation of SEM, Flare, and Flat*

  More features should be studied to differentiate these segments. As shown in the latter part of the project, the failure to correctly identify these segments will devastate the epoch classification.

- *Coordinate Features for Segment Classification*

  While significant effort was put into optimizing each feature's utility, more coordinated effort would be beneficial. For instance, it was noticed that in the segment classification, the Delta rule was hurting the sensitivity of the Flat rule. Individually, peak value can provide reasonable sensitivity to identify a subset of Flat segments. The very small lack in specificity in the Delta rule misclassifies the subset to which the peak value is sensitive. Therefore, when the features are combined, the sensitivity to identify Flat was nearly nil.

  One way of approaching this task is to use multi-variable in the assessment of these combined features. Large scripts can be used to automatically generate various combinations of feature settings within boundaries set through manual observation. These combinations can tested on the data, such that their combined utility is optimized.

- *Incorporate prospective performance* In this project, we only used retrospective performance indicators. Prospective indicators, meaning performance evaluated on a data set foreign to the system. These indicators would be more representative of the algorithm's true performance. Aside from extracting more data from the same patient's EOG data. Testing on other patient's sleep data will allow the algorithm to adapt to all types of data quality and patient-to-patient variances.

- *Continuous data* While most of the testing was conducted on segments or epochs that are already extracted, the algorithm should also be applied to continuous stream of data. Using a window that continuously move through time, it can extract the epochs in real time. Using such a method, we can consider incorporating contextual information, as well as improving on the efficiency by reducing redundant computation.

5.5 Conclusion

This section designed a method of extracting sleep staging information based on time frequency representation. The objective was to identify EOG features that can complement EEG features to successfully differentiate various sleep stages. This method emphasizes the identification of the type of EOG activity presented during each epoch. Due to the two component nature of EOG signal during REM sleep, the 30-second epochs are analyzed in 5-second segments.

At the segment level, to distinguish between Awake versus SEM, the best feature was Count above 1000 set at a threshold of 5. It has 100% sensitivity and 97.5% specificity. To identify EOG Delta activity, the features Area above 1000 and High Frequency Content must be used together to get a sensitivity of 95.0% and a specificity of 93.75%. To differentiate Flare activity, the feature Count above 1000 has a sensitivity of 90.0% and a specificity of 87.5%. Due to Flat activity's similarity with SEM, none of the features could perform better than 70% sensitivity and 85% specificity.

Using a discrete rule system, the various features were combined and it was found that Awake and EOG Delta patterns can be differentiated reliably. Awake had 85% sensitivity and 95% specificity. Delta had 100% sensitivity and 82.5% specificity.

It was determined that 9 segments would determine a dominant epoch.

Again using a set of discrete rule at the epoch level, it was found that Awake had 90% sensitivity and 56.67% specificity. SEM performed very bad with 0% sensitivity and 56.67% specificity. Delta was the best at 100% sensitivity and 91.67% specificity. REM which includes Flare and Flat had 45% sensitivity and 96.67% specificity.

The lower performance in the overall system is primarily caused by the weaker rules used for Flat segments. Also, the discrete rules did not provide sufficient flexibility to permit addition performance enhancement. Therefore, while EOG features based on TFR have been shown to be useful, additional improvements on specific features are necessary to improve the overall system performance.

In addition to EEG and EOG data, the third source of major information is context. The next section looks at the incorporation of contextual information.

Chapter 6

Context Feature Extraction

EEG and EOG are two signals that are primary indicators of sleep stage. However, context also influences the sleep scorer's decision. Some context information is derived from R&K definitions and some are based on bio-variability patterns. These information generally fall into the following 2 categories,

1. *Time-based Context*

   The most basic time-based context is the definition of certain sleep stages based on the past occurrence of transient characteristic waves. For instance, scoring NREM II to NREM I to REM would depend on the last occurrence of sleep spindles or K complexes. This type of context information can be easily built into an automated expert system. Not very much data is required as there will be multiple instances of these occurrences even within one night's sleep study data.

   More complicated time-based context puts each epoch of data within the context of a night's sleep. For instance, the first few cycles of sleep each night has deep sleep, and this pattern leads the scorer to expect to score NREM III and IV after NREM II. Since entire night of data is required, this type of context needs more data sets, but a few can at least build a pattern.

2. *Patient- or Environment-based Context*

   Patient-based context include patient's age, fatigue level, whether patient has a sleep disorder, etc. Environmental context can be the temperature. These factors influence the human scorer to expect certain

patterns. For instance, an elderly patient generally has less deep sleep, so the scorer will have different expectation of the scored architecture. These context information cannot be modeled with a very limited data set. To draw patterns, a large range of examples are needed for each context parameter while preferably holding other parameters constant. Therefore, with limited resource, the expert knowledge from a sleep specialist would be used to approximate the actual pattern.

Due to the limited numbers of data set, this section looks at time-based context only. Part of the information will be modeled after the standard sleep architecture and the other part derived from the data sets. This section will first take a closer look at sleep architecture. Then, this section documents the process of selecting a proper model to represent the context as relevant to time-based context analysis. Next, it looks at potential ways of implementing the selected models. Finally, it provides an example of context information improving the performance of EEG features.

6.1 Sleep Architecture

The sleep stages follow a specific temporal pattern, generally shown in a hypnogram as in Figure 6.1 and Figure 6.2. As can be observed from the diagrams, the stages generally go from Awake to NREM I, II, III, IV and back to III, II, I. Generally it should go from Stage I to REM. This pattern is ideally shown in Figure 6.1.

However, Figure 6.2 shows that often the sleep stage do not change ideally. This departure is caused either by inaccurate sleep staging or by too short a duration for some stage to be recognizable. The two stages most likely to be lost is Stage I and Stage III, because of their short duration and similarities to Stage II and IV, respectively.

6.1.1 Data

To study the transitions that occur during sleep, overnight sleep study data derived from 24 subjects are used. Of these subjects 10 are labeled under CPAP[1], 8 are labeled as OSA[2], and 6 are normal. The number of each

[1]CPAP stands for Continuous Positive Air Pressure.

[2]OSA stands for Obstructive Sleep Apnea.

Figure 6.1: Hypnograms are used to show the transitions between sleep stages and the duration in each stage.

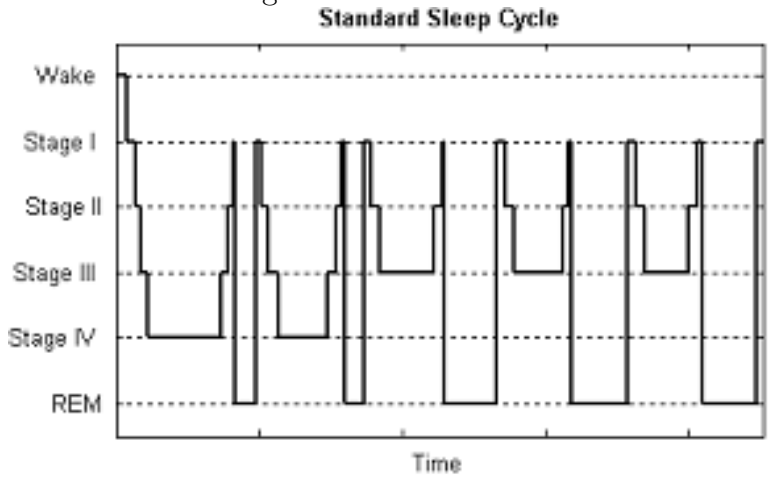
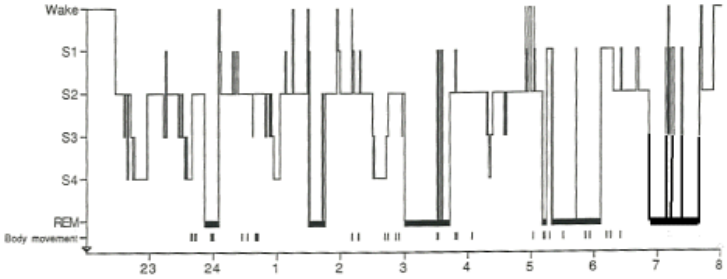


Figure 6.2: Typical hypnogram for a healthy young subject.



type of transition are tallied and presented in Table 6.1. The rows represent the starting stage and the columns divide the end stage. From the tally, the probability of each transition is derived and shown in Table 6.2. As a comparison, the same probability is calculated for normal subjects only and the results are in Table 6.3. Though the general trend is maintained, the values have small differences depending on the subject pool. In order to generate analysis on normal transitions, the next set of analysis is based on Table 6.3.

From these tables, the following observations can be made,

1. *Awake:* Significant amount of time is spent in the Awake state due to its high probability of transiting into its own state. When leaving Awake

Table 6.1: Number of each type of sleep stage transition based on all subjects.

States	Awake	Stage I	Stage II	Stage III	Stage IV	REM	Total
Awake	3158	335	118	2	2	37	3652
Stage I	98	557	399	0	0	11	1065
Stage II	265	161	9472	124	3	89	10114
Stage III	3	0	50	432	94	0	579
Stage IV	29	3	45	21	2308	1	2407
REM	98	10	30	0	0	3364	3502
Total	3651	1066	10114	579	2407	3502	21319

Table 6.2: Sleep stage transition probabilities based on all subjects.

States	Awake	Stage I	Stage II	Stage III	Stage IV	REM
Awake	0.8177	0.1179	0.0460	0.0006	0.0008	0.0172
Stage I	0.0740	0.4909	0.4211	0.0000	0.0000	0.0139
Stage II	0.0262	0.0173	0.9343	0.0127	0.0003	0.0092
Stage III	0.0033	0.0000	0.1711	0.5942	0.1480	0.0000
Stage IV	0.0129	0.0027	0.0244	0.0069	0.7858	0.0006
REM	0.0291	0.0027	0.0087	0.0000	0.0000	0.9178

Table 6.3: Sleep stage transition probabilities based on normal subjects.

States	Awake	Stage I	Stage II	Stage III	Stage IV	REM
Awake	0.8112	0.1387	0.0352	0.0000	0.0000	0.0150
Stage I	0.0664	0.4796	0.4276	0.0000	0.0000	0.0264
Stage II	0.0219	0.0093	0.9430	0.0154	0.0004	0.0101
Stage III	0.0044	0.0000	0.0868	0.7266	0.1822	0.0000
Stage IV	0.0097	0.0013	0.0222	0.0073	0.9595	0.0000
REM	0.0247	0.0058	0.0102	0.0000	0.0000	0.9593

state, the most likely transition goes to Stage I. On rare occasion, due to short Stage I, the transition moves directly into Stage II.

2. *Stage I:* Stage I is short and its reflected by the relatively small probability of transiting into itself. It is most likely to go to Stage II, but on rare occasions return to Awake and REM.

3. *Stage II:* Stage II is a very long stage. As expected, it has a tendency to move into Stage III. On the other hand, it may also directly exit to

Awake, move back to Stage I, or enter into REM.

4. *Stage III:* Stage III is generally not a very long stage in that it is between Stage I and Stage II's duration ranges. Stage III predominantly proceed to Stage IV or back to Stage II. Being in deep sleep, it is more difficult to go to Awake.

5. *Stage IV:* This stage is also very long. Due to Stage III's transitional nature, Stage IV often enters Stage II directly. Note that neither of the deep sleep stages were associated with REM.

6. *REM:* REM can also be a long stage especially later in the night. As mentioned previously, REM is generally associated with the light sleep stages. It may also exit directly to Awake.

Knowing that these are the probable transition is not enough for contextual analysis. It is also important to derive the likely duration of each stage and the temporal influence on specific probabilities. In fact, it is also important to analyze the change in probabilities depending on the NREM-REM cycle. Therefore, the next step in this research is to find a model that considers both NREM-REM cycle and duration in current state.

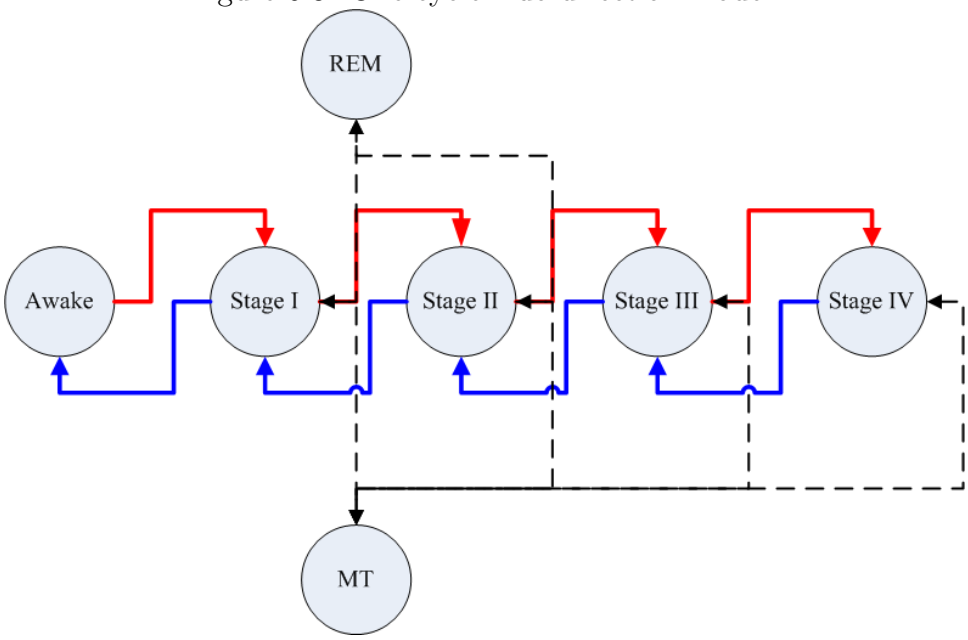
6.2 Model Selection

Several sleep cycle models were considered. In general, as the models are improved in terms of better simulating reality, its complexity also increases. Rise in complexity translates to higher training data requirements, which is an issue discussed for each model.

6.2.1 One-Cycle-Duo-Direction Model

The first model, shown in Figure 6.3, is the basic model that is most commonly adopted. The patient starts Awake and proceeds through the NREM stages. Only Stage I and Stage II can precede to REM. All stages, except for Awake, can move into and leave from MT. This model is sufficient to generate the sleep stage transition graphs shown in Figure 6.1, but will not definitively show that pattern.

Figure 6.3: One-cycle-Duo-direction model.



Assumption

The model can handle skipped states. It is fairly common to see transient states such as Stage I and Stage III to be so brief that they are not scored at all. In which case, this model assumes that it can pass right through the transient states.

Occasional waking up during the night are ignored. Patients may wake up for various reasons for short bits of time. The patient generally return directly to the stage they were in. For example, a patient woken from Stage IV will relapse to Stage IV right after instead of starting in Stage I again. Due to the short duration and minimal interruption in the overall cycle, these segments can be ignored.

Transitions from a state into MT always returns to the first state. Subjects may move during any stage of their sleep but it generally doesn't bring about a change in states. Therefore, in actuality, a separate MT state should be observed for each state, except Awake, but in this model, they are lumped together. It may be worthwhile to count the probability of MT occurring under each state as well.

Shortcoming

This model does not track the direction of the NREM stage transitions. For instance if the patient is going through AWAKE→Stage I→Stage II, the next stage is most likely Stage III. However, if the sequence is Stage IV→Stage III→Stage II, the chance that Stage III occurs next is very low. However, this model cannot differentiate the difference between the two sequences because it does not record the direction. This issue also applies to entries into REM. The sequence Stage II→Stage I→REM is far more likely than Awake→Stage I→REM. Therefore, the model should be adjusted to look at the directions.

Data Requirement

An advantage of this model is its low requirement for data. Firstly, this model has very few parameters to set. Keeping to the assumptions listed above, there are 22 transitions to parametrize. (Awake and NREM states have a total of 8 one-direction transitions. There are 2 two-direction transitions with REM, and 5 two-direction transitions with MT.) In these 22 transitions, the 5 transitions from MT back to the source state should be considered as 1 based on the assumptions. Therefore, only 17 transitions needs to be found.

Each normal sleep study can expect to see 4 to 6 transitions between Awake and NREM stages. There are more transitions into REM and MT. These transitions will provide a good average.

The second reason for low data requirement is that the amount of ambiguity in the model surpasses errors from parameters. For instance, the transitions, whose probability change dramatically depending on the direction, would not be accurate regardless the amount of data provided. These transitions would observe one direction's probability as 90%+ and another direction's as 10%-. Since this model lumps them together, the data would suggest a 50% probability, which is not at all indicative of the actual two values. Therefore, the errors would come from the model as opposed to the parameters.

