

# Reducing the Computational Demands of Medical Monitoring Classifiers by Examining Less Data

by

Eugene Inghaw Shih

Bachelor of Science in Computer Engineering  
University of Washington, Seattle, 1998

Master of Science in Electrical Engineering and Computer Science  
Massachusetts Institute of Technology, 2001

Submitted to the Department of Electrical Engineering and Computer Science in  
Partial Fulfillment of the Requirements for the Degree of

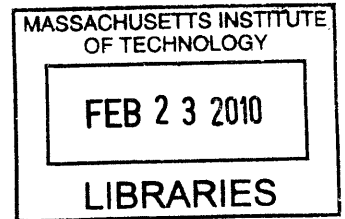
Doctor of Philosophy in Electrical Engineering and Computer Science  
at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY


February 2010

© 2010 Massachusetts Institute of Technology. All rights reserved.


**ARCHIVES**



Signature of Author \_\_\_\_\_

  
Eugene I. Shih  
Department of Electrical Engineering and Computer Science  
February 1, 2010

Certified By \_\_\_\_\_

  
John V. Guttag  
Professor of Electrical Engineering and Computer Science  
Thesis Supervisor

Accepted By \_\_\_\_\_

  
Terry P. Orlando  
Chairman, Department Committee on Graduate Students



# **Reducing the Computational Demands of Medical Monitoring Classifiers by Examining Less Data**

by

Eugene Inghaw Shih

Submitted to the Department of Electrical Engineering and Computer Science  
on February 1, 2010, in Partial Fulfillment of the  
Requirements for the Degree of  
Doctor of Philosophy in Electrical Engineering and Computer Science

## **Abstract**

Instrumenting patients with small, wearable sensors will enable physicians to continuously monitor patients outside the hospital. These devices can be used for real-time classification of the data they collect. For practical purposes, such devices must be comfortable and thus be powered by small batteries. Since classification algorithms often perform energy-intensive signal analysis, power management techniques are needed to achieve reasonable battery lifetimes.

In this thesis, we describe software-based methods that reduce the computation, and thus, energy consumption of real-time medical monitoring algorithms by examining less data. Though discarding data can degrade classification performance, we show that the degradation can be small. We describe and evaluate data reduction methods based on duty cycling, sensor selection, and combinations of the two.

Random duty cycling was applied to an online algorithm that performs risk assessment of patients with a recent acute coronary syndrome (ACS). We modified an existing algorithm that estimates the risk of cardiovascular death following ACS. By randomly discarding roughly 40% of the data, we reduced energy consumption by 40%. The percentage of patients who had a change in their risk classification was 3%.

A sensor selection method was used to modify an existing machine learning based algorithm for constructing multi-channel, patient-specific, delay-sensitive seizure onset detectors. Using this method, we automatically generated detectors that used roughly 60% fewer channels than the original detector. The reduced channel detectors missed seven seizures out of 143 total seizures while the original detector missed four. The median detection latency increased slightly from 6.0 to 7.0 seconds, while the average false alarms per hour increased from 0.07 to 0.11.

Finally, we investigated the impact of approaches that combine duty cycling with sensor selection on the energy consumption and detection performance of the seizure onset detection algorithm. In one approach, where we combined two reduced channel detectors to form a single detector, we reduced energy consumption by an additional 20% over the reduced channel detectors.

Thesis Supervisor: John V. Guttag  
Title: Professor



## Acknowledgements

The contents of this thesis are the result of many years of hard work. However, the manuscript you see before you would not have been possible without the support and assistance of people from both my professional and personal life.

First, I would like to thank my thesis advisor and mentor, Professor John Guttag, for guiding me through the thesis writing process. I will always remember the invaluable advice and wisdom he offered.

I'm also grateful to the other members of my thesis committee, Professor Sam Madden and Professor Anantha Chandrakasan, for providing honest feedback that helped to improve the final manuscript.

In addition, I would like to acknowledge my undergraduate advisor at the University of Washington, Professor Gaetano Borriello. Without his encouragement, I may have never pursued graduate studies. Moreover, his unwavering belief in me at various times helped me overcome some of the more difficult periods of my graduate career.

I am also grateful to a number of fellow graduate students and research staff who helped me during my graduate school years: Magda Balazinska, Ari Benbasat, Manish Bhardwaj, Vladimir Bychkovsky, Ben Calhoun, SeongHwan Cho, Dorothy Curtis, Jakob Eriksson, Gartheeban Ganeshapilai, Lew Girod, Michel Goraczko, Wendi Rabiner Heintzelman, Bret Hull, Nathan Ickes, Kyle Jamieson, Al Kharbouch, Fred Lee, Rex Min, Allen Miu, Asfandyar Qureshi, Ali Shoeb, Anima Singh, Amit Sinha, Zee-shan Syed, Godfrey Tan, Naveen Verma, Alice Wang, and Jenna Wiens.

Of these individuals, I would like to extend a special thanks to Ali Shoeb, Manish Bhardwaj, and Vladimir Bychkovsky. Each of them helped me both technically and personally throughout the process of writing my thesis. I will never forget the late nights in the lab we spent working on a problem set or on our own research projects—all while commiserating about our rough lives as graduate students. I am sure in a few years from now, I will look back upon those days with fondness.

This thesis also would not have been possible without the dedication and kindness of Donna Kaufman, Sheila Marian, and Margaret Flaherty. Their assistance in coordinating room reservations and scheduling meeting times was always helpful. I will also miss Sheila's wonderful homemade butterscotch scones.

I'd also like to thank Jennifer Schmitt for being a strong supporter of me during some of the lowest times. I will always cherish her exuberance and loyalty.

I also have to thank two close friends of mine: Yee Wong and Adam Halpern. I thank Yee for entertaining me with stories about her life and Adam for injecting humor into my life.

I would be extremely negligent if I did not thank the close friends who I met through Boston Progress Arts Collective: Pen Khek Chear, Hsindy Chen, Victor Chien, Ash Hsie, Theresa Hwang, Sophea Khem, Dave Kong, Giles Li, and Sopheak Tek. You all showed me that even I could be an artist and that there is much more to life than 1's and 0's. Thanks for all your love—working on my thesis would not have been possible without taking a break for Supper Club and Poker Nights.

I also have to thank my future stepdaughter, Tae Cheung Hong, for reminding me that everyday is a day to celebrate. In addition, I wish to thank my future in-laws, Gim Hom and Mildred Cheung, for treating me so kindly. I have felt like a part of their family since day one.

I am also deeply indebted to my parents, Tzu-Jung Sun and Chien-Hsiung Shih, and my sister, Grace, for their support and sacrifice. Thank you all for tolerating my absence during my studies. San Diego may be physically distant from Boston, but you will always be close to my heart.

Finally, I would like to thank my fiancée, Delia Cheung Hom. Thank you for always filling up my half-empty outlook on life. Thank you for supporting me through this long and seemingly, endless process and for giving me the extra spark that I needed to finish this journey.

Eugene Shih  
Brookline, Massachusetts  
January 29, 2010

# Contents

<b>Contents</b>	<b>7</b>
<b>List of Figures</b>	<b>10</b>
<b>List of Tables</b>	<b>13</b>
<b>1 Introduction</b>	<b>15</b>
1.1 Applications . . . . .	16
1.1.1 Seizure Onset Detection . . . . .	16
1.1.2 Real-time Risk Stratification of Post-ACS Patients . . . . .	17
1.2 State-of-the-Art Ambulatory Monitors . . . . .	17
1.3 Reducing Energy Consumption . . . . .	18
1.3.1 Duty Cycling . . . . .	19
1.3.2 Feature or Sensor Selection . . . . .	19
1.3.3 Combining Channel Selection with Duty Cycling . . . . .	20
1.4 Thesis Contributions . . . . .	20
1.5 Thesis Outline . . . . .	21
<b>2 Background</b>	<b>23</b>
2.1 Seizure Onset Detection . . . . .	23
2.2 Risk Stratification of Post-ACS Patients . . . . .	24
2.2.1 Risk Stratification Using ECG Recordings . . . . .	25
2.3 Online Ambulatory Medical Monitoring Algorithms . . . . .	25
2.3.1 Data Acquisition . . . . .	25
2.3.2 Windowing . . . . .	25
2.3.3 Segmentation and Pre-processing . . . . .	26
2.3.4 Feature Extraction . . . . .	27
2.3.5 Analysis and Classification . . . . .	27
2.3.6 Aggregate Analysis . . . . .	28
2.4 Architectures of Wearable Ambulatory Monitoring Systems . . . . .	28
2.5 Overview of Energy Saving Methods . . . . .	29
2.6 Related Work . . . . .	30
2.6.1 Relation to Compressive Sampling . . . . .	31
<b>3 Random Duty Cycling</b>	<b>33</b>

3.1	The Morphological Variability Algorithm . . . . .	34
3.1.1	Beat Segmentation . . . . .	34
3.1.2	ECG Noise Filtering . . . . .	34
3.1.3	Morphologic Difference . . . . .	36
3.1.4	Spectral Energy of Morphologic Differences . . . . .	37
3.2	Online Monitoring of MV-DF . . . . .	38
3.2.1	The Online MV-DF Algorithm . . . . .	38
3.2.2	Discussion . . . . .	39
3.3	Comparing Approaches . . . . .	39
3.3.1	Data . . . . .	39
3.3.2	Evaluation Metrics . . . . .	40
3.4	Results . . . . .	40
3.4.1	Evaluation of Risk Assessment . . . . .	42
3.5	Effect of Duty Cycling . . . . .	44
3.6	Evaluation on TIMI MERLIN Patients . . . . .	47
3.6.1	Effect of Duty Cycling on TIMI MERLIN patients . . . . .	49
3.7	Discussion . . . . .	51
<b>4</b>	<b>Automatic Sensor Selection</b> . . . . .	<b>53</b>
4.1	Notation . . . . .	54
4.2	A Seizure Onset Detection Algorithm . . . . .	55
4.2.1	The Patient-Specific SVM-Based Detector . . . . .	56
4.3	Evaluation Metrics . . . . .	56
4.4	Hardware Prototype of a Wearable Seizure Monitor . . . . .	57
4.4.1	Energy Measurements . . . . .	59
4.4.2	Alternative Architectures . . . . .	62
4.5	Constructing a Reduced Channel Detector . . . . .	65
4.5.1	Feature Selection Algorithms . . . . .	65
4.5.2	The Wrapper Approach . . . . .	66
4.5.3	Example . . . . .	68
4.6	Evaluation Methodology . . . . .	68
4.6.1	Data Description . . . . .	69
4.6.2	Testing Procedure . . . . .	69
4.7	Results for All Patients . . . . .	72
4.7.1	Number of Channels . . . . .	72
4.7.2	Seizure Onset Detection Performance . . . . .	73
4.7.3	Detection Latency . . . . .	74
4.7.4	False Events per Hour . . . . .	74
4.7.5	Energy Consumption . . . . .	75
4.7.6	Discussion . . . . .	76
4.8	Case Studies . . . . .	77
4.9	Patient 8 . . . . .	77
4.9.1	Detection Performance and False Alarm Rate . . . . .	79
4.9.2	Latency . . . . .	82
4.9.3	Sensitivity to Channels Selected . . . . .	82
4.10	Patient 16 . . . . .	83



4.10.1	Detection Performance, False Alarm Rate, and Latency . . . . .	84
4.11	Application of Channel Selection Approach . . . . .	86
4.12	Other Channel Selection Approaches . . . . .	88
4.12.1	Fisher Criterion . . . . .	88
4.12.2	Support Vector Machine Recursive Feature Elimination . . . . .	89
4.12.3	Comparing Different Channel Selection Approaches . . . . .	90
4.13	Summary . . . . .	91
<b>5</b>	<b>Combining Duty Cycling and Sensor Selection</b>	<b>93</b>
5.1	Constructing Screening Detectors . . . . .	93
5.2	Evaluation . . . . .	95
5.2.1	Number of Channels . . . . .	96
5.2.2	Seizure Onset Detection Performance . . . . .	96
5.2.3	False Events Per Hour . . . . .	98
5.2.4	Detection Latency . . . . .	98
5.2.5	Impact on Energy and Lifetime . . . . .	98
5.3	Impact of Duty Cycling on Reduced Channel Detectors . . . . .	100
5.4	Evaluation . . . . .	102
5.5	Case Studies . . . . .	103
5.5.1	Patient 15 . . . . .	103
5.5.2	Patient 3 . . . . .	105
5.5.3	Summary . . . . .	107
5.5.4	Screening Approach . . . . .	107
5.5.5	Duty Cycling . . . . .	108
<b>6</b>	<b>Summary and Conclusions</b>	<b>109</b>
6.1	Data Reduction Through Random Duty Cycling . . . . .	109
6.2	Data Reduction Through Sensor Selection . . . . .	110
6.3	Data Reduction Through Duty Cycling and Sensor Selection . . . . .	110
6.4	Future Work . . . . .	111
	<b>Bibliography</b>	<b>113</b>

## List of Figures

2.1	Abstract structure of a prototypical ambulatory medical monitoring algorithm. . . . .	26
2.2	Two examples of window of samples . . . . .	26
2.3	Example classifier architectures . . . . .	28
2.4	Architectures for ambulatory medical monitoring systems. . . . .	29
3.1	A diagram of a QRS complex. The letters Q, R, and S marks deflection points in the signal. . . . .	35
3.2	The online MV-DF and original MV-DF distributions . . . . .	41
3.3	Online MV-DF error histogram . . . . .	42
3.4	Patients classified as <i>low risk</i> by the online MV-DF algorithm that were marked as <i>high risk</i> by the original MV-DF algorithm . . . . .	44
3.5	Patients classified as <i>high risk</i> by the online MV-DF algorithm that were marked as <i>low risk</i> by the original MV-DF algorithm . . . . .	45
3.6	The root mean squared difference (RMSD) of the duty cycling algorithm decreases exponentially with increasing $p$ . At $p = 0.6$ , the RMSE is 1.3. . . . .	46
3.7	The fraction of patients that are reclassified compared to the non-duty cycled algorithm as a function of $p$ . At $p = 0.5$ , only 22 patients have been reclassified. . . . .	47
3.8	Classification error of the duty cycled algorithms using CVD as the ground truth as a function of $p$ . . . . .	48
3.9	Online MV-DF error histogram for TIMI MERLIN cohort . . . . .	48
3.10	The root mean squared difference between various $p$ -duty cycled algorithms and the non-duty cycled algorithm as a function of $p$ for the TIMI MERLIN cohort. . . . .	49
3.11	The fraction of TIMI MERLIN patients that are reclassified compared to the non-duty cycled algorithm as a function of $p$ . . . . .	50
3.12	Classification error on the TIMI MERLIN cohort when comparing the risk assignment of the $p$ -duty cycled algorithms to patient death outcome as a function of $p$ . . . . .	50
4.1	The 10/20 electrode montage . . . . .	55
4.2	Schematic of single channel EEG analog signal acquisition circuit. . . . .	58
4.3	Prototype single channel EEG acquisition system . . . . .	58

4.4	Alpha waves captured by prototype EEG system . . . . .	59
4.5	Power consumption of CPU during radio transmission . . . . .	60
4.6	Power consumption of CC2500 radio during transmission . . . . .	61
4.7	Energy consumption per channel as a function of number of channels for various portable monitoring systems . . . . .	64
4.8	Backward elimination algorithm . . . . .	67
4.9	The average number of channels for all patients was 6.4. . . . .	72
4.10	Per-patient seizure onset detection latencies of the two types of detectors .	74
4.11	The per-patient median latency for seizure onsets detected by both detectors. For most patients, the median latency increased. . . . .	75
4.12	The average false events per hour increased slightly from 0.07 to 0.11. .	76
4.13	The average energy savings achieved by the SUBSET detection algorithm was 60%. . . . .	77
4.14	Lifetime of various types of batteries as a function of the number of channels. . . . .	78
4.15	The physical channels selected for patient 8 when $T_1$ was input as training data. . . . .	79
4.16	Two seizures from patient 8 . . . . .	80
4.17	Number of false positive windows and false events achieved on Test Set 1 for patient 8 as a function of the number of channels . . . . .	81
4.18	False event declared by SUBSET detector on Test Set 1 . . . . .	82
4.19	False positives and false events achieved on Training Set 1 for patient 8 as a function of the number of channels . . . . .	83
4.20	Average training and test detection latencies as a function of the number of channels . . . . .	84
4.21	Two seizures from patient 16 . . . . .	85
4.22	For patient 16, on Training Set 1, the number of false positives and false events decrease as a function of the number of channels . . . . .	87
4.23	A segment during a non-seizure record. A seizure is declared at 3566 seconds into the recording. . . . .	88
4.24	Average training and test detection latencies as a function of the number of channels . . . . .	89
5.1	A two-level detector. . . . .	94
5.2	The average number of channels used by the subset detectors and screening detectors for all patients where screening could be applied . . . . .	97
5.3	The average per-patient false events per hour decreased slightly from 1.2 to 1.1 when we added a screening detector. . . . .	98
5.4	Per-patient seizure onset detection latencies of the SUBSET and combined detectors . . . . .	99
5.5	The per-patient median latency for seizure onsets detected by both detectors. For Patient 6, we have no latency to report since the combined detector did not detect any seizures. . . . .	99
5.6	The energy consumed by the combined detectors was 69% less than the energy consumed by the original 18-channel detectors. In contrast, the subset detectors used only 50% less energy. . . . .	100

5.7	Adaptive duty cycling algorithm . . . . .	101
5.8	The energy consumed as a percentage of the energy consumed by the non-duty cycled detector as a function of $p$ , the duty cycling parameter. . . . .	102
5.9	Average change in detection rate as a function of $p$ . . . . .	103
5.10	The average change in latency as a function of $p$ . . . . .	104
5.11	The average change in false alarm rate as a function of $p$ . Since any window that is skipped is considered a negative, lower false alarm rates can be achieved with lower values of $p$ . . . . .	104
5.12	The average energy consumed as a percentage of the energy consumed by the non-duty cycled reduced channel detector and the detection rate plotted as a function of $p$ for patient 15. As expected, increasing $p$ improves the detection rate and also increases the amount of energy consumed. . . . .	105
5.13	Latency and false alarm rate per hour plotted versus $p$ for patient 15. . . . .	106
5.14	The average energy consumed as a percentage of the energy consumed by the non-duty cycled reduced channel detector and the detection rate plotted as a function of $p$ for patient 3. . . . .	106
5.15	Latency and false alarm rate per hour plotted versus $p$ for patient 12. . . . .	107

# List of Tables

1.1 Energy consumption of the different components in our head-mounted prototype seizure monitor . . . . .	19
3.1 Patients with the largest absolute change in the MV-DF measure in ranked order . . . . .	43
4.1 Measured values for the parameters used in our energy model. The supply voltage for all the components was 3.0 V. The dominant source of power consumption is the radio during transmit and transfer modes. . . . .	62
4.2 Power consumption of an optimized wireless EEG acquisition chip [57]. . . . .	63
4.3 The detector with the highest true negative rate is chosen from all the candidate 17-channel detectors. . . . .	69
4.4 The list of the best $k$ -channel detectors, $k = 1, 2, \dots, 17$ and the optimal SUBSET detector . . . . .	70
4.5 Recording duration and number of seizures for each patient . . . . .	71
4.6 Detection performance of detectors compared. . . . .	73
4.7 Common battery specifications . . . . .	76
4.8 Channels selected for patient 8 using the channel selection algorithm . . . . .	78
4.9 Test performance of constructed detectors for patient 8 . . . . .	79
4.10 Performance of detectors constructed using the different channel subsets with different test data. All seizure onsets were detected without significant change in the detection delay and number of false events. . . . .	86
4.11 Average test performance of constructed detectors for patient 16. . . . .	86
4.12 Performance of other channel selection approaches compared to the wrapper approach . . . . .	91
5.1 Detection performance of the combined, ALLCHANNEL, and SUBSET detectors on a subset of patients. . . . .	97



---

# Introduction

Instrumenting patients with small, wearable sensors will enable physicians to continuously monitor patients outside clinical environments while they go about their daily routines. By analyzing the physiological signals that are collected, these portable devices can be programmed to track or detect interesting changes or anomalies in a patient's medical condition. If the detected event is potentially harmful, the device could be further used to alert a caregiver or to initiate a therapy to relieve symptoms or prevent injury.

To enable continuous monitoring applications, designers of portable medical monitoring devices must address the issues of patient comfort and convenience. Since ambulatory devices must be worn or implanted, small batteries with limited energy capacities must power them. Unfortunately, biomedical signal processing algorithms tend to be computationally and energy intensive and will quickly drain these batteries. Thus, careful power management is needed to extend battery lifetime.

In this thesis, we describe software-based methods that reduce the energy consumption of real-time ambulatory medical monitoring algorithms. In many cases, our methods reduce energy consumption by collecting or analyzing less data without significantly degrading performance. We evaluate these methods in the context of two medical monitoring applications: seizure onset detection and risk stratification of patients with a recent acute coronary syndrome (ACS).

To detect seizures, data from 16 to 21 electroencephalographic (EEG) channels are first collected. The data is then processed and analyzed for seizure activity. To reduce energy consumption, we show that a subset of the input signals can be used instead. To find this subset, we use a sensor selection algorithm for constructing patient-specific seizure onset detectors. When we applied this approach to 16 patients with epilepsy, the resulting detectors identified 136 out of 143 seizures using only 36% of the data and 60% of the energy used by the original algorithm—only three fewer seizures than we would have detected had we used all the data. In addition, the number of false detections increased only slightly, from 0.07 to 0.11 false declarations per hour, while the median detection delay increased from 6.0 to 7.0 seconds.

In the post-ACS risk stratification problem, an algorithm is used to distinguish between patients who have a high or low risk of cardiovascular death using electro-

cardiogram (ECG) data. To reduce the energy consumed by this algorithm, we used random duty cycling method to avoid examining 40% of the data, and thus, reduced energy consumption by about 40%. Using this duty cycling approach to analyze the input data, only 22 out of the 753 patients were reclassified relative to the non-duty cycled algorithm.

In an effort to further increase battery lifetime of the seizure detection algorithm, we also explored the use of two hybrid energy saving strategies—methods that combine duty cycling with sensor selection. The first method reduces energy consumption by gating the operation of a reduced channel detector with a *screening detector*, an algorithm that approximates the behavior of the base detector but with even fewer channels. This approach resulted in an additional 20% reduction in energy consumption with a marginal degradation in detection performance. In the second approach, we adaptively duty cycle the reduced channel detector on a window-by-window basis. This approach reduced energy consumption of the reduced channel detector by 25% while degrading detection performance by only 2%.

In the rest of this chapter, we describe the two example ambulatory applications that we will focus on. Next, we give an overview of the methods that we use to decrease energy consumption. After surveying the state-of-the-art in medical monitoring systems, we provide an overview of the rest of the thesis.

## 1.1 APPLICATIONS

The medical applications we will consider analyze physiological data to provide a quantitative medical assessment of a patient or to track temporal changes in a patient’s physiological state. In this thesis, we focus on two specific medical monitoring applications: seizure onset detection and real-time risk assessment of patients who recently suffered an acute coronary syndrome (ACS).

### 1.1.1 Seizure Onset Detection

Epilepsy is a neurological disorder characterized by recurring unprovoked seizures. Seizures typically cause no direct harm to a patient with epilepsy. Instead, patients often become injured due to the symptoms of seizures, *e.g.*, involuntary convulsions. Despite advances in the management of epilepsy, more than 20% of individuals will continue to experience seizures. A portable seizure monitor could benefit these individuals by alerting caregivers of an impending seizure or by automatically initiating a therapeutic intervention that may reduce seizure intensity or duration [58]. For these monitors to be useful however, such a device must be able to reliably detect a seizure with low latency. In addition, the number of false detections must be kept low to avoid alarming caregivers and the patient unnecessarily.

To detect seizures, most state-of-the-art algorithms process multiple channels of electroencephalographic (EEG) data. Shoeb *et al.* have designed a patient-specific seizure onset detector that can operate in real time [44]. In this detector, EEG data is sampled from multiple EEG channels. The samples from each channel are grouped to form windows where the beginning of each window overlaps with the previous window by one second. Spectral features are extracted from each window and then aggregated to form a single feature vector. The feature vectors are then labeled to



indicate whether a window contains seizure activity. If persisting seizure activity is detected, a seizure is subsequently declared.

### 1.1.2 Real-time Risk Stratification of Post-ACS Patients

Acute coronary syndrome (ACS) is used to describe the clinical signs and symptoms of sudden myocardial ischemia—a sudden reduction in blood flow to the heart. ACS is often used to describe patients who present with either unstable angina (UA) or acute myocardial infarction (MI).

One study has shown that people who have been diagnosed with ACS have a higher risk for future adverse events, such as myocardial infarction and sudden cardiac death [2]. However, the risk levels for adverse cardiac events can vary among patients diagnosed with ACS. Thus, accurate risk assessment is important, not only to identify the high risk patients, but also to determine the appropriate treatments to administer to each patient.

One class of risk stratification techniques involves using long-term electrocardiograph (ECG) recordings [30, 47, 51]. Using long-term ECG for risk stratification can be beneficial since these recordings are routinely obtained during hospital stays or by portable Holter monitors. Moreover, ECG data can provide a wealth of diagnostic information about cardiovascular function.

In this thesis, we will focus on one ECG-based risk stratification algorithm: the Morphological Variability (MV) algorithm [51]. The MV algorithm performs risk stratification by quantifying the variation of morphologies in the heartbeats captured in an ECG signal. This algorithm, like all ECG-based risk stratification methods, operates in batch mode where analysis of the ECG signal is performed after sufficient data—typically 24 hours worth—has already been collected.

Instead of processing the signal offline, we propose to perform *online analysis* of the ECG signal. From a medical perspective, processing ECG in an online fashion could provide fine-grained information about how cardiovascular health varies throughout a day. While the benefits of online risk assessment using MV is not known, we believe that the MV value and the intermediate values generated by the MV algorithm, may have utility for cardiovascular health assessment.

In the online algorithm, data is processed one window at time, instead of as a single 24 hour recording. As we will show in Chapter 3, the output of the online algorithm provides a reasonably good approximation of the output of the original batch algorithm.

## 1.2 STATE-OF-THE-ART AMBULATORY MONITORS

Both implantable and non-invasive portable, ambulatory medical monitoring devices are routinely used during clinical practice. Most of these monitoring devices have recording and monitoring capabilities, but only a few perform online processing and/or initiate treatment in response to an event. Since these devices are designed for long-term ambulatory monitoring, long battery lifetimes are important.

For cardiac ailments, portable long-term monitoring devices have been helpful not only for diagnosis of cardiac conditions, but also for treatment of harmful arrhythmias. Implanted devices such as pacemakers ensure that a heart continues to beat at

a normal rate by issuing periodic electrical pulses. Implantable cardioverter defibrillators (ICDs) are similar devices, but can issue higher energy electrical shocks to control life-threatening arrhythmias. Both devices sample and process sensor data, but because computations are fairly simple, these devices have battery lifetimes of up to seven years.

For patients who are being diagnosed for an underlying heart condition, patients can be given a non-invasive Holter monitor or cardiac event monitor. Holter monitors continuously measure the electrical activity of the heart using electrodes attached to the body. Holter monitors are recording devices only and typically, are designed to continuously record data for a period of 24 hours. Data is then analyzed offline for various rhythm abnormalities. On the other hand, a cardiac event monitor is designed to detect intermittent events. They do not continuously collect ECG data, but instead record short segments of ECG data surrounding the occurrence of a patient-detected event. Each recorded segment is approximately 60 seconds in duration.

Non-invasive and implanted devices for monitoring abnormal neurological activity also exist. Recently, companies such as NeuroVista and NeuroPace have introduced inter-cranial implants that can monitor EEG for seizures and stimulate the brain using electrical pulses or shocks.

Non-invasive ambulatory recorders for EEG are also available for clinical use [17]. The majority of these recorders are only capable of capturing and storing EEG for later review by clinicians. While a few models are equipped with real-time seizure event detectors, they make no guarantees on how quickly seizures are detected. Thus, these recorders are unsuitable for alerting or initiating therapies. Integrating seizure onset detectors [20, 59] with current commercial EEG recorders will make delay-sensitive applications a possibility.