Implementation

This model is generally implemented as a discrete time Markov chain with 30-second epoch. The transition probability matrix is derived by counting the number of each type of transition leaving a certain state and calculating the ratio of each type to the total, which is the probability of that type

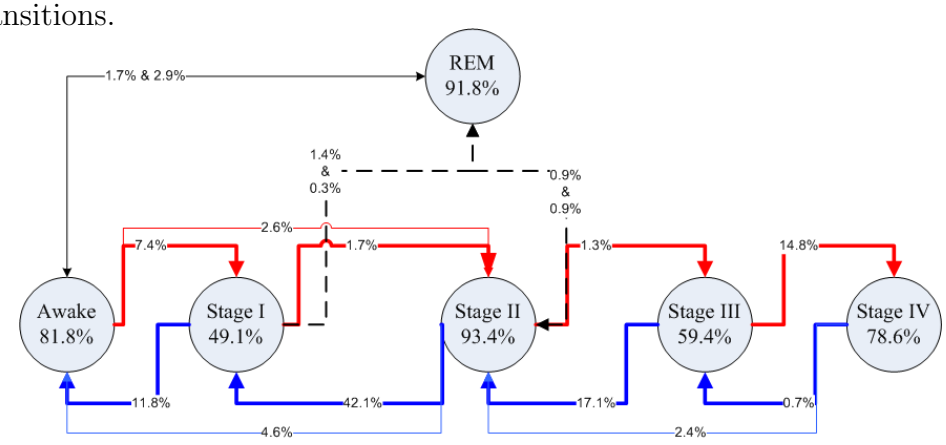
Table 6.4: Sleep stage transition probabilities based on all subjects.

States	Awake	NREM I	NREM II	NREM III	NREM IV	REM
Awake	0.818	0.118	0.046	0.001	0.001	0.017
NREM I	0.074	0.491	0.421	0.000	0.000	0.014
NREM II	0.026	0.017	0.934	0.013	0.000	0.009
NREM III	0.003	0.000	0.171	0.594	0.148	0.000
NREM IV	0.013	0.003	0.024	0.007	0.786	0.001
REM	0.029	0.003	0.009	0.000	0.000	0.918

of transition. Though the one-cycle-duo-direction model only considers a subset of the transitions represented by the transition probability matrix, the transitions outside of the model will almost equal zero.

Table 6.4 contains the transition probability matrix extracted from the data sets provided by Dr. C. George of Western Ontario University. Figure 6.4 demonstrates the same example graphically. In this diagram, the transition arrows are weighted according to their significance relative to all transitions.

Figure 6.4: State transitions probabilities calculated by ratio of each type of transitions.



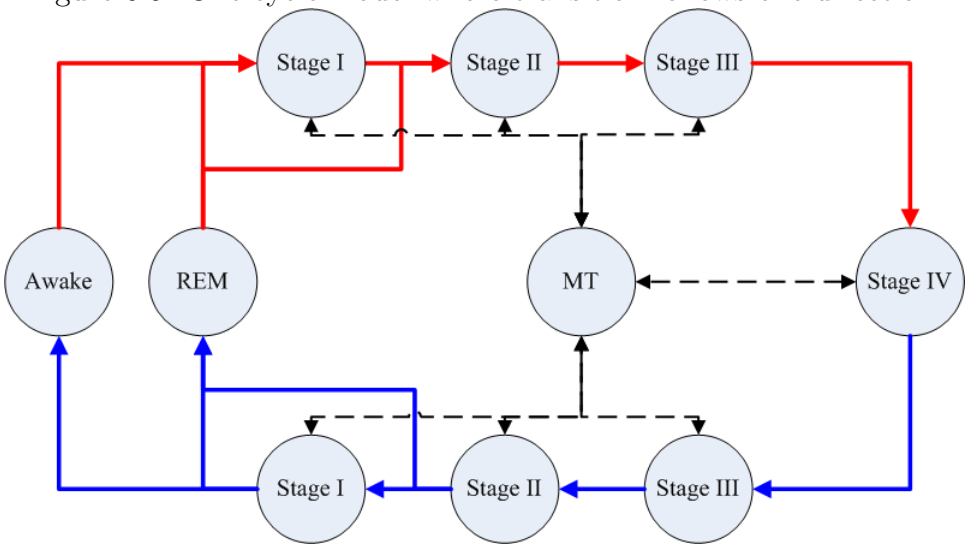
Despite the simplicity of this model, observations can be made to better understand the sleep cycle. Each state shows the highest probability of remaining in itself. As stated in the assumptions, NREM I and NREM III are transient states that are either short or skipped. NREM I's short dura-

tion is reflected by a lower probability of remaining in itself. The transition probability from Awake directly to NREM II indicates that NREM I may be skipped. Similar observations can be made about NREM III.

6.2.2 One-Cycle-One-Direction Model

The model in Figure 6.5 corrects the two direction issue in the previous section. It uses red to indicate the transition direction is towards deeper sleep and blue to indicate the opposite. Therefore, it differentiates the probability of advancing into the next sequential state from the probability of doubling back to the previous state. For instance, the probability $P(II \rightarrow I)$ in the previous model would be broken down to $P_{red}(II \textit{ relapsing into } I)$ and $P_{blue}(II \rightarrow I)$.

Figure 6.5: One cycle model where transition follows one direction.



Shortcoming

This model does not look at the amount of time elapsed during the night. In this respect, subjects generally pass through this cycles 4 to 6 times. As the cycles progress, the configuration of the cycles change moderately. However, this model fails to look at these issues and possibly introduce high levels of error by lumping various cycles together.

Data Requirement

The data requirement for this model is relatively similar to the previous model. It has the same number of parameters to set. The only difference being that the training data must first be examined to identify the direction in which the transitions are destined. This process would be highly manual and time-consuming. Therefore, an implementation will not be provided for this model.

6.2.3 Four-Cycle and Six-Cycle Model

The one-cycle-one-direction model addressed the issue of useful direction information missing from one-cycle-duo-direction model. However, it fails to record the cycle in the context of the entire night's sleep. The four-cycle and six-cycle model aims to address that issue. Since it is well recognized that at least four cycles are experienced by most healthy subjects, this model looks at the first four cycles. Figure 6.6 shows the transition diagram. The six-cycle model just goes through two more cycles before returning to awake.

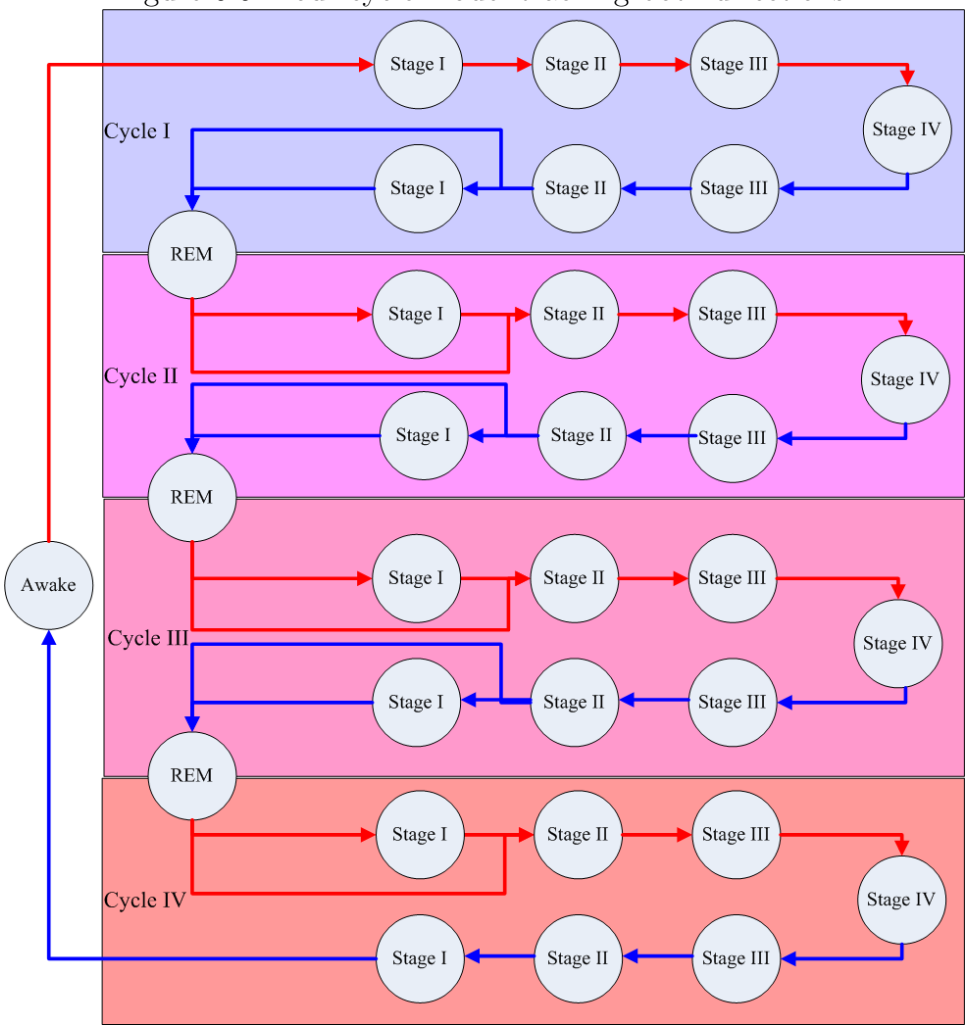
Despite that the diagram does not show the MT state, it is still considered in the model. The MT state is merely not shown graphically due to complications in display. Again, it should be noted that there should be an MT state for each non-Awake states. The reason for individual MT states is that following state i$\rightarrow$ MT, returning to state i is far higher than going into another state.

Shortcoming

The four-cycle model ignores the last two potential cycles. While it is difficult to obtain consistent patterns from the last two potential cycles, their frequent presence means it is important to include them in the model. However, a six-cycle model means that the last two cycles must be considered whether they occur. This issue can be dealt with by allowing optional cycles, but then the model would be more complex to construct.

These two models are very cumbersome. In order to make the model manageable, only the transitions vital to the cycle would be determined. Unlike the earlier models, such a complex model cannot have each state connecting to each other state, where many connections would never occur. For instance, there would not be a cycle 1 state 4 transition into cycle 3 state

Figure 6.6: Four cycle model tracking both directions.



3. However, the design choice, to only study the standard transitions, would mean that events such as skipped states, or doubling back to a previous, or waking intrusions, etc., are all being ignored. Therefore, some thought should be put into the model such that most possible and non-zero transitions are included in the model.

Data Requirement

The data requirement for these models is significantly higher. The number of parameters is approximately four and six times that of the one-cycle-duo-direction model. Since each sleep study can only provide one set of information, the quantity requirement jumps by a multitude. In terms of quality, the data sets must contain at least 4 continuous and relatively standard cycles.

Also the choice of data must look at the tradeoff between good sample space of training data and similarity to the base model. A large and representative collection of sample sleep studies will make the model more generalized. Their average will be a good starting point. However, if the samples are vastly different from the base model, then the variance exceeds the usefulness of the average value. Therefore, it is important to choose a representative yet coherent set of training data.

For the six-cycle model, further difficulty arises due to the inconsistent occurrence of the last two cycles. These cycles are also likely to be marred by multiple waking and dreaming.

6.2.4 Multi-Cycle-Connected Model

The last two models fail to contain the minor connections between states. For instance, state 1 is often skipped before entering REM, which means a direct transition from stage 2 to REM would be reasonable. As discussed previously, a fully connected model would be overly complicated and containing significant useless information. Therefore, this model aims to connect only the possible connections.

This model can be adapted to 4 or 6 cycles. It will include all the connection to Awake and to MT. It should assume possibilities of skipped states, of states that double back, etc.

Shortcoming

Many of the transitions added in this model are rare and they do not contribute significantly to the contextual information. For instance, consider a patient in NREM II for several minutes, shifts to I for one epoch, and returns to II immediately afterwards. The detection of that NREM I is not particularly vital. Therefore, to model the relatively insignificant transitions as equals to the main transitions would be placing undue emphasis on the

added connections. Furthermore, due to their rarity, their transition probabilities will be too small to notice in the whole picture. One solution to this problem is to model these connections separately.

Data Requirement

It should be noted that the data requirement for this model is even higher. Since it not only models the basic transitions shown by physicians, it also models secondary transitions. It is clearly difficult to collect representative data sets to model the secondary transitions, which occurs in rare cases. Therefore, this model will require the significant preprocessing of the data sets. In particular, the secondary transitions must be identified to parametrize separately.

6.2.5 Duo-Layer-Multi-Cycle Model

This model improves upon previous models by modeling main transitions separately from secondary transitions. The duo-layer approach is achieved by building a model with two levels of states, *cycle-level states* and *epoch-level states*.

Cycle-Level States

The cycle-level states demonstrate the overall behavior. It takes the form of the six-cycle model but acts like a continuous-time Markov chain. A plot of the transitions in cycle-level states would correspond to the Figure 6.1. The four-cycle equivalent transition diagram is in Figure 6.6[3].

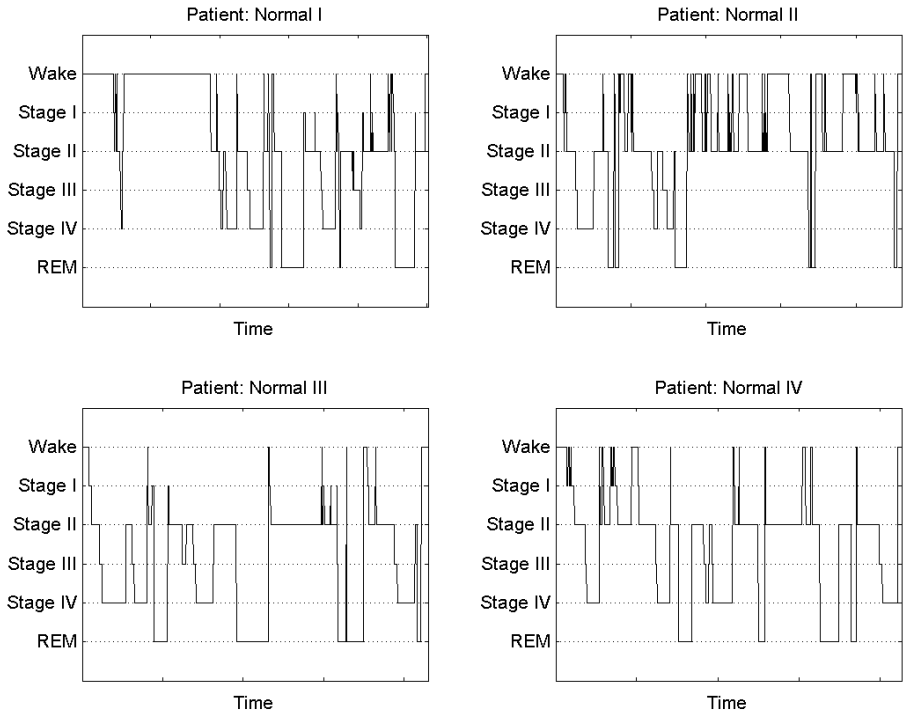
Ideally, sleep data should be analyzed to identify the distributions that most suit each state. However, in this study, a normal distribution is assumed for the duration in each state. Therefore, each state must be associated with an average duration and a standard deviation. Only the transitions linked in the figure are assumed possible. In order to deal with transient states that are too short to score, these states can have duration of 0.

[3]Note that the model being designed will assume it is not possible to skip NREM I between Awake and NREM II or REM and NREM II.

Epoch-Level States

From actual sleep studies, the plot in Figure 6.1 is not realistic. Figure 6.7 shows that aside from the main state transitions, many small spikes may occur in the behavior. These spikes may be short transitions into the previous states, next state, Awake, or MT. However, the cycle-level states will not be able to account for these spikes. Therefore, inside the cycle-level states, a 5-state structure is constructed to account for this kind of behavior. This structure is shown in Figure 6.8.

Figure 6.7: The sleep architecture of 4 normal subjects.



For these 5-state structures, the overall sleep study will still consider the subject as being in the *current state*. However, the subject may for short periods transition into the 4 transient states. The model contains a probability for such transitions and the expected duration in those states.

Figure 6.8: 5-state structure.

Shortcoming, Data Requirements, and Implementation

This model is designed to overcome the shortcoming of previous models. Despite encapsulating more information, its data requirements will be equivalent to that of the six-cycle model, because it can reuse the same data sets to extract the epoch-level states. Due to its comprehensive nature, this model is ultimately considered the most representative. Its implementation is significantly more complex than previous models, and it will be discuss in section on implementation.

6.3 Context Implementation

In the previous section, various models were considered. The Duo-Layer-Multi-Cycle Model was selected because it has the most potential to represent the context information properly. This section designs a method of implementing this model. First, the implementation challenges are analyzed followed by an algorithm design.

6.3.1 Problem Analysis

An algorithm that would develop the sleep cycle model described in the previous section must resolve several issues. In order to parameterize the average duration in the cycle-level states, the algorithm must overcome the lack uniformity between subjects. The algorithm also need to smooth individual curves such that individual durations can be extracted for variance calculation. The most significant obstacle is to select a set of data with sufficient quantity and adequate quality from which to build this model. First, take a closer look at the source of these problems and attempt to identify some solutions.

Lack of Uniformity

While the medical community presents a generic sleep cycle pattern, shown in Figure 6.1, high levels of inter-subject difference still exists. This fact is demonstrated in Figure 6.7. In fact, if the sleep cycle of the same subject on two different nights were observed, the cycles are still likely to be different.

Sleep Onset Variance The first reason for the variance is that the onset of sleep may be different. One subject might have fallen asleep within 5 minutes of closing their eyes, while another patient can take 2 hours. This manifestation is related to the degree of tiredness the subject feels, whether the subject is excited or worried about the study and the new environment. There exist studies that determine the average wait before sleep onset. However, the number of uncertainty factors relevant to this issue is too high to model efficiently.

However, if the onset is different, the first transition between Awake and Stage I would be offset. In order to correct that issue, all data sets should pre-

processed to start at the same epoch, meaning the first transitions occurring in a fixed epoch. The entire data segment would be shifted accordingly.

Fatigue Level Physical and mental fatigue levels will determine partly the sleep cycle pattern. If subjects feel physically tired, they may have longer NREM IV periods or multiple occurrences of NREM IV. Emotionally burdened subjects may experience more REM periods. The idea behind these observation is that REM and NREM IV provide the best stages to get psychological and physiological rests, respectively.

There is no method by which to assess the degree of tiredness associated with the subject in each data set. Of course, a particular data set that contain tiredness information can be generated. For instance, a survey can be filled out by the subjects before going into the sleep study, or the subjects can be requested to perform certain tasks associated with a particular type of fatigue. Clearly, this type of data sets would be expensive and time-consuming. Though in a real continuous sleep monitoring situation, the fatigue information would be available through recent history analysis.

In the case where no fatigue level can be ascertained, this factor will be considered as an uncertainty.

Subject Profile The sleep cycle also vary depending on the patient's profile in terms of age and sleep disorders. It is well-documented that as subjects become older, their sleep cycles change. The older subjects tend to get lighter sleep as their circadian rhythm control mechanisms weaken. More specifically they experience shorter NREM III-IV and longer REM in the first cycle. In some cases, medications taken by elderly patients may affect sleep. Also, sleep disorders, such as the condition sleep apnea and restless legs are more common in older subjects.

This type of profile information can be derived relative easily from the medical records of the sample data set subjects. However, it is difficult to get significant number of cases with a particular profile, meaning that deriving a pattern is not easy. Another issue is that this information may not be available when the automatic sleep staging algorithms are used. One way of circumventing that issue is to adjust all data sets into a standard mold. However, such an adjustment function would not be easy to generate. Therefore, these factors must be considered again as a source of uncertainty.

Interpersonal Differences Finally, even when two subjects with similar profile put under the same physiological stress, their sleep cycle still will not coincide with each other exactly. There will simply always be some interpersonal differences. There is no way to account for them. However, these differences should be minor enough that a prominent pattern can still be derived despite the differences. In the general situation, however, these interpersonal differences will be so small that they would not significantly contribute to the standard deviation.

Spikes in Data

Most scored sleep studies will contain spikes in data. These spikes occur for reasons including issues in sleep scoring, data selection, and natural causes.

Sleep Scoring Sleep scoring attempts to impose a discrete set of states on a continuously and gradually changing biological process. This set of states are not proven to be perfectly correct, though they are widely accepted. In order to avoid ambiguity, the rules through which these states are meant to be scored are not always intuitive. Furthermore, the scoring accuracy depends on the scorer's experience. Of course being a tedious task, mostly technicians with little experience are given this task. These technicians may not understand the biological process or the nature of the rules such that they would score with error or inconsistency. Under such circumstances, spikes may occur either because the scoring rules failed to provide smooth transitions in the scoring, or because the scorer did not classify the borderline cases properly.

This issue is inherent to the sleep scoring problem, and it will exist both in the manual or the automated versions. As the goal of the automated version is to mimic the manual version, modeling with this issue is not a particularly important issue.

Data Selection This issue is associated with the previous problem. Without selecting scores that were provided by an experience authority, the scores may contain inconsistencies or even errors. Also, without selection some of the particularly spiky data sets might be included as source data. This problem can be resolved by filtering the source data sets. In this process, scores that appear particularly noisy or inconsistent should be eliminated. Particular attention should be extended to data sets from experienced sleep scoring

experts. Furthermore, it is worthwhile to select data sets that exhibit a range of behavior. One must be careful not to involve a data set that is so unique that the model is too biased from the standard.

Natural Causes Aside from the external factors to cause spikes in sleep cycle, nature does play an important role as well. For instance, the subject may be disturbed at night by noise or movement causing the subject to transition from REM to Awake. When the stimulus is removed, the subject will most likely transition back. These will also appear as spikes. Similar explanation can be provided to transitions into movement time, into previous, and logical next states.

Since the model is supposed to capture this biological process, the manifestation of natural causes should not be ignored. The 5-state structure was designed to capture this kind of spiky behavior. Refer to the previous section for details on this structure and the specifics of how this structure can model these natural spikes. However, steps to prevent data sets that are artificially spiky should be taken in order to minimize the bias towards this behavior.

Data Set Shortage

The last problem facing modeling the sleep cycle is that there is a shortage of data sets. Each sleep study will only result in one set of data. Therefore, if getting a good statistical average requires some hundreds of samples, significant resources would have to be devoted to the data collection and data processing. Currently only two dozen data sets are available for construction and for validation. Therefore, this data shortage will present significant issues.

One issue is that there will be a bias to one or two scorers. While the eventual model may simply reflect the style of the participating scorers, it may be a problem if they are not particularly accurate. Another issue is that if the two scorers have contrasting methods, converging on a reasonable model may be difficult. The resulting model may also be inaccurate since there is no way to measure the goodness of the approximate averaging effect.

The serious nature of data set shortage means that this report will not be able to present a reasonably viable model. The algorithm will be designed and tested to deal with as many of the above issues as possible. It will be tested with the limited resources available to indicate the algorithm can be expanded to develop a real model with the right data sets.

6.3.2 Algorithm Design

The modeling process will identify the parameters of cycle-level and epoch-level states. The top view of the modeling process is Part A in Figure 6.9. The figure shows that there are 3 processes stemming from the same source data. Each process is aimed to identify a subset of the parameters. The first and second processes extract the average and variance in the cycle-level states duration, respectively. The third process extracts the epoch-level parameters. The output of these processes form the final model.

The most important step in this algorithm are the *average* procedure at the beginning of the first process. Its output feeds into all the processes. Given the lack of uniformity in data, a generic averaging algorithm would cause the sleep cycle pattern to be flattened, as seen in Figure 6.10. The resulting curve would not fit into the discrete sleep stages or the six-cycle model.

Since the average curve is meant to have the minimum distance to each of the samples, then a search algorithm can approximately find the optimal curve. This problem can be solved using genetic algorithms. Genetic algorithms are a class of algorithms that searches for optimization solutions by mimicking biological evolution. The algorithms start by generating a set of solutions. Then the algorithm iteratively follow the steps below,

1. Select the best solutions from the set.

2. If the best solutions are not good enough, modify these best solutions and put the modified solutions into the set.

Two common methods of modification is mutation - slightly altering a solution or crossover - combining two solutions. Its flowchart is Part B in Figure 6.9. Using this method, the same function can be used by the second and third processes to smooth out each data set.

Search algorithm using genetic algorithms requires the following points to be addressed: the definition of the individuals, the initial population, the health measurement, the termination condition, and the birth process.

Definition of Individuals

The individuals in this context represents each possible solution. These solutions are assumed all viable but with varying fitness levels. In this context,

Figure 6.9: Flowchart (a) of overall modeling process. (b) of average procedure.

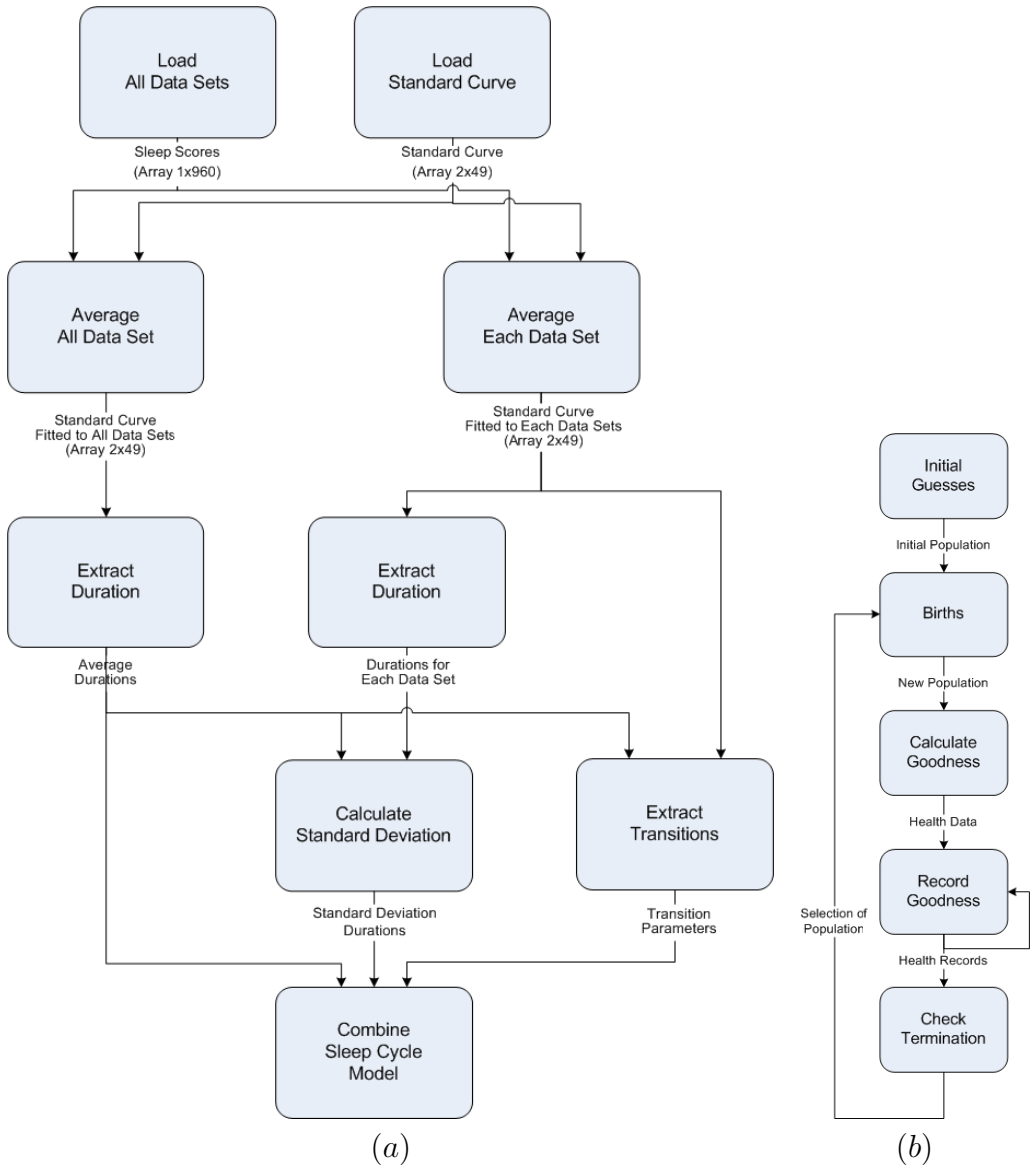
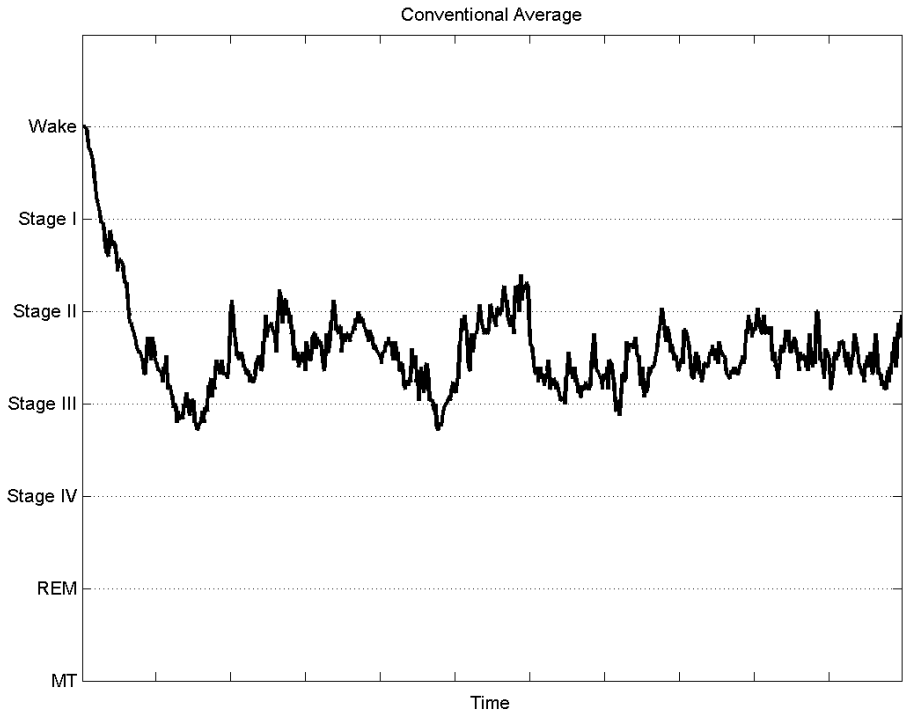


Figure 6.10: Conventional average algorithm flattens sleep cycle.



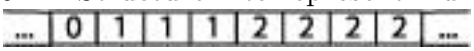
these individual's fitness level means their distance to the data sets in general. The structure of the individuals determine the method by which the fitness can be found.

Epoch-Stage-based Definition

This definition identifies each individual based on an array of $2epoch/minute \times 60min/hour \times 8hours/study = 960$ elements, where each element indicates the stage scored. Figure 6.11 shows a small segment of this array. One advantage of this model is that the data sets are already stored in this format. The health can be easily defined as the minimum distance between each corresponding point-pairs from an individual and the reference data set. The disadvantage is that no obvious birthing processes can be applied. Since the individual must retain the six-cycle model, the design for good mutation and crossover algorithms is seriously limited.

Duration-based Definition

Figure 6.11: Structure 1 to represent individuals.



A different structure is to keep an array of duration spent in each states. The six-cycle model has a 49-element array. A sample structure is shown in Table 6.5. This structure simplifies the design of mutation and crossover algorithms, and those algorithms will be discussed in Section 6.3.2. Calculating the health directly from this structure is more complex than the epoch-stage-based definition. However, this form can easily be converted to the first structure.

Table 6.5: Duration table for the standard curve.

Stage	0	1	2	3	4	3	2	1	5
Cycle I	2	2	10	2	40	2	10	2	20
Cycle II		2	10	10	20	10	10	2	26
Cycle III		2	15	12	6	12	15	2	26
Cycle IV		2	18	5	0	5	18	2	40
Cycle V		2	5	0	0	0	5	2	46
Cycle VI		2	5	0	0	0	5	2	46

Initial Population

There are three options in the design of the initial population.

1. The first option takes the sleep cycle curves provided by various medical sources as the base population. It is beneficial to start with the curve that approximates the ideal behavior, because it is already likely to be in excellent health. But these curves would be highly similar and they would not have enough variation to encourage new sample spaces. In this case, radical mutation or crossover can be attempted.

2. The second option is to randomly generate a population. The randomness in this step allows coverage by a significant sample space. However, it is possible that none of the future population will be close to the solution.

3. The final option is to visually analyze a subset of the data files, and record the pattern manually. Such an initial population starts closer to the data, but will migrate to converge all the data sets. Similarly, it may fall into the problem of being too close to some sets of data and never leaving the local minima in its area. Good design of mutation and crossover algorithms may overcome this issue.

Health Measure

Section 6.3.2 explained the basic method of determining the health by calculating the distance between an individual and a data set. While this method provides a basic measure, it is not most reflective of the health of an individual. The sleep stages are converted to discrete values on the computer, and this is a casual assignment. Therefore, the values are not reflective of the relative distance of one state to another.