Lowering the energy consumption of medical monitoring devices would reduce battery replacement frequency or allow the use of smaller batteries to achieve reasonable battery lifetimes. For example, without applying our methods to the seizure onset detection algorithm, a lightweight Li-Ion coin cell battery would last only 18 hours. In contrast, by applying our data reduction methods, we could extend battery lifetime to three or four days without degrading detection performance significantly.

### 1.3 REDUCING ENERGY CONSUMPTION

Ambulatory medical monitoring systems generally consist of the same set of electronic components. There are sensors to measure physical phenomena, analog-to-digital converters (ADC) to transform analog signals into samples, microprocessors or digital signal processors to compute or process the data, and wireless radios for communication. A good energy saving strategy should target the components that consume the most energy to realize the largest savings. In general, communication components have the highest power consumption and computation components have the lowest power consumption. However, system energy consumption depends not only on power consumption, but also on how often a component is used.

To determine the energy consumption of a wearable monitoring system, we built a prototype seizure monitor. In our device, sensor components are placed on or near the head to gather EEG data. To limit the weight of the devices worn on the head,

we avoid including a complex microprocessor or large battery. Instead, to process the data, a radio is used to transmit the data to a computer located at the belt or within the room, where more complex operations can be performed and more energy capacity is available. Table 1.1 shows the energy required to send 18 channels of EEG data using the radio—the energy consumed by the radio dominates.

To extend battery lifetime, reducing the energy consumed by the component that uses the most energy will yield the largest energy savings. Optimizing hardware components or using more efficient algorithms are two common approaches for reducing energy consumption. The software-based methods that we consider in this thesis reduce energy consumption by examining less data. These methods can provide enormous energy savings, since they affect all levels of the architecture. The two primary data reduction methods that we will consider are *duty cycling* and *feature or sensor selection*.

### 1.3.1 Duty Cycling

One method to reduce the energy consumption of real-time medical monitoring algorithms is to *duty cycle* the entire device or components of the device periodically. While this method is guaranteed to reduce energy consumption, turning a device off at the wrong moment or for too long can cause a delay in the detection of an event or cause an event to be missed. These situations may not matter in applications where the goal is to derive statistical information from the data, such as event frequency and mean heart rate.

In Chapter 3, we describe and apply a duty cycling approach to reduce the energy consumption of the online risk stratification algorithm.

### 1.3.2 Feature or Sensor Selection

In applications where data from multiple sensors are needed to detect events, we can reduce energy consumption by processing data from fewer sensors. Depending on the architecture of the device, fewer sensors will allow less data to be acquired, less processing to be performed, and less data to be transmitted. The risk of using less data is that detection performance may degrade—specificity and sensitivity may decrease and detection latency may increase.

Applications that acquire a signal from a single sensor, but extract multiple features from the data stream, can also benefit from this approach. By extracting fewer

**Table 1.1:** Energy consumption of the different components in our head-mounted prototype seizure monitor to process one second’s worth of 18 channels of EEG data. The energy consumed by the radio dominates. See Chapter 4 for more details.

Component	Energy Consumed (mJ)
EEG sensor	5.4
CPU	6.7
Radio	18.6

features from the data, energy savings can be realized. However, this is possible if the computational and energy cost of feature extraction is high.

In Chapter 4, we use this method to reduce the number of channels needed to build an energy-efficient seizure onset detector with high sensitivity, high sensitivity and reasonable detection latency.

### 1.3.3 Combining Channel Selection with Duty Cycling

Applications that use multiple sensors for event detection can benefit from both sensor selection and duty cycling methods to realize additional energy savings. In theory, by combining duty cycling and sensor selection, any desired level of detection performance can be achieved using minimum energy.

One way to combine the two methods is to organize classifiers into a tree with the lowest energy classifier at the root and the highest energy classifiers at the leaves. Classifiers near the root are consulted first, while classifiers closer to the leaves are consulted as needed. If the classifier at the root is able to quickly and accurately classify a large number of windows using low energy, then the overall energy consumption will be smaller than when using a classifier from a leaf. In some sense the lower energy classifier duty cycles the higher energy classifier based on the data it analyzes. We call this hybrid method *screening*.

Another way to combine the two methods is to simply duty cycle the reduced channel detector and determine the impact on detection performance. In Chapter 5, we apply these techniques to the problem of seizure onset detection and measure their impact on detection performance and energy consumption.

## 1.4 THESIS CONTRIBUTIONS

In this thesis, we made the following contributions:

- (a) We identified real-time monitoring applications where software methods that examine and process less data can be used to reduce energy consumption without impacting the quality of the monitoring.
- (b) We devised a systematic method for transforming an offline algorithm that analyzes ECG data using batch processing into an online algorithm that performs the same task while processing the data in real-time as it is collected. The goal of both algorithms was the same: to risk stratify post-ACS patients into two groups—those with high risk of cardiovascular death and those with low risk of cardiovascular death. The online version of the algorithm achieved an average error of 2.0 with a standard deviation of 6.3 for a test population of 753 patients. This error resulted in only 38 out of 753 patient reclassifications.
- (c) We applied duty cycling to the online risk stratification algorithm to selectively process portions of the ECG data stream. The data that remained was used to approximate the output of the online algorithm. We demonstrated that by sampling 60% of the windows and thus, reducing energy consumption by approximately 40%, only 3% of the patients or 22 out of 753 patients were assigned different risk levels as compared to the non-duty cycled online algorithm.

- (d) We devised a patient-specific machine learning-based method for automatically reducing the number of sensors needed to perform real-time event detection on an energy-constrained device.

Using our channel reduction method, we were able to reduce the average number of channels needed to perform seizure detection for 16 patients from 18 to 6.4. This corresponds to an average energy savings of 60% or an increase in battery lifetime of  $2.5\times$ . The impact on detection performance was minimal. Of the 143 total seizure onsets, the reduced channel detectors were able to find 136 seizure onsets compared to 139 detected by the full 18-channel detectors. The median detection latency increased slightly from 6.0 to 7.0 seconds, while false alarm rate per hour increased only marginally from 0.07 to 0.11.

- (e) We designed and built a prototype of a single channel wireless EEG acquisition system. The energy consumption of the acquisition system was measured to determine how energy consumption varied with the number of sensors.
- (f) We devised and evaluated the impact of combining a screening detector with a reduced channel seizure onset detector on energy consumption and detection performance.

On average, we reduced the energy consumed by the reduced channel detectors by 69%. Each screening detector used 2.8 channels on average. In terms of detection performance, 78 out of the 80 seizures detected by the subset detectors were detected when screening was added. The median latency increased slightly from 6.2 to 6.3 seconds, but the false alarm per hour decreased from 1.2 to 1.1.

- (g) Finally, we evaluated the impact of duty cycling the reduced channel detectors on detection performance.

In expectation, our random duty cycling scheme was able to reduce energy consumption by about 25% while degrading detection performance by only 2%. As expected, average latencies increased, but only by 1.1 seconds. Moreover, the false alarm rate per hour decreased by 0.3 events per hour. In general, detection performance is patient-dependent.

## 1.5 THESIS OUTLINE

The rest of the thesis is organized as follows: In Chapter 2, we provide additional background material on the medical monitoring applications that are described in this thesis. In addition, we describe the general structure of the algorithms used in real-time ambulatory monitoring systems and discuss how these algorithms can be modified to reduce energy consumption. Next, in Chapter 3, we analyze the performance of a duty cycling strategy for reducing the energy consumption of a risk stratification algorithm designed for patients who have recently suffered an ACS. In Chapter 4, we devise a patient-specific feature selection algorithm for selecting useful channels for detecting seizure onset. We then evaluate the detection performance of the reduced channel detector constructed using the channels suggested by the algorithm. The energy consumption of the resulting detectors is determined using an energy model

based on a prototype EEG acquisition system, which is also described in this chapter. Finally, in Chapter 5, we describe methods that employ both duty cycling and channel selection for reducing energy consumption in the context of seizure onset detection. The performance of the detectors that utilize these hybrid approaches is compared to the detectors described in Chapter 4.

---

## Background

Ambulatory medical monitoring devices have the potential to give physicians a better understanding of a patient's medical condition by collecting and processing physiological data as the patient goes about his or her daily routine. Typically, these devices are designed to be worn on or implanted in the patient and thus, must be powered by batteries.

A device designed for long-term ambulatory monitoring must be comfortable and convenient to wear. One important factor that influences comfort is device weight and size. Since batteries can be a large contributor to the size of the device, batteries need to be small. Unfortunately, small batteries have limited energy capacities. Since biomedical signal processing algorithms often involve energy-intensive operations, careful power management is needed to extend the battery lifetime of these devices and to avoid frequent battery recharging or replacement.

In this chapter, we describe the types of applications and algorithms of interest to us in this thesis. First, we describe two example monitoring applications that are representative of real-time medical monitoring algorithms: *seizure onset detection* and *online risk stratification of post-ACS patients*. These applications will be used throughout the thesis to illustrate how the methods are applied and their effectiveness. Next, we characterize the general structure of real-time medical monitoring algorithms and describe possible architectures of ambulatory medical monitoring devices. Finally, we provide an overview of the methods we use for reducing energy consumption and also describe other related methods.

### 2.1 SEIZURE ONSET DETECTION

Epilepsy is a chronic neurological disorder characterized by recurrent unprovoked seizures. According to the Epilepsy Foundation of America, epilepsy affects over 3 million Americans of all ages, at an estimated annual cost of \$15.5 billion in direct and indirect costs [18]. Despite recent advances in the management of epilepsy, more than 20% of individuals experience refractory seizures, seizures that cannot be completely managed with anti-epileptic drugs. A portable non-invasive seizure monitor could benefit these individuals tremendously. For instance, a monitor might be used

to alert patients or caregivers of seizure onset before symptoms of the seizure cause injury [56]. In addition, an ambulatory seizure onset monitor might be used to automatically initiate delivery of a therapy to reduce both seizure intensity and duration [58]. Both alerting and therapeutic applications require seizures to be detected reliably with low latency. In addition, the number of false detections must be minimized to avoid alarming caregivers and the patient unnecessarily.

Seizure onset is defined as the point in time where a patient's EEG transitions from a non-ictal to ictal state. A seizure is the period in time following this event and typically lasts from one to several minutes. One way to detect seizure onset is by processing EEG data that has been collected using non-invasive scalp electrodes. To perform highly specific and sensitive seizure onset detection, data from multiple EEG channels is usually required.

Portable, digital EEG recorders are used routinely to record the EEG of patients outside the hospital [17]. However, few are equipped with seizure event detectors. The integration of seizure onset detectors [20,59] with current commercial EEG recorders would make delay-sensitive applications a possibility. A working implementation of a real-time patient-specific seizure onset detector on a digital signal processor (DSP) was described by Shoeb *et al.* [46].

## 2.2 RISK STRATIFICATION OF POST-ACS PATIENTS

Acute coronary syndrome (ACS) is a term used to describe the clinical signs and symptoms of sudden myocardial ischemia—a sudden reduction in blood flow to the heart. ACS is often used to describe patients who present with either unstable angina (UA) or acute myocardial infarction (MI). Both of these conditions are highly prevalent—an estimated 16.8 million Americans have experienced either myocardial infarction or unstable angina in their lifetime [29].

Individuals who have been diagnosed with ACS have an elevated risk for future adverse events. In the GUSTO-IIb trial, patients who experienced a non-ST elevation acute coronary syndrome (NSTEMI) had mortality rates of 5.7% within 30 days and 11.1% within one year [2]. For unstable angina patients, the mortality rates were 2.5% and 7.0% respectively. In addition, 9.8% of NSTEMI patients and 6.2% of unstable angina patients experienced an myocardial infarction within six months.

There are two reasons for this increased risk. First, the underlying cause of the ACS—cardiovascular disease—may cause future events. Second, ACS often leads to myocardial damage or necrosis, the permanent damage of cardiac tissue. Myocardial necrosis can interfere with the normal electrical conduction pathways of the heart, which can further lead to fatal arrhythmias such as ventricular tachycardia.

The risk levels for death and adverse cardiac events can vary among patients diagnosed with ACS [2, 7, 60]. Because the appropriate treatment to administer to each patient depends on a patient's risk level, accurate risk assessment is important. The most invasive treatments, which are typically more expensive and risky, are most beneficial to high-risk patients. Low-risk patients, on the other hand, may be treated successfully with more conservative therapies. For ACS patients, treatments may include anti-arrhythmic drugs, surgical or catheterization procedures, or the implantation of a



device such as an implantable cardioverter defibrillator (ICD), a device that can detect arrhythmias and administer an electrical shock to restore a normal cardiac rhythm.

### 2.2.1 Risk Stratification Using ECG Recordings

To assess a patient's risk for future adverse events, clinicians can use a number of post-ACS risk stratification methods. Some of these methods also can be used to predict the expected benefit of a given intervention to a patient. One class of risk stratification techniques involves using long-term electrocardiograph recordings. Using long-term ECG for risk stratification would be beneficial since these recordings are routinely obtained during hospital stays or by portable Holter monitors. Moreover, ECG data can provide a wealth of diagnostic information about cardiovascular function. Currently, none of these portable devices provide online risk assessment. Online risk assessment allows physicians to continuously track cardiovascular health.

Various ECG-based risk stratification methods have been proposed, including heart rate variability [30], deceleration capacity [4], and T-wave alternans [47]. It has been shown that risk stratification techniques using long-term ECG collected in the hours after admission can refine an initial prognosis made at the time of presentation. Syed *et al.* [51] recently introduced an ECG-based post-ACS risk stratification method that uses *morphologic variability* (MV) to accurately risk stratify patients. MV-based techniques, such as MV-LF/HF and MV-DF [50] estimate risk by quantifying the variation of morphologies in the heartbeats captured in an ECG signal. MV-based techniques have successfully been applied to ECG signals for assessing the risk of cardiovascular death by Syed *et al.* [54].

## 2.3 ONLINE AMBULATORY MEDICAL MONITORING ALGORITHMS

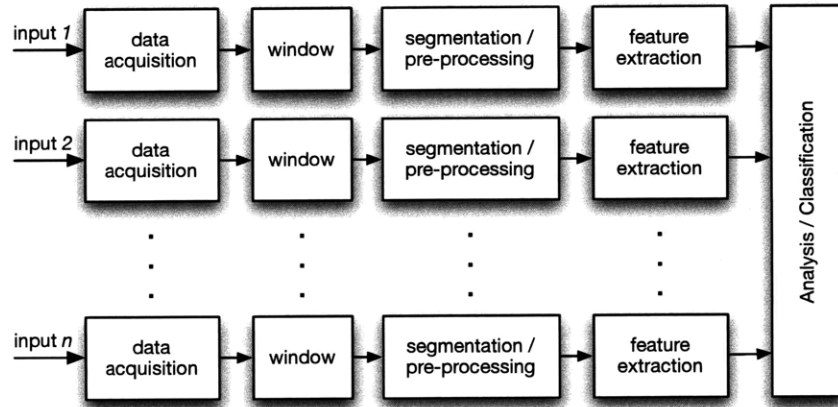
Real-time ambulatory medical monitoring algorithms can be characterized abstractly using a common structure. Figure 2.1 shows the components of a prototypical real-time medical monitoring algorithm. Note that we will use the notation  $x_{ij}$  to represent the  $j^{\text{th}}$  element of a sequence  $x$  obtained from the  $i^{\text{th}}$  sensor. When there is only a single sensor, we omit the first subscript.

### 2.3.1 Data Acquisition

In the first stage, data is acquired from one or more sensors that convert physical phenomena into electrical signals. Examples of sensors that could be used in an ambulatory system include pulse oximeters, ECG, EEG, and accelerometers. Each electrical signal is optionally filtered and/or amplified and then quantized and sampled using an analog-to-digital converter (ADC) at a fixed rate to form one or more sample sequences or data streams. Once a signal is quantized, a microcontroller or DSP can be used to process the data. In general, each signal is treated independently and can be filtered and sampled in different ways.

### 2.3.2 Windowing

In the next stage, the samples from each data stream  $i$  are grouped in time to form a sequence of windows,  $w_{i1}, w_{i2}, \dots$ . A window is defined as a substring of  $x[t]$ , where  $x[t]$  is a single sequence of quantized samples.

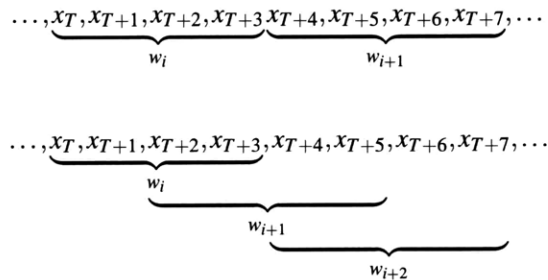


**Figure 2.1:** Abstract structure of a prototypical ambulatory medical monitoring algorithm.

The windows may overlap. Given a single sensor, two consecutive windows,  $w_i$  and  $w_{i+1}$ , overlap if  $w_i \cap w_{i+1}$  is a non-empty substring of  $x[t]$ . Windows may also consist of a variable number of samples, though typically, window sizes are constant. Figure 2.2 shows examples of how a sample stream could be windowed to form a sequence of windows.

### 2.3.3 Segmentation and Pre-processing

Some physiological signals have an underlying structure that is related to a recurring physical phenomenon. For these signals, once a window of samples has been accu-



**Figure 2.2:** Examples of how an input sample stream is divided into *overlapping* and *non-overlapping* windows. Both streams consist of eight samples beginning at time  $T$ . The top sample stream is divided into two non-overlapping windows of four samples each. The bottom sample stream is divided into three overlapping windows, each consisting of four samples.

mulated, the data stream can be further divided into segments. For example, ECG signals contain QRS complexes that are correlated with the depolarization of the right and left ventricles of the heart. Thus, an algorithm that analyzes ECG signals, such as the online risk stratification algorithm, will often extract segments (beats) from the signal before further processing.

Some algorithms may also need to pre-process the signal, *e.g.*, to filter out noise, to prepare the signal for further analysis.

#### 2.3.4 Feature Extraction

Features are then extracted from each window of data. Features can be time-based and/or frequency-based. Examples of features include: signal mean, signal variance, energy in specific frequency bands, etc.

Though Figure 2.1 shows features being extracted from each sequence of windows independently, features can be extracted from multiple window sequences at the same time depending on the nature of the application.

Also, more than one feature can be extracted per window of data. Typically, when multiple features are extracted from a single window, they can be combined into a single feature vector. As an example, consider a single sequence of windows,  $w_1, w_2, \dots, w_n$ . We can extract features from each window by applying a set of application-specific functions  $g_1, g_2, \dots, g_k$  as shown:

$$\begin{aligned} f_1 &= [g_1(w_1) g_2(w_1) \cdots g_k(w_1)]^T \\ f_2 &= [g_1(w_2) g_2(w_2) \cdots g_k(w_2)]^T \\ &\vdots \end{aligned}$$

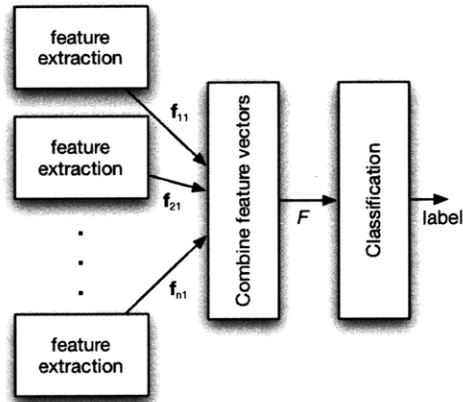
#### 2.3.5 Analysis and Classification

After features have been extracted, a typical real-time medical monitoring algorithm will take one of two actions: classify the features or simply store the features for future analysis.

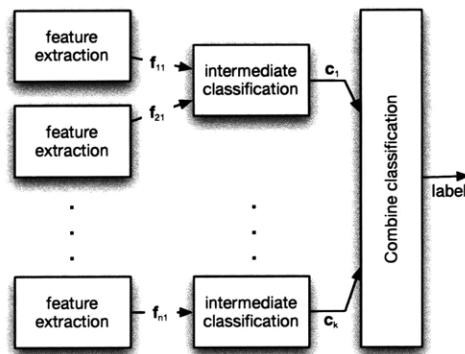
Often, the extracted features will need to be classified, *e.g.*, for event detection. Labels, which are typically discrete elements, are assigned to each window by a classifier when a specific set of criteria is met. Many different classification algorithms can be used to label each window. Some examples include decision trees and support vector machines. In addition, a classification algorithm may label a window by comparing information contained in one window with information contained in previous windows.

Figure 2.3 shows how feature vectors extracted from multiple streams could be classified. In Figure 2.3(a), the feature vectors from each of the streams are combined into a single feature vector and then labeled by one classifier. In Figure 2.3(b), the label may be generated using hierarchical classification, that is, labels for a subset of the feature vector are generated first. Then, the intermediate labels are combined to generate a single label.

In the case where classification is not needed, the extracted features can be stored for offline or direct analysis.



(a) Combined feature vector



(b) Hierarchical

**Figure 2.3:** Example classifier architectures

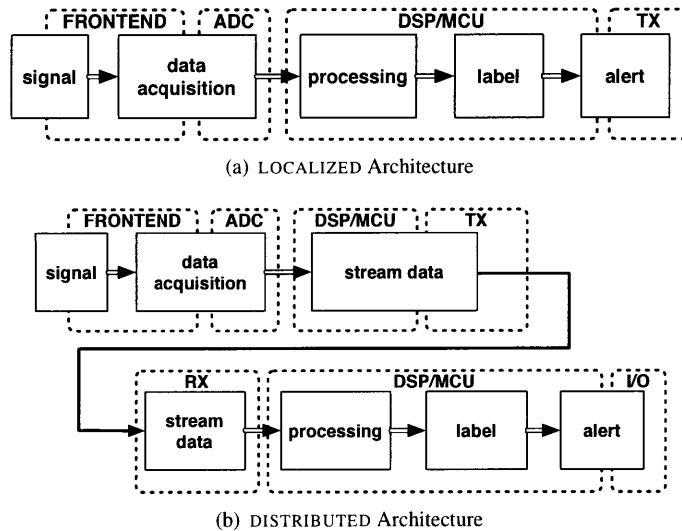
### 2.3.6 Aggregate Analysis

In the last step of an online medical monitoring algorithm, the labels or feature vectors are analyzed in aggregate to declare in event. Once an event has been detected, further action can be taken if the event is potentially harmful.

## 2.4 ARCHITECTURES OF WEARABLE AMBULATORY MONITORING SYSTEMS

The energy consumed by a wearable medical monitoring algorithm depends on algorithm complexity and system architecture. A more complex algorithm requires more computation and thus, will likely require more energy to perform.

Figure 2.4 shows two different architectures for structuring an ambulatory monitoring system. Depending on the processing requirements of the detection algorithm, the size of the sensors and computation components, and the goal of the application, one architecture may be preferred over the other.



**Figure 2.4:** Architectures for ambulatory medical monitoring systems. Both the labeling and communication components are optional.

Figure 2.4(a) shows a block diagram of the LOCALIZED architecture. In this architecture, data acquisition, processing components, storage and communication are all co-located with the sensor. In this architecture, most of the signal processing occurs near the sensor. The communication component is only necessary if a remote signal (*e.g.*, to send an alert) needs to be initiated. An example of a device with this architecture is an ICD.

Figure 2.4(b) shows a diagram of the DISTRIBUTED architecture. In this architecture, only the data acquisition components are co-located with the sensor. Thus, the amount of hardware placed near the sensor can be minimized. In such an architecture, data is delivered to a remote device—such as a belt-worn or desktop computer—where more computation and more energy resources are available. An example of a device with this kind of architecture is a wireless Polar Heart Rate Monitor. Alternatively, one could stream the data from the sensors to a remote computer via the Internet using 802.11-based wireless technologies. Feature extraction and/or classification could occur at the remote location.

Using a belt-worn or local computing device instead of a remote computer for computation has the advantage of being able to alert the patient if an event were to occur. In addition, local computations are resilient to remote network or server failures.

## 2.5 OVERVIEW OF ENERGY SAVING METHODS

To reduce the energy consumed by a real-time medical monitoring device, a system designer could implement the algorithms used with an application-specific integrated circuit (ASIC). Hardware-based methods can achieve many orders of magnitude of energy savings. However, custom integrated circuits are expensive to design, manu-

facture, and test. In addition, hardware is inflexible. Therefore, improvements to the signal processing algorithms cannot be easily incorporated into a system built around an ASIC.

Alternatively, system designers can make algorithmic transformations to reduce energy consumption. Since software-based methods are easier and cheaper to implement than hardware-based methods, systems can take advantage of the energy savings offered by software-based methods with much lower implementation effort.

One way to decrease the energy consumed by the algorithm is to reduce its time complexity since energy consumption is proportional to complexity. In this thesis, we reduce computation and thus, energy consumption by reducing the amount of data processed. The risk of collecting and processing less data is that detection performance may degrade—specificity and sensitivity may decrease and detection latency may increase. In this thesis, we consider three different data reduction methods: duty cycling, feature or sensor selection, and hybrid duty cycling-sensor selection methods.

## 2.6 RELATED WORK

Software-based algorithms that reduce the energy consumption of embedded systems utilize a variety of mechanisms. These mechanisms include [15]:

- (a) *Duty cycling*: Turning off power to a component of the system can reduce average power draw.
- (b) *Batching*: Buffering operations and then executing them in a burst amortizes high start-up or overhead energy costs.
- (c) *Hierarchy*: Arranging operations by their energy consumption can reduce consumption if low-energy operations are invoked before high-energy operations and if the result of the low-energy operation eliminates the invocation of the high-energy ones.
- (d) *Redundancy reduction*: Reducing redundancy through compression or elimination of data allows systems to perform fewer processing, computation, and storage operations.

The impact of these methods on energy consumption will depend on the underlying hardware.

Many software-based energy saving strategies employ these mechanisms. The most commonly-used mechanism is duty cycling. For example, energy-efficient MAC protocols use duty cycling to reduce the energy consumption of a radio listening for an incoming message [12, 39, 55]. These protocols adapt the duration the device spends in the on or off state based on observations of the communication method. Another energy saving strategy that uses duty cycling is predictive shutdown. Predictive shutdown methods [48] use data about the usage patterns of a computing device to predict the occurrence and duration of future idle times of a device. Idle times give an energy saving strategy opportunities to duty cycle.

The batching mechanism is most often used to amortize the overhead of writing to storage devices or of transmitting data wirelessly. For flash storage devices, there is a

fixed energy cost for accessing a page and a per-byte cost for reading or writing each byte within that page [31]. The fixed cost is ten to one thousand times the cost of the per-byte cost. The fixed cost can be amortized by reading or writing a large number of bytes at one time. For wireless transmission, physical and link layers require additional non-payload bits to be transmitted along with each packet of data. During the transmission of these bits, energy is consumed. Sending payloads that contain many bytes of data at once can amortize the cost of transmitting these overhead bytes.

Hierarchy methods can be useful for sensing based systems where low-power sensors are sampled before higher-power sensors. Higher-power sensors are turned on only if they can provide information not provided by the low-power sensors [16]. Hierarchy will reduce energy consumption when higher-power sensors are consulted infrequently. Hierarchy methods can also be used to reduce the idle power consumption of radios by using low-power radios to activate high-power radios when necessary [42].

Removing redundancy reduces energy consumption by reducing the amount of data that is transmitted, stored or processed. Various researchers have conducted studies that compare the impact of various off-the-shelf compression methods on energy consumption of embedded systems [3, 41]. Compression methods result in energy savings when the cost of computation exceeds the cost of transmission or storage.

The energy saving methods that we employ in this thesis use duty cycling, redundancy reduction and hierarchy mechanisms. Sensor selection is a method that removes redundancy with respect to a specific task, namely seizure onset detection. The hybrid methods are hierarchical approaches. Note that our methods can be used in conjunction with existing methods—of which there are many—that reduce energy consumption.

Duty cycling the sensors and processor also offers opportunities to dynamically scale the voltage and frequency of the CPU in order to further reduce energy consumption. If we are processing the data online and we know when and for long our device will be turned off, we can lower the CPU frequency and voltage to potentially achieve more energy savings. For example, suppose we know that we will sample the next two seconds worth of data and then shut-off for four seconds. Instead of processing the two seconds of data as fast as possible and then going into the idle state before the device needs to sample again, we can lower the frequency and spread the processing over six seconds. Lowering the frequency will allow lowering of the voltage. Decreasing the voltage can result in energy savings if the idle power consumption is large.

### **2.6.1 Relation to Compressive Sampling**

The feature selection and duty cycling methods we discuss in this thesis reduce the amount of data collected without adjusting sampling rates. Implicit in our approach is the idea that there is some redundancy in the data and thus, removing the redundant data does not substantially degrade detection performance.

A related approach that also leverages the idea of redundancy is compressive sampling [8]. Compressive sampling theory asserts that signals can be recovered using fewer samples than traditional methods. This implies that the information in a signal may be much smaller than suggested by its bandwidth and thus, can be represented

much more compactly.

Thus, compressive sampling is not a feature selection method, but a feature extraction method since the goal is to represent a signal compactly. However, because it is a paradigm that is able to “compress” the signal by using a lower average sampling rate, compressive sampling can reduce energy consumption dramatically.



---

## Random Duty Cycling

Portable medical monitoring devices designed for long-term use must be comfortable and convenient for a patient to wear. Since battery lifetime can impact the convenience of a body-worn device, reducing energy consumption is critical.

One approach for reducing energy consumption is to duty cycle, *i.e.*, to turn off components of the device periodically. The risk of duty cycling is that key segments of the input signal will not be processed. In event detection systems, turning a device off at the wrong moment or for too long can result in a delay in the detection of an event or an event to be missed.

Duty cycling strategies can be classified into two broad categories: data dependent and data independent duty cycling. A data dependent duty cycling approach makes decisions to turn components on or off based on analysis of input data. Conversely, a data independent strategy does not examine the input data at all. For example, consider a device designed to detect anomalous events using a single sensor. If components are put to sleep probabilistically or after a predetermined duration has elapsed, then the system employs a data independent strategy. On the other hand, if the decision to put the system to sleep is based on detection of a pattern in the input signal, then the system employs a data dependent strategy.

Data dependent strategies are useful for applications where the input signals are correlated to the occurrence of a future event. Data dependent strategies can be used in applications where latency is important, such as seizure onset detection, and also where latency is less important, *e.g.*, event enumeration. On the other hand, data independent strategies are primarily useful where the goal of the application is to derive statistical information from the data, such as event frequency, mean heart rate, etc., using sampling. For deriving aggregate information, sampling methods are effective if the values derived by processing the entire input signal form a distribution that can be approximated by sampling a subset of the input signal.

In this chapter, we describe and evaluate a random data independent duty cycling method for reducing the energy consumption of an existing ECG-based algorithm for risk stratification of patients who have suffered an acute coronary syndrome (ACS). Since the existing algorithm is a batch program, its original form is not suitable for ambulatory medical monitoring applications and cannot be duty cycled. Thus, we first

describe a redesigned version of the original algorithm that processes ECG signals in an online fashion and compare it to the original. Then, we describe the impact of duty cycling on the risk stratification ability of the online algorithm.

We also explore the impact of duty cycling on energy consumption. Typically, when determining the impact of an energy savings strategy, an energy model is needed. For duty cycling, we report energy savings as the percentage of time a device spends in the off state—we assume that a device that is off consumes no energy. In truth, this model is too simplistic. First, idle power consumption is not 0 since timers and memory needs to be powered. To switch a device from the off state to the on state, low-power timing or watchdog circuitry is needed. Second, state changes do not occur instantaneously. Thus, some amount of energy is consumed during the act of switching. Since we assume that these factors are small, reporting the energy savings as a percentage of time a device is off is sufficient.

### 3.1 THE MORPHOLOGICAL VARIABILITY ALGORITHM

Acute coronary syndrome (ACS) is a term used to describe the clinical signs and symptoms of sudden myocardial ischemia. Some individuals who have suffered an ACS are at higher risk for future adverse cardiac events, such as sudden cardiac death. Thus, accurate risk assessment is important. Although there are many risk stratification methods, we focus on the Morphological Variability-Diagnostic Frequency (MV-DF) algorithm that performs risk stratification using long-term ECG recordings [50]. MV-based techniques have successfully been applied to ECG signals for assessing the risk of cardiovascular death (CVD) by Syed *et al.* [54].

The original algorithm for determining MV-DF accepts 24 hours of three channels of pre-recorded ECG data from a single patient as input. Typically, the ECG data is sampled at 128 or 256 samples per second. There are four phases to the algorithm. All three channels are used in the first two stages, while only a single channel is needed in the remaining stages.

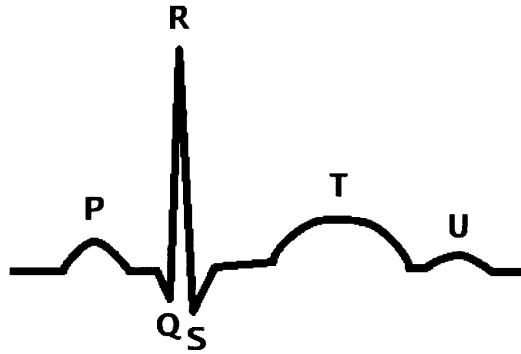
#### 3.1.1 Beat Segmentation

The ECG signal is segmented into beats using two QRS detection algorithms with different noise sensitivities. A QRS complex is a distinctive waveform visible in an ECG signal that is produced as a result of depolarization of the ventricles. Figure 3.1 shows a simple diagram of a QRS complex. Both algorithms are part of the Physionet SQI software package. The first algorithm uses digital filtering and integration [23], while the second is based on a length transform after filtering [61]. QRS complexes are marked only at locations where both algorithms agree.

#### 3.1.2 ECG Noise Filtering

Characterizing morphology requires using information from the parts of the ECG that are low amplitude and where small amounts of noise can significantly affect the signal-to-noise ratio. To minimize this effect, automatic methods for noise removal and signal rejection are first applied to the ECG.

Noise removal is carried out in three steps. In the first step, baseline wander is subtracted from the ECG signal. The baseline signal is determined by applying median filtering to the signal [13]. In the second step, the baseline corrected ECG signal



**Figure 3.1:** A diagram of a QRS complex. The letters Q, R, and S marks deflection points in the signal.

is processed using wavelet de-noising with a soft-threshold to remove high frequency noise [14]. Finally, sensitivity to calibration errors is decreased by normalizing the entire ECG signal by the median R-wave amplitude.

Next, ectopic beats and “noisy” beats are removed. An ectopic beat is a disturbance of the electrical conduction system of the heart where beats arise from the wrong part of the heart muscle. Ectopic beats are determined using the beat classification algorithm included in the Physionet SQI package. Noisy beats are identified by using four analysis methods [28]: disagreement between multiple beat detection algorithms on a single ECG lead, disagreement between the same beat detection algorithm on different ECG leads, the kurtosis of a segment of ECG, and the ratio of power in the spectral distribution of a given ECG segment between 5–14 and 5–50 Hz. In addition to removing noisy and ectopic beats, the beats occurring immediately before and after each ectopic beat were also removed.

After “bad” beats are removed, a signal rejection algorithm is applied to the remaining data. Unlike noise removal, which helps to remove artifacts commonly encountered in long-term ECG records, the signal rejection process is designed to remove longer sections of the ECG signal where the signal-to-noise ratio is so low that meaningful analysis of the morphology is challenging even after noise removal.

First, the data is divided into half-hour windows and the standard deviation of the R-waves during each half-hour window is calculated. Any window with a standard deviation greater than 0.2887 is discarded. Given the earlier normalization of the ECG signal, under a model that allows the R-wave amplitude to uniformly vary between 0.5 and 1.5 every beat (i.e., up to 50% of its mean amplitude), we expect the standard deviation of the R-wave amplitudes to be less than 0.2887 for any half-hour window. This heuristic identifies windows that are likely corrupted by significant non-physiological additive noise, and where the morphology of the ECG cannot be meaningfully analyzed. In addition, any half-hour window that contains fewer than 1500 beats is also discarded.

### 3.1.3 Morphologic Difference

Next, the *morphologic difference* or MD time series is generated for one channel of ECG (Lead I) by performing dynamic time-warping (DTW) [34] between the sequences of samples of adjacent beats in the recording. At a high level, DTW takes the two input sequences and aligns them in a way that minimizes an objective function. DTW outputs a sequence of pairs that describes the optimal alignment and the optimal value of the objective function achieved. A detailed description of the DTW algorithm, adapted from [50] and [54], follows.

Let  $x_1$  be the sequence of samples contained in the first beat and  $x_2$  be the sequence of samples contained in the second beat. Assume that  $x_1$  has length  $M$  and  $x_2$  has length  $N$ . Using these two sequences, an  $M \times N$  distance matrix  $D$  is first constructed where  $D(i, j) = (x_1(i) - x_2(j))^2$ . An *alignment*  $\alpha$  of the two sequences  $x_1$  and  $x_2$  is a sequence of integer pairs of some length  $K$ .

$$\alpha = (\phi_1(1), \phi_2(1)), (\phi_1(2), \phi_2(2)), \dots, (\phi_1(K), \phi_2(K)), \quad (3.1)$$

where  $\phi_1$  and  $\phi_2$  represent row and column indices of the distance matrix  $D$ . Intuitively, each ordered pair  $(\phi_1(i), \phi_2(i))$  matches two elements to be aligned. Any feasible sequence will satisfy the following boundary conditions:

$$\begin{aligned} \phi_1(1) &= 1, & \phi_2(1) &= 1, \\ \phi_1(K) &= M, & \phi_2(K) &= N, \end{aligned}$$

as well as the following continuity and monotonicity conditions:

$$\begin{aligned} \phi_1(j) \leq \phi_1(j+1) \leq \phi_1(j) + 1 & \quad \forall j: 1 \leq j < K, \\ \phi_2(j) \leq \phi_2(j+1) \leq \phi_2(j) + 1 & \quad \forall j: 1 \leq j < K. \end{aligned} \quad (3.2)$$

In other words, both sequences can only contain non-decreasing sample numbers. In addition, every sample in each beat must be matched with a sample in the other beat.

Given an alignment  $\alpha$  between beats  $x_1$  and  $x_2$ , the cost of the alignment,  $C$  can be computed as:

$$C(\alpha, x_1, x_2) = \sum_{j=1}^K (x_1(\phi_1(j)) - x_2(\phi_2(j)))^2 \quad (3.3)$$

The optimal cost or DTW distance is the minimum cost  $C$  over all feasible alignments.

$$C(x_1, x_2) = \min_{\alpha} C(\alpha, x_1, x_2) \quad (3.4)$$

Dynamic time-warping captures both amplitude and timing differences between the signals.

The search for the optimal path can be carried out efficiently using dynamic programming [11]. Let  $\gamma(i, j)$  be minimum cost of aligning the subsequences  $x_1(1, \dots, i)$  and  $x_2(1, \dots, j)$ . When  $i = 1$  and  $j = 1$  then the two inputs only have one sample each, so  $\gamma(1, 1) = D(1, 1) = (x_1(1) - x_2(1))^2$ . Using the continuity conditions Equation 3.2,

we can then use the following recurrence relation to determine  $\gamma(i, j)$ :

$$\gamma(i, j) = D(i, j) + \min \begin{cases} \gamma(i-1, j) \\ \gamma(i, j-1) \\ \gamma(i-1, j-1) \end{cases} . \quad (3.5)$$

Formulated in this way, the optimal cost  $C(x_1, x_2)$  is equal to  $\gamma(M, N)$ . An implementation of Equation 3.5 can be used to compute the optimal total cost in  $O(MN)$  time.

The recurrence 3.5 allows a single sample in either sequence to be aligned with an arbitrarily large number of consecutive samples in the other sequence. Syed *et al.* restrict the local range of the alignment path to prevent biologically implausible alignments of large parts of one beat with small parts of another. For an entry  $(i, j)$  in the distance matrix  $D$ , only paths passing through  $(i-1, j-1)$ ,  $(i-1, j-2)$ ,  $(i-2, j-1)$ ,  $(i-1, j-3)$  and  $(i-3, j-1)$  are permitted. To account for this restriction, we use the following recurrence:

$$\gamma(i, j) = D(i, j) + \min \begin{cases} \gamma(i-1, j-1) \\ D(i-1, j) + \gamma(i-2, j-1) \\ D(i, j-1) + \gamma(i-1, j-2) \\ D(i-1, j) + D(i-2, j) + \gamma(i-3, j-1) \\ D(i, j-1) + D(i, j-2) + \gamma(i-1, j-3) \end{cases} . \quad (3.6)$$

This algorithm is executed using all adjacent pairs of beats in the original ECG recording as input to produce a sequence of energy differences. This new signal is then smoothed using an 8-tap median filter. The median filter ensures that high values in the MD time series correspond to persistent changes in morphology.

### 3.1.4 Spectral Energy of Morphologic Differences

In the final stage of the algorithm, the MD time series is grouped into five minute windows. The power spectral density of the MD values within each five minute window is computed using the Lomb-Scargle periodogram [26]. The Lomb-Scargle periodogram is designed to measure the spectral content of an irregularly sampled signal by taking into account both the signal value and the time of each sample. Unlike other spectral estimation techniques, this method can be used without using interpolation to replace missing data. Next, the energy in the frequency range of 0.30–0.55 Hz is computed for each window from the power spectrum. We refer to this number as the per window MV value. Finally, to obtain a single risk measure, the 90<sup>th</sup> percentile of the per window MV values is computed. This value is known as MV-DF.

Syed *et al.* demonstrated the ability of MV-DF to discriminate between patients who have a high risk for cardiovascular death and patients who have low risk [53]. Six hundred patients randomly selected from the placebo population for the TIMI MERLIN [32] trial were tested. On this population, MV-DF was computed for each patient and compared to a threshold. According to the authors, the threshold value of 52.5 was determined by computing the MV-DF for a subset of patients in the TIMI DISPERSE-2 trial and finding the seventy-fifth percentile of the resulting distribution.

Using the dichotomized risk variables, Syed *et al.* show that high MV-DF was strongly associated with death over a 90 day period following NSTEMI with a hazard ratio of 10.45 ( $p < 0.001$ ). In addition, the  $c$ -statistic of the MV-DF values of patients from the TIMI MERLIN study was 0.85, which exceeds the threshold of 0.8 associated with genuine clinical utility [35].

### 3.2 ONLINE MONITORING OF MV-DF

In its current form, the MV-DF algorithm provides only a snapshot of the risk level of a patient following an ACS—and only after 24 hours of data has been collected. Since cardiovascular health may change over time, using the algorithm for continuous risk assessment could be beneficial. For example, continuous risk assessment using MV-DF or the MD time series could help monitor the effectiveness of anti-arrhythmic drugs prescribed to patients who have had an ACS event. In addition, continuous monitoring could enable clinicians to obtain information about a patient during less frequent physical activities, such as exercise.

Currently, there is no concrete evidence demonstrating that online risk stratification is clinically valuable. However, the purpose of this thesis is not to demonstrate the utility of online risk stratification. Rather, we use online MV as an example to illustrate the usefulness of random duty cycling.

The current algorithm for computing MV-DF is not structured for online monitoring. One issue is that 24 hours of data needs to be collected before processing can proceed. One consequence of this is that more storage is required. If we can process data as it is collected, we can reduce the storage requirements of the algorithm. Another consequence is that the per window MV values are not available until after 24 hours of data has been collected and processed.

In the rest of this chapter, we describe a restructured, online version of the original MV-DF algorithm. We test both algorithms on two patient populations and compare the resulting MV-DF values. In addition, we use this new value, the online MV-DF, to risk stratify a patient population. We show that the online algorithm is a good approximation of the original algorithm.

#### 3.2.1 The Online MV-DF Algorithm

To restructure the algorithm, we process the data on a window by window basis, where each window is five minutes in duration. For each window, we perform the following steps:

- (a) First, the ECG signals are segmented into QRS complexes. The same segmentation algorithms used in the original algorithm are used here. Next, the baseline wander is removed and wavelet de-noising is used to clean the signal. As before, the signal is normalized using the median of the R-wave amplitude.
- (b) Next, ectopic beats and “noisy” beats are removed. The same methods that are used to identify ectopic beats and noisy beats are used here.
- (c) After these beats have been removed, we compute the standard deviation of the R-waves in the five minute window. The window is discarded if the standard deviation of each window is greater than 0.2887. We assume that the distribution of

the amplitude of the R-waves does not change even when less data is considered. Windows that contain fewer than 100 beats are also discarded.

- (d) For any window that is not discarded, the MD time series is derived from the remaining beats. The MV value for the window is computed by applying the Lomb-Scargle periodogram to the MD time series and summing the energy in the frequency band 0.3 to 0.55 Hz. After 24 hours worth of data has been processed, the online MV-DF value for the 24 hour period is computed as the 90<sup>th</sup> percentile of the per window MV values.

### 3.2.2 Discussion

In the online risk stratification algorithm, the ECG is processed in five minute windows. We elected to use five minute windows since the original MV-DF algorithm also sliced the data into five minute windows. However, we believe that there is a range of acceptable window sizes that could produce similar estimates of the MV-DF value. Further experiments will need to be conducted to better understand the impact of changing the window size.

## 3.3 COMPARING APPROACHES

The two algorithms are compared by examining the results produced when ECG data recorded from identical patient populations are given as input. Given the differences between the two algorithms, the algorithms will process a different set of beats. Thus, we expect the outputs to be different. However, since no formal specification for MV-DF exists, we cannot evaluate the quality of either output.

### 3.3.1 Data

For input to the algorithms, we use recorded ECG data collected from patients who participated in two different trials: the TIMI DISPERSE-2 trial and the TIMI MERLIN trial. The DISPERSE-2 trial compared the efficacy and safety of AZD6140 and clopidogrel in patients admitted following NSTEMI. Patients in the study had continuous Holter ECG data recorded at 128 samples per second within 48 hours of admission due to NSTEMI. A total of 990 patients were enrolled in the trial. After filtering out patients with recordings lasting less than 24 hours and those with low quality data using the method described in [52], 753 patients remained. This data set was used to guide the development of the online algorithm and to compare the online estimate to the original output. Of the 753 patients, fourteen deaths were observed over a follow-up period of 90 days.

The second data set consists of Holter ECG recordings from 2,302 patients in the placebo group of the TIMI MERLIN trial. The MERLIN trial compared the safety and efficacy of ranolazine versus placebo in patients post-NSTEMI. All patients in this study had at least 24 hours of continuous ECG data, again sampled at 128 samples per second. The data was preprocessed in a similar way as for DISPERSE-2. From the 2,302 patients, we randomly selected 800 patients for inclusion in our second test set. Fifteen patients died of cardiovascular causes within 90 days in this group. This data set was withheld during development of our online algorithm and thus, was used to validate our results.

### 3.3.2 Evaluation Metrics

For a patient  $i$ , let  $MV_A(i)$  denote the value output by the original algorithm and  $MV_B(i)$  be the value output by the online algorithm. Given these values, we then used the following metrics to compare the two algorithms:

- (a) To compare how well the online algorithm approximates the original algorithm, we computed the absolute difference,  $MV_B(i) - MV_A(i)$ , and percent difference,  $(MV_B(i) - MV_A(i))/MV_A(i) \times 100$  for all patients and examined the resulting distribution.
- (b) We computed the  $c$ -statistic to measure how well the online MV-DF can distinguish between patients who have a high risk of cardiovascular death and patients who have a low risk. For a binary classification problem, such as ours, the  $c$ -statistic is equivalent to the area under the receiver operating characteristic (ROC) curve. The area under the curve is equal to the probability that the MV-DF measure will rank a randomly chosen high risk patient higher than a randomly chosen low risk one. An algorithm that can correctly order all pairs of high-risk and low-risk patients has a  $c$ -statistic of 1.0.
- (c) We also computed the risk assigned by the two different algorithms to all the patients in a given cohort, *e.g.*, the TIMI DISPERSE-2 or TIMI MERLIN data sets, and compared the risk assignments to the death outcome of each patient. A “1” indicates that the patient died within 90 days and a “0” indicates otherwise.

Let  $\delta_A(i) \in \{0, 1\}$  be the risk class assigned to patient  $i$  where  $\delta_A(i) = (MV_A(i) > \tau_A)$ . Similarly, let  $\delta_B(i)$  be the risk class assigned to patient  $i$  where  $\delta_B(i) = (MV_B(i) > \tau_B)$ . A “1” denotes high risk and a “0” denotes low risk.  $\tau_A$  and  $\tau_B$  are the threshold MV-DF values used to separate low risk patients from high risk patients.  $\tau_A$  is determined by computing the 75<sup>th</sup> percentile of the original MV-DF values of all patients in the TIMI DISPERSE-2 cohort, while  $\tau_B$  is the 75<sup>th</sup> percentile of the online MV-DF values in the same cohort.

Patients marked as low risk, but died within 90 days are considered false negatives, while patients marked as high risk, but survived past 90 days are considered false positives.

- (d) Finally, we compared the risk assignments  $\delta_A(i)$  to  $\delta_B(i)$  for all patients  $i$ . For this comparison, we treated the risk class assigned by the original algorithm,  $\delta_A(i)$  to be ground truth. A patient who is marked as low risk by the online algorithm ( $\delta_B(i) = 0$ ), but marked as high risk by the offline algorithm ( $\delta_A(i) = 1$ ) is considered a *relative false negative*. Similarly, a patient who is marked as high risk by the online algorithm ( $\delta_B(i) = 1$ ), but marked as low risk by the offline algorithm ( $\delta_A(i) = 0$ ) is considered a *relative false positive*.

## 3.4 RESULTS

Figure 3.2 shows the distributions of the MV-DF values computed using the original and online algorithms. Visually, it is apparent that the distributions are similar, but not

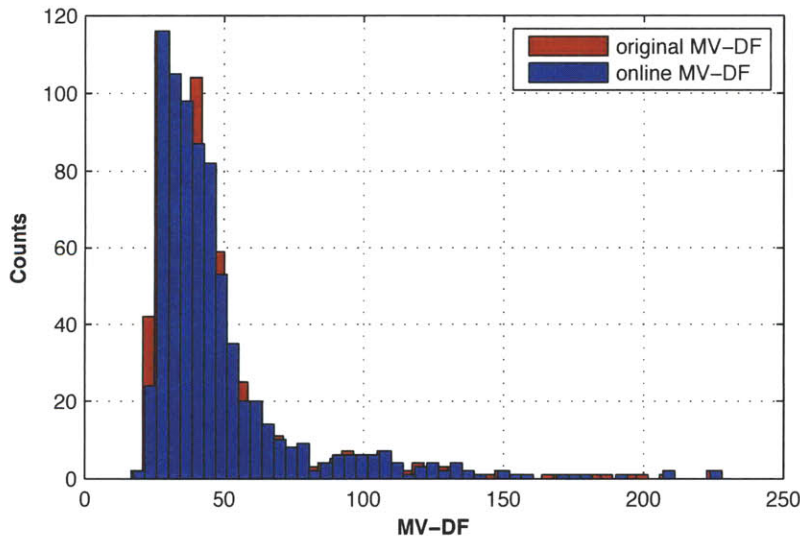


identical. The two-sample Kolmogorov-Smirnov (K-S) test yielded a K-S statistic of 0.045 ( $p = 0.418$ ) between the two sample distributions with their means removed. In the K-S test, the null hypothesis is that the underlying cumulative distribution functions (CDFs) are equal for all samples. Thus, the large  $p$  value suggests we cannot reject the null hypothesis, since we can only reject the null hypothesis when the significance level, typically 0.05, is greater than or equal to  $p$ .

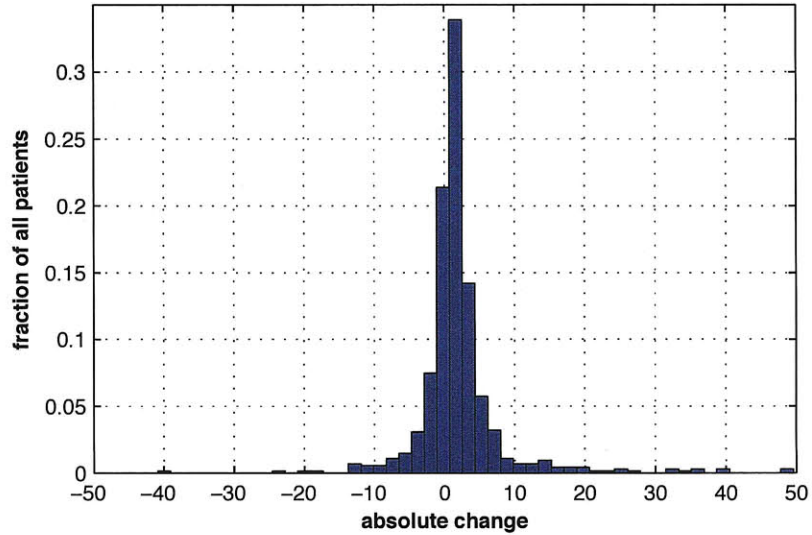
Figure 3.3 depicts a histogram of the absolute difference, which shows us how much the MV-DF value changes for each patient. The mean absolute difference is 2.0 and the standard deviation divided by the mean of the original MV-DF value—known as the coefficient of variation—is 0.14. In addition, the mean percent difference is 5%. This suggests that the online MV-DF value is on average higher than the original MV-DF value for a given patient. Using the distributions, we computed  $\tau_A$  and  $\tau_B$ . Using the original MV-DF values,  $\tau_A$  was computed to be 48.6. Because there is a positive bias in the difference between the online MV-DF and the original MV-DF values, we recomputed the threshold for risk stratifying the online MV-DF values ( $\tau_B = 50.9$ ) instead of setting  $\tau_B$  to  $\tau_A$ .

In Figure 3.3, we observe that the magnitude of the difference is large for a few of the patients. Table 3.1 shows the difference in the original MV-DF values and the online MV-DF values for the 20 patients with the largest absolute change in ranked order. There are a few possible sources for these large differences.

- (a) The first potential source is the number of five minute windows that are processed. The last two columns in Table 3.1 show the number of non-zero five minute windows that are processed by the original algorithm and the number of non-zero



**Figure 3.2:** The online MV-DF and original MV-DF distributions for 753 patients in the TIMI DISPERSE-2 study are similar, but not identical.



**Figure 3.3:** The histogram shows the distribution of the absolute change of the online MV-DF value. The ideal histogram is one where all patients have zero difference. In this histogram, the mean change is equal to 2.0.

five minute windows that are processed by the online algorithm. For 19 out of 20 patients, the online algorithm processed more five minute windows than the original algorithm. The reason for this difference lies in how signal rejection is performed. In the original algorithm, entire half-hour chunks are discarded when the signal is deemed too noisy, even if the noise is isolated to only parts of the signal. Because the online algorithm performs signal rejection on a window-by-window basis, more windows will be retained if noise is isolated.

- (b) Secondly, though both algorithms try to filter out noisy and ectopic beats, the algorithms produce different results because they have access to different portions of the signal at a time. The offline algorithm obtains a noise estimate by examining the entire 24 recording and may get a better overall view of noise. However, if the noise is not stationary, obtaining an estimate of noise using short windows may yield better local estimates.

One way to compensate for the first source of change is to discard the current window when the previous 30 minutes is too noisy or contains too few beats. While this idea is promising, the impact of this method on the number of windows retained and mean difference will need to be investigated in future work.

### 3.4.1 Evaluation of Risk Assessment

We calculated the  $c$ -statistic, using online MV-DF as the classifier and occurrence of death within 90 days as the outcome, to be 0.76. The  $c$ -statistic computed using the original MV-DF value as the classifier was computed to be 0.75. Since the  $c$ -statistics

**Table 3.1:** Patients with the largest absolute change in ranked order. The last two columns in the table show the number of non-zero five minute windows that is processed by the two algorithms.

Patient I.D.	Original MV-DF	Online MV-DF	Change	Original Window Count	Online Window Count	
1	468	47.8	97.3	49.5	210	297
2	481	47.6	95.3	47.8	144	299
3	466	110.4	69.6	-40.8	72	143
4	732	41.1	79.9	38.8	150	265
5	19903	49.7	88.5	38.8	198	288
6	20069	39.2	75.8	36.6	234	278
7	237	44.2	80.1	35.9	185	259
8	20150	75.2	109.9	34.7	288	296
9	20138	29.6	62.1	32.6	138	300
10	19975	89.3	120.9	31.7	114	297
11	20089	38.9	65.6	26.7	240	269
12	20065	49.3	74.5	25.3	222	259
13	19933	77.2	101.9	24.7	276	279
14	19892	67.9	91.7	23.8	246	299
15	19771	93.1	70.1	-23.0	210	228
16	19952	29.4	51.0	21.5	258	295
17	19954	91.0	111.6	20.6	246	232
18	19934	48.6	69.0	20.5	174	215
19	20094	148.1	127.7	-20.4	216	288
20	303	74.8	93.9	19.0	150	297

are approximately the same, one can conclude that the online MV-DF score has about the same likelihood of distinguishing a high risk patient from a low risk one as the original MV-DF for this set of patients.