Table 6.6: Goodness weights.

Stages	0	1	2	3	4	5
0	0	1	1	1	1	1
1	1	0	1	1	2	1
2	1	1	0	1	2	1
3	1	2	1	0	1	2
4	1	2	1	1	0	3
5	1	1	1	2	3	0

For instance, a REM state may be classified as 5, but it is naturally next to NREM I, whose value is 1. So their distance should be 1 instead of 4. Furthermore, MT is classified as 6, but it is never more than 1 away from any non-awake states. Therefore, a weighting table that reflects the physiological distance is constructed as in Table 6.6. In practice, this table is normalized and de-referenced each time the distance between a data point and a curve point must be determined.

The method of aggregating the distances of a data set to an individual is not determined. It is possible to use the average distance or the sum of all distances. The average method would give me a figure independent of the length of data set after lining up the arrays. On the other hand, it might be useful to have an indication of the amount of sleep based on the data length.

Termination Condition

There are multiple termination conditions that would make sense in this case. For instance, the algorithm can terminate once a pre-specified level of goodness is reached or when a certain number of generations was reached. A better criterion would probably be number of generations where performance has not been improved upon.

In this implementation, a particular method is used. An array of n elements is kept as a goodness record for the solutions already tested. Each element contains a field for the associated distance measure and a field for the array defining the individual. When a new individual with a shorter distance than the worst individual in the array, the new one is inserted in sorted order and the one with the worst distance is removed.

Birth Process

The birth process must be designed to complement the definition of individual. Using the duration-based definition, both mutation and crossover algorithms can be designed. Figure 6.12 shows one design of the birth process. This design random selects a subset of the population as input to *mutation one*, one subset for *mutation two*, and the rest for *crossover*. Each of these procedures will output its own next generation and *check for instant death* is executed to remove those individuals that fail the current definition of an individual. For instance, the individual that does not translate to the appropriate number of epochs would be instantly killed. Through this process only viable individuals are "born".

There are 5 aspects of this process that require further study, the implementation of *mutation one*, *mutation two*, *crossover*, the combination of the latter three procedures, and the selection of parents.

Mutation One

Mutation One is designed to carry out drastic change from the parents to the offspring. It cuts the parents into segments and reattach the pieces. The location of the cut, the number of cuts, and the location of the reattachments would be randomly generated. The Figure 6.13 shows how one parent bore three different and healthy offsprings.

Mutation Two

Mutate Two aims to adjust the genetics by small amounts, for final tuning near the end. This method shifts the boundary between two neighboring

Figure 6.12: Flowchart for Birth Process.

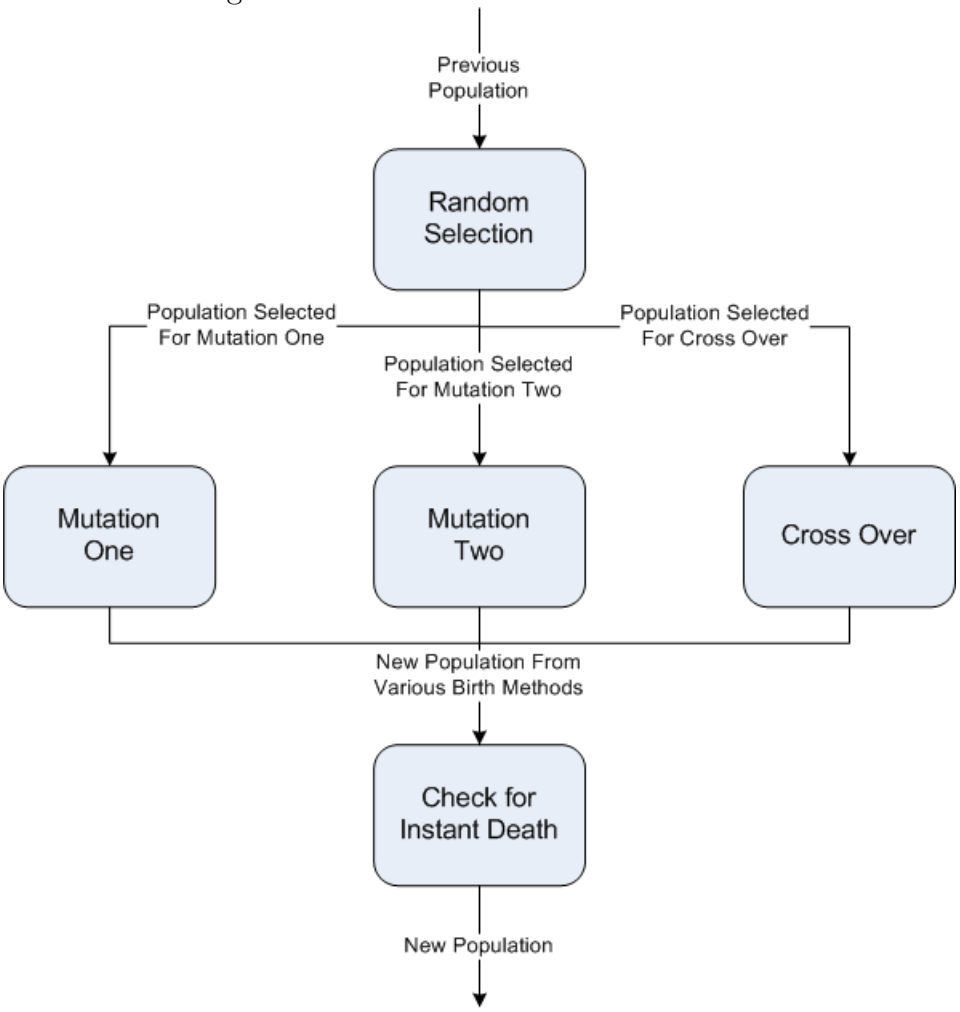


Figure 6.13: Mutation by cut and reattach segments.

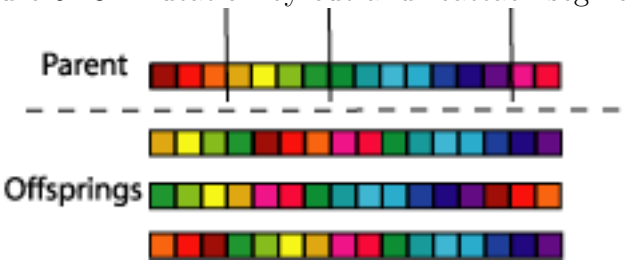
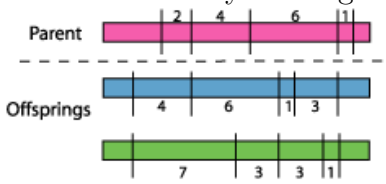


Figure 6.14: Mutation by shifting boundary.

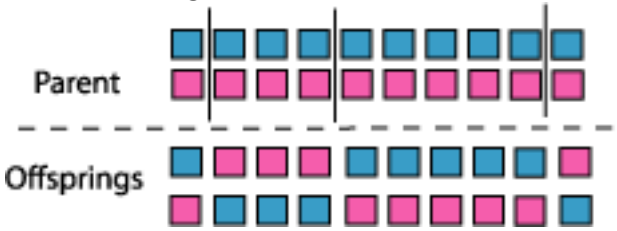


segments. Figure 6.14 shows that but adjusting the boundary, many new individuals, highly related to the previous generation, will be generated. While this process is slow, it analyzes a region of sample space closely. To that end, it would be the best mechanism to use near the end of the search. Its attention to detail is likely to identify the closest solution.

Crossover

Crossover brings together two parents to produce the offsprings. The parents are cut at random points, and the segments are shuffled and rejoined. Figure 6.15 demonstrates crossover. The resulting offspring is also likely to explore a bigger sample space.

Figure 6.15: Crossover.



Mechanism Combination

The design in Figure 6.12 uses the three genetic algorithms in parallel. However, it is also possible to set these mechanisms in series. When these genetic operations occur in series, even more random or variation is created in the population. This type of set up would be suitable for a randomly created initial population, which is the second option described in Section 6.3.2. However, the initial population based on first or third option would lose performance with too much variation. The initial population was already close to the target, and the excessive variation may cause the algorithm to completely miss the nearby target.

Parent Selection The selection of parents will have significant effect on the offspring. With the parallel layout of the genetic operations, two of the mechanisms are attempting to provide variation, while the third aims to refine the search. Therefore, it may be worthwhile to replace *random select* with a deterministic algorithm. This algorithm would direct very good parents to *mutation two*, and direct a combination of other parents to the other two mechanisms. Note that randomness is preserved because the genetic algorithms inherently contain the random factor.

Parameter Extraction

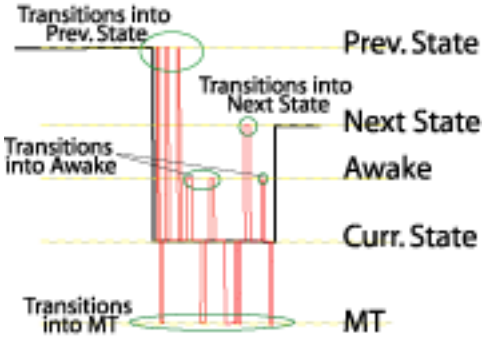
Though *average* described above does not provide a conventional average between a list of data sets, it does result in a pattern with minimum overall distance to all the data sets, while maintaining the standard sleep cycle shape. As indicated previously, this function is used also to smooth each data sets. Following these processing, the parameters needed in the duo-layer-multi-cycle model can be extracted.

Cycle-level State - Average Duration The pattern resulting from *average* applied on all data sets will be an "average" of all patient data. The pattern is recorded as a set of durations, which can be extracted directly for the cycle-level states.

Cycle-level State - Standard Deviation For each data set, the duration of the cycle-level states can be extracted from the smoothed data sets. These durations form a range, and based on the average determined previously, a distribution can be established.

Epoch-level State The smoothed data sets are used to delimit the cycle-level states. With in each cycle-level state, the frequency of transitions into Awake, MT, previous, and next states can be calculated easily. At the same time, the average duration in each type of transition can be determined. Figure 6.16 demonstrates the process of identifying the transitions. It should be noted that for the cases where Awake is the previous or next state, the frequency and duration are split between the two situations.

Figure 6.16: Epoch-level states determination.



6.3.3 Algorithm Design

The design in Figure 6.9 was implemented in Matlab v6.5. Due to a shortage of data sets, the full capability of this sleep cycle modeling algorithm cannot be illustrated. However, the available data sets were used to experiment with the basic functionality of the program. It should be noted that only *mutate two* was implemented, because the initial population was generated based on option one and three. Also, time limitations only allowed a small subset of the various design considerations discussed in Section 6.3.2 to be explored. The rest of this section demonstrate the function of this algorithm and its results. Then this section provides some analysis of the design considerations that were studied. It would be recommended to study the remaining design considerations at a later date.

Cycle-level State Parameterizations

Despite the lack of sufficient data sets, the algorithm was tested on existing data sets. Figure 6.17 shows the resulting curve compared with the curve provided as an initial guess. As can be seen from the diagram, the averaged curve has moved significantly from the initial guess. From the curve, the average durations are extracted and presented in Table 6.7.

Interestingly, the characteristics discussed earlier can be observed in this averaged curve. As the night progresses, REM's duration increases and NREM IV's duration decreases. In fact, in the last two cycles, NREM IV is absent. NREM I and NREM III both have very short durations throughout the night reflecting that they are transient states.

The standard deviation is also generated from the same data set. The

Figure 6.17: Average sleep cycle versus pattern from literature.

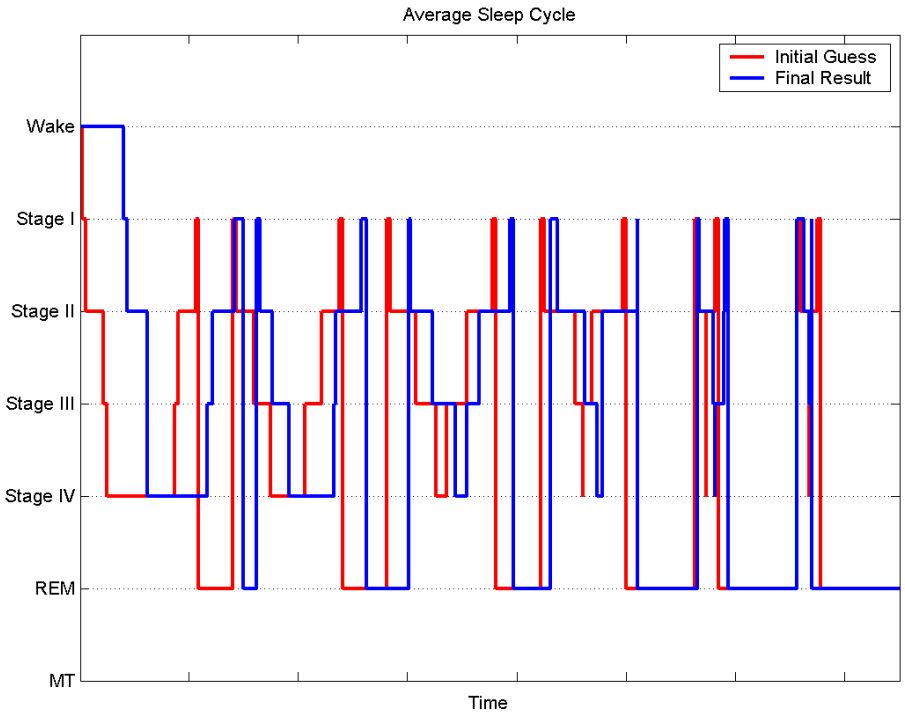


Table 6.7: Average duration.

Stage	0	1	2	3	4	3	2	1	5
Cycle I	26	2	12	0	35	3	13	5	8
Cycle II		2	7	10	26	1	15	3	25
Cycle III		1	13	13	7	7	18	2	22
Cycle IV		4	16	7	3	0	21	0	35
Cycle V		1	8	1	0	5	1	2	40
Cycle VI		4	3	2	0	0	0	0	51

values are presented in Table 6.8. By observation, the magnitude of the standard deviation in duration are proportional to the average duration. This observation indicates that the model for these cycle-level states can be simplified into an exponential distributions with only the average duration as parameter.

Table 6.8: Standard deviation in duration.

Stage	0	1	2	3	4	3	2	1	5
Cycle I	21.9547	12.557	10.344	4.2622	11.654	2.7767	8.0108	3.7563	8.7630
Cycle II		4.7039	8.9994	5.6883	7.6241	5.6666	10.063	2.1016	8.7936
Cycle III		2.7881	6.3048	5.5308	3.6955	4.8795	5.9279	2.8798	6.8418
Cycle IV		2.2121	3.9946	3.7430	3.2419	2.1602	8.3702	2.1856	5.9461
Cycle V		1.4059	1.9149	1.3441	2.2531	1.0801	4.8108	1.5631	7.5613
Cycle VI		1.6753	2.1409	0.52281	0.58310	0.77028	2.1276	0.47258	2.7970

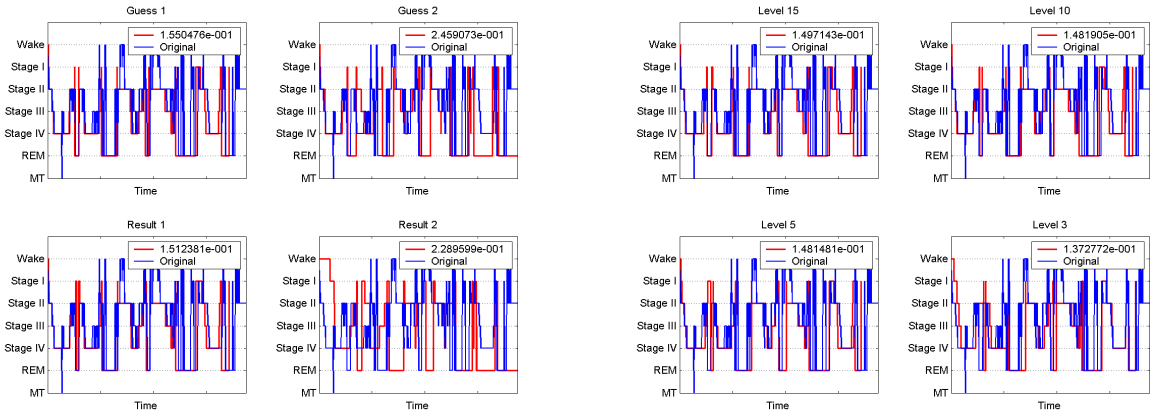
Quality of Initial Population

A good initial guess of the standard curve will affect the smoothing performance. Figure 6.18 Part A demonstrates this effect. Guess 1 was generated based on manual observation and Guess 2 comes from a standard curve in literature. Guess 1 outperforms Guess 2 by a significant margin. Since the algorithm using the *mutate two* mechanism, can only improve the solution, both results shows improvement from the guess. However, Result 2 cannot match the performance of Guess 1 or Result 1. Therefore, unless the other genetic algorithms are implemented to bring in variance, the current algorithm relies heavily on a good initial guess.