We compared the risk stratification ability of the two different algorithms by computing  $\delta_A(i)$  and  $\delta_B(i)$  for all patients  $i$ . We then compared these values to the death outcome. Both algorithms correctly placed 571 out of 753 patients in the appropriate risk category. Of the 14 patients who died, ten were correctly identified as high risk. And of the 739 patients who did not die, 561 were correctly identified as low risk. Though the risk assignment for individual patients may differ, this result suggests that despite the differences in the computed MV-DF values, the online algorithm is as discriminative as the offline algorithm for the DISPERSE-2 population.

We also compared the risk assessments assigned by the two methods to each another. A total of 38 patients were classified differently—19 patients classified as low risk by the original algorithm were reclassified as high risk (relative false positives) and 19 patients classified as high risk by the original algorithm were reclassified as low risk (relative false negatives).

Figures 3.4 and 3.5 show the original and online MV-DF values for these patients. In both figures, the red squares represent the original MV-DF values, while the blue

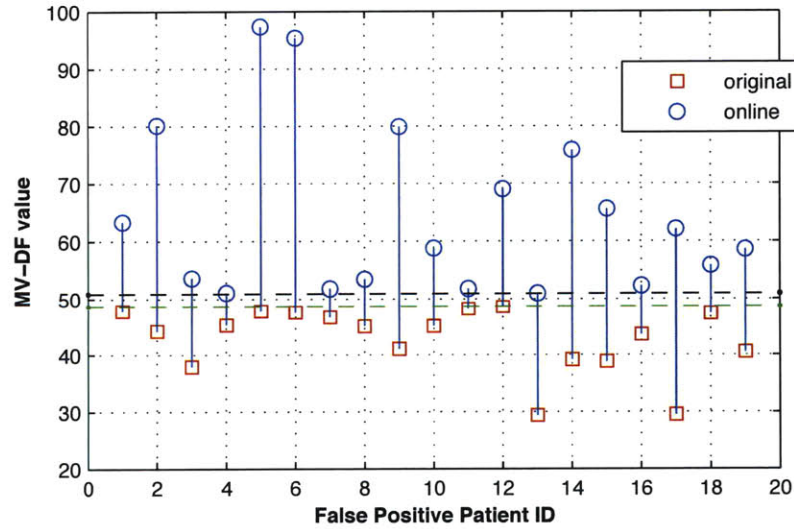
circles represent the online MV-DF value. The green dotted line marks  $\tau_A$  and the black dotted line marks  $\tau_B$ . In Figure 3.4, a large increase in MV-DF is likely due to the online algorithm retaining parts of the signal that were rejected as noise by the original algorithm. Noisy parts an ECG signal with dramatically changing beat-to-beat MD values will increase the per window MV values since the per window MV is concerned with energy in a specific frequency band.

For the relative false negative patients, the magnitude of the change in MD-value, is not as large. The reason for these changes is not well-understood. The figure also shows that, in some instances, even small changes in the MV-DF value can result in reclassification. The reclassification occurs because the online threshold  $\tau_B$  used to separate the population into high and low risk classes is greater than the offline threshold. Thus patients once above the line can now be below it.

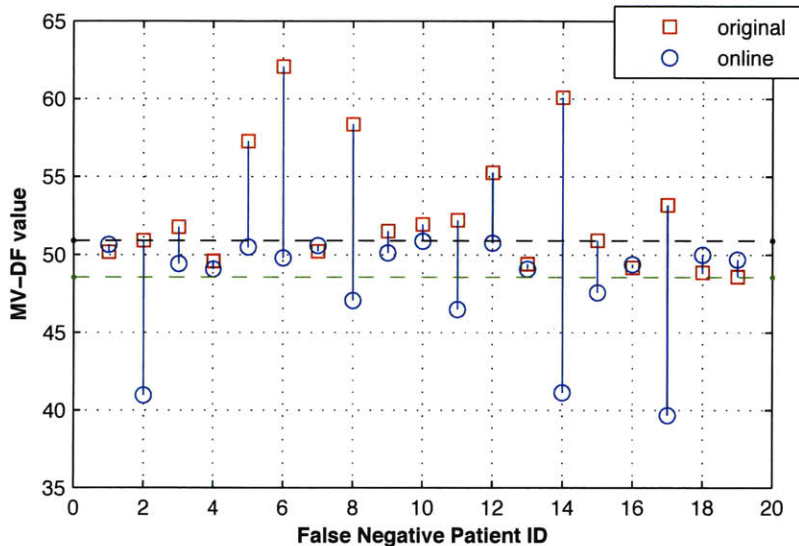
### 3.5 EFFECT OF DUTY CYCLING

One issue with the online MV-DF algorithm is that the processing performed by the algorithm, in particular dynamic time-warping, is computationally and thus, energy intensive. We use duty cycling to reduce the amount of processing consumption without dramatically affecting the diagnostic power of the algorithm.

In this section, we report the effect of applying random data independent duty cycling to the online MV-DF algorithm. We process each five minute window with a



**Figure 3.4:** Patients classified as high risk by the online MV-DF algorithm that were marked as low risk by the original MV-DF algorithm. The original MV-DF values are marked using red squares. For the most part, the significant change in MV-DF is likely due to the online algorithm retaining parts of the signal that were rejected by the original algorithm.



**Figure 3.5:** Patients classified as low risk by the online MV-DF algorithm that were marked as high risk by the original MV-DF algorithm. The original MV-DF values are marked using red squares. Several patients have minor changes in their MV-DF value. However, because the threshold value used for stratification is different, the patients are reclassified anyway.

probability  $p$  to achieve energy savings of approximately  $1 - p$ . We denote the algorithm that processes each window with a probability  $p$  as a  $p$ -duty cycled algorithm.

The duty cycling approach we describe and evaluate may not be the best possible duty cycling algorithm for this application. We introduce it only to show what is possible and to demonstrate an example of an application where it can be applied successfully. A duty cycling approach that is more tailored to the underlying application may result in more energy savings. After evaluating this approach on the MV-DF algorithm, we describe the necessary properties of applications for which a random duty cycling approach will work.

The subset of per window MV values that remain after duty cycling can be used to compute an estimate of the MV-DF value. Since MV-DF is computed as the 90<sup>th</sup> percentile of the per window MV values, if the subset of windows is a representative sample of the distribution of per window MV values, then the estimate given by a duty cycled version of the algorithm will have a small error.

To evaluate this method, we varied  $p$  from 0.05 to 1 in 0.05 intervals and computed the 90<sup>th</sup> percentile of the remaining windows for each value of  $p$ . We simulated the duty cycling algorithm ten times and took the median of the estimates. Let  $MV_B(i, p)$  be the median MV-DF value for patient  $i$  while duty cycling with a probability of  $p$ .

To determine how well the  $p$ -duty cycled algorithm approximates the output of the online algorithm, we computed the root mean square difference (RMSD) between

$MV_B(i, p)$  and  $MV(i)$ , the non-duty cycled MV-DF value of each patient, as

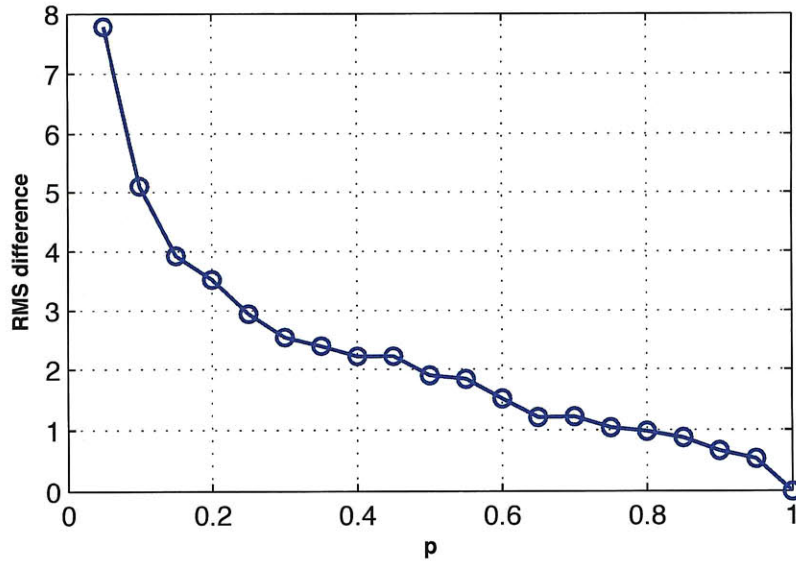
$$\sqrt{\frac{1}{N} \sum_{i=1}^N [MV_B(i, p) - MV(i)]^2}$$

where  $N$  is the number of patients.

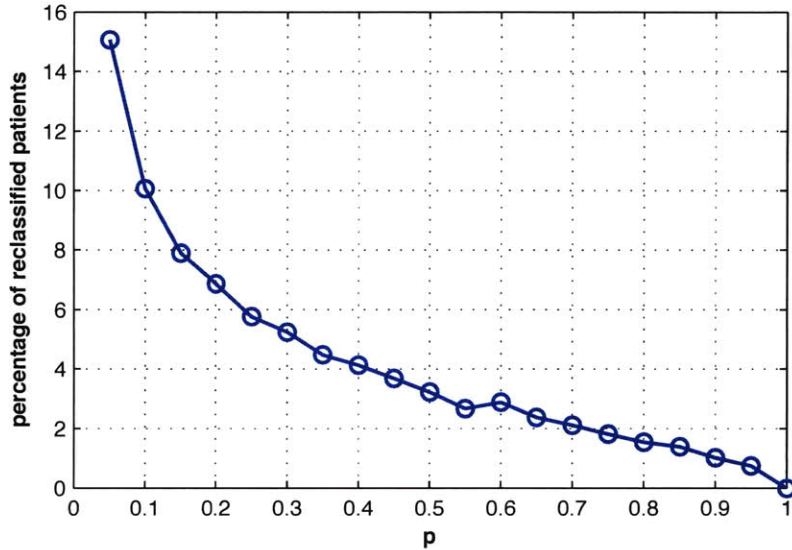
Figure 3.6 shows the RMSD as a function of  $p$ . As expected with  $p = 1$ , the difference drops to 0. At  $p = 0.6$ , the RMSD increases to 1.3.

The differences tell us the impact of duty cycling on the raw MV-DF value—it does not say much about how risk assessment is impacted. We compared  $MV_B(i, p)$  to the threshold  $\tau_B$  to obtain  $\delta_B(i, p)$ , the risk assessment of patient  $i$  using a  $p$ -duty cycled algorithm. We then compared  $\delta_B(i, p)$  to the non-duty cycled risk assignment  $\delta_B(i)$ . We use the term reclassification change to refer to a change that results when we compare the risk assignment of a  $p$ -duty cycled algorithm to the risk assignment of the non-duty cycled online algorithm. Figure 3.7 plots the reclassification change as a function of  $p$ . As expected, as  $p$  increases, the reclassification change decreases. Moreover, the shape of the curve essentially follows the error curve. At  $p = 0.6$ , the curve indicates that approximately 3% or 22 out of the 753 patients are reclassified relative to the non-duty cycled version of the algorithm. A  $p$  of 0.6 implies that we can achieve energy savings of approximately 40%.

Finally, we computed the classification error, which is determined by comparing the risk assignment  $\delta_B(i, p)$  to the death outcome of each patient. Figure 3.8 plots the classification error as a function of  $p$ . For reference, the classification error of the



**Figure 3.6:** The root mean squared difference (RMSD) of the duty cycling algorithm decreases exponentially with increasing  $p$ . At  $p = 0.6$ , the RMSE is 1.3.



**Figure 3.7:** The fraction of patients that are reclassified compared to the non-duty cycled algorithm as a function of  $p$ . At  $p = 0.5$ , only 22 patients have been reclassified.

original MV-DF algorithm is shown as a dotted black line. The duty cycled algorithms use the  $\tau_B$  threshold for dividing the population into the two risk classes while the original MV-DF algorithm uses  $\tau_A$ . The graph shows that the classification error of the duty cycled algorithm quickly converges to the classification error of the original algorithm for small  $p$ .

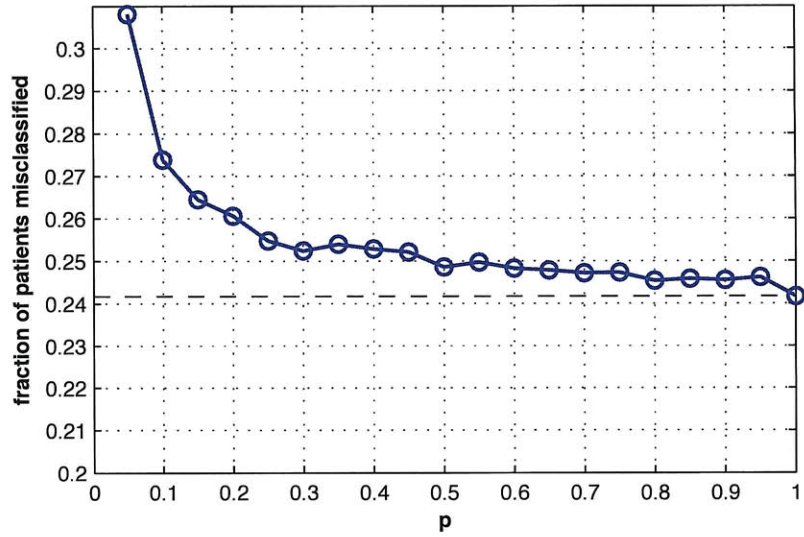
### 3.6 EVALUATION ON TIMI MERLIN PATIENTS

To validate the performance results of the online MV-DF algorithm and duty cycled versions of the online algorithm, we repeated the experiments on 800 randomly selected patients from the placebo group of the TIMI MERLIN study.

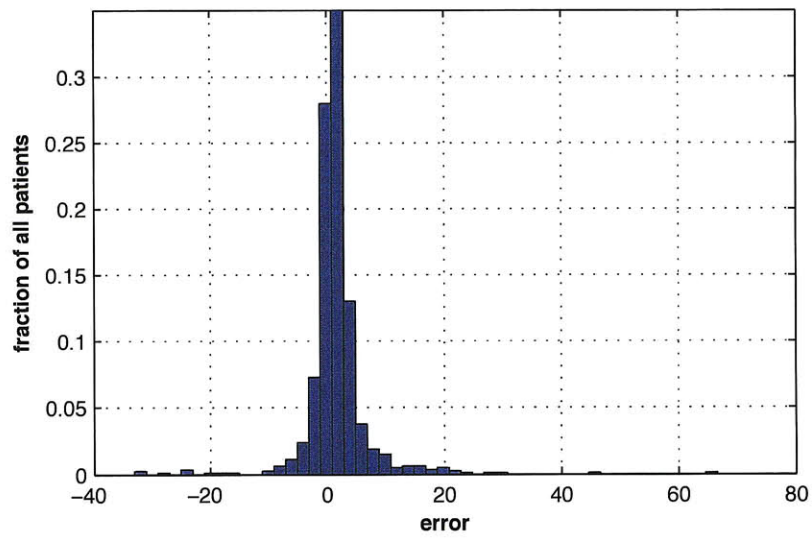
First, we compared the histograms of the original and online MV-DF values to verify that the online MV-DF could be used to get good approximations of the original MV-DF value. The error distribution in Figure 3.9 shows a non-zero, but small average error (1.7).

Next, we computed the  $c$ -statistics of the original and online algorithm on the TIMI MERLIN database to be 0.76 and 0.73 respectively. Again, the two algorithms seem to have similar discriminative power.

We then compared the MV-DF values to the thresholds derived from the TIMI DISPERSE-2 data set ( $\tau_A = 48.6$ ,  $\tau_B = 50.9$ ) to obtain the risk assignments  $\delta_A(i)$  and  $\delta_B(i)$  for all patients  $i$  in the TIMI MERLIN cohort. We then compared these risk assessments to the death outcome of each patient. The offline algorithm assigned 606



**Figure 3.8:** The classification error of the duty cycled algorithm quickly converges to the classification error of the original algorithm as  $p$  increases.



**Figure 3.9:** The histogram shows the distribution of the absolute error of the online MV-DF value computed per patient from the TIMI MERLIN study. The mean of the distribution is 1.7.



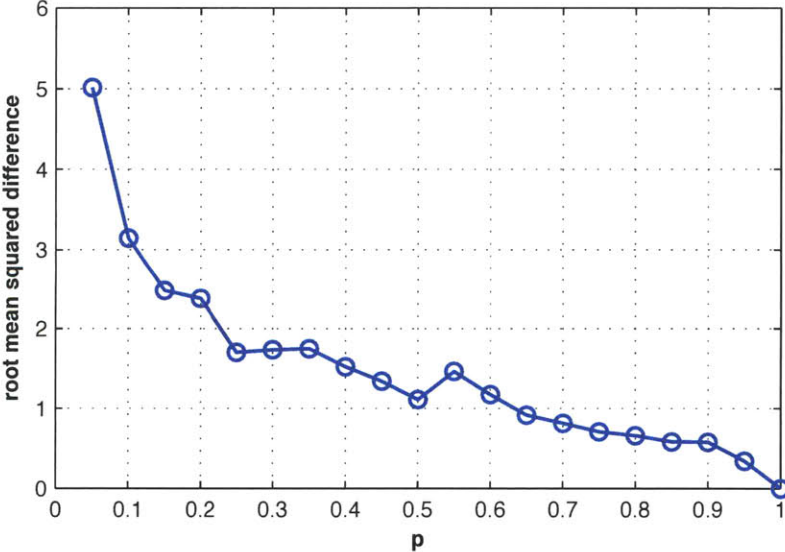
out of 800 patients to the appropriate risk level, while the online algorithm assigned 612 out of 800 patients to the appropriate risk level.

**3.6.1 Effect of Duty Cycling on TIMI MERLIN patients**

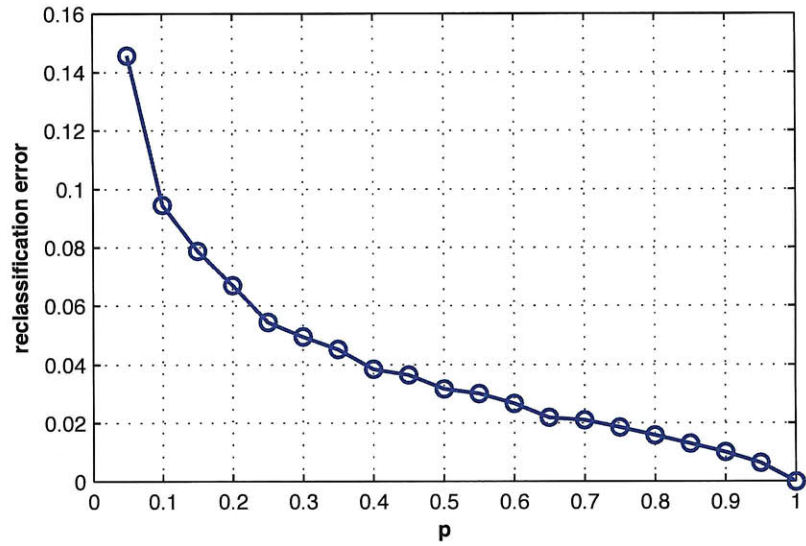
The random duty cycling strategy outlined in Section 3.5 was also applied to patients in the TIMI MERLIN data set. Figure 3.10 shows the root mean squared difference between the duty cycled online algorithm and non-duty cycled algorithm on the TIMI MERLIN data set as a function of  $p$ .

The impact of duty cycling on the risk stratification of the TIMI MERLIN patients is shown in Figure 3.11 where the reclassification error is plotted as a function of  $p$ . Recall that the reclassification error compares the risk assignment  $\delta_B(i, p)$ , determined by comparing  $MV_B(i, p)$  to the threshold  $\tau_B$ , to ground truth. Ground truth for each patient  $i$  is taken to be  $\delta_B(i)$ , the risk assignment resulting from comparing the non-duty cycled online MV-DF value to  $\tau_B$ . The general shape of the curve is similar to that achieved on the TIMI DISPERSE-2 data set and gives us confidence that the duty cycling approach is applicable to other patients and that we have not tailored the algorithm to a specific population.

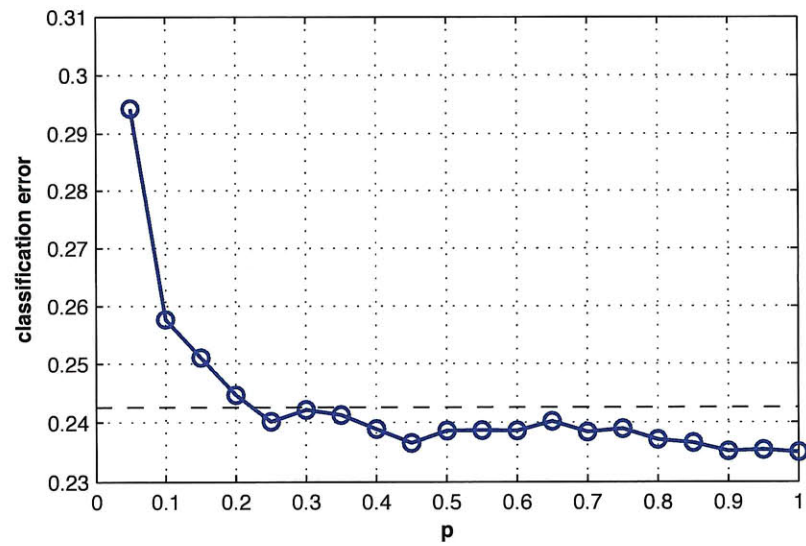
Finally, we computed the classification error by comparing the risk assignment resulting from each  $p$ -duty cycled algorithm to the death outcome of each patient in the TIMI MERLIN cohort. Figure 3.12 plots the classification error as a function of  $p$ . As before, the classification error of the original MV-DF algorithm is shown as a dotted black line. In the graph, we observe a quick convergence of the classification error of the duty cycled algorithm. However, since the classification error (188/800)



**Figure 3.10:** The root mean squared difference between various  $p$ -duty cycled algorithms and the non-duty cycled algorithm as a function of  $p$  for the TIMI MERLIN cohort.



**Figure 3.11:** The fraction of TIMI MERLIN patients that are reclassified compared to the non-duty cycled algorithm as a function of  $p$ .



**Figure 3.12:** As in the TIMI DISPERSE-2 population, the classification error of the duty cycled algorithm converges quickly for the patients in the TIMI MERLIN cohort.

of the online algorithm is slightly lower than the classification error (194/800) of the original algorithm on the TIMI MERLIN cohort, as  $p$  approaches 1, the classification error begins to dip below the error achieved by the original algorithm.

### 3.7 DISCUSSION

In this chapter, we have demonstrated how random data independent duty cycling can be used to reduce the number of windows processed by an online ECG-based risk stratification algorithm. Adding duty cycling did not substantially change the diagnostic capabilities of the original algorithm.

Random duty cycling worked well for our application because the goal of risk stratification is not to isolate a particular event in time, but rather to summarize a distribution using a statistical value, in this case, the 90<sup>th</sup> percentile of the per window MV values. As long as a sufficient number of samples is acquired to accurately model the underlying distribution, random duty cycling can be an easy and effective approach to reducing energy consumption for long-term medical monitoring applications. We believe that this method of duty cycling can be used to reduce energy consumption for other applications where localizing an event in time is not critical. One such medical application is event enumeration, *e.g.*, counting the number of ectopic beats per hour. In event enumeration, the frequency and number of events is more important than the occurrence of a single event in time. If event frequency is sufficiently high, then sampling or duty cycling can be employed to reduce the amount of data processed without degrading event count estimation. If the cost of detecting an event occurrence is high, then this duty cycling approach will result in energy savings.

Our approach can also be used to reduce the energy consumption of a portable ECG monitoring system that collects and stores the data for later analysis. We have shown that only a subset of the data is needed to obtain an estimate of the MV-DF value. Thus, we can safely ignore segments of the input stream by not sampling or storing the data. Since the energy cost of storing data can be significant [31], this approach can reduce the energy consumed by a monitoring device.



---

## Automatic Sensor Selection

In medical monitoring applications, physiological data is often gathered using multiple sensors. For example, in ECG monitoring, multiple leads are used to capture electrical signals generated by the heart. For gait analysis, to monitor an individual's movement patterns, patients can be instrumented with a collection of accelerometers, gyroscopes, and tilt sensors placed at different locations on the body. In EEG monitoring, the electrical signals of the brain are measured using multiple electrodes.

To enable long-term ambulatory medical monitoring applications that use multiple sensors, designers of portable medical devices must address two key issues: patient comfort and convenience. Several factors can have an impact on these issues including:

- (a) *Number of sensors.* Reducing the number of sensors that are worn on the body can improve the comfort of a wearable device. For implantable devices, reducing the number of sensors could simplify the surgical implanting procedure.
- (b) *Number of wires.* Using wires to connect sensing components to computation components is cumbersome for a patient to wear.
- (c) *Device size and weight.* To improve long-term comfort, the size and weight of the overall device should be minimized as much as possible.
- (d) *Battery lifetime.* If a device is implanted or worn continuously, frequent recharging of the battery or frequent battery replacement is inconvenient. Thus, reducing the energy consumed by the processing or collecting of data is critical. One additional benefit of reducing overall energy consumption is that battery size and weight could also be decreased.

In this chapter, we focus on the problem of selecting sensors for a real-time portable medical monitoring application: *epileptic seizure onset detection*. An epileptic seizure is a transient symptom of excessive or synchronous neuronal activity in the brain. The EEG recorded during a seizure is said to be *ictal*. Seizure onset is defined as the point in time where a patient's EEG transitions from a non-ictal to an ictal state. A seizure is the period in time following this event and typically lasts from

a few seconds to a few minutes. As mentioned in Chapter 2, giving patients who have uncontrolled seizures a portable seizure monitor that can alert caregivers or the patient in a response to a detected seizure, can have many benefits. For such a device to be useful, however, seizures need to be detected reliably with low latency. In addition, the number of false detections must be minimized to avoid alarming caregivers and the patient unnecessarily.

To select channels, we use a machine learning based algorithm that constructs reduced channel seizure onset detectors starting from an existing algorithm for constructing 18-channel patient-specific detectors. The algorithm we use for selecting channels is based on the wrapper approach [25], a feature selection technique. Since EEG signals of epilepsy patients can vary considerably, channels are selected on a patient-by-patient basis. Indeed, Shoeb *et al.* [43]. have demonstrated that a seizure onset detector trained using data from an individual patient will tend to have better sensitivity and specificity than a non-patient specific detector.

Using our approach, the average size of the selected channel subsets was reduced from 18 to 6.4. These channel subsets were used to build reduced channel seizure onset detectors. When we ran these detectors on recorded EEG data, we observed only a small degradation in performance compared to the original 18-channel detector. The number of seizures detected decreased slightly from 139 to 136. However, for 10 out of the 16 patients, the reduced channel detectors identified all of the seizures detected by the original algorithm. The median detection latency increased slightly from 6.0 to 7.0 s, while the average number of false events per hour increased slightly from 0.07 to 0.11.

To determine the impact of reducing the number of channels on energy consumption, we developed a simple energy model to relate energy consumption to the number of channels used by a seizure onset detector. The parameters for the model were derived by measurement of a prototype implementation of an ambulatory EEG monitoring device. Using this model, we determined that our approach reduced the energy consumption by an average of 60%—an increase in battery lifetime of  $2.5\times$ .

The rest of this chapter is organized as follows. First, we describe the original 18-channel patient-specific seizure onset detection algorithm used to detect seizures. Next, to understand the impact of reducing channels on energy consumption, we develop an energy model that relates the number of channels to energy consumption.. We then describe our approach for building reduced channel seizure onset detectors. Finally, we evaluate our method on recorded EEG data from several patients by comparing the performance of our subset detectors to the baseline classifier that uses all channels.

#### 4.1 NOTATION

Throughout this chapter, we will often use the following notation to simplify the discussion.

Let  $X$  be the space of possible unlabeled examples, and  $Y$  be the set of possible labels. Each example  $\mathbf{x} \in X$  is a vector of  $m$  input features or variables, *i.e.*,  $\mathbf{x} = \langle x_1, x_2, \dots, x_m \rangle$ . Let  $D = \{d_1, d_2, \dots, d_n\}$  be a data set consisting of  $n$  labeled instances where  $d_i = \langle \mathbf{x}_i \in X, y_i \in Y \rangle$ .

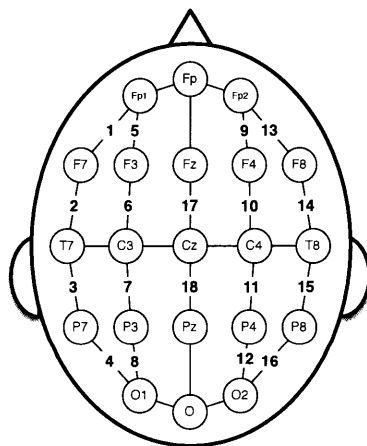
A classifier  $C$  maps an unlabeled instance  $\mathbf{x} \in X$  to a label  $y \in Y$ . A detector is a classifier where the number of unique elements in  $Y$  is two. An inducer or learning algorithm  $I$  maps a data set  $D$  to a classifier  $C$ ; it is used to construct a classifier from a given labeled data set.

To refer to a specific classifier, we will often use the notation  $I(D)$  instead of merely  $C$  since the former notation allows us to track the training data that was used to construct a specific classifier. In addition, we use the notation  $I(D, \mathbf{x})$  to denote the label assigned to an unlabeled instance  $\mathbf{x}$  by the classifier built by inducer  $I$  on data set  $D$ , that is,  $I(D, \mathbf{x}) = (I(D))(\mathbf{x})$ .

#### 4.2 A SEIZURE ONSET DETECTION ALGORITHM

Continuously monitoring individuals with epilepsy using a portable, monitoring device can offer a number of benefits. The monitor can warn patients of seizure onset before symptoms cause injury or it could be used to initiate delivery of a therapy. Current state-of-the-art seizure onset detection algorithms require 21 scalp electrodes to be worn by the patient. The electrodes are used to generate 18 to 20 data streams called channels that are analyzed with software.

To detect seizures, scalp EEG data is acquired using electrodes arranged on the scalp as shown in Figure 4.1. EEG data is typically sampled at 200 to 250 samples per second. Researchers have used machine learning to construct seizure detection



**Figure 4.1:** The 10/20 montage arrangement of scalp electrodes used for EEG monitoring. A channel is formed by measuring the difference between two adjacent electrodes (e.g. P7-O1). Instead of using the electrode names, we will frequently refer to the channels using the numbers indicated in the diagram. In this thesis, only the channels that are labeled with a number are used.

algorithms from data collected using similar montages [40, 44]. Typically, several hours of EEG data are recorded from multiple channels for each patient. The data is

then annotated manually to mark salient features in the recording, such as the start and end of each seizure.

#### 4.2.1 The Patient-Specific SVM-Based Detector

Shoeb *et al.* constructed patient-specific support vector machine (SVM) based classifiers to detect seizure onsets in real-time [44]. In their paper, the authors showed how their algorithm could be used to initiate a delay-sensitive procedure following the detection of seizure onset.

To construct a patient-specific algorithm, first, EEG data from a single patient was divided into two second windows. Using the annotations marking the location of seizure onsets, each window was assigned a label of +1 if the window was considered part of seizure onset and  $-1$  if it was a non-seizure window. Seizure windows were left unlabeled. Next, features were extracted from each labeled window on a per channel basis for all 18 channels. Seven elements corresponding to the energies in overlapping frequency bands were extracted from each window. The frequency bands chosen were 3 Hz wide and collectively spanned the 0–20 Hz range. The seven features from all channels were concatenated to form a 126 dimensional feature vector  $\mathbf{x}$ . Each feature vector  $\mathbf{x}$  was paired with its assigned label  $y$  to form a data set  $D$ . With  $D$  as input, a support vector machine  $\mathbf{C} = \mathbf{I}(D)$  was trained to determine whether an observed two second EEG window resembles an individual’s seizure onset EEG window or not.

This classifier was then used as a component of a real-time algorithm to detect previously unseen seizure onsets. Every second, features are extracted from two seconds of EEG samples, for each channel separately. A 126 element feature vector  $\mathbf{x}$  is constructed and passed to the classifier  $\mathbf{C}$  for labeling. Any epileptiform activity lasting more than four seconds is declared as the start of a seizure.

### 4.3 EVALUATION METRICS

In this section, we define a few key metrics that we will use to quantify the performance of the detectors. For our purposes, a window is defined as two seconds of consecutively sampled EEG data.

- (a) *false negative rate*: The false negative rate refers to the fraction of true positive windows that are misclassified as negative windows.
- (b) *false positive rate*: The false positive rate refers to the fraction of true negative windows that are misclassified as positive windows.
- (c) *fraction of seizure onsets detected*: The fraction of seizure onsets detected is the number of seizures correctly declared over the total number of seizures present in a recording. All detectors we consider in this chapter will declare seizure onset when the underlying classifier finds four consecutive windows that contain seizure activity. Note that a high false negative rate may not necessarily translate into a low fraction of seizures detected, since seizures typically last longer than five seconds.



- (d) *false events per hour (false alarm rate)*: If a seizure onset is declared during a period that does not contain an actual seizure, we call this declaration a false alarm or false event. Note that a high false positive rate may not necessarily result in a high false alarm rate.
- (e) *detection latency*: The detection latency is the difference in time between when a detection algorithm declares seizure onset and the electrographic onset time of the seizure as marked by a human expert. The detection latency depends in part on the minimum number of consecutive positive windows that needs to be labeled before a seizure is declared. Thus, if we reduce the number of consecutive positive windows required before declaring an event, the detection latency would decrease.
- (f) *energy consumption*: The energy required to label an EEG recording. In this chapter, the energy consumed by our algorithms is determined using a model based on our prototype hardware.

The seizure onset detection algorithm described in [44] was able to achieve high specificity, high sensitivity, and low latency for many of the patients in their study. We use this detector’s performance and energy consumption as a baseline for comparison purposes throughout this chapter.

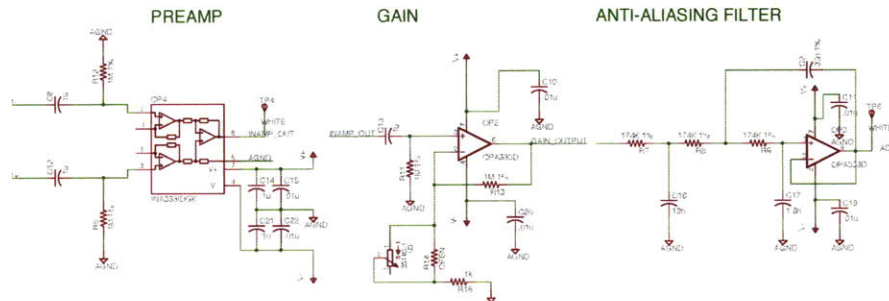
#### 4.4 HARDWARE PROTOTYPE OF A WEARABLE SEIZURE MONITOR

We built a prototype using off-the-shelf parts and measured the energy consumption of the main components of the system to develop a model that relates energy consumption of an ambulatory monitor to the number of channels used.

The prototype transmits data to a remote device where more intensive computations can be performed. Our prototype is an instance of the LOCALIZED architecture as described in Section 2.4. While the system is capable of acquiring data from more than one channel, for the purposes of developing an energy model, building a single channel acquisition system is sufficient. Since the system is designed for wearable monitoring, we made an effort to select low-power, lightweight components.

In the prototype system, signals from two scalp electrodes are input into an instrumentation operational amplifier (in-amp) which amplifies the signal difference and rejects the DC and AC common-mode signals present on both channels [24]. In the prototype, we chose the Texas Instruments INA333 low-noise, low-power in-amp, which has a 100 dB minimum common-mode rejection ratio and draws no more than 80  $\mu A$  to amplify the differential signal. We set it to amplify the signal by five times. After passing the signal through an RC high-pass filter with a cutoff below one Hz, the signal is then amplified by an additional factor of 100 using a single gain stage. Additional amplification is needed because EEG signals gathered using scalp electrodes have a peak-to-peak amplitude of 5 to 300  $\mu V$  [36]. The signal is then anti-aliased using a third order Butterworth filter with a cutoff frequency of 100 Hz. Coupled with the  $1/f$  filtering provided by the scalp, this filter provides sufficient anti-aliasing. Figure 4.2 shows the schematic of the analog portion of the design.

After filtering, the signal is sampled at 200 samples per second using the on-board 12-bit analog-to-digital converter (ADC) of a microcontroller, the Texas Instruments



**Figure 4.2:** Schematic of single channel EEG analog signal acquisition circuit.

MSP430FG4618 [33], operating at 8 MHz at 3.0 V. The output of the analog frontend is connected to the MSP430FG4618 present on the MSP430 Experimenter's Board. After sampling, each data point is truncated to 8 bits. Once a full second's worth of data has been collected, *i.e.*, 200 bytes, the data is split into 50 byte packets and delivered to a CC2500 transceiver [9], a 2.4 GHz radio. The data is split because the size of the transmit and receive FIFOs in the radio is 64 bytes. Along with the payload data, an 8-bit sequence number is also sent to help with detecting packet loss and ordering the incoming packets. Data is transmitted at 250 kbps using minimum shift-key (MSK) modulation and forward error correction. A photo of the prototype system is shown in Figure 4.3.

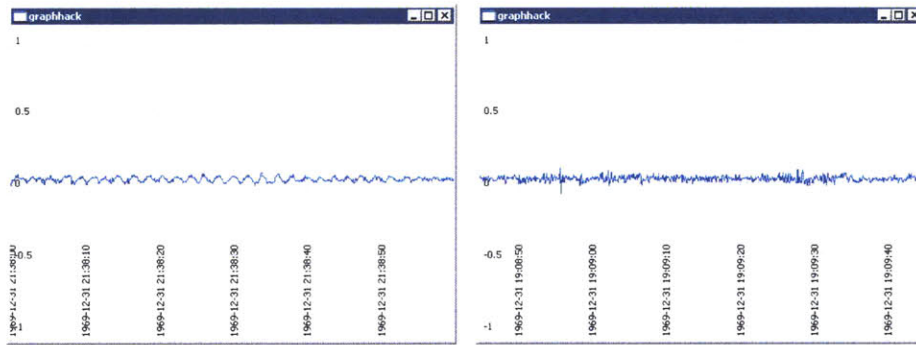


**Figure 4.3:** Prototype single channel EEG acquisition system

In our prototype, the data is delivered to a computer via an EZ430-RF2500 module, which consists of a CC2500 transceiver as well as another MSP430. The module stores a full second's worth of data before sending the data via a USB serial port to the computer, where further processing of the signals can be performed. In a deployment,

the data would be transmitted to a device located at the patient’s belt.

Figure 4.4 shows two screenshots of a single channel (O1-REF) of EEG captured using electrodes placed on the author’s scalp. In Figure 4.4(a), the screenshot on the left, alpha wave patterns are being captured by the system. Alpha waves are electromagnetic oscillations with a frequency of approximately 10 Hz that arise from synchronous and coherent electrical activity of thalamic pacemaker cells in the human brain. Typically, alpha waves appear when the subject closes his or her eyes and enters a state of relaxation. Figure 4.4(b) shows what happens when the subject opens his eyes—the alpha waves disappear.



(a) Eyes closed; alpha waves present

(b) Eyes open; alpha waves absent

**Figure 4.4:** Screenshots of a single channel (O1-REF) of EEG using electrodes placed on the author’s scalp. The date shown in both screenshots is incorrect.

#### 4.4.1 Energy Measurements

To measure the power consumption of the three main components in the system, sense resistors were placed in series with the power supply of each of the components. By measuring the voltage drop across a resistor using an oscilloscope, the current draw and power consumption can be determined using Ohm’s Law.

In our prototype, the data acquisition circuit remains fully powered at all times. Duty cycling the front-end is possible, however, aggressive duty cycling may lead to data corruption. When powered on, the data acquisition circuit has a current consumption of 100  $\mu\text{A}$  at 3.0 V. The energy needed to acquire  $N_c$  channels of data can be modeled as

$$E_{\text{sample}} = N_c P_{\text{sensor}}^{(\text{on})} T_{\text{sensor}}^{(\text{on})} \quad (4.1)$$

where  $T_{\text{sensor}}^{(\text{on})}$  is the time the data acquisition unit is active and  $P_{\text{sensor}}^{(\text{on})}$  is the active power consumption of the sensor.

The signal is sampled at 200 samples per second using the onboard ADC of the MSP430. We can place the CPU into Low Power Mode 0 (LPM0) during idle periods since we configured the ADC to use an on-board timer clocked by the SMCLK signal

to trigger conversions. Note that a lower power mode can be achieved by taking advantage of other sampling features of the MSP430. Sampling a single EEG channel can be accomplished by operating the ADC in single-channel, repeated conversions mode; multiple channels can be sampled by using the ADC in multiple-channel, repeated conversions mode. Doing so will increase the on-time of the ADC, but the increase will have a negligible impact on the energy consumption because power consumed by the ADC is small compared to the CPU operating in LPM0 mode.

The energy consumed by the MSP430 can be approximated as:

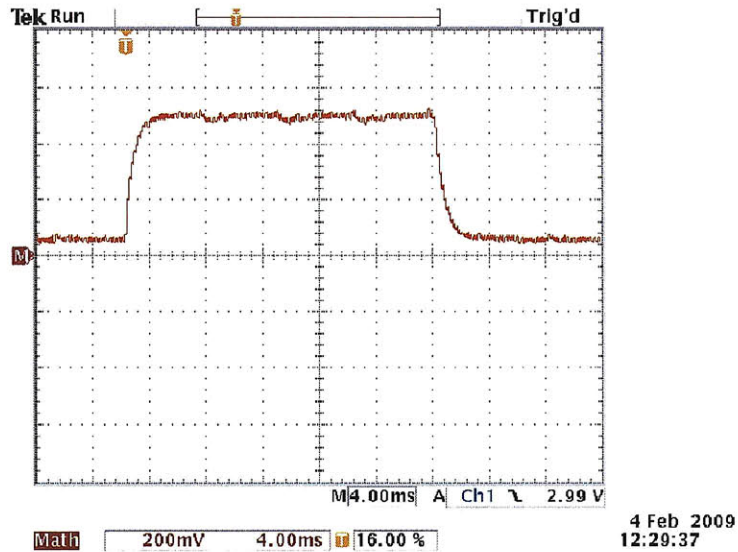
$$E_{\text{CPU}} = P_{\text{CPU}}^{(\text{on})} T_{\text{CPU}}^{(\text{on})} + P_{\text{CPU}}^{(\text{idle})} T_{\text{CPU}}^{(\text{idle})}$$

where  $T_{\text{CPU}}^{(\text{on})}$  and  $T_{\text{CPU}}^{(\text{idle})}$  are the durations for which the CPU is active and idle respectively and  $P_{\text{CPU}}^{(\text{on})}$  and  $P_{\text{CPU}}^{(\text{idle})}$  are the power consumed by the CPU when active and idle. In our system, the CPU is activated once per second to transmit data. Therefore, for every one second interval, the CPU is idle for  $1 - T_{\text{CPU}}^{(\text{on})}$  seconds. The energy consumed in one second by the CPU can be written as:

$$E_{\text{CPU}} = P_{\text{CPU}}^{(\text{on})} (N_c T_{\text{active}}) + P_{\text{CPU}}^{(\text{idle})} (1 - N_c T_{\text{active}}). \quad (4.2)$$

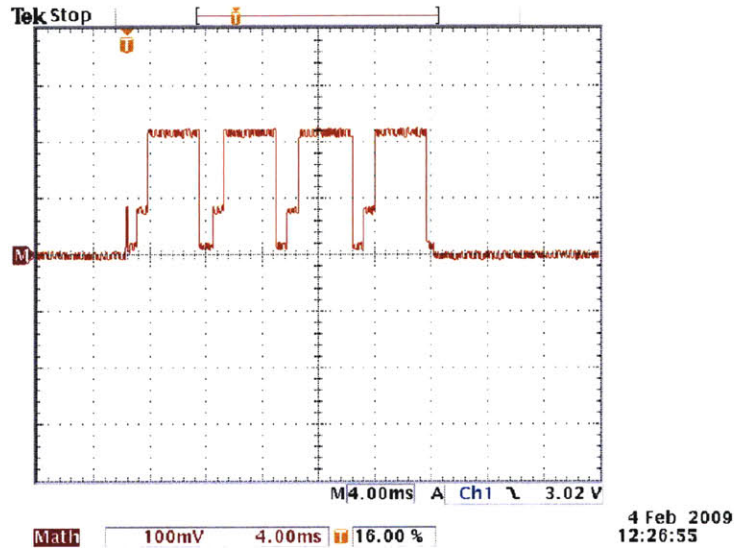
where  $T_{\text{active}}$  is the duration of time that the radio is active when sending one channel's worth of data and  $N_c$  is the number of channels.

Figure 4.5 shows the change in voltage across the sense resistor as the CPU transitions from idle to active state. We use this screenshot to determine the active and idle power consumptions of the CPU.



**Figure 4.5:** The change in voltage across the  $110 \Omega$  CPU sense resistor as the MSP430 transitions from idle state to active state was 60 mV. Using Ohm's Law, the current consumption is equivalent to  $550 \mu\text{A}$ .

To transmit data to a nearby remote device (*e.g.*, on the belt), we use the CC2500 [9] transceiver. Figure 4.6 shows a trace of the radio sending 200 bytes of data and is used to determine the power consumption of the radio in various states of operation. There are several things to note. First, the transmission of each 50 byte



**Figure 4.6:** A trace of the voltage change across the 10 Ω resistor used to sense the power consumption of the CC2500. The CC2500 is transmitting data from a single channel of EEG collected over a one second window. The total size of the payload data is 200 bytes. Because of the limited size of the CC2500 transmit FIFO, we transmit 50 bytes at a time.

packet takes about 3.7 ms. We pause between each packet to give the receiver a chance to process the previous packet of data. Also, notice that the radio transitions between various power states in order to transmit. We can identify five different power states from the diagram.

Initially, the radio is in *idle* state, while it waits for data. When the CPU needs data to be transmitted, it puts the radio in a *wake-up* state and then transfers data from its memory to the transmit FIFO on the radio during the *transfer* state. Once enough data has filled the FIFO, the radio enters a *transmit* state, during which data is delivered out of the radio front-end. The radio then goes into a *listen* state while it gives time for the receiver to process the just sent packet. The radio cycles between the transfer, transmit, and listen states, until all the data in the CPU memory has been transmitted. Then, the radio returns to the idle state.

As more channels are sampled, more data will need to be sent. However, only the time spent in the transmit, wait, and transfer states will increase. The wake-up cost and idle time are constants. Therefore, during one second, the energy consumed by

the radio can be modeled as

$$\begin{aligned}
 E_{\text{radio}} &= \sum_{i \in R} P_{\text{radio}}^{(i)} T_{\text{radio}} \\
 &= N_c \sum_{j \in S} P_{\text{radio}}^{(j)} T_{\text{radio}}^{(j)} + \sum_{k \in T} P_{\text{radio}}^{(k)} T_{\text{radio}}^{(k)}
 \end{aligned} \tag{4.3}$$

where  $S = \{\text{transmit, wait, transfer}\}$ ,  $T = \{\text{idle, wake-up}\}$  and  $R = S \cup T$ . The duration of time the radio spends in the idle mode in one second is simply one minus the sum of the duration spent in the other modes.

Table 4.1 summarizes the values of the parameters of components used in our prototype system. Our model does not take into account the energy needed to switch from idle state to on state, but this energy is negligible. Combining Equations 4.1, 4.2, and 4.3, and using the values in Table 4.1, we can compute the energy consumed to collect and transmit one second's worth of data as a function of the number of channels  $N_c$ :

$$E(N_c) \approx 1.58 \text{ mJ} \cdot N_c + 2.12 \text{ mJ}. \tag{4.4}$$

Therefore, a system that uses 18 channels has a power consumption of 30.6 mW. If we are able to reduce the number of channels required by a seizure onset detector to one channel, we would consume 3.71 mW, a reduction of 88%.

#### 4.4.2 Alternative Architectures

As we have shown, varying the number of channels in a system that performs no local computation can have a huge impact on energy consumption. Using a different

**Table 4.1:** Measured values for the parameters used in our energy model. The supply voltage for all the components was 3.0 V. The dominant source of power consumption is the radio during transmit and transfer modes.

Component	Symbol	Name	Value
EEG sensor	$P_{\text{sensor}}^{(\text{on})}$	sensor power consumption	300 $\mu\text{W}$ @ 3.0 V
MSP430	$P_{\text{CPU}}^{(\text{on})}$	active power consumption	14 mW @ 8 MHz
	$P_{\text{CPU}}^{(\text{idle})}$	idle power consumption	1.6 mW in LPM0
	$T_{\text{active}}$	active time	23.2 ms
CC2500	$P_{\text{radio}}^{(\text{transmit})}$	transmit power consumption	64 mW @ 0 dBm
	$P_{\text{radio}}^{(\text{wait})}$	wait power consumption	4.8 mW
	$P_{\text{radio}}^{(\text{transfer})}$	transfer power consumption	24 mW
	$P_{\text{radio}}^{(\text{wake-up})}$	wake-up power consumption	24 mW
	$P_{\text{radio}}^{(\text{idle})}$	idle power consumption	0.5 mW
	$T_{\text{radio}}^{(\text{transmit})}$	transmit time	14.7 ms
	$T_{\text{radio}}^{(\text{wait})}$	wait time	0.64 ms
	$T_{\text{radio}}^{(\text{transfer})}$	transfer time	3.2 ms
$T_{\text{radio}}^{(\text{wake-up})}$	wake-up time	0.05 ms	

architecture to implement a portable monitoring system changes the ratio between the per-channel energy consumption and fixed energy consumption and therefore, affects the energy savings achievable using channel selection. Figure 4.7 plots the energy consumed per second by three different portable monitoring architectures as a function of number of channels. We only consider the energy consumed by the device mounted on the scalp.

The REMOTE architecture line shows the energy consumption of the prototype we just described. All EEG data is sent to a remote device where the computation can take place.

To reduce the energy consumed by the radio, we can perform the feature extraction locally at the head. Unfortunately, the MSP430 is too underpowered to perform the filtering necessary to extract features. To do this computation, we can either use an application-specific integrated circuit (ASIC) to acquire, filter, and process the EEG stream or we can use a more sophisticated, but higher-power CPU, such as the Marvell PXA270.

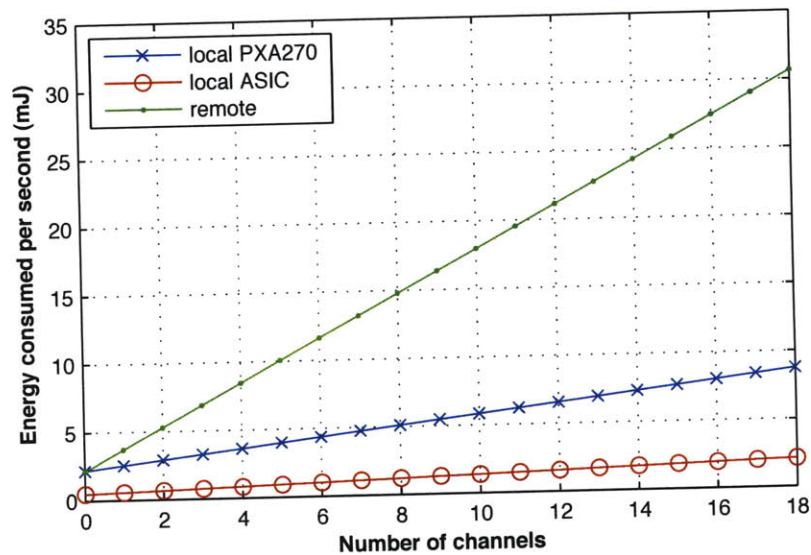
The PXA270 can operate at 104 MHz while consuming 116 mW. In standby mode, 1.7 mW are consumed. Based on our estimates, the PXA270 can extract features for a single channel using tens to hundreds of microseconds. After features are extracted, seven 16-bit values would be transmitted per channel. Since the total time needed to transmit and extract features is a small fraction of the window duration, most of the energy is consumed while the radio and CPU are idle. Thus, reducing the number of channels, which would reduce active power consumption, would have little impact on overall power consumption.

If we used an ASIC to do local computations, we could use a design that integrates the instrumentation amplifier, ADCs, and the feature extraction processor for a single channel into a single chip [57]. Each chip computes at a clock rate that allows the features for each two second window to be generated just before the next two second's worth of data needs to be gathered. This reduces the idle power consumption of the system to a minimum.

To monitor 18 channels, 18 of these circuits could be combined. Table 4.2 shows the power consumption of a wireless 18-channel EEG acquisition system based on combining 18 of these circuits. After features have been extracted for each channel, the features would need to be transmitted to a remote device for classification. Either each circuit would be equipped with a separate radio or the data would be combined at a central point where a single radio would be used to transmit the features from all channels.

**Table 4.2:** Power consumption of an optimized wireless EEG acquisition chip [57].

Component	Power consumption ( $\mu$ W)
I-Amp	72
ADC	3
Digital processor	2.1



**Figure 4.7:** Energy consumption per channel versus number of channels for various portable monitoring systems. Though energy savings are still achievable, performing all processing locally reduces the impact of channel reduction on overall energy consumption.

From the lines marked LOCAL PXA270 and LOCAL ASIC in Figure 4.7, we see that performing feature extraction locally reduces per-channel energy consumption. However, reducing the number of channels has a smaller effect on overall energy consumption because the fixed power consumption becomes dominant. When we reduce the number of channels, the time spent actively computing is reduced and idle time increases. To mitigate this effect, dynamic voltage scaling could be employed. When fewer channels are used, instead of idling, we can reduce the voltage and CPU clock rate. This can result in tremendous energy savings since a linear scaling of voltage results in a quadratic reduction in power consumption.

Instead of only extracting features locally, we can perform classification on the head as well. However, reducing the number of channels has a marginal effect on the classification computation. Computing the radial basis kernel requires computing an exponential—typically a compute intensive operations—which is independent of the number of channels.

Another modification we can make to our design is to use a different radio. The CC2500 is a low-power radio that is comparable to a Zigbee or IEEE 802.15.x radio. If we instead used an IEEE 802.11x radio, which are high power devices, the energy used by communication would be even larger. Thus, reducing the number of channels would have a huge impact on energy consumption. Using the body as a communication medium would likely lower the energy consumed to transmit data, but would require more complex computational methods for encoding and decoding the data.

From the graph, it appears that a localized architecture is a better approach than



our distributed approach. However, our channel selection methods can still benefit systems with a localized architecture by (a) further reducing energy consumption, and (b) improving the wearability of portable monitoring devices.

#### 4.5 CONSTRUCTING A REDUCED CHANNEL DETECTOR

From Equation (4.4), we see that using fewer channels can reduce energy consumption significantly. In this section, we address the question of how to find good subsets of channels, *i.e.*, subsets that will allow us to construct reduced channel detectors that can achieve detection performance similar to that of the 18-channel detector.

##### 4.5.1 Feature Selection Algorithms

To find channel subsets, we use feature or variable selection algorithms. Consider a set of  $n$  examples  $\langle \mathbf{x}_i, y_i \rangle$ ,  $i = 1, 2, \dots, n$ , where  $\mathbf{x}_i$  is a vector consisting of the  $m$  elements  $\langle x_{i1}, x_{i2}, \dots, x_{im} \rangle$  and  $y_i \in Y$ , a set of labels. Let  $\mathbf{A}$  denote a feature selection algorithm that maps a data set  $D$  to a feature subset  $F \subset \{1, 2, \dots, m\}$  where  $m$  is the total number of original features. Typically, once a subset of features has been selected, a new data set  $D'$  is derived from the existing data set. The elements of the new data set  $D'$  are unchanged except that only the features specified by  $F$  are retained. Formally, let  $f$  be a selector function that maps an input data set  $D$  and feature subset  $F$  to a new data set  $D'$ . For every element  $\langle \mathbf{x}, y \rangle \in D$ , there will be an element  $\langle \mathbf{x}', y \rangle \in D'$  such that  $\mathbf{x}' = \langle x_f \rangle$  where  $f \in F$ .