Mutation Parameters

The current version of *mutate two* incorporates two parameters. The first parameter *mutation number* determines the average number of mutation spots to occur in each generation. The second parameter *mutation level* determines the average amount of boundary shift. Figure 6.18 Part B demonstrates the effect of changing *mutation level*. The levels used are 3, 5, 10, and 15. Note that the larger levels correspond to poorer performance. Since this mutation algorithm is meant to refine the search towards the target, large shifts in boundary will cause the algorithm to miss the target. However, the factor that cannot be seen from the graph is that the smaller levels require far more generations to find a local minima, where as larger levels work much faster. Experiment with changes in *mutation number* has similar effect.

Though these two parameters are adequate for this simple mutation algorithm, better adjustment parameters are required to improve the performance of the algorithm. One idea is to use adaptive parameters to introduce both speed and performance gains. Also, the parameters for the other two



(a) (b)

Figure 6.18: Design considerations.(a) Quality of initial guess. (b)Mutation parameter.

proposed genetic algorithms will have significantly different dynamics. These design consideration need to be studied at a later date.

6.4 Context Application

While this chapter has selected the Duo-Layer-Multi-Cycle Model to represent the sleep structure, the lack of sufficient data prevents this model from being implemented well enough to demonstrate its application. Therefore, this section uses the One-Cycle-Duo-Direction Model discussed in Section 6.2.1 to demonstrate an example of the effect context would have on features developed earlier in this thesis.

6.4.1 EEG Band Features

The set of features considered in this method is the Beta Power discussed in Section 4.1.3. Specifically the feature is relative power with a scaling factor of 2 and its ability to differentiate Awake from other stages.

6.4.2 Formulating in Context

The formula by which the context is considered for a feature that differentiates Stage i and Stage j is

$$f_{adjusted} = f \times \left(\frac{P\{\text{Current Epoch is i}\}}{P\{\text{Current Epoch is j}\}} \right)^n,$$

where f is the raw feature value and n is the emphasis factor. This formula considers the context, in that it uses the previous epoch's stage to provide a probability of current epoch being some stage i or j. The value n allows us to control how much context can influence the feature value.

6.4.3 Results and Discussion

The usefulness of this type of contextual information is demonstrated in Table 6.9. This table considers the Ratio feature's ability to differentiate the Awake state from the other states. The left side of the table shows the sensitivity and specificity of the feature on its own. The right side shows the

Table 6.9: Ratio feature to differentiate Awake from other stages.

Stage	Orig. Sens.	Orig. Spec.	New Sens.	New Spec.
NREM I	0.9644	0.8475	0.9666	0.9153
NREM II	0.9936	0.9651	0.9920	1.0000
NREM III	0.9973	0.9787	0.9984	1.0000
NREM IV	1.0000	0.9803	1.0000	1.0000
REM	0.9352	0.8279	0.9947	0.9953

performance of this featured adjusted with contextual knowledge. It should be noted that the context maintained or improved the performance.

Without context, this feature had low accuracy in terms of differentiating from NREM I and REM. One reason is that these two stages both bear some resemblance to the Awake stage. However, the context is able to improve awake and REM differentiation by nearly 16.7%.

It also improves the performance for differentiating NREM I. However, the improvement is not as noticeable as REM, because NREM I, the transient state, may not have enough occurrence to have significant representation in contextual analysis.

Therefore, it is clear that even with such a simple method of including the context information, the features designed based on EEG and EOG can be biased to perform better. Note however, since this particular model is fairly simple, it cannot improve the performance better because the transitions being represented change over the course of the night. Furthermore, the use of bias work for majority of cases but may trigger some false alarms where bias would be working against a specific decision.

6.5 Summary

This chapter studied various aspects of context analysis. First, it studied various models that can represent the complex stage transition sequence. It was determined that the most effective model would be Duo-Layer-Multi-Cycle model because it can both account for the true and transient transitions. The true transitions corresponds to the progression of sleep stages over the span of the night. The transient transitions, on the other hand, correspond to the temporary shifts into neighboring stages either caused by ambiguous scoring or by short-lived events.

Next, this chapter identified a method of building the Duo-Layer-Multi-Cycle model based on an artificial intelligent optimization algorithm to parametrize the epoch- and cycle-levels. This approach builds an average model while maintaining the discrete nature of the stages. A preliminary model was trained based on the limited sets of data. However, its representation of a realistic sleep architecture is clear.

Finally, this chapter attempted to demonstrate the ability for context information to influence the classification. Specifically, it used a simple One-Cycle-Duo-Direction model with an EEG band feature to demonstrate an improved ability to differentiate Awake from other stages. It was shown that in some instances, the performance can be raised over 10% depending on the ability for the feature's original performance.

To fully utilize the power of contextual information, significant more resources will be necessary to encapsulate all the expert knowledge tied into context analysis. Just to fulfill the selected model, many more sets of data sets will be needed. If patient-based and environment-based context were to be included, additional sets of data would be necessary to develop trend of the model's parameters compared against each variable.

Chapter 7

Conclusion

This thesis analyzed various ways to improve automated sleep staging. It looked at the optimization of the classifier, primarily for accurate Awake versus Sleep distinction. It found that the neural network MLP can be more easily trained and fine tuned to optimize a quick response to wakefulness detection. Such a system is versatile in the type of input, is light-weight, and can produce sufficiently accurate results in near real-time. Specifically, the MLP optimized in this thesis with spectral features from various EEG band activity has an approximate 91.91% accuracy when validated across 3 patients. Aside from MLP, Hidden Markov Model was trained with Hjorth parameters to an approximate accuracy of 77.36%. During the design of HMM, it was found that separating Awake and Sleep is more effectively accomplished by actually differentiating all the sleep stages.

The performance limitations of the classifiers were in part contributed to the quality of features used as input and in part of the quantity of training data. Improved features were examined primarily based on EEG and EOG signals, because a significant portion of the sleep stage definitions involved activities in them. In both instances, using time frequency representation was able to analyze these two signals in both time and frequency domain more closely mimicking the thought process of human scorers.

For EEG, spectrograms and scalograms are used for band activity feature extraction. A spectrogram feature was optimized to differentiate Awake with 93.52% sensitivity and 94.60% specificity, and differentiate REM with 96.12% sensitivity and 93.63% specificity. A scalogram feature was optimized to differentiate Stages II and III with 96.81% sensitivity and 89.28% specificity, and differentiate Stages III and IV with 93.60% sensitivity and 90.43% speci-

ficity. In addition to band features, characteristics waves were investigated by using MLP for pattern recognition. For K-complex, Vertex and Sawtooth waves the accuracy were 82.14%, 63.89%, and 68.18%, respectively.

For EOG, it was found that the 30 second epochs needed to be first analyzed in 5 second segments. Features were optimized to differentiate various EOG activities at the segment level with 70% to 100% sensitivity and 85% to 97.5% specificity. Using a set of discrete rules based on 9 segments as dominant in an epoch, Awake can be identified with 90% sensitivity and 56.67% specificity, Slow Rolling Movement with 0% sensitivity and 56.67% specificity, EOG Delta with 100% sensitivity and 91.67% specificity, and Rapid Eye Movement with 45% sensitivity and 96.67% specificity. The rules can be adjusted to reduce the performance for Delta in favor of the other 3 activities.

In addition to EEG and EOG signals, it was found that contextual information also plays a big role in sleep staging. Due to limitation in data sets, the type of contextual information possible to consider in this thesis is time-based, meaning relative to the overall sleep cycles per night. Since there are multi-facets of sleep architecture information to be captured, it was necessary to identify a model to represent the context. A Duo-Layer-Multi-Cycle model was selected because it can both capture the cycle-level behavior as well as transient epoch-level behavior. A genetic algorithm was designed to parametrize this model. However, due to the lack of sufficient data, an application of this model could not be demonstrated. Instead, using a simple One-Cycle-Duo-Direction model, it was demonstrated that a contextual bias can be inserted into the rule sets. Using this approach, it was shown that performance of EEG features can be improved by up to 10%.

Since the accuracy of the average human scorer is 88% to 94%, some of the accuracies achieved by several EEG band features were comparable if not superior. With the additional performance improvement from the context information, the automated algorithms used in this thesis would surpass the human scorer in terms of accuracy. In addition to the benefits of better performance, the automated algorithms can score the data in real-time and repeat the same results reliably.

Overall, this thesis analyzed the various aspects of sleep staging automation that can be improved. While the full potential of these improvements could not be demonstrated due to a limitation on resources, it explored the various promising aspects and the rest of this chapter will list the additional work to further realize these improvements.

7.1 Additional Resources

In some sections of this thesis, it was clear that the ability to fully test certain algorithms were limited by the resources available. Below is a brief list of some additional resources that would benefit future research

1. In general, bigger data sets are necessary to enable effective training and testing. These data sets need to be collected over a variety of subjects, both health and with specific conditions. Furthermore, the data sets need to be scored by at least 2 experts who were trained separately.

2. Data sets representing transient stages such as NREM I and MT need to be built. Since these stages occupy a very small portion of each night's study, the ratio of their representation in the overall training data set is very small. Therefore, matching the number of training epochs among the various stages will help eliminate the problem of over-fitting to the dominant stages.

3. Data sets containing only specific characteristic waves are necessary to build better detectors for these waves. These data sets should be built by expert scorers with a grading of the quality of each sample. Specifically, data sets for characteristic waves such as sleep spindle, K complex, vertex waves, and sawtooth waves are necessary for EEG. For EOG, the data sets should be classified according the the EOG wave type rather than deduced from the stage classification.

4. Expert knowledge needs to be further documented in a manner accessible to the automation researchers. Aside from the R&K manual, human scorers use a variety of more modern journal articles to supplement the original rules as well as their personal experiences. These information need to be documented and built into a format that can be incorporated into the classifier. Such information include how certain patient-based context is considered in the system.

5. Expert validation is necessary throughout the design of various intermediate mechanism in the overall algorithm. For instance, formulations of certain features should correspond to aspects considered by human scorers. If fuzzy inference is used, membership functions should be validated.

7.2 Future Research

The list below only capture some of the aspects that should be investigated further.

1. Develop rules based on the R&K manual. These rules should be supplemented by modifications made since 1969 in various journal articles. The rules should be transferable between discrete and fuzzy logic.

2. Develop additional EEG features, in particular those that cut across epoch divisions. For instance, tracking the change in amplitude of the EEG signal over several epochs or the ratio of amplitude between stages are useful information used by scorers.

3. Build better detectors for EEG characteristic waves. For example, instead of inputting the entire signal waveform, an intermediate step where useful features are used as input to the neural network is beneficial. Also, instead of outputting a binary decision whether the segment is or is not a certain characteristic wave, it should output a continuous grading. In such a case, there should be a range of ideal characteristic waves and then examples of gradual decline in quality.

4. Build a detector for the EEG characteristic wave, sleep spindle.

5. Link the EEG characteristic wave detectors with patient-based context. For instance, the strength and frequency of sleep spindles and K-Complexes are influenced by patient's age and fatigue level.

6. Investigate the dual channels of EOG, because human scorers often look at the symmetry between the two channels to indicate certain stages. For instance, in SRM and REM the LEOG and REOG are symmetric, whereas in BEEG the two channels are nearly identical.

7. Investigate the effect of EOG features on the simple rule set based on EEG features. It should be noted that features that are designed to perform well individually may not contribute the most to a combined system. Therefore, different combinations of EEG and EOG features should be tested.

8. Build a full version of the Duo-Layer-Multi-Cycle model. Formulate ways to link the detailed context information to the basic rule system.

9. Investigate whether overall algorithm are robust across a variety of situations, such as different data collection equipment and settings, across subjects and environmental factors.

Bibliography

[1] N. McGrogan, E. Braithwaite, and L. Tarassenko. Biosleep: A comprehensive sleep analysis system. Istanbul, Turkey, October 2001.

[2] J. Hasan. Past and future of computer-assisted sleep analysis and drowsiness assessment. *Journal Clinical Neurophysiology*, 13(4):295–313, 1996.

[3] A. Rechtschaffen and A. Kales, editors. Number NIH Publication No. 204. US Government Printing Office, Washington, DC, 1968.

[4] T. Roth. Characteristics and determinants of normal sleep. *Journal of Clinical Psychiatry*, 65(Suppl. 16):8–11, 2004.

[5] R. L. Williams, I. Karacan, and C. Hursch. *EEG of human sleep: clinical applications*. John Wiley and Sons, New York, 1974.

[6] E. O. Bixler, A. Kales, J. A. Jacoby, C. R. Soldatos, and A. Vela-Bueno.

[7] H. H. Jasper. The 10-20 electrode system of the internation federation. *Electroencephalography Clinical Neurophysiology*, 10:370–375, 1958.

[8] T. Penzel and R. Conradt. Computer based sleep recording and analysis. *Sleep Medicine Reviews*, 4(2):131–148, 2000.

[9] T. Akgl, M. Sun, R. J. Sclabassi, and A. E. etin. Characterization of sleep spindles using higher order statistics and spectra. *IEEE Transactions on Biomedical Engineering*, 47(8):997–1009, August 2000.

[10] S. Kubicki, W.-M. Herrmann, and L. Hller. *Critical comments on the rules by Rechtschaffen and Kales concerning the visual evaluation of EEG records.*

[11] J. Hasan. Automatic analysis of sleep recordings: A critical review. *Ann. Clin. Res.*, 17:280–287, 1985.

[12] S-L. Himanen and J. Hasan. Limitations of rechtschaffen and kales. *Sleep Medicine Review*, 4:149–167, 2000.

[13] Lengelle R. Schaltenbrand, N. and J.-P. Macher. Neural network model: Application to automatic analysis of human sleep. *Computers and Biomedical Research*, 26:157–171, 1993.

[14] J. T. Kelley, E. L. Reilly, J. E. Overall, and K. Reed. Reliability of rapid clinical staging of all night sleep eeg. *Clinical Electroencephalography*, 16(1):16–20, 1985.

[15] S. Roberts and L. Tarassenko. New method of automated sleep quantification. *Medical and Biological Engineering and Computing*, 5:509–517, 1992.

[16] T. Penzel, K. Stephan, S. Kubicki, and W. M. Herrmann. Integrated sleep analysis, with emphasis on automatic methods. *Epilepsy Research*, Suppl. 2:177–204, 1991.

[17] V. Brezinova. The number and duration of episodes of the various eeg stages of sleep in young and older people. *Electroencephalography and clinical neurophysiology*, 39:273–279, 1975.

[18] W. B. Webb and L. M. Dreblow. A modified method for scoring slow wave sleep of older subjects. *Sleep*, 5:195–199, 1982.

[19] W. B. Webb. Sleep in older persons: sleep structure of 50- to 60-year-old men and women. *Journal of Gerontology*, 37(5):581–586, 1982.

[20] A. J. Rowan and E. Tolunsky. *Primer of EEG with a Mini Atlas*. Butterworth Heinemann Health, 2003.

[21] T. Deboer. Brain temperature dependent changes in the electroencephalogram power spectrum of humans and animals. *Journal of Sleep Research*, 7:254–262, 1998.

[22] J. Hu and B. Knapp. Electroencephalogram pattern recognition using fuzzy logic. In *Conference record of the Twenty-fifth Asilomar Conference on Signals, Systems, and Comoputers*.

[23] M. Grozinger, C. Wolf, T. Uhl, C. Schaffner, and J. Roschke. Online detection of rem sleep based on the comprehensive evaluation of short adjacent eeg segments by artificial neural networks. *Prog Neuro psychopharmacol Biol Psychiatry*, 21(6):951–963, August 1997.

[24] O. R. Pacheco and F. Vaz. Integrated system for analysis and automatic classification of sleep eeg.

[25] A. Akin and T. Akgul. Detection of sleep spindles by discrete wavelet transform. In *Proceedings of the IEEE 24th Annual Northeast Bioengineering Conference*, pages 15–17, April 1998.

[26] J. Muthuswamy and N. V. Thakor. Spectral analysis methods for neurological signals. *Journal of Neuroscience Methods*, 83:1–14, 1998.

[27] T. Malina, A. Folkers, and U. G. Hofmann. Real-time eeg proceeding based on wavelet transform. June 2002.