The feature selection algorithms that are most useful to our problem are those that select subsets of features that can be used to build a good predictor. A survey of various feature selection methods is described in [21]. In general, feature selection algorithms can be divided into three broad categories: filters, wrappers, and embedded methods.

Filters rank the  $m$  input variables using a scoring function computed from both the input elements and the output label. Typically, a high score indicates that a variable is useful for building a highly predictive classifier. For classification, a simple scoring function could be the performance of a single variable used as a discriminant function, where performance can be measured as a function of the error rate, the false positive rate, and/or the false negative rate.

Wrapper methods [25] use the performance of a learned classifier to assess the usefulness of different subsets of features. When implementing the wrapper approach, one needs to decide (a) how to search the space of possible feature subsets, (b) how to assess the performance of each learned machine to guide the search, and (c) which learning algorithm to use to select features.

Embedded methods incorporate variable selection as part of the model training process. Simultaneous feature selection and model training offers a few advantages. The learning process no longer needs to be split into a selection stage and then a training stage. As a consequence, the time needed to select features is dramatically reduced. Unfortunately, only a few machine learning algorithms perform feature selection and model training simultaneously. Moreover, these algorithms often restrict the shape of the separating boundary.

A few researchers have used various feature selection methods to reduce the complexity or energy consumption of a detector or classifier. For example, Benbasat *et al.* describe a framework for automatically generating power-efficient decision tree classifiers for wearable gait monitoring nodes. By modifying the learning algorithm to take into account energy cost, a tree that is “energy-aware” can be built. Once a decision tree is constructed, the system dynamically consults only the sensors that are necessary to determine the system state. However, because sensors are activated sequentially over time, the detection latency of such systems will be greater than those that examine all the data simultaneously.

The authors of [37] explore ways to scale the data or feature set used by an ECG-based application for use on a mobile device. The authors show how to use their methods to trade off classification accuracy for bandwidth. Their feature selection approach uses mutual information to “scale” the data set for computation on a mobile device. Other researchers [19, 27] have applied feature selection techniques such as Recursive Feature Elimination for reducing channels in EEG-based applications. In [27], feature selection is performed to reduce the inputs to the learning algorithm and hopefully improve classification, while in [19], the motivation is primarily user comfort. Neither paper considers energy consumption.

For the rest of this chapter, we will describe how we used the wrapper method to select good channels for seizure onset detection. We selected the wrapper method because it allows users to specify a custom objective function with which to evaluate feature subsets. In addition, because wrapper methods can be adapted to work with any machine learning approach for feature selection, we can continue to use SVM classifiers in our detectors.

#### 4.5.2 The Wrapper Approach

As mentioned, the wrapper method [25] uses the performance of a learned classifier to assess the usefulness of different subsets of features. When using the wrapper approach, one needs to decide how to search the space of possible feature subsets for the subset that minimizes (or maximizes) an objective function. The simplest search is an exhaustive search, where a learned algorithm is built for all feature subsets. Unfortunately, for  $n$  features, there are  $2^n - 1$  possible subsets. Such a search could take a enormous amount of time since training a model for each subset involves solving a difficult optimization problem. For example, to train an SVM, a quadratic programming optimization problem is solved. In our experience, training and testing an SVM for each channel subset takes on the order of a few minutes on a 2.8 GHz Intel Core 2 Duo laptop. For  $n = 18$ , where  $n$  is the number of channels, building an SVM for each of the  $2^{18} - 1$  subsets would require a month or more.

An alternative approach is a greedy search. In a greedy approach, the subsets are constructed and evaluated incrementally. There are two basic strategies: *forward selection* or *backward elimination*. In forward selection, variables are incorporated incrementally into larger and larger subsets. In backward elimination, we start with the set of all features and gradually remove or eliminate the least useful ones.

Figure 4.8 shows the greedy backward elimination algorithm we use to select channels. The input to the algorithm is a training data set  $T$  and a validation data set  $V$ . Both contain labeled feature vectors.

---

```

1:  $F \leftarrow \{1, 2, \dots, m\}$  { $F$  is the set of all channels}
2:  $R \leftarrow []$  { $R$  stores the list of best channel subsets, each of different size}
3: repeat
4:    $S \leftarrow []$  { $S$  is a temporary variable used to store objective function values}
5:   for  $i \leftarrow 1$  to  $|F|$  do
6:      $F' \leftarrow \text{FEATURES}(F \setminus F[i])$  {Get set of features after removing channel  $F[i]$ }
7:      $T' \leftarrow f(T, F')$  {Reduce dimensionality of each example in training data  $T$ }
8:      $V' \leftarrow f(V, F')$  {Do similar for validation data  $T$ }
9:      $C \leftarrow \mathbf{I}(T')$  {Train a classifier using data set  $T'$ }
10:     $S[i] \leftarrow J(C, V')$  {Score subset by running classifier  $C$  on test data set  $V'$ }
11:  end for
12:   $k \leftarrow \text{FINDWORSTFEATURE}(S)$  {Get index of worst channel}
13:   $F \leftarrow F \setminus F[k]$  {Remove channel from list}
14:   $R[i] \leftarrow (F, S[k])$  {Store best channel subset and score}
15: until  $F = \emptyset$ 
16: return  $\text{SELECTBEST}(R)$  {Return best channel subset}

```

---

**Figure 4.8:** The backward elimination algorithm. As input, the algorithm takes as input, a training data set  $T$  and a validation data set  $V$ . The functions  $J$ ,  $\text{SELECTBEST}()$ , and  $\text{FINDWORSTFEATURE}()$  are defined by the user.

We adapted the wrapper approach to select channel subsets instead of feature subsets. When a channel is removed, all of the features associated with the channel—in our case, seven—are removed from the feature vector. To find the best channel subset, we incrementally found the best channel subset of a given size. The optimal channel subset is selected from the resulting set of channel subsets.

In the seizure onset detection problem, we started with 18 channels. All channel subsets of size 17 were created by removing each of the channels with replacement from this initial set. Each of these channel subsets were used to prune the feature vectors of the input training and test sets to generate new data sets where each feature vector has seven fewer features. Then, each of the training data subsets were used to train a classifier with fewer channels. The classifier is then incorporated into a seizure onset detector similar to that described in Section 4.2.1.

Each 17-channel subset is then assigned a score using an objective function  $J$ . In our case, the value of the objective function is based on the performance of the corresponding detection algorithm on the test data  $V'$ . A channel subset is assigned a score of zero if the resulting detector has a lower seizure onset detection rate than the 18 channel classifier constructed using the same set of data. Otherwise we assign each subset a score corresponding to the false positive rate of the SVM classifier constructed using that subset.

Once each channel subset is assigned a score, we then use the function  $\text{FINDWORSTFEATURE}()$  to determine an optimal 17-channel subset. An optimal 17-channel subset is one that corresponds to the detector with the lowest false positive rate that detects as many seizures as the  $\text{ALLCHANNEL}$  detector. If there are no detec-

tors that meet this criteria, a channel subset with the highest seizure onset detection rate is designated as the optimal subset instead. This channel subset, along with the value of objective function, is recorded for later use in  $R$ . The least useful channel is the one channel that is not present in this channel subset.

The best channel subset of size 17 is then used as input to find an optimal channel subset of size 16. The algorithm continues in this fashion until no further channels can be removed. After all rounds have completed, the algorithm examines  $R$ , which will contain a list of channel subsets with a decreasing number of channels, and selects one to be the optimal subset. In our implementation, the optimal channel subset chosen is one with smallest number of channels that detects as many seizures as the ALLCHANNEL algorithm without exceeding the false positive rate of the ALLCHANNEL algorithm by more than two times. We allow for a reduced channel detector to make more false positive rates to allow for performance degradation due to fewer channels.

### 4.5.3 Example

For illustration purposes, we outline an example execution of the channel selection algorithm for one patient (Patient 13). For input data, the training and test data set consists of the same set of feature vectors. Seizure and non-seizure feature vectors from an individual's EEG recording are included in the input data. Table 4.3 shows the seizure event detection rates and the true negative rates of the constructed 17-channel detectors after the first round of the algorithm has been executed.

After the first execution of the loop, we then select the subset of channels that corresponds to the SUBSET detector with the lowest false positive rate that detects as many seizures as the ALLCHANNEL detector. In this case, the ALLCHANNEL detector was able to detect all the seizures in the training set with a false positive rate of 0.0011. From Table 4.3, we see that the SUBSET detector that does not include channel 12 is the optimal 17-channel detector. We update  $F$  to  $F \setminus x$  and store the channel subset in  $R$ .

This process continues until  $F$  is empty. Now,  $R$  contains a list of the best  $k$ -channel detectors,  $k = 1, 2, \dots, 17$ . Table 4.4 shows the contents of  $R$ . From this list of channel subset detectors, we then select the detector that uses the fewest number of channels without degrading the seizure detection rate. We allow the false positive rate to be twice that of the ALLCHANNEL detector, that is, less than or equal to 0.0012. In this case, the optimal subset is  $\{3\}$  or using the standard electrode names,  $\{T7-P7\}$ .

## 4.6 EVALUATION METHODOLOGY

To evaluate the channels output by the channel selection algorithm, we constructed reduced channel detectors and measured the performance of the resulting detectors on real data. A good subset of channels will result in the construction of a detector with reasonably good specificity, sensitivity, and latency relative to the original 18-channel detector. In this section, we describe the data we used for input to the channel selection algorithm. In addition, we outline the testing procedure used to evaluate the channels selected.

**Table 4.3:** The seizure event detection rate and the false positive rate of the 17-channel detectors constructed after the first iteration of the loop has been executed. Since all of the detectors have equal detection rates, selection is based on the false positive rate. The detector with the *lowest* false positive rate, is the one that does not include channel 4.

Channel subset used by detector																	Detection rate	True pos. rate	
	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	1.0	0.000578
1		3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	1.0	0.000541
1	2		4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	1.0	0.000606
1	2	3		5	6	7	8	9	10	11	12	13	14	15	16	17	18	1.0	0.000392
1	2	3	4		6	7	8	9	10	11	12	13	14	15	16	17	18	1.0	0.000531
1	2	3	4	5		7	8	9	10	11	12	13	14	15	16	17	18	1.0	0.000513
1	2	3	4	5	6		8	9	10	11	12	13	14	15	16	17	18	1.0	0.000513
1	2	3	4	5	6	7		9	10	11	12	13	14	15	16	17	18	1.0	0.000485
1	2	3	4	5	6	7	8		10	11	12	13	14	15	16	17	18	1.0	0.000466
1	2	3	4	5	6	7	8	9		11	12	13	14	15	16	17	18	1.0	0.000447
1	2	3	4	5	6	7	8	9	10		12	13	14	15	16	17	18	1.0	0.000587
1	2	3	4	5	6	7	8	9	10	11		13	14	15	16	17	18	1.0	0.000503
1	2	3	4	5	6	7	8	9	10	11	12		14	15	16	17	18	1.0	0.000541
1	2	3	4	5	6	7	8	9	10	11	12	13		15	16	17	18	1.0	0.000503
1	2	3	4	5	6	7	8	9	10	11	12	13	14		16	17	18	1.0	0.000513
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15		17	18	1.0	0.000569
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16		18	1.0	0.000475
1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17		1.0	0.000401

#### 4.6.1 Data Description

For channel selection and detector evaluation, we used pre-recorded pediatric EEG data from 16 patients collected in a previous study [43]. All patients in the study were being evaluated for potential cerebral resection surgery, and had their medications stopped. Therefore, they had seizures more often than is typical. Table 4.5 shows the number of seizures and number of hours of recorded EEG data for each patient. The onset and duration of each of the seizures for each patient was labeled by a human expert.

#### 4.6.2 Testing Procedure

Ideally, to test our algorithm, we would use all available data recorded from a given patient to select channels and to construct a detector. The performance of the detector would then be evaluated prospectively in real-time using live EEG from the same patient.

We did not have the opportunity to do this, so instead, we divided the data for each patient into training and test data sets. Both training and test data sets contained some of the seizure data and some of the non-seizure data. Ideally, the training data set would be further divided into two subsets—one subset would act as the training data input and the other would serve as the validation data input used by the channel

**Table 4.4:** The list of the best  $k$ -channel detectors,  $k = 1, 2, \dots, 17$ . From this list of SUBSET detectors, the best SUBSET detector was found to be the one using  $\{3\}$  or,  $\{T7-P7\}$ . This detector (highlighted) uses the fewest number of channels without degrading the seizure detection rate and has a false positive rate that is less than twice that of ALLCHANNEL detector, that is, less than 0.0012.

Channel subset used by detector																		Detection rate	True pos. rate
1 2 3	5 6 7 8 9	10 11 12 13 14 15 16 17 18	1.0	0.000392															
1 2 3	5 6 7 8 9	11 12 13 14 15 16 17 18	1.0	0.000326															
1 2 3	5 6 7 8 9	11 12 13 14 15 16 18	1.0	0.000317															
1 2 3	5 6 7 8	11 12 13 14 15 16 18	1.0	0.000308															
1 2 3	5 6 7 8	11 12 13 14 15 16	1.0	0.000308															
1 2 3	5 6 7 8	11 12 13 14 16	1.0	0.000261															
1 2 3	5 7 8	11 12 13 14 16	1.0	0.000252															
1 2 3	7 8	11 12 13 14 16	1.0	0.000252															
1 2 3	7	11 12 13 14 16	1.0	0.000261															
2 3	7	11 12 13 14 16	1.0	0.000233															
2 3		11 12 13 14 16	1.0	0.000252															
2 3		11 12 13 16	1.0	0.000252															
2 3		11 13 16	1.0	0.000289															
3		11 13 16	1.0	0.000252															
3		11 16	1.0	0.000392															
3		16	1.0	0.000485															
3			1.0	0.000895															

selection algorithm. Unfortunately, we did not have enough seizure data per patient to be able to divide the data into three subsets. Thus, for channel selection, we use the entire training data set as input. The risk of using training performance to select channels is that performance results may not generalize.

A detailed description of our evaluation procedure follows. First, the EEG data belonging to each patient was divided into consecutive, one-hour records. Each one-hour record consisted of approximately 3600 two second windows where each window overlapped the previous window by one second. Each window contained EEG samples recorded from all 18 channels and labels for each window (+1 or -1) were derived from annotations made by a human expert.

Next, we partitioned the available data from an individual patient into training and test sets using leave-one-seizure-record-out cross validation (LOOCV) [49]. If  $M$  is the number of one-hour records containing seizures, this procedure results in  $M$  different training and test set combinations. Each training set consists of  $M - 1$  seizures. Using each training set, two different patient-specific classifiers were constructed:

- (a) an SVM classifier that uses all 18 channels and
- (b) an SVM classifier that uses a subset of the channels, where the channels were chosen using the wrapper approach with the same training data set as input for the

**Table 4.5:** Data for each patient was obtained from a previous study [43]. The table shows the number of seizures and number of hours of EEG data recorded for 16 patients. The number of seizures is more than the number of one-hour records with seizures, because some records contained more than one seizure.

Index	Patient I.D.	Number of seizures	Records with seizures	Recording duration (h)
1	40	24	8	16.0
2	25	4	3	64.7
3	22	7	5	55.0
4	36	3	3	27.5
5	10	3	3	35.2
6	24	2	2	65.0
7	31	5	5	18.0
8	38	3	3	34.8
9	41	9	7	32.0
10	3	5	5	41.5
11	12	2	2	40.0
12	20	4	4	40.0
13	35	7	7	50.0
14	43	8	7	25.0
15	45	38	6	28.0
16	47	19	13	35.0

training and validation set.

We used the LIBSVM package [10] to train the SVM classifiers. The parameters  $C$  and  $\gamma$  were set to 10 and  $10^{-2}$  respectively based on a grid search conducted using a subset of the patients.

Using the resulting classifiers, we then constructed two real-time seizure detection algorithms, ALLCHANNEL and SUBSET respectively, using the constructed classifiers. For the ALLCHANNEL detector, the 18-channel classifier was used as the classifier component of the algorithm. For the SUBSET detector, the reduced channel detector was used as the classifier component.

To determine detector performance, we executed the detector on the union of the seizure-free records and the one-hour record containing seizure data that was excluded during training. For each of the detectors, we recorded the following metrics and used them to compare the performance of the detectors.

- (a) *the number of true seizures detected:* A seizure onset is declared when four consecutive windows are labeled as positive.
- (b) *the number of false events declared per hour of monitoring:* If a seizure onset is declared during a period that does not contain an actual seizure, we call this declaration a false alarm or false event. In this method, two runs of four positive windows separated by a single negative window would be counted as two events.

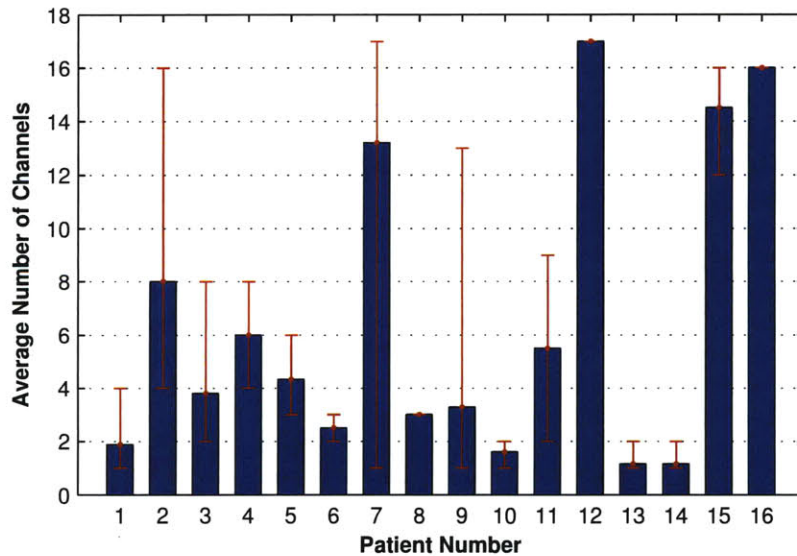
- (c) *the detection delay for each detected seizure in the recording*: The detection latency is the difference in time between when the detector declares seizure onset and the electrographic onset of the seizure as marked by a human expert.
- (d) *the number of channels used and energy consumed by each detector*: To quantify the energy consumed by a detector with  $N_c$  channels for one second, we used Equation (4.4). The total energy consumed is then this number multiplied by the number of total windows in the EEG recording.

#### 4.7 RESULTS FOR ALL PATIENTS

In this section, we compare the performance and energy consumption of the SUBSET detectors to the performance and energy consumption of the ALLCHANNEL detectors in aggregate.

##### 4.7.1 Number of Channels

Figure 4.9 shows the average number of channels that were selected by the greedy backward elimination wrapper algorithm for each patient. Averaging is done over the different trials. Each trial produced a different subset detector since different training data subset is used as input. The error bars show the maximum and minimum size of the channel subsets selected for each patient. All of the SUBSET detectors used fewer than 18 channels on average and overall, the average number of channels used by the reduced channel detectors was 6.4—a significant reduction from 18. For patients 7,



**Figure 4.9:** The average number of channels (6.4) used by the reduced channel detectors was significantly smaller than 18. The error bars show the maximum and minimum number of channels selected for each patient.



12, 15, and 16, the reduction was small, but for the remaining patients, the number of channels used by the SUBSET detector was reduced on average to fewer than 9 channels. The graph shows high inter-patient variability in the number of channels. High inter-patient variability can be explained by the variability in the spatial distributions of seizures among patients. Intra-patient variability, on the other hand, seems to differ for each patient. For some patients, when given a different training subset, the variance in the number of channels selected remains small (Patients 6, 8, 10, 13, and 14).

Note that a constant number of channels does not necessarily mean that the *same* channels are being selected. We will investigate the intra-patient variability when we consider some of the patients individually.

#### 4.7.2 Seizure Onset Detection Performance

The ALLCHANNEL SVM-based seizure onset detectors identified 139 out of 143 seizures. The SUBSET detection algorithm detected 136. Table 4.6 compares the detection performance of the ALLCHANNEL seizure onset detectors to the detection performance of the SUBSET seizure onset detectors for each of the patients in the study. In some cases, the SUBSET detectors detect more seizures than the ALLCHANNEL detector (*e.g.*, for patients 4 and 14). For 10 out of 16 patients, the subset detectors

**Table 4.6:** The detection performance of 18-channel detectors is compared to the detection performance of reduced channel detectors. We will discuss the results for the highlighted patients in more detail later in this section.

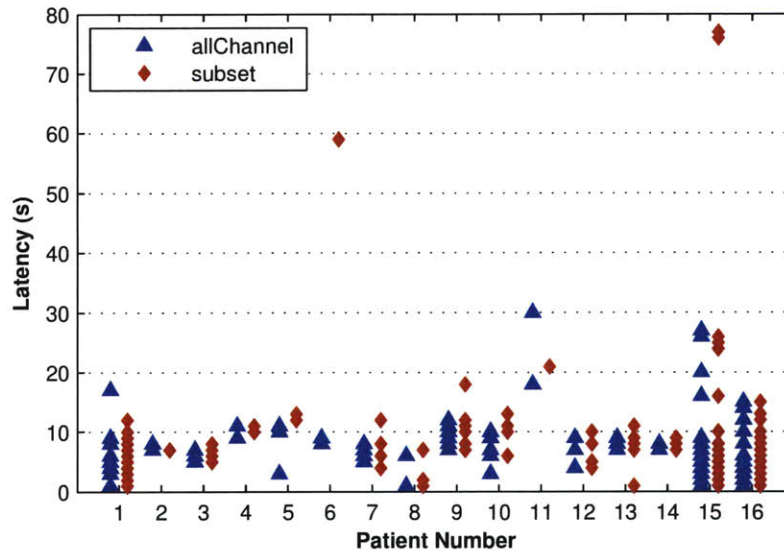
Patient	Total seizures	ALLCHANNEL	SUBSET
1	24	24	24
2	4	3	3
3	7	7	6
4	3	2	3
5	3	3	2
6	2	2	1
7	5	4	4
8	3	3	3
9	9	9	9
10	5	5	5
11	2	2	1
12	4	4	4
13	7	7	7
14	8	7	7
15	38	38	38
16	19	19	19
Sum	143	139	136

detected all of the seizure onsets detected by the ALLCHANNEL detector. For patients where a seizure was missed, we attribute the drop in detection rate to intra-patient seizure variation. Thus, where one seizure might be easily detected using one set of reduced channels, other seizures may need a different set to be detected successfully.

The fact that fewer channels can be used to detect seizures reliably for many patient is not surprising—neurologists can often identify a small number of active channels from an EEG trace. What is surprising is that sometimes the algorithm also selected channels that appear inactive during a seizure. We will see this effect when we consider the results for a single patient. We speculate that including a few non-active channels can reduce the false positive rate.

### 4.7.3 Detection Latency

Figure 4.10 plots the detection delay for the ALLCHANNEL and SUBSET detectors for each patient. Each point in the graph represents the detection delay of either

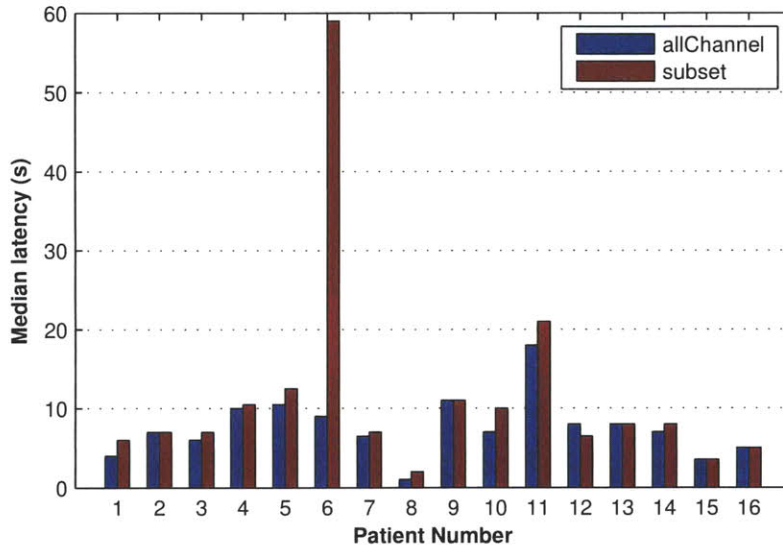


**Figure 4.10:** The distribution of detection delays for the SUBSET and ALLCHANNEL detectors appears similar for all patients except for Patient number 6. Each point represents the detection delay of a single seizure. For some patients, some seizures were detected with the same latency and thus, overlap in the graph.

the ALLCHANNEL or SUBSET detector for a single seizure. In aggregate the detection latencies of the SUBSET detectors are comparable to those of the ALLCHANNEL detector—the median detection delay for all patients increased slightly from 6.0 to 7.0 s. If we compare the per-patient median latencies of the seizure onsets detected by both detectors (Figure 4.11), we see that for many patients, the median latency increased. We speculate that in reducing the number of channels used for detection, the algorithm omits channels that may exhibit ictal activity earlier than the others.

### 4.7.4 False Events per Hour

The average number of false events per hour increased from 0.07 to 0.11. Figure 4.12 plots the false events per hour for each of the patients in the study. We can see that



**Figure 4.11:** The per-patient median latency for seizure onsets detected by both detectors. For most patients, the median latency increased.

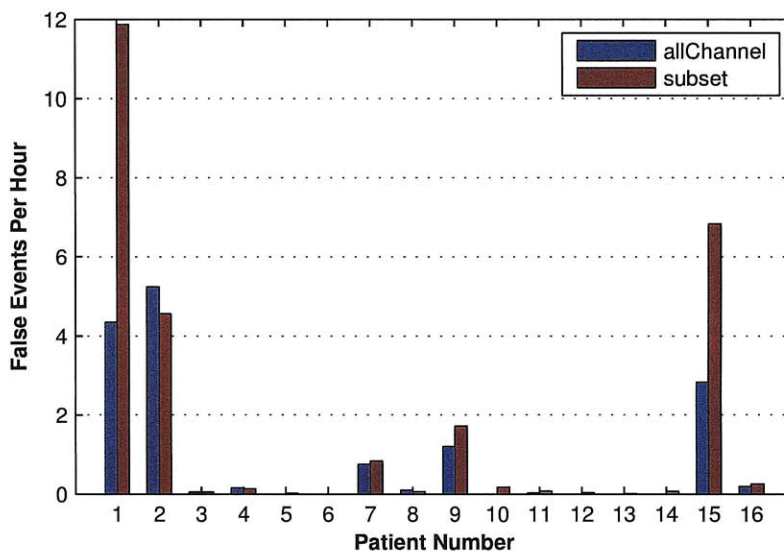
for 15 out of 16 patients, the false events per hour increased. We will explore the relationship between false events per hour and number of channels when we consider the performance for individual patients.

#### 4.7.5 Energy Consumption

Figure 4.13 illustrates the average energy savings achieved by the SUBSET detector for each patient. Using Equation (4.4) to determine the energy consumed by all of the detectors, we determined that the algorithm achieved an average savings of 60%.

To determine how this impacts battery lifetime, we considered the energy density of a few real batteries. The energy density of a battery varies depending on the type of material used as the electrolyte. Some common battery types include: lithium-ion (Li-Ion), lithium-polymer (Li-Poly), alkaline, and nickel-metal hydride (NiMh). Table 4.7 shows the energy densities of various common batteries.

Figure 4.14 shows the estimated battery lifetime of our prototype system as a function of the number of channels used by the detector. We have assumed that the batteries do not self-discharge and can maintain the same voltage level regardless of the remaining battery capacity. For ambulatory seizure onset detection, the use of a small battery is preferred because the acquisition hardware will be worn on or near the head. The Li-Ion coin cell which has one-third the weight of the other batteries seems the most appropriate. If all channels are used, the Li-Ion coin cell has a battery lifetime of only 0.78 days. If we can reduce the number of channels to three channels, the battery lifetime can be increased to three or four days.