[28] E. Oropesa, H. L. Cycon, and M. Jobert. Sleep staging classification using wavelet transform and neural network. Technical Report TR-99-008, Berkeley, California, March 1999.

[29] J. Fell, J. Roschke, K. Mann, and C. Schaffner. Discrimination of sleep stages: a comparison between spectral and nonlinear eeg measures. *Electroencephalography Clinical Neurophysiology*, 98(5):401–410, 1996.

[30] M. Grozinger, J. Fell, and J. Roschke. Neural net classification of rem sleep based on spectral measures as compared to nonlinear measures. *Biological Cybernetics*, 85(5):335–341, 2001.

[31] E. Huupponen, A. Vrri, S-L. Himanen, J. Hasan, A. Saastamoinen, M. Lehtokangas, and J. Saarinen. Eeg alpha activity detection by fuzzy reasoning. pages 411–416, Vancouver, Canada, July 2001.

[32] P. Van Hese, W. Philips, J. De Koninck, R. Van de Walle, and I. Lemanhieu. Automatic detection of sleep stages using the eeg. pages 1944–1947, Istanbul, Turkey, October 2001.

[33] I. Rezek, S. Roberts, and P. Sykacek. Complexity features for sleep stage analysis. pages 1650–1651, 1999.

[34] S. J. Roberts, I. A. Rezek, W. D. Penny, and R. M. Everson. The use of advanced information processing methods in eeg analysis. June.

[35] P.Y. Ktonas and A. P. Gosalia. Spectral analysis vs. period-amplitude analysis of narrow band eeg activity: a comparison based on the sleep delta frequency band. *Sleep*, 4:193–206, 1981.

[36] B. Hjorth. Eeg analysis based on time domain properties. *Electroencephalography and Clinical Neurophysiology*, 29:306–310, 1970.

[37] Micheloyannis S. Vourkas, M. and G. Papadourakis. Use of ann and hjorth parameters in mental-task discrimination. pages 327–332, 2000.

[38] Sun M. Liu, Q. and R. J. Sclabassi. Decorrelation of multichannel eeg based on hjorth filter and graph theory. volume 2, pages 1516–1519, August 2002.

[39] I. N. Bankman, V. G. Sigillito, R. A. Wise, and P. L. Smith. Feature-based detection of the k-complex wave in the human electroencephalogram using neural networks. *IEEE Transaction on Biomedical Engineering*, 39(12):1305–1310, 1992.

[40] E. Jovanov and V. Radivojevic. Software support for monitoring eeg changes in altered states of consciousness. pages 129–141, Belgrad, Yugoslavia, September 1997.

[41] T. Shimada, T. Shiina, and Y. Saito. Sleep stage diagnosis system with neural network analysis. volume 20, pages 2074–2077, 1998.

[42] L. Rabiner and B.-H. Juang. Prentice Hall, Englewood Cliffs, NJ, 1993.

[43] A. Flexer, P. Sykacek, G. Dorffner, and I. Rezek. Using hidden markov models to build an automatic, continuous and probabilistic sleep stager. In *Proceedings of the IEEE-INNS-ENNS International Joint Conference on, IJCNN 2000*, volume 3, pages 627–631, Como, Italy, July 2000.

[44] A. Flexer, G. Gruber, and G. Dorffner. Improvements on continuous unsupervised sleep staging. In *Proceedings of the 2002 12th IEEE Workshop on Neural Networks for Signal Processing*, pages 687–695, September 2002.

[45] C. J. C. Burges. A tutorial on support vector machines for pattern recognition. *Data Mining and Knowledge Discovery*, 2(2):121 – 167, 1998.

[46] D. Gorur, U. Halici, H. Aydin, G. Ongun, F. Ozgen, and K. Leblebicioglu. Sleep spindles detection using short time fourier transform and neural networks. pages 1631–1636, Honolulu, HI, May 2002.

[47] T. Shimada, T. Shiina, and Y. Saito. Detection of characteristic waves of sleep eeg by neural network analysis. *IEEE Transactions on Biomedical Engineering*, 47(3):369–379, March 2000.

[48] S. Kubicki, L. Hoeller, I. Berg, C. Pastelak-Price, and R. Dorow. Sleep eeg evaluation: A comparison of results obtained by visual scoring and automatic analysis with the oxford sleep stager. *Sleep*, 12(2):140–149, 1989.

[49] V. Pohl and E. Fahr. Neuro-fuzzy recognition of k-complexes in sleep eeg signals. pages 789–790, Montreal, Quebec, Canada, September 1997.

[50] D. Henry, D. Sauter, and O. Caspary. Comparison of detection methods: Application to k-complex detection in sleep eeg. volume 2, pages 1218–1219, Baltimore, MD, November 1994.

[51] B. Kemp. A proposal for computer-based sleep/wake analysis. *Journal of Sleep Research*, 2:179–185, 1993.

[52] Laird N.M. Dempster, A.P. and D.B. Rubin. Maximum likelihood from incomplete data via the em algorithm. *Journal Royal Statistical Society*, 39(1):1–38, 1977.

[53] D. Ogilvie, R. The process of falling asleep. *Sleep Medicine Reviews*, 5 (3):247–270, 2001.

[54] R. J. A. Laing. A sleep spindle detection algorithm. Master's thesis.

[55] S. Roberts and L. Tarassenko. Analysis of the human eeg using self-organising neural nets. *IEE Colloquium on Neurological Signal Processing*, pages 6/1–6/3, March 1992.

[56]

[57] J. Pardey, S. Roberts, and L. Tarassenko. A review of parametric modelling techniques for eeg analysis. *Medical Engineering Physics*, 18:2–11, 1996.

[58] S. J. Roberts, W. Penny, and I. Rezek. Temporal and spatial complexity measures for eeg-based brain-computer interfacing. *Medical and Biological Engineering and Computing*, 37(1):93–99, 1998.

Glossary

10/20 System	EEG electrode placement standard defined by International Federation of Societies for EEG and Clinical Neurophysiology
ACNN	All connecting neural network
AI	Artificial intelligence
ANN	Artificial neural network
AR	Autoregressive modelling
BPN	Backpropagation Neural Network
DFT	Discrete Fourier Transform
EEG	Electroencephalogram
Electrodes	Metal disks attached to the skin with adhesives and used to detect electric potential on the skin.
EOG	Electrooculogram
EMG	Electromyogram
FIS	Fuzzy inference system
FFT	Fast Fourier Transform
GUI	Graphical user interface
HMM	Hidden Markov Model
KCD	K complex detected
KCT	Time to last K complex
MBA	Mixed band activity
MLP	Multilayer perceptron
MT	Movement time
NREM	Non rapid eye movement
PA	Period analysis
PR	Pattern recognition
PWR	Relative power
REM	Rapid eye movement

SOFM	Self-organizing feature map
SSD	Sleep spindle detected
SST	Time to last sleep spindle
SVM	Support Vector Machine
SWD	Saw tooth wave detected
TDNN	Time delayed neural network
TFR	Time-frequency representation
VWD	Vertex wave detected

Appendix - Sleep Onset

This small section will be devoted to explaining the events in the sleep onset process, because it will found one criticism against the current sleep staging standard.

Sleep onset received similar attention in the 30's. It was widely recognized that the onset of sleep occurred in Stage II. One group of researchers, Loomis, Harvey, and Hobart, were particularly interested in this transition. They proposed in 1937 that there are three stages involved in this process, A - alpha or slow waves, B - low voltage with rolling eyes, and C - Spindles[53]. These three stages corresponded to awake, drowsy, and asleep.

The first significant change came in the 50's. Kleitman and Dement proposed that Loomis' Stage A and B can be combined into the Stage I used in the rest of sleep research. Similarly Loomis' Stage C fell into Stage II.

In 1961, another researcher, Roth produced a very detailed description of sleep onset. It listed the following,

- *Stage 1*: This is called disintegrated alpha, which is the Stage A in Loomis' definition. It corresponds to the Awake state.
- *Stage 2a*: This is the beginning of Loomis' Stage B. It is characterized by flattening EEG and a decrease in alpha blocking.
- *Stage 2b*: This is also part of Loomis' Stage B. It has 5 to 6 Hz frequency and 10 to 40 μV amplitude. It shows the beginning of alpha waves.
- *Stage 2c*: The last part of Loomis' Stage B, it has 3 to 4 Hz frequency and 50 to 80 μV amplitude. It demonstrates a mixture of alpha waves and vertex waves.
- *Stage 3*: This is Loomis' Stage C and it is accepted as clinical sleep. It contains sleep spindles, K-complexes, and vertex waves.

The most comprehensive segmentation of sleep onset is proposed in 1994 by Hori, Hayashi, and Morikawa. They defined 9 stages as follows,

- *Stage 1 - Alpha Wave Train*: Epoch composed of continuous alpha activity with minimum 20 μV.
- *Stage 2 - Alpha Wave Intermittent (A)*: Similar to above but more than 50%.
- *Stage 3 - Alpha Wave Intermittent (B)*: Similar to above but less than 50%.
- *Stage 4 - EEG Flattening*: Suppressed wave with less than 20 μV.
- *Stage 5 - Ripples*: Low voltage theta wave (20 to 50 μV) burst suppression.
- *Stage 6 - Vertex Wave (Solitary)*: Epoch contain one well-defined vertex wave.
- *Stage 7 - Vertex Wave (Train)*: At least two well-defined vertex wave.
- *Stage 8 - Vertex Wave (Wave)*: At least one well-defined vertex wave and one incomplete spindle.
- *Stage 9 - Spindle*: At least one well-defined spindle.

Hori's system is very well received because it provides a detailed index to describe sleep onset in a subject.

PSG Diagrams

The following diagrams are real screen shots of the data presented to the sleep scorers. The sleep scorer has more than the basic three channels of physiological data at their disposal. Each screen shot shows a 30-second epoch. In most instances, the 75 μV boundaries are shown around the better EEG channel. These diagrams demonstrate some of the basic R&K rules and show sample characteristic waves.

These diagrams are provided by the sleep lab at London Health Sciences Center. This contribution from all the technicians in particular Ms. Patricia A. Clements is sincerely appreciated.

Figure 1: NREM I with alpha, theta, vertex waves and slow rolling eye movement.

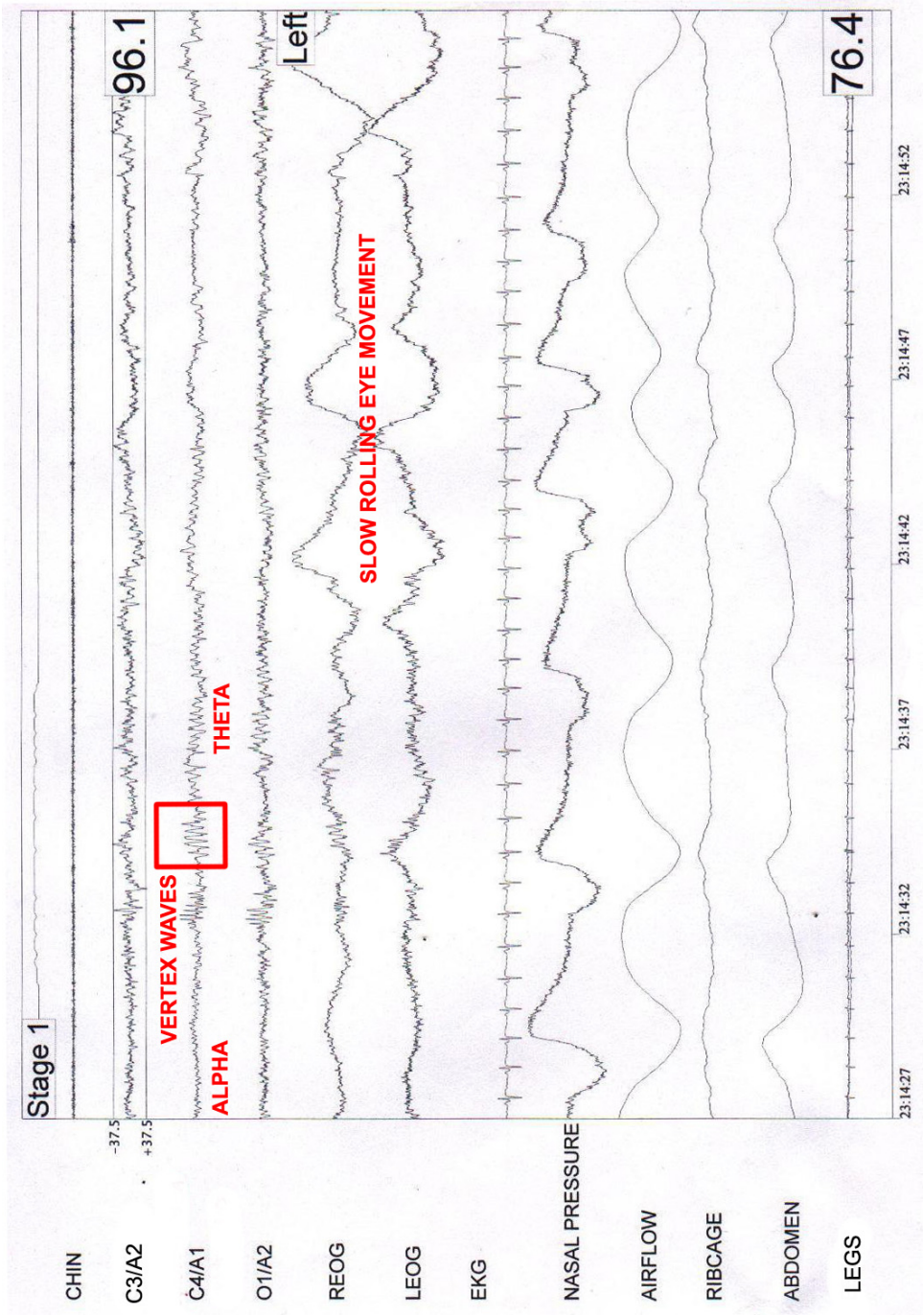


Figure 2: NREM II with K complex, sleep spindle, and background EEG.

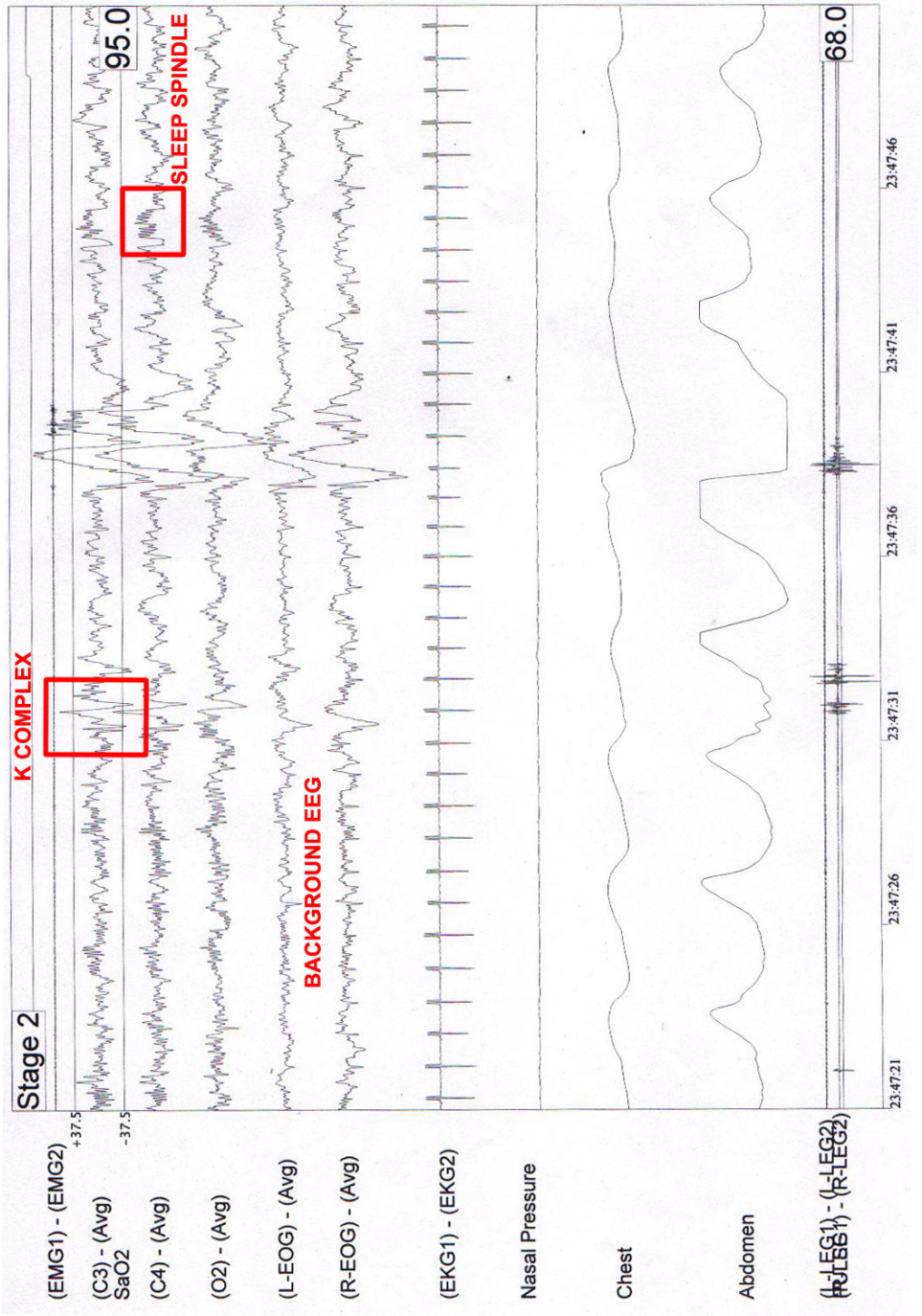


Figure 3: NREM II with elevated EMG.