**Figure 4.12:** The average false events per hour increased slightly from 0.07 to 0.11.

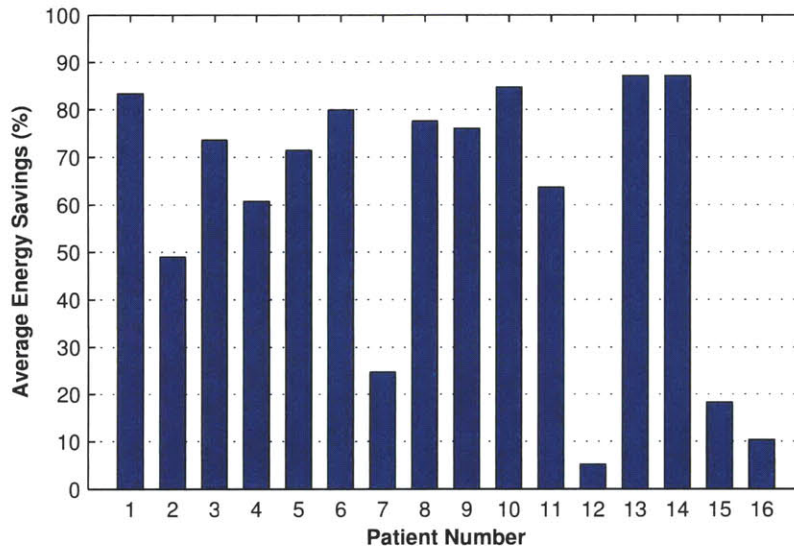
**Table 4.7:** Common battery specifications. The Li-Ion batteries are rechargeable batteries, while the AAA battery is a non-rechargeable alkaline battery. For long-term seizure onset detection where the device is placed on or near the head, the use of a smaller lightweight battery is preferred.

	Li-Ion (phone)	AAA	Li-Ion (coin)
Weight (g)	24	23	7.8
Capacity (mAh)	850	900	160
Nominal Voltage (V)	3.7	3.0	3.6
Ideal Capacity (kJ)	11.3	9.7	2.1

#### 4.7.6 Discussion

The results demonstrate that the channel selection algorithm finds reasonably good reduced channel detectors for the majority of patients. However, there are two potential issues with our approach. First, for a patient, does the spatial distribution of the seizure focus change over time? Though published information about how seizures change over years or decades is not available, clinicians generally believe that adults with epilepsy tend to have stable EEGs. In one clinical study [45], a patient-specific detector trained using EEG data collected eight or nine months ago is shown to successfully detect a new previously unseen seizure in real-time.

Another issue is that there is high intra-patient channel subset variability for some patients. In other words, for the same patient, different optimal subsets are output when different training data is input. The most likely reason for this variability is



**Figure 4.13:** The average energy savings achieved by the SUBSET detection algorithm was 60%.

inter-seizure variation—a patient may have several types of seizures that originate from slightly different locations in the brain. If a electrographically unique seizure is left out of the training set, channels that are active during that seizure will have a lower ranking than channels that are active.

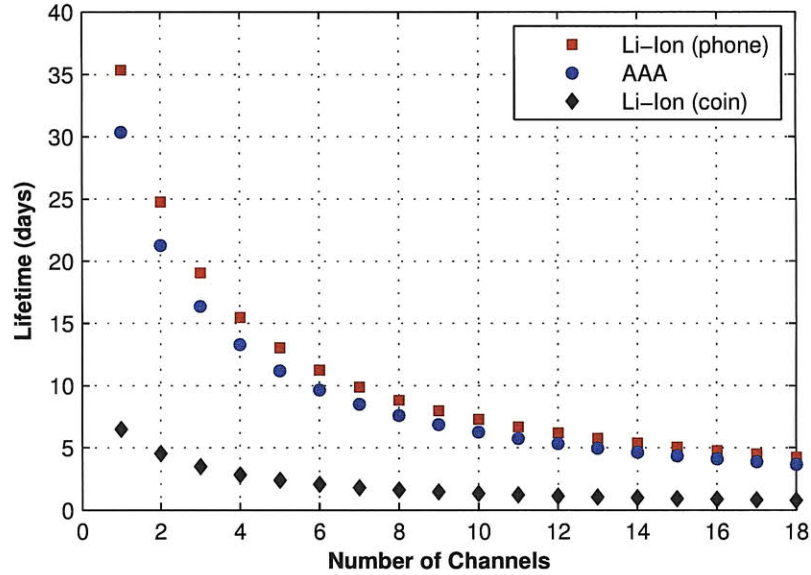
#### 4.8 CASE STUDIES

To better understand the behavior of our channel selection method, we analyze the performance of the reduced channel detectors for two patients. For the first patient, the reduced channel detectors achieved better than average performance. For the second patient, the reduced channel detectors achieved worse than average performance. In the course of this patient-specific analysis, we will address the following questions:

- (a) Is there a physiological explanation for the selected channels?
- (b) What is the performance trade-off of reducing the number of channels?
- (c) Training on different subsets of data may yield different channel subsets for each patient. How does one select a final set of channels for use in deployment? Does using different channel subsets yield significant differences in performance?

#### 4.9 PATIENT 8

Patient 8 experienced three seizures in 34.8 hours. Following the procedure outlined in Section 4.6.2, we generated three training subsets— $T_1$ ,  $T_2$  and  $T_3$ —and used them



**Figure 4.14:** With all 18 channels active, the heavier Li-Ion battery has an ideal battery lifetime of about 4.3 days. The AAA battery has an ideal battery lifetime of 3.7 days and the Li-Ion coin has an ideal battery lifetime of 0.78 days.

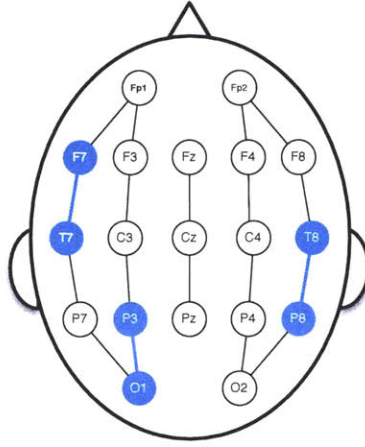
as input to the channel selection algorithm. Table 4.8 shows the selected channel subsets and Figure 4.15 shows the physical locations of those channels on the scalp. To determine whether the channels selected have a physiological basis, we visually

**Table 4.8:** Channels selected for patient 8 by the channel selection algorithm. F7-T7 is used by all detectors.

Training subset	Channels Selected		
$T_1$	F7-T7	P3-O1	T8-P8
$T_2$	F7-T7	C4-P4	Cz-Pz
$T_3$	F7-T7	F3-C3	T8-P8

inspect the EEG signals during a couple of seizures. Figure 4.16 illustrates the onset of two recorded seizures from patient 8.

The seizures begin with the appearance of 8 Hz rhythmic activity most prominently on the left hemispheric channels (Fp1-F7, F7-T7, T7-P7, P7-O1, Fp1-F3, F3-C3, C3-P3, P3-O1). Notice that there is limited or no involvement of right hemispheric channels (Fp2-F4, F4-C4, C4-P4, P4-O2, Fp2-F8, F8-T8, T8-P8, P8-O2). This explains why the channels from the left hemisphere (F7-T7, P3-O1, F3-C3) were selected. However, all detectors also included a channel from the right hemisphere, e.g., T8-P8. These channels were selected even though no ictal activity is observed on



**Figure 4.15:** The physical channels selected for patient 8 when  $T_1$  was input as training data.

these channels during the seizure. We speculate that the addition of these non-active channels allows the detector to ignore false events better.

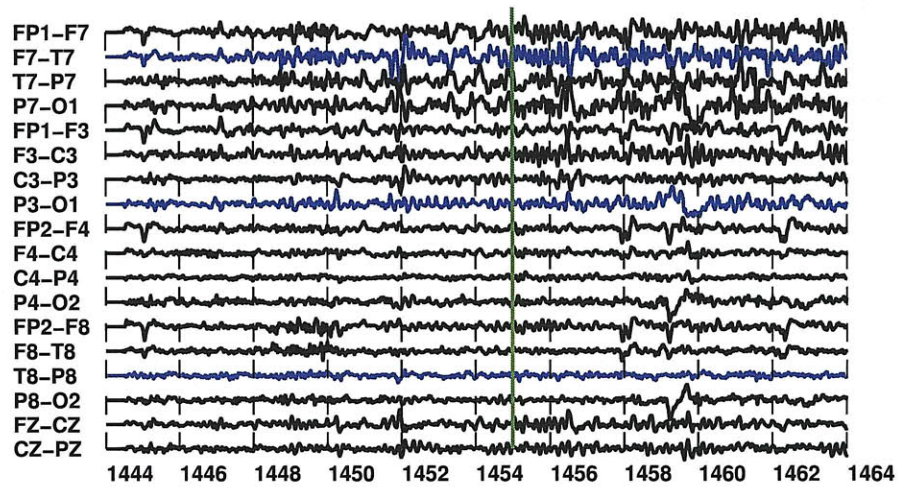
#### 4.9.1 Detection Performance and False Alarm Rate

Each of the three sets of channels were used to construct three SUBSET detectors. Table 4.9 shows the test performance of the SUBSET detectors compared to the test performance of the ALLCHANNEL detectors.

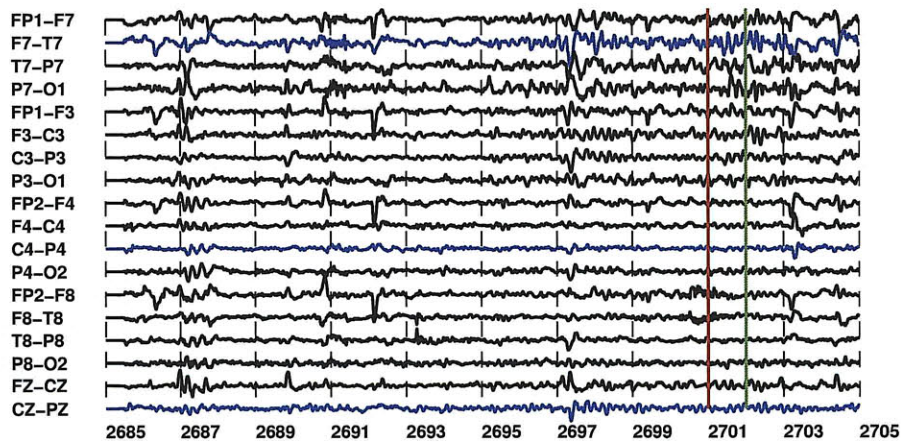
**Table 4.9:** Overall, the SUBSET detectors constructed for patient 8 had good performance on the test data. All of the seizures in the recording were detected using three channels. This amounts to energy savings of approximately 80%.

	Test Set 1		Test Set 2		Test Set 3	
	ALL	SUBSET	ALL	SUBSET	ALL	SUBSET
Seizure detected?	Y	Y	Y	Y	Y	Y
False positives	13	35	9	42	33	18
False events	2	2	0	1	3	0
Latency (s)	1	1	6	7	1	2
Number of channels	18	3	18	3	18	3

From the table, we see that the reduced channel detectors were able to detect all the seizures. We believe this is due to the stereotypical nature of the seizures. In addition, the table shows that the SUBSET detectors achieved a comparable false alarm rate as the ALLCHANNEL detector, even though the number of false positives was higher. More false positives will not result in more false alarms as long as the false positive windows do not form sequences of 4 or more positive windows. One question that



(a) The EEG signals recorded during a seizure for patient 8. The seizure begins at 1454 s.



(b) A second seizure from patient 8. The seizure begins at 2695 s into the recording.

**Figure 4.16:** Prominent 8 Hz rhythmic activity is present on the selected channels, highlighted in blue, in two recorded seizures from patient 8. The vertical lines mark the times at which the ALLCHANNEL (red) and SUBSET (green) detectors declare seizure onset.

arises is: What is the relationship between the number of false events and the number of channels?

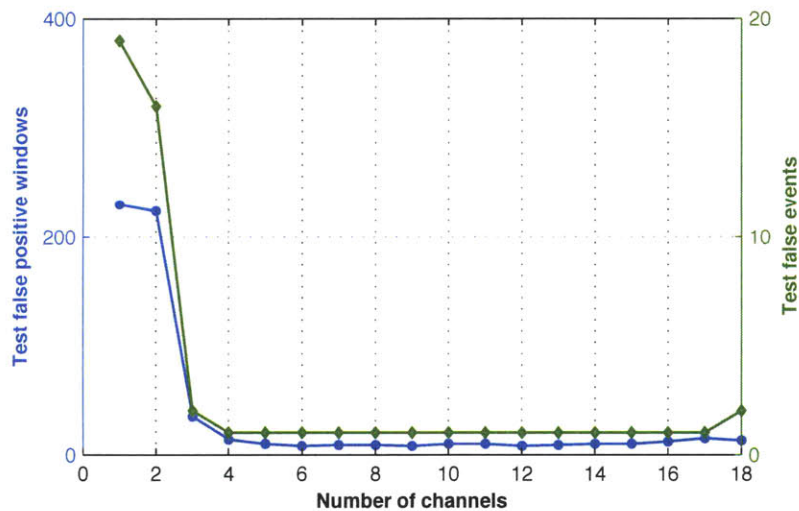
Figure 4.17 plots the number of false alarms and the number of false positive windows on Test Set 1. The number of false events decreases dramatically as you increase the number of channels from one channel to four channels. The decrease suggests that the channel selection algorithm provided a good ranking of the channels. Thus, when



the most discriminative channels are included, specificity increases dramatically. After four channels has been added, the number of false events remains unchanged as more channels are added. This suggests that the remaining channels are irrelevant and thus, we observe no change in the false event rate.

We located the false alarms triggered by the SUBSET and ALLCHANNEL detectors in the EEG recording1. Figure 4.18 shows a false event declared by both detectors. The channels selected are highlighted in blue. This false alarms occurred prior to the onset of the seizure in the recording. Thus, we may decide that this declaration is not a false alarm.

To determine why the three channel detector was selected for this data set, we need to examine the training results. Figure 4.19 plots the number of *training* false positive and false events as a function of the number of channels used by the detector constructed using  $T_1$ , Training Set 1, as input data. The horizontal line marks the upper bound on the acceptable number of false positive windows for the reduced channel detector and explains why the three channel detector was selected. From the graph, it is surprising that, unlike in the test data, the number of false positive windows and false alarms increase as the number of channels increases beyond four. This increase is likely due to the following reasons. As irrelevant channels are added, the activity on those channels may be viewed as seizure activity. Thus, in segments of the recording where those patterns emerge, the detector will make false declarations. If we had more seizure data for training, it is likely that these patterns would be ignored. Also, the increase observed is somewhat deceptive since these false alarms and false positives occurred during the one-hour records that contained seizures. Note that the increase in false positives does not immediately impact the number of false events since the additional false positives do not result in formations of sequences of four or more.



**Figure 4.17:** False positive and false event rates achieved on the Test Set 1 for patient 8 as a function of the number of channels.

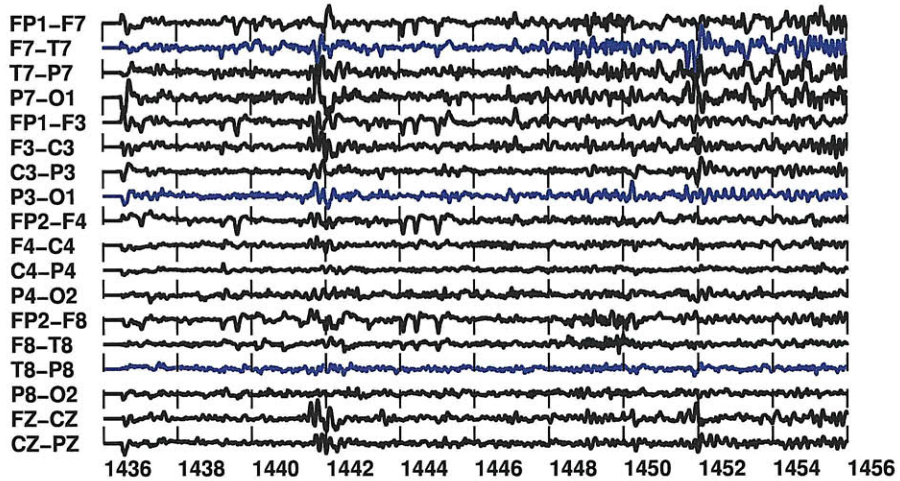


Figure 4.18: A false event at the 1446 s mark is declared by both detectors.

#### 4.9.2 Latency

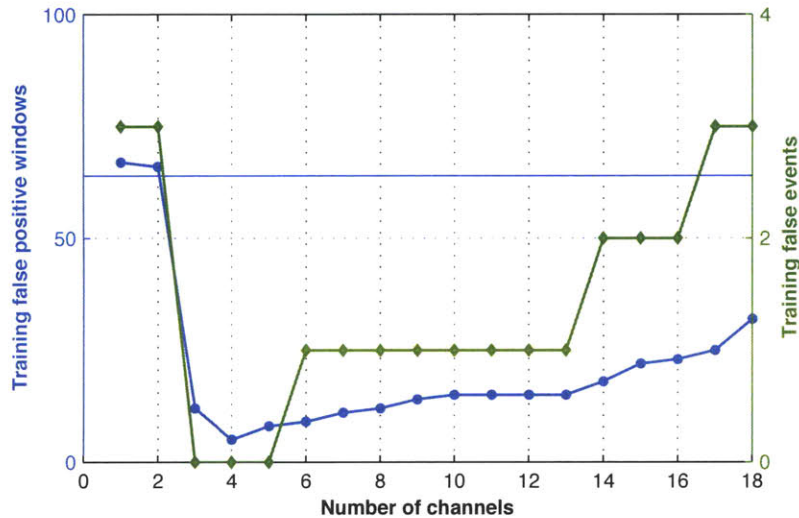
In all training and test data sets, the detection latencies of the SUBSET detectors were higher. For this patient, one possible explanation for this increase is that some of the channels that were not included in the SUBSET detectors may have exhibited ictal activity earlier. However, this is not obvious in the seizures shown in Figure 4.16.

Figure 4.20 plots the training and test detection latencies as a function of the number of channels averaged over the three data sets. From this graph, we can make the following observations. First, after the top three or four channels are included in the detector, the latency decreases with increasing number of channels. Second, the training false alarm rate is inversely related to latency, *i.e.*, a decrease in false alarm rate correlates with an increase in latency. We can explain the inverse relationship between the two metrics and their relationship to the number of channels as follows. As more channels are used, irrelevant information will increase the number of false positive windows. Inevitably, these incorrect labels will lead to more false events. However, since non-ictal windows in the recording near a seizure have a higher probability of being mislabeled, when the number of false positives increases, latency decreases.

This inverse relationship between false alarm rate and latency is not apparent in the test data. However, in the test data, only one seizure is present. Thus, we do not have enough test data to see this relationship.

#### 4.9.3 Sensitivity to Channels Selected

One final concern with the channel selection algorithm is that different channels are selected when the algorithm is presented with a different training set. This raises the question of how sensitive the performance is to the choice of channels. To address this question, we performed the following experiment. We took all channel subsets returned by the channel selection algorithm when a specific training data set was used



**Figure 4.19:** Number of false positives and false alarms achieved on the Training Set 1 ( $T_1$ ) for patient 8 as a function of the number of channels. The horizontal line marks the upper bound on the acceptable fraction of false positive windows for the reduced channel detector.

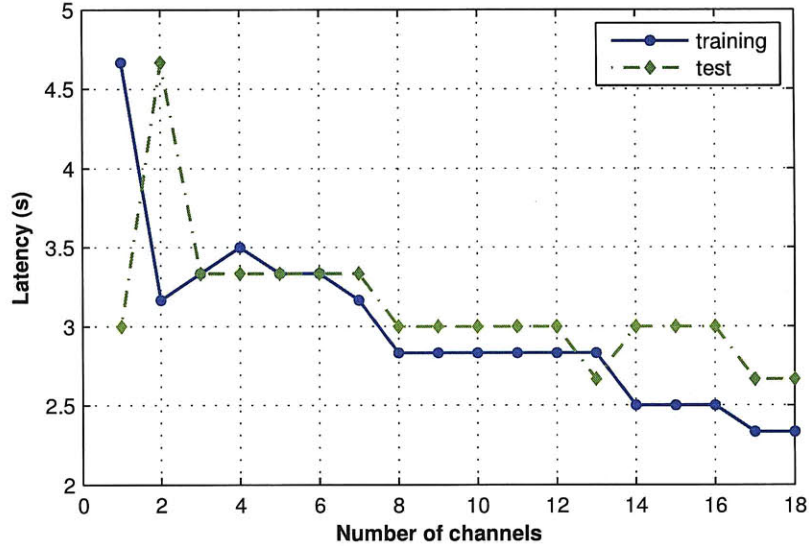
and then trained an SVM using those channels with a different training data set. We then evaluated the resulting detectors on the appropriate test file. For example, when we input data subset  $T_1$  into the wrapper algorithm, channels F7-T7, P3-O1, and T8-P8, were selected. We represent this channel subset using the notation  $F_1$ . We then constructed the SVM-based classifier  $\mathbf{I}(T_1, F_1)$  and used it to build a new subset detector. We then executed this detector on test data.

Table 4.10 shows the results of this experiment. The results indicate that all seizure onsets in the recording were detected no matter which files were used for training and which files were left out for testing. Moreover, the mean detection delay and number of false events did not change significantly when different data sets were used for training.

#### 4.10 PATIENT 16

For patient 16, the channel selection algorithm only reduced the average number of channels from 18 to 16. Following the procedure discussed in Section 4.6.2, 13 training subsets were generated and used as input. Since so many channels were included, we do not show all the channel subsets output by the algorithm when different training data sets were used as input.

Figure 4.21 shows two of the recorded seizures for this patient. From the EEG waveforms, we can make a few observations. First, this patient appears to have at least two different kinds of seizure onsets. Second, in the seizure shown in Figure 4.21(b), almost all of channels become involved in the seizure, though, the seizure is most



**Figure 4.20:** Average training and test detection latencies as a function of the number of channels. Latency is inversely related to the false alarm rate.

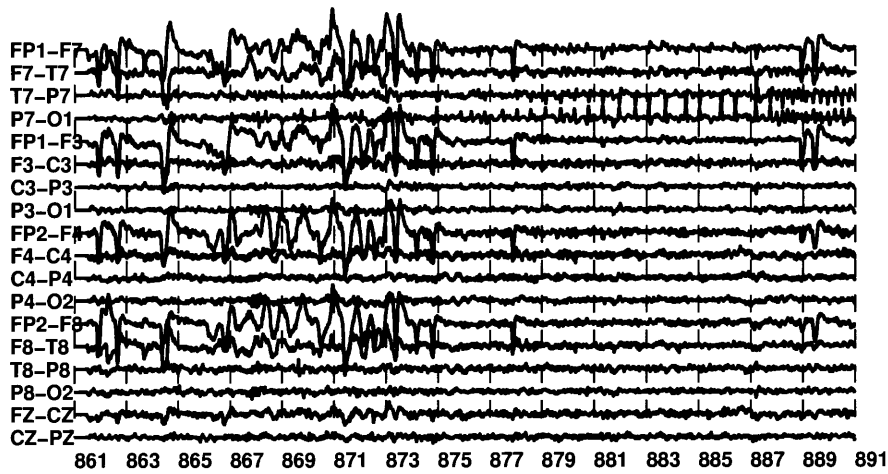
prominent on channels T7-P7 and P3-O1. Given that the seizure is evident on multiple channels, it may seem odd that the channel selection algorithm did not reduce the number of channels significantly. However, by examining the non-seizure regions of the recording, we discovered that ictal discharges are evident on multiple channels. Thus, to differentiate between seizure and non-seizure, many channels are needed.

Figure 4.22 plots the number of *training* false positives and false events as a function of the number of channels used by the reduced channel detector for patient 16 on Training Set 1. The figure shows how the reduced channel detector was selected—the 16-channel detector had a false positive rate that was no more than twice the false positive rate of the 18-channel detector, which is indicated by the horizontal line.

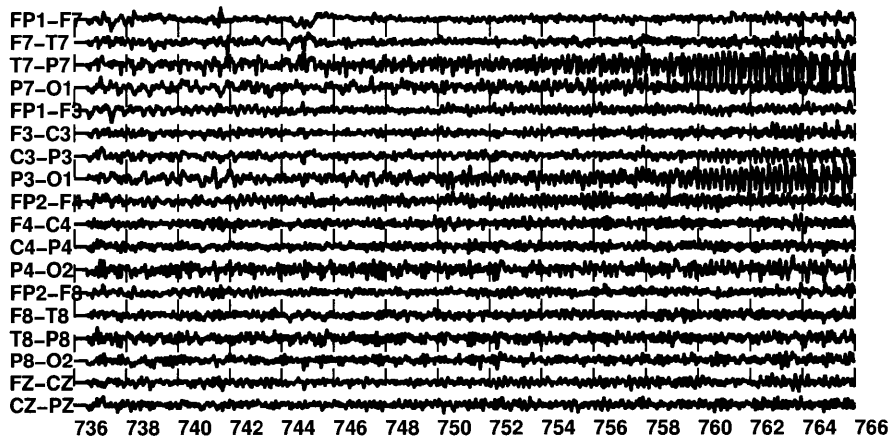
#### 4.10.1 Detection Performance, False Alarm Rate, and Latency

The 13 sets of channels were used to construct 13 different SVM classifiers  $\mathbf{I}(T_k), k \in \{1, 2, \dots\}$ , which were then used to build  $\text{SUBSET}(k)$  detectors. Table 4.11 summarizes the average test performance of these detectors and compares them to the average test performance of the  $\text{ALLCHANNEL}$  detectors. All of the  $\text{SUBSET}$  detectors were able to detect as many seizures as the  $\text{ALLCHANNEL}$  detectors. On the test data, the average false events per hour increases, while the latency decreases. A large number of channels is needed by the detector because we suspect that during the non-seizure portions of the recording, many channels are involved. Thus, as many channels as possible are needed to distinguish seizure from non-seizure. Figure 4.23 shows a segment during a non-seizure record. Comparing the EEG signal leading up to the false declaration at 3566 seconds, we see how similar this segment is to Figure 4.21(b).

Given our experience with patient 8, we believed that latency would be inversely



(a) This seizure begins at 876 s into the recording.



(b) This seizure begins at 751 s into the recording.

**Figure 4.21:** Two seizures from patient 16. Several of the channels are involved during seizure onset, but selection of a subset of channels is based on the false positive rate of the detector.

**Table 4.10:** Performance of detectors constructed using the different channel subsets with different test data. All seizure onsets were detected without significant change in the detection delay and number of false events.

Channels	Metric	Test Set 1	Test Set 2	Test Set 3
F7-T7, P3-O1, T8-P8	Seizures detected?	Y	Y	1
	False positives	35	36	30
	False events	1	2	2
	Mean latency (s)	1	7	2
F7-T7, C4-P4, Cz-Pz	Seizures detected?	Y	Y	Y
	False positives	9	42	22
	False events	1	1	0
	Mean latency (s)	1	7	2
F7-T7, F3-C3, T8-P8	Seizures detected?	Y	Y	Y
	False positives	17	24	18
	False events	1	1	0
	Mean latency (s)	1	7	2

**Table 4.11:** Average test performance of constructed detectors for patient 16.

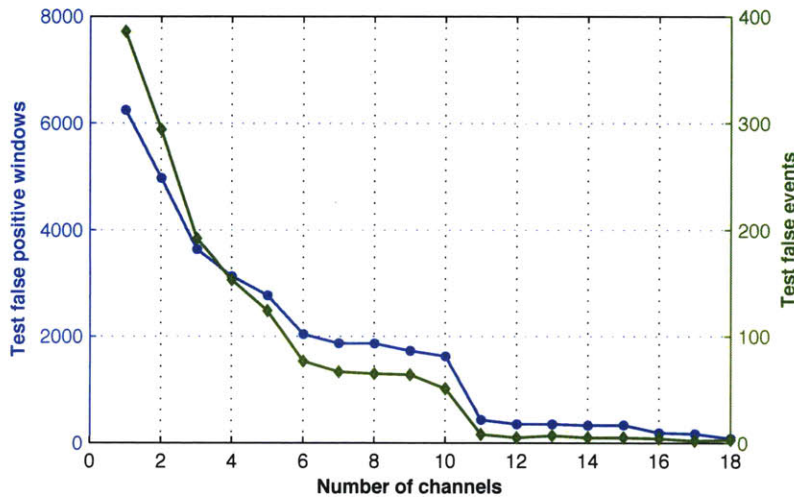
Metric	ALLCHANNEL	SUBSET
Detection	19 / 19	19 / 19
False events per hour	0.2	0.25
Average latency (s)	6.8	6.2
Average number of channels	18	16
Energy savings	-	10%

related to the number of false positives and false events for patient 16. Figure 4.24 plots the average training and test detection latencies as a function of the number of channels. For this patient, latency decreases with increasing number of channels. One possible reason that both latency and false alarms decrease with increasing number of channels is that the detectors that use fewer channels wait until definitive ictal activity is evident on the channels before declaring seizure. This suggests that the detectors that use fewer channels miss the windows that are at the beginning of a new seizure. Further investigation will be required to understand this phenomenon.