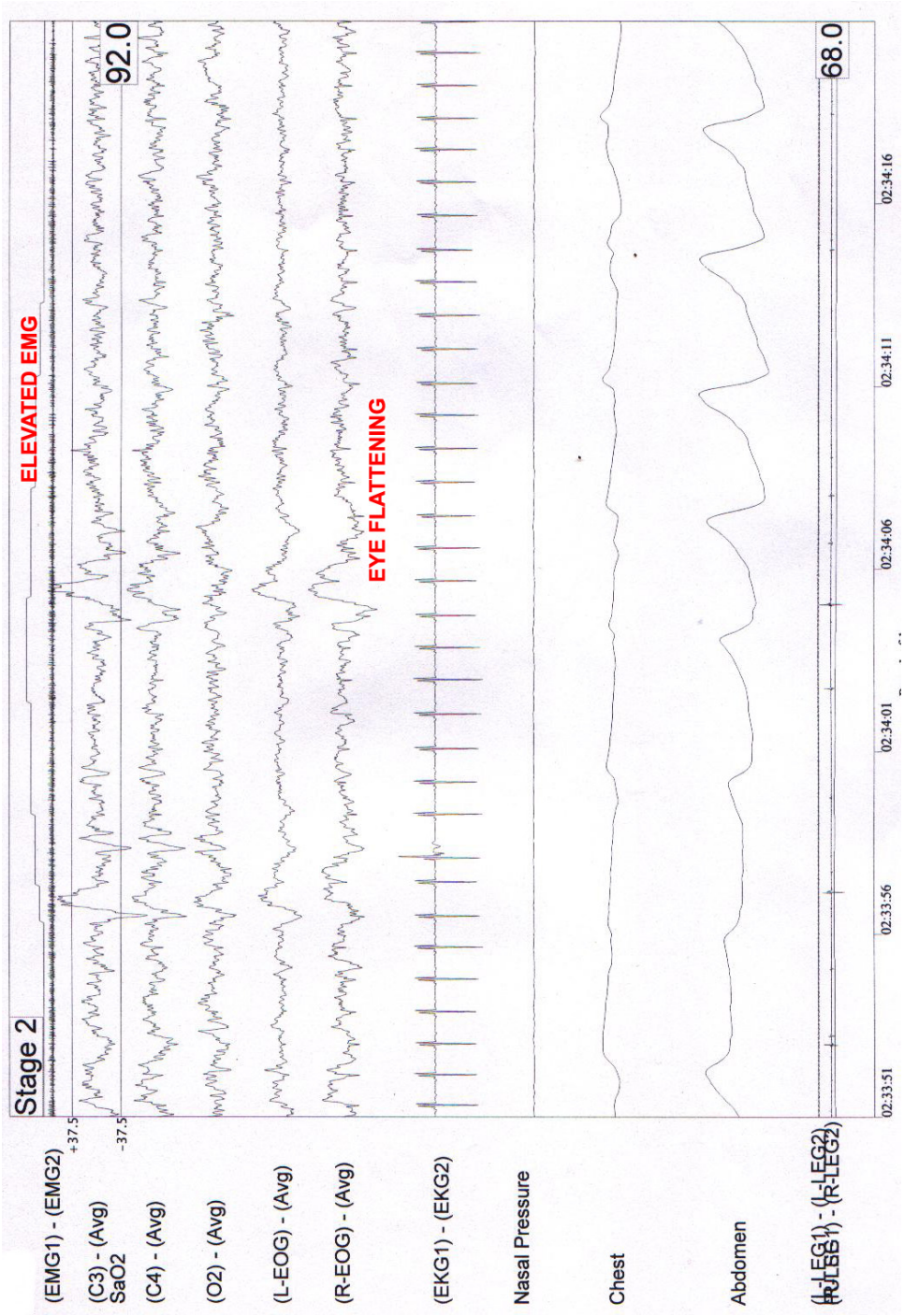


Figure 4: NREM IV with delta activity.

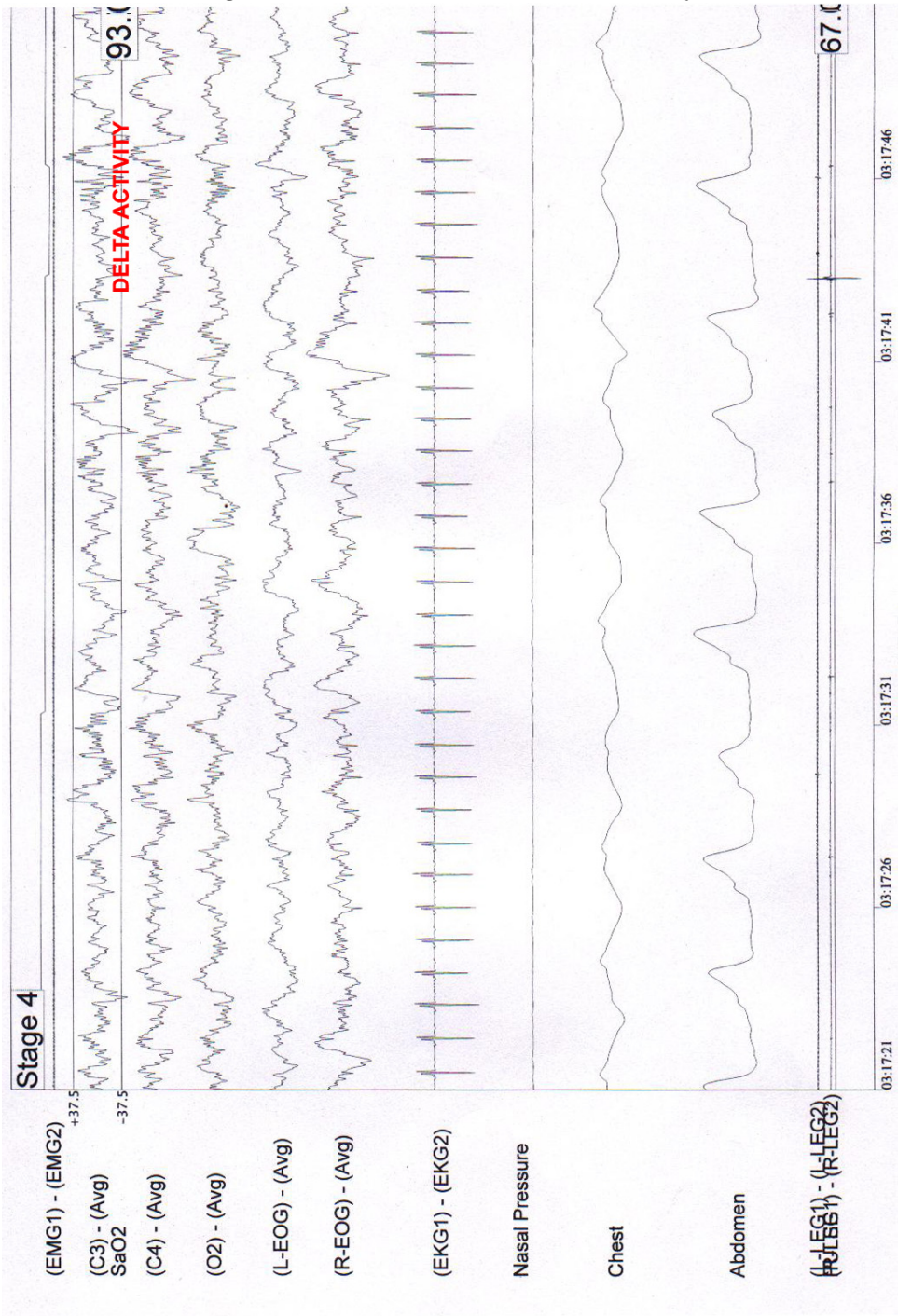


Figure 5: NREM IV with alpha intrusion.

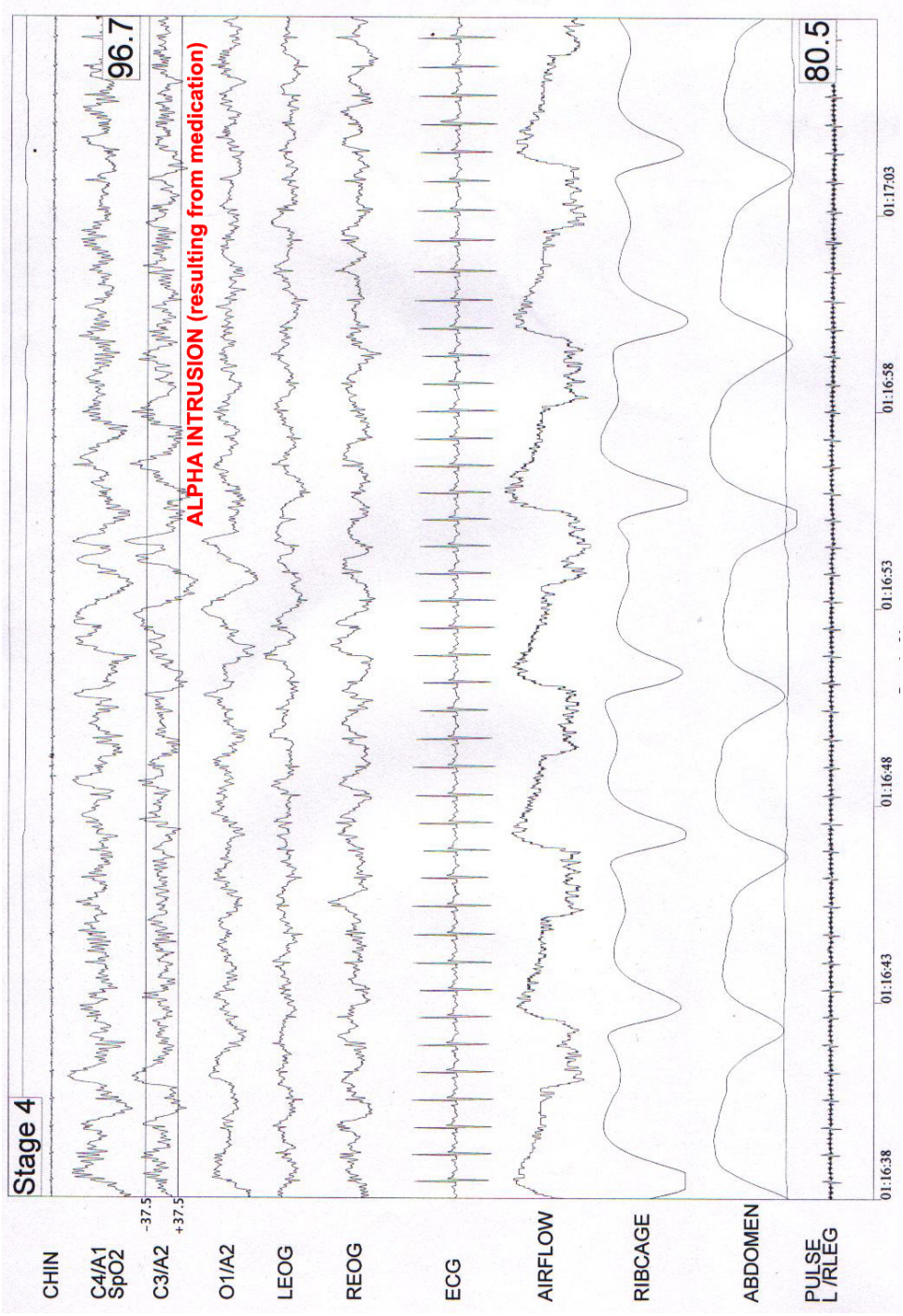


Figure 6: REM with tonic REM and flat EMG.

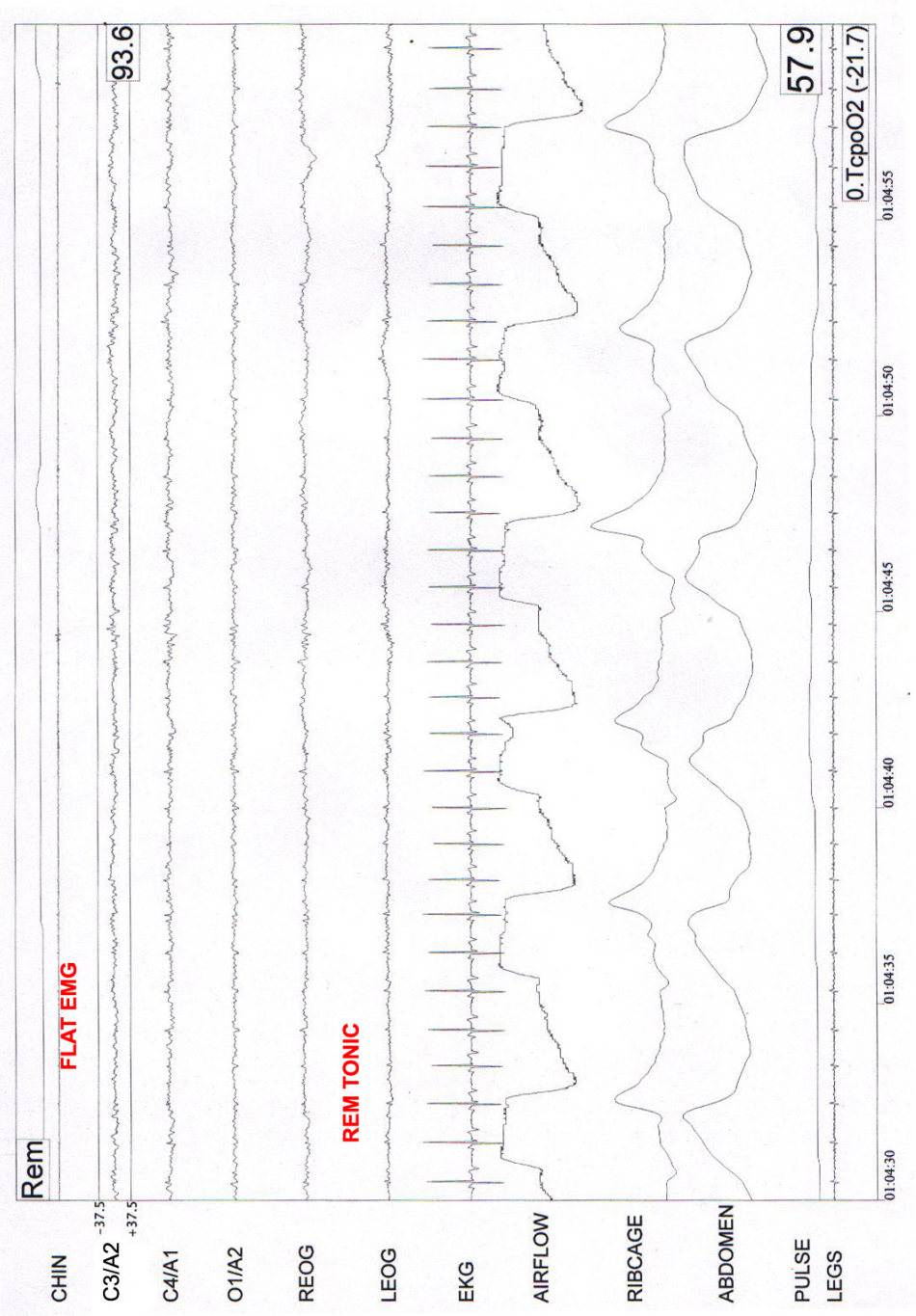
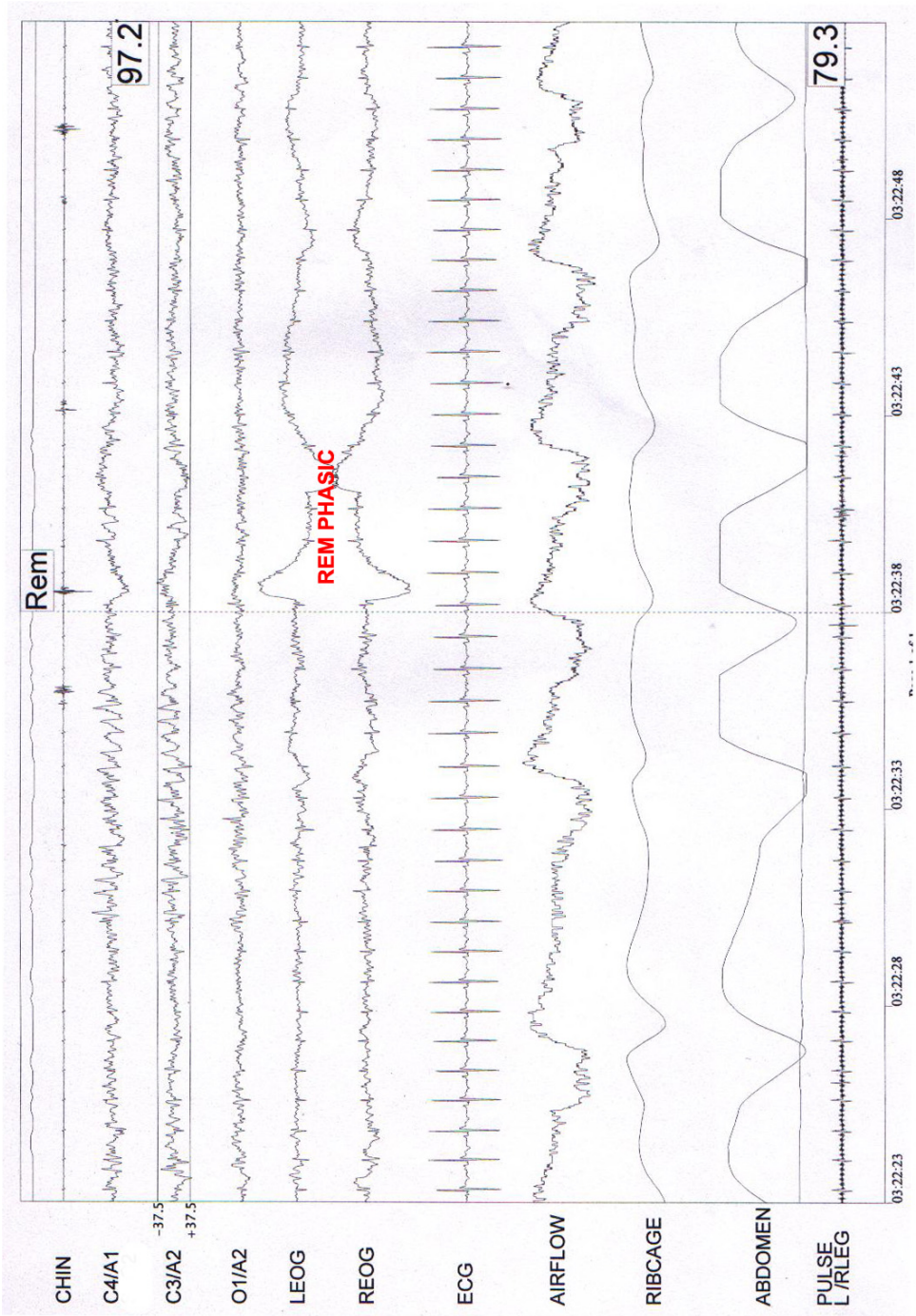


Figure 7: Beginning of REM with phasic REM.



Equations

Transforms

In this section, the definition and characteristics of spectrograms and scalograms are briefly introduced. This information is extracted from Laing's master thesis [54].

Spectrogram

Spectrograms are basic TFR, where signals are transformed into the frequency domain in segments and realigned against time. The segments of the signal are of equal length and they may overlap. Generally a windowing function $w(t)$ is used to isolate the segment. Each segment $s(t)$ is transformed by Short-Time Fourier Transform,

$$STFT_x(t,f) = \int_{-\infty}^{\infty} x(\tau)w^*(\tau - t)e^{-\imath 2\pi f\tau}d\tau. \tag{1}$$

Then the energy density spectrum is

$$SPEC(t,f) = |STFT_x(t,f)|^2 \tag{2}$$

and the result is called a spectrogram.

The spectrogram follows the Heisenberg Uncertainty Principle, meaning that it is impossible to have perfect resolution in both time and frequency. Spectrogram is time-dependent, meaning that the resolutions are dependent on the window size, in other words the length of signal segment. The wider the window, the better the frequency resolution and the worse the time resolution.

The center of the window function, t^*, and that of the bandpass filter, f^*, can be determined with

$$t^* = \frac{\int t|w(t)|^2 dt}{\int |w(t)|^2 dt} \tag{3}$$

and

$$f^* = \frac{\int f|W(f)|^2 df}{\int |W(f)|^2 df}. \tag{4}$$

These values may indicate the time that an event occurred or the frequency at which some activity was concentrated. The time resolution, Δt, and the frequency resolution, Δf, are

$$\Delta t = \left(\frac{\int t^2|w(t)|^2 dt}{\int |w(t)|^2 dt} \right)^{\frac{1}{2}} \tag{5}$$

and

$$\Delta f = \left(\frac{\int f^2|W(f)|^2 df}{\int |W(f)|^2 df} \right)^{\frac{1}{2}}. \tag{6}$$

Time events closer than Δt and frequencies closer than Δf cannot be differentiated. Note that once the time-window size is determined, the resolutions are the same across all windows used to compose the spectrogram.

Scalogram

Unlike the spectrogram, scalogram is based on continuous wavelet transform (CWT). CWT uses time-windows that are dependent on the frequency, where higher frequency resolution is available at lower frequencies. The CWT is defined as

$$CWT_x(t, s) = \frac{1}{\sqrt{s}} \int_{-\infty}^{\infty} x(\tau)\psi^* \left(\frac{\tau - t}{s} \right) d\tau, \tag{7}$$

where s is the scale and ψ is the mother- wavelet. The scale s is related to the frequency by

$$s = \frac{f_o}{f}, \tag{8}$$

where f_o is the frequency of the mother wavelet. Depending on the time and scale, the mother-wavelet is being compressed and shifted such that it

provides the appropriate time sampling step for a desired frequency. Similar to spectrogram's relation to STFT, scalogram is defined as

$$SCAL(t,s) = |CWT_x(t,s)|^2. \tag{9}$$

The relative bandwidth at time t and scale s can be derived from

$$\Delta f_r(t,s) = \frac{\Delta f_o}{s} = \frac{1}{s}\left(\frac{\int f^2|\Psi(f)|^2 df}{\int |\Psi(f)|^2 df}\right)^{\frac{1}{2}}, \tag{10}$$

where the Δf_o is the bandwidth of the mother-wavelet.

Spectral Features

The following equations are taken from [29]. Relative spectral power

$$S(\omega) = F(\omega) \times F^*(\omega)/S_{total} \tag{11}$$

Spectral entropy SEN

$$SEN = -\int S(\omega)\log S(\omega)/\log n \tag{12}$$

where $S(\omega)$ is normalized spectral power and n is the number of discrete frequencies.

Spectral edge (EDGE)

$$\int_{0.5}^{EDGE} S(\omega)d\omega = 0.9\int_{0.5}^{45} S(\omega)d\omega \tag{13}$$

First spectral moment $M1$

$$M1 = \int \omega \cdot S(\omega)d\omega \tag{14}$$

Entropy of amplitude ENA

$$ENA = -\sum_i P_i \times \log P_i/log(n) \tag{15}$$

where P_i is normalized amplitude distribution and n is the number of equidistant boxes.

Correlation dimension $D2$

$$D2 = \lim_{R \to 0} (\ln C(R, 20) / \ln R) \tag{16}$$

where

$$C(R, 20) = \sum_{i,j=1; i<j}^{N} \Theta \left(R - \|y_i - y_j\| \right) \tag{17}$$

Kolmogorof entropy

$$K2 = (1/5) * log_2[C(R, 19)/C(R, 20)] \tag{18}$$

Harmonic parameters [32] are defined below. Central frequency f_c

$$f_c = \frac{\int_{f_L}^{f_H} f S_{xx}(f) df}{\int_{f_L}^{f_H} S_{xx}(f) df} \tag{19}$$

Bandwidth frequency f_σ

$$f_\sigma = \sqrt{\frac{\int_{f_L}^{f_H} (f - f_c)^2 S_{xx}(f) df}{\int_{f_L}^{f_H} S_{xx}(f) df}} \tag{20}$$

Value at center frequency S_{f_σ}

$$S_{f_\sigma} = S_{xx}(f_c) \tag{21}$$

Temporal Features

Hjorth features [32] are defined as

$$Activity = \sigma_0^2 \tag{22}$$

$$Mobility = \sigma_1 / \sigma_0 \tag{23}$$

$$Complexity = \sqrt{(\sigma_2/\sigma_1)^2 - (\sigma_1/\sigma_0)^2}, \tag{24}$$

where σ_i is the variance of the ith derivative of the signal.

Summary of Past Automation Research

This section contains a table summarizing the automation research referenced in this document. The seven columns of the table are

- Research *Group* contains the names of main researchers, the years and the references of the publications.
- Source *Data* quantifies the data sets according to the data channels, the number of subjects, and the number of epochs.
- *Goals* identifies the main research objectives.
- *Features* lists the feature sets investigated in the research. Transformation used to derive these features are included in this column.
- *Classifiers* lists the classification methods investigated.
- *Context* analysis discusses whether context was considered in the investigation.
- *Performance* of the system is listed for comparison. Unless otherwise indicated, the default performance measure is accuracy. See Appendix 7.2 for performance measure definitions.

Table 2: Previous research review

Research Group	**Source Data**	**Goals**	**Features**	**Classifier**	**Context**	**Performance**
Akin, Akgul (98-00) [25][9]	EEG (3/NA)	Detect sleep spindle	Wavelet transform & higher order statistics & spectra & bispectrum	NA	NA	NA
Bankman (92) [39]	EEG (200 K complexes & 200 non K complexes)	Detect K complex	Amplitude & duration features	MLP	NA	Sensitivity 90% & specificity 91.9%
Fell (96) [29]	EEG (12/5(2:44 min epoch))	Study features for sleep staging	Spectral power & non-linear	Multi- & univariate analysis of variance	NA	79.2%
Flexer (00-02) [43][44]	2 Channels of EEG (C3 & C4) & EMG (9/8460)	New definition	Reflection coefficients, stochastic complexity of C3 & C4, measure of EMG	HMM	Inherent to HMM	$\sim 80\%$

Continued on next page

Table 2 – continued from previous page

Research Group	Source Data	Goals	Features	Classifier	Context	Performance
Gorur (02) [46]	EEG (1064 sleep spindles & 1600 non sleep spindle)	Detect sleep spindle	32 coefficients between 2 and 64 Hz	MLP & SVM	NA	88.7% & 95.4%
Grozinger (97) [23]	EEG (5/NA)	Identify REM stage	Relative power in activity bands	ANN	NA	$> 90\%$
Grozinger (01) [30]	EEG (13/1440)	Study features to identify REM stage	Spectral power & non-linear	ANN	NA	88% (spectral)
Henry (94) [50]	Synthesized K complexes	Compare decomposition by matched filter, in a set of orthonormal function, & by wavelet analysis	Amplitude & duration features	NA	NA	sensitivity $\sim 70\% - 97\%$ depending on noise levels
Hu, Knapp (91) [22]	EEG (NA/1101)	Sleep staging	Amplitude & duration features	FIS	NA	77%
Huupponen (01) [31]	MSLT (15/600)	Identify alpha activity	Spectral power ratio	FIS	NA	TP 85% FP 13%

Continued on next page

Table 2 – continued from previous page

Research Group	Source Data	Goals	Features	Classifier	Context	Performance
Kubicki (89) [48]	PSG (10/9360)	Evaluate Oxford Sleep Stager	NA	NA	NA	95.7%
McGrogan (01) [1]	EEG (9/8502)	Correlate new definition with R&K	Reflection coefficients	MLP	NA	72.2%
Oropesa (99) [28]	EEG (2/1690)	Sleep staging	Spectral features by wavelet transform	ANN	NA	77.6%
Pacheco, Vaz (98) [24]	EEG (8/2400)	Modified sleep staging	Elementary patterns, Background activity	MLP	Look at previous stage	90.5%
Pohl (95) [49]	EEG (106 K complexes)	Detect K complex	Amplitude & duration features	Neuro-fuzzy detector	NA	$> 50\%$
Rezek (99) [33]	EEG (1/960)	Study features for sleep staging	Complexity features	NA	NA	NA
Roberts, Tarassenko, Pardey (92-98) [15][55][56][57][34][58]	EEG (9/8100)	Redefine significant event	AR model or Kalman filter for parametrization	SOFM	NA	NA

Continued on next page

Table 2 – continued from previous page

Research Group	Source Data	Goals	Features	Classifier	Context	Performance
Schaltenbrand (93) [13]	EEG, EOG, & EMG (11/11906)	Sleep staging	Spectral features	MLP	NA	87.9%
Shimada (98) [41][47]	EEG (3/272 minutes)	Customizing MLP for sleep staging	Layer 1 - TFR, Layer 2 - Characteristic waves	Cascaded MLP	Look at up to 15 epochs in a MLP	82.0%
Van Hese (01) [32]	EEG (NA)	Sleep staging	Hjorth and spectral features	K-means clustering	NA	NA

Performance Measures

Based on the needs of a project, researchers apply different techniques to study the performance of their algorithm. This section defines the popular measures used in the sleep staging automation area.

The Confusion Matrix is a analysis utility to study classification errors in terms of its nature and frequency. The columns represent reference classification and the rows represent trial classification. In the case of sleep staging automation, the reference classification comes from human scorers, and the trial classification comes from the algorithm in research. Perfect agreement requires that only the diagonal be inhabited with non-zero elements. Table 3 shows a reduced confusion matrix, where all the NREM stages are combined.

Table 3: Reduced confusion matrix used in sleep staging automation.

Trial \ Reference	Wake	NREM	REM
Wake	23	3	3
NREM	2	15	4
REM	6	4	76

The confusion matrix contain all the performance information, but it is difficult to identify the key details. Therefore, performance measures are calculated to summarize the confusion matrix.

Overall Accuracy measures the percentage of classifications that are correct.

$$Overall\ Accuracy = \frac{\sum diagonal\ elements}{\sum all\ elements} = 83.8\% \tag{25}$$

True Positive, False Positive, True Negative, and False Negative combines the information in a confusion table such that the analysis is focused on one classification. Table 4 shows these values with respect to Wake.

Table 4: Performance for Wake.

Trial \ Reference	Wake	Asleep
Wake	TP=23	FP=6
Asleep	FN=8	TN=99

Sensitivity and Specificity are a pair of measures commonly presented together. The equation below defines these measures with respect to Wake.

$$Sensitivity_{wake} = \frac{TP}{TP + FN} = 74.2\% \tag{26}$$

$$Specificity_{wake} = \frac{TN}{TN + FP} = 94.3\% \tag{27}$$

Negative and Positive Predictive Values are an alternative pair to sensitivity and specificity.

$$Negative\,Predictive\,Value\,(NPV) = \frac{TN}{FN + TN} = 92.5\% \tag{28}$$

$$Positive\,Predictive\,Value\,(PPV) = \frac{TP}{TP + FP} = 79.3\% \tag{29}$$

Receiver Operating Characteristic Curve (ROC) is a common visual performance measure. The curve has specificity on x-axis and sensitivity as the y-axis. Figure 8 shows 3 sample ROC curves. Curve A would be considered better than curve B, which is in turn better than curve C. Curve C suggests that the classifier is performing as well as random guessing when the two classes are equally likely; therefore, anything curving more towards the origin would be useless as a classifier. The best ROC has more area under it or has a shortest distance from $(1, 1)$. Often in parameterizations, the ROC is plotted with respect to the parameter.

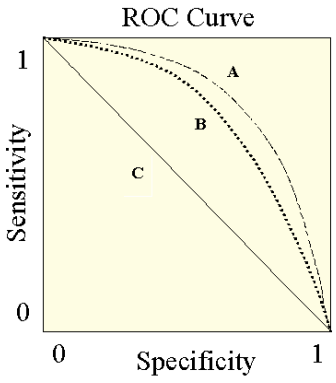


Figure 8: Sample ROC Curves