#### 4.11 APPLICATION OF CHANNEL SELECTION APPROACH

We have demonstrated the utility of the wrapper channel selection algorithm to build reduced channel, patient-specific seizure onset detectors using pre-recorded EEG data. In this section, we discuss how one might apply the wrapper method to build a reduced-channel seizure onset detector for a new patient with epileptic seizures.

First, ictal and non-ictal EEG data must be recorded from the patient. This data can be obtained by admitting the patient to a hospital or by using an ambulatory EEG



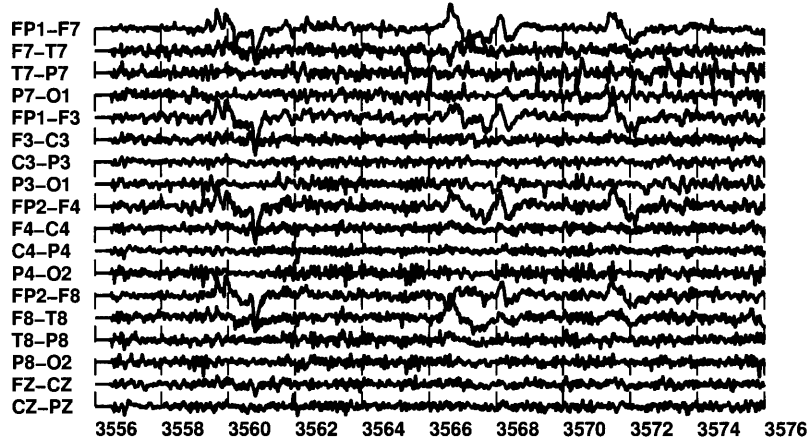
**Figure 4.22:** For patient 16, on Training Set 1, the number of false positives and false events decrease with an increasing number of channels. The horizontal line indicates the maximum false positive rate allowable; it illustrates why a channel subset of size 16 was selected for the reduced channel detector.

recorder. We then use this data to build a reduced channel detector for detecting future seizure onsets. To train a detector, we first organize the data as follows. We divide the recording into one-hour records. Each one-hour record is then further subdivided into two second windows and features are extracted from each window. The labels for each window are derived automatically using seizure onset markings provided by a human expert.

Next, using this data, our goal is to determine the appropriate channel subset for detecting future seizures and to evaluate how well a reduced channel detector built using those channels will perform. There are several possible approaches for identifying an appropriate channel subset. We will describe one possible approach.

The first step is to construct multiple training and test sets using the leave-one-record-out procedure we described in Section 4.6. For each training set, we use the wrapper method to find a channel subset. If there are  $N$  total seizure records, this procedure will yield  $N$  different channel subsets. If the  $N$  channel subsets are identical, then we can be confident that the reduced channel subset will perform well on future data and thus, deploy using that subset.

If the  $N$  channel subsets are not identical, we could select one of the subsets to deploy. For each channel subset, we would train a reduced channel detector using data from all the training subsets (used to find the reduced channel detectors) and test each detector on the withheld records. Using the test results, we could then select the channel subset that achieves the best performance. This channel subset could then be deployed as long as the performance of this reduced channel detector meets a user-defined level, otherwise, a reduced channel detector should not be deployed. One



**Figure 4.23:** A segment during a non-seizure record. A seizure is declared at 3566 seconds into the recording.

advantage of this approach is that the selection of the final channel subset is conducted at the same time as the evaluation of the subsets.

Note that this is only one out of many other possible methods to select the best channel subset. To determine the best possible approach however, will require more investigation and analysis.

#### 4.12 OTHER CHANNEL SELECTION APPROACHES

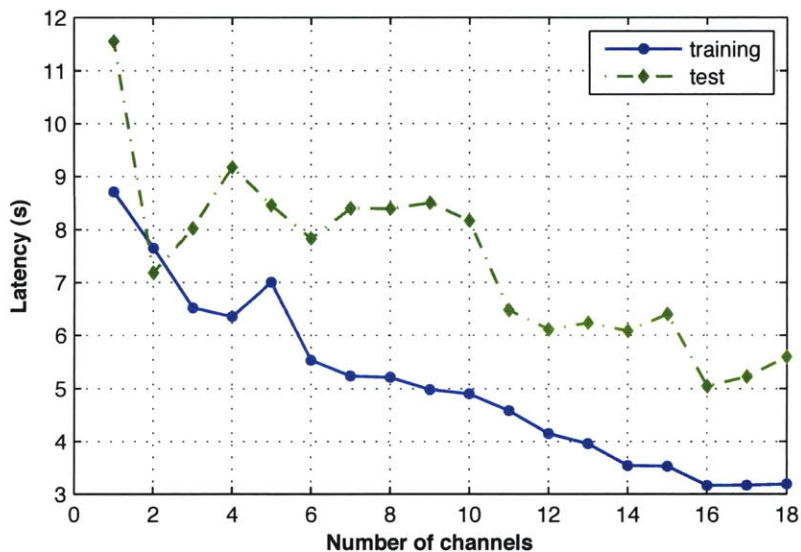
As mentioned in Section 4.5.1, in addition to the wrapper method, there are other feature selection algorithms that we can apply to the channel selection problem. In this section, we describe these approaches and use them to select channels and to construct SVM-based seizure onset detectors. Following this section, we compare the performance of these detectors to the reduced channel detectors constructed using the wrapper algorithm.

##### 4.12.1 Fisher Criterion

One way to select channels is to rank features. A simple ranking function is the Fisher Criterion. The Fisher Criterion is a measure of how strongly a feature is correlated with the labels [6]. The Fisher Criterion can be used to measure how well each feature can separate the training examples into two classes, non-ictal and ictal. Let  $X$  be the set of feature vectors where each element  $\mathbf{x}$  is a vector of  $m$  input features or variables, *i.e.*,  $\mathbf{x} = \langle x_1, x_2, \dots, x_m \rangle$ . Define  $\mu_j(X)$  and  $V_j(X)$  to be the mean and variance of feature  $j$ . Thus, for a feature  $j$ , the Fisher Criterion score  $R_j$  can be computed as

$$R_j(X) = \frac{[\mu_j(X^+) - \mu_j(X^-)]^2}{V_j(X^+) + V_j(X^-)} \quad (4.5)$$





**Figure 4.24:** Average training and test detection latencies as a function of the number of channels. The average test latency decreases with increasing number of channels.

where  $X^+ = \{\mathbf{x}_i \in X \mid y_i = 1\}$  and  $X^- = \{\mathbf{x}_i \in X \mid y_i = -1\}$ . Equation (4.5) shows how to compute the Fisher Criterion for a single feature. Since a channel consists of multiple features, we use the mean score of the features that correspond to that channel as the Fisher Criterion of a channel. The authors of [27] use this ranking criterion to perform channel selection for a brain-computer interface.

To evaluate the effectiveness of the Fisher Criterion, we applied it on a patient-specific basis. For each patient, we then constructed a set of SVMs. An SVM was constructed using the top  $k$  channels where  $k = 1, 2, \dots, 17$ . Each of these SVMs were then used as part of a seizure onset detector. From this list of channel subset detectors, we then selected the detector that uses the fewest number of channels that correctly detected as many seizure onsets and had the same or better true negative rate as the original ALLCHANNEL detection algorithm.

#### 4.12.2 Support Vector Machine Recursive Feature Elimination

Support Vector Machine Recursive Feature Elimination (SVM-RFE) is a specific instance of backward feature elimination that uses the model parameters from a trained SVM in the objective function. In this section, we give a brief overview of the algorithm adapted for use with non-linear kernels. For more details, refer to [22].

To evaluate feature subsets, SVM-RFE uses an objective function that looks at the change in a cost function that occurs when a feature is removed. In each iteration of the SVM-RFE algorithm, the feature that is removed is the one that results in the smallest change in this cost function.

In linear binary classification problems, the cost function is the size of the margin,

essentially, the distance between the positive and negative classes. In any iteration, the feature that results in the smallest decrease in the margin is, in some sense, the least useful feature. Thus, of the remaining features, it is the best feature to remove.

In the non-linear case, the concept of a margin is not well-defined for all non-linear boundaries. Instead, the cost function must be derived from the dual formulation of the optimization problem used to construct SVMs. In the dual formulation, the following problem is solved:

$$\begin{aligned} \operatorname{argmax}_{\mathbf{a}} \quad & \tilde{L}(\mathbf{a}) = \sum_{i=1}^N a_i - \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N a_i a_j t_i t_j k(\mathbf{x}_i, \mathbf{x}_j) \quad \text{subject to} \\ & 0 \leq a_i \leq C, \\ & \sum_{i=1}^N a_i t_i = 0 \end{aligned}$$

where  $\mathbf{a} = \langle a_1, \dots, a_N \rangle$  and  $\mathbf{x}_1, \dots, \mathbf{x}_N$  are  $m$  dimensional feature vectors with corresponding target values  $t_1, \dots, t_N$  where  $t_i \in \{-1, +1\}$ . In general,  $k$  is a kernel function. In our work, we used the radial basis kernel function  $k(\mathbf{x}_i, \mathbf{x}_j) = \exp(-\gamma \|\mathbf{x}_i - \mathbf{x}_j\|^2)$  where  $\gamma$  is a parameter that determines the “radius” of the function.

Define the objective function,  $J(k)$  to be the change in  $\tilde{L}(\mathbf{a})$  that results when the feature  $k$  is removed. To compute  $J(k)$ , we remove feature  $k$ , leave the  $a_i$ 's unchanged, and re-compute  $\tilde{L}(\mathbf{a})$ . Formally,

$$\begin{aligned} J(k) &= \tilde{L}(\mathbf{a}) - \tilde{L}(\mathbf{a}, -k) \\ &= \frac{1}{2} \sum_{i=1}^N \sum_{j=1}^N a_i a_j t_i t_j [k(\mathbf{x}_i(-k), \mathbf{x}_j(-k)) - k(\mathbf{x}_i, \mathbf{x}_j)] \end{aligned}$$

where the notation  $(-k)$  indicates that feature  $k$  has been removed. Note that since  $\mathbf{a}$  is left unchanged, we do not train a classifier every time we remove a feature. The feature that results in the smallest  $J(k)$  is the one that we remove before continuing on with the algorithm. Note that in the solution to the SVM problem, only a subset of the  $a$ 's will be non-zero. Thus, computing  $J(k)$  will involve adding much fewer than  $N^2$  terms.

The SVM-RFE approach was adapted for the channel selection problem. Thus, instead of removing features, entire channels were removed. We used SVM-RFE to rank the 18 channels used for seizure onset detection. For each patient, we then constructed a set of SVMs. An SVM was constructed using the top  $k$  channels where  $k = 1, 2, \dots, 17$ . Each of these SVMs were then used as part of a seizure onset detector. From this list of channel subset detectors, we then selected the detector that uses the fewest number of channels that correctly detected as many seizure onsets and had the same or better true negative rate as the original ALLCHANNEL detection algorithm.

### 4.12.3 Comparing Different Channel Selection Approaches

Table 4.12 compares the average performance of the detectors for each patient that was constructed as a result of various feature selection methods. The table suggests that other approaches can be used to find reasonable channel subsets. However, the

**Table 4.12:** Performance of other channel selection approaches compared to the wrapper approach. Other methods seem to produce reasonably good subsets. However, in terms of number of channels, the wrapper algorithm seems to do the best.

Method	False events per hour	Detected events	Mean latency (s)	Number of channels	Energy saved (%)
Baseline	0.07	139	6.5	18.0	–
Wrapper	0.11	136	8.2	6.4	60%
Fisher	0.08	135	6.5	11.7	33%
SVM-RFE	0.14	139	6.4	8.6	44%

wrapper approach seems to favor choosing smaller channel subsets while achieving comparable detection performance.

#### 4.13 SUMMARY

In this chapter, we described a simple channel selection technique for reducing the energy consumption of medical monitoring devices that use multiple sensors. and conducted a comprehensive study of the impact of using fewer channels. From these results, we want to emphasize two key points:

- (a) *Optimal channel subset depends on objective function.* Through the course of analyzing the results, we discovered that the subsets suggested by the algorithm depend on the objective function used to evaluate the subsets. The objective function we used was a function of the training performance parameters of learned detectors and preferred those detectors that used a smaller number of channels.
- (b) *Optimization based on training performance can generalize.* The objective function we used to evaluate channel subsets was defined in terms of parameters based on the *training* performance of detectors. Optimization based on training error or other training parameters does not always result in good test performance since the resulting detectors can be overfit to the training data—indeed, this is likely true for the patients where there was high variance in the size of the channel subsets. However, in this application, optimization based on training parameters was not an issue for many patients, because the seizure onsets the detectors were trained on were fairly similar in spectral content to the seizure onsets the detectors were validated on.



---

## Combining Duty Cycling and Sensor Selection

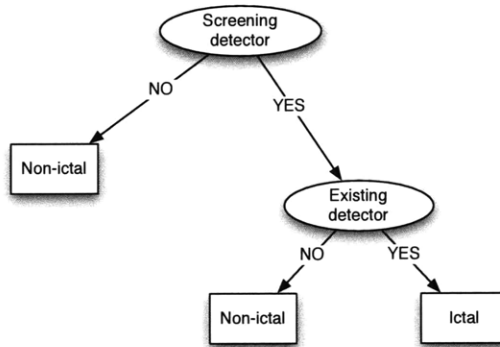
In a multi-sensor event detection application, energy consumption can be reduced both by careful sensor selection and by duty cycling the components in the system. Both duty cycling and sensor selection methods achieve energy savings by reducing the amount of data collected or processed. If the removed data contains important information, then the risk of using less data is that detection performance—specificity, sensitivity, and detection latency—may be degraded. In this chapter, we explore two energy reduction methods that combine duty cycling with sensor selection in the context of seizure onset detection.

The first method reduces energy consumption by gating the operation of a good, but energy inefficient detector with a *screening detector*, an algorithm that approximates the behavior of the base detector while using fewer resources. Usually, designing an appropriate screening detector requires application-specific knowledge. In this chapter, we use the wrapper method described in Chapter 4 to automatically generate screening detectors. In the second approach, we duty cycle a reduced channel detector on a window-by-window basis. The duration and frequency of off periods can be adapted based on the input during on periods.

Both approaches assume that the input is non-stationary and moreover, that there are indicators within the input signal that can be used to signify an impending event onset. In the rest of this chapter, we describe how we applied the screening and duty cycling methods to the problem of seizure onset detection.

### 5.1 CONSTRUCTING SCREENING DETECTORS

Figure 5.1 shows how a screening detector could be used to duty cycle an existing “good” detector in the context of seizure onset detection. In general, however, the screening detector could be designed to label windows of any type that are relatively easy to identify. The screener forwards ambiguous windows to the original detector, which can then distinguish ictal data from non-ictal data. In this example, the screening detector is designed to label examples that are definitively non-ictal. If the



**Figure 5.1:** A two-level detector.

screeener suspects that a window contains ictal activity, it forwards the window to the original detector. At this point, the original detector takes over analysis of future windows until a non-ictal window is observed. Note that, by construction, the combined detector will have no false positives relative to the original algorithm, but could have false negatives.

In constructing a screener, to achieve good overall detection performance, the screener must have low false negatives relative to the original detector; false positives, however, are acceptable. If the screener uses less energy per label than the original detector, and the number of windows that the screener does not know how to label is small enough such that the cost of processing the window twice does not dominate the savings obtained on other windows, then the use of such a detector will result in energy savings.

Our concept of screening is a software-based idea that is similar to the idea of using a lower energy, but lower fidelity device in order to gate the operation of a higher power device. In [1, 38, 42], the authors use low-power radios to listen for incoming messages. This allows the rest of the device, including high-power radios, to remain off when the device is not being used. By using the low-power radio to handle incoming messages, dramatic energy savings can be realized.

Similarly, the authors of [5] describe a framework that is used to automatically generate a power-efficient decision tree classifiers for wearable gait monitoring nodes. Essentially, they reduce the energy consumption of the sensors by putting them in various sleep states. Their system can dynamically adjust the activation and sampling rate of sensors such that only the data necessary to determine the system state is collected at any time.

Screening can be viewed as a data dependent duty cycling method since a screening detector must analyze input data to determine when to activate the second detector. For general detection or monitoring applications, people typically use a manual approach for screener construction. For seizure onset detection, we describe semi-automatic approach for screener construction. We start with a reduced channel detec-

tor built using the approach described in Chapter 4. Then, we use a slightly modified version of the backward elimination algorithm to construct a screening detector that uses fewer channels than the reduced channel detector. In the modified version, we do the following.

- (a) First, instead of selecting a subset from all 18 channels, we select a subset of the channels used by the reduced channel detector.
- (b) Second, we expect that a classifier that uses fewer channels may have lower detection rates. To avoid having too many false negatives relative to the reduced channel detector, we bias the SVM algorithm to make fewer mistakes on the seizure windows. We set the penalty for making an error on a true positive example four times higher than the penalty for making an error on a true negative ( $C_{\text{positive}} = 40$  as opposed to  $C_{\text{negative}} = 10$ ).
- (c) At the end of each round, we determine which channel to discard by determining the performance of the detector formed by combining each candidate screener with the original detector. We remove the channel that produces a combined detector that exhibits the smallest degradation in the false negative rate.

After all rounds, we must select one final screener from the candidate screeners. At this point, we combine each candidate screener with the original subset detector. From this set, we select the combined detector that detects as many seizures as the original reduced channel detector with the lowest false negative rate.

Note that there are a few cases where we cannot use a screener at all. If the original subset detector uses only a single channel for detection, then we cannot perform further channel reductions. In addition, if there are no combined detectors that achieve the detection rate of the original subset detector, then we will also choose not to screen.

## 5.2 EVALUATION

We used the same evaluation method described in Section 4.6 to evaluate the combined detectors. For each patient, the entire EEG recording was split into one-hour records. From these records, multiple training data sets were constructed and used as input to the screener construction algorithm. The resulting screener was combined with the reduced channel algorithm constructed using the same training data set as input.

To determine detector performance, we executed the detector on the union of the withheld seizure-free records and the one-hour record containing seizure data that was excluded during training. For each of the detectors, we recorded the same metrics as before: (a) the number of true seizures detected, (b) the number of false events declared per hour of monitoring, (c) the detection delay for each detected seizure in the recording and (d) the number of channels used and energy consumed by each detector. For each patient, we compared the performance of the combined detectors to the performance of the subset detector to assess the performance of screening.

To determine the energy consumed by the combined detector, we used the following energy model:

$$E_{\text{total}} = N_{\text{screen}}E_{\text{screen}} + N_{\text{reduced}}E_{\text{reduced}} + N_{\text{combined}}E_{\text{combined}}$$

where  $E_{\text{screen}}$  is the energy consumed per second by just the screening detector,  $E_{\text{reduced}}$  is the energy consumed per second when just the reduced detector runs, and  $E_{\text{combined}}$  is the energy consumed per second when both detectors are executed. The total energy depends on how often we run each of the detectors.

The impact of screening on energy consumption depends on the architecture of the system. In our prototype, computation occurs remotely (*e.g.*, at the belt). Thus, data must be transmitted from the head. When the screening detector is able to classify an EEG window, only the data from the channels used by the screening detector needs to be transmitted, however, the channels that are used by the combined detector still need to be sampled, in case the screening detector needs to consult the original, reduced channel detector. In this architecture,  $E_{\text{screen}}$  is equal to  $E(N_c)$  (Equation 4.4) where  $N_c$  is equal to the number of channels in the screening detector plus the energy required to sample the additional channels of data. On the other hand,  $E_{\text{reduced}}$  can be determined just using Equation 4.4. Finally,  $E_{\text{combined}}$  is equal to  $E_{\text{screen}}$  plus the energy required to transmit the channels that are not used by the screening detector, but are used by the original, reduced channel detector.

A different way to structure a combined detector is to use the screener to determine whether or not the *next window* should be analyzed by the screening detector or the combined detector. The advantage of this structure is that only the data needed by the screener would need to be sampled. If storage and sampling costs are high, this combined detector would use less energy. The downside is that specificity and sensitivity could degrade. In this thesis, we did not evaluate the performance of such a detector.

### 5.2.1 Number of Channels

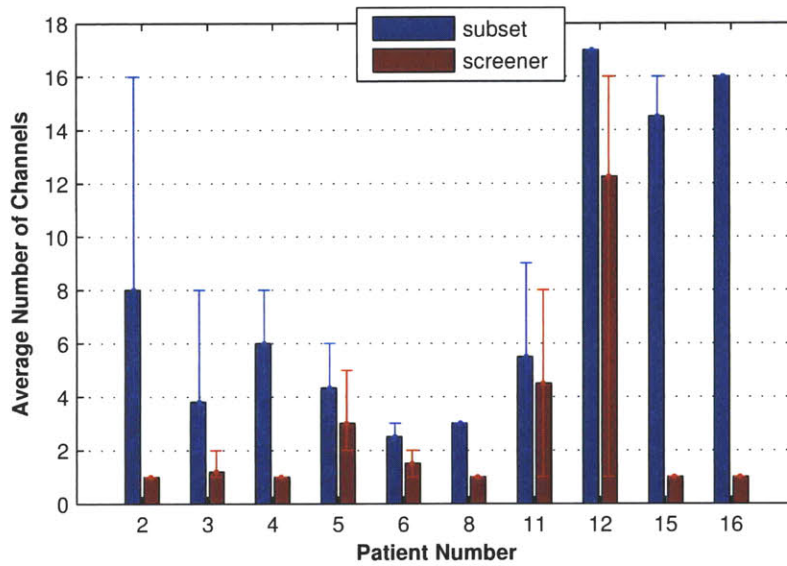
We were unable to construct a screener for only 6 out of the 16 patients, since we were able to detect seizures using a one-channel detector. A one-channel detector cannot be further reduced. Figure 5.2 shows the average number of channels used by the screening detectors for each patient. The error bars show the maximum and minimum size of the channel subsets selected for each patient. Overall, the average number of channels used by the screener detectors was 2.7 channels as opposed to 8.0 channels used by the subset detectors. The high variance in the average number of channels across patients can be attributed to variability of seizures across patients.

### 5.2.2 Seizure Onset Detection Performance

Table 5.1 compares the detection performance of the combined detectors to the detection performance of both the ALLCHANNEL and SUBSET seizure onset detectors. Overall, for the patients where we could apply screening, detection performance slightly degraded. 78 of the 80 seizures detected by the SUBSET were detected by the combined detectors.

For one of the seizures missed by the combined detector—the seizure for patient 6—the detection latency of the SUBSET detector was greater than 50 seconds. Since detecting a seizure after more than 50 seconds could be considered a miss, it is not clear that the miss by the screened detector should be counted. In the other case (Patient 4), adding a screener caused a seizure to be missed. However, the missed seizure was the same seizure that the ALLCHANNEL detector missed.





**Figure 5.2:** The average number of channels used by the subset detectors and screening detectors for all patients where screening could be applied. The error bars show the maximum and minimum size of the channel subsets. High inter-patient variability is due to variation in the spatial distribution of seizures across patients.

**Table 5.1:** Detection performance of the combined, ALLCHANNEL, and SUBSET detectors on a subset of patients. Overall, for the patients where we could apply screening, detection performance was degraded. 78 of the 80 seizures detected by the SUBSET were detected by the combined detectors.

Patient	Total seizures	ALLCHANNEL	SUBSET	combined
2	4	3	3	3
3	7	7	6	6
4	3	2	3	2
5	3	3	2	2
6	2	2	1	0
8	3	3	3	3
11	2	2	1	1
12	4	4	4	4
15	38	38	38	38
16	19	19	19	19
Sum	85	83	80	78

### 5.2.3 False Events Per Hour

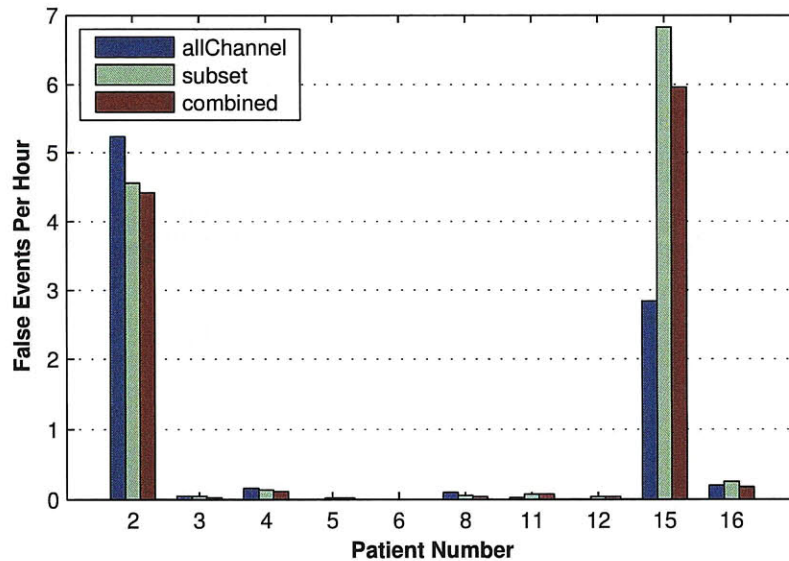
Figure 5.3 compares the per-patient false event per hour for the three different detectors. We expected the false event per hour to be no worse when we added a screening detector. In fact, the false alarm rate decreased for all ten patients. This implies that the screening detectors, by using fewer channels, were able to filter out the false positives of the subset detector. For the patients that we were able to screen, the false alarm rate decreased slightly from 1.2 to 1.1. For patient 15, the decrease was almost one per hour.

### 5.2.4 Detection Latency

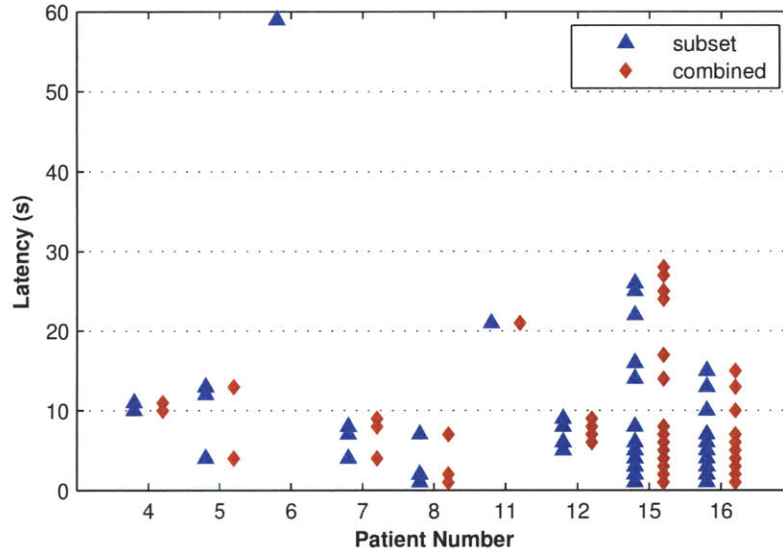
Figure 5.4 plots the detection delay for the SUBSET and combined detectors for each patient. Each point in the graph represents the detection delay of either the SUBSET or combined detector for a single seizure. In aggregate, the detection latencies of the combined detectors are comparable to those of the SUBSET detector. We observed a slight increase in the median latency— from 6.2 to 6.3 seconds. For most patients, the median latency stayed the same or increased slightly.

### 5.2.5 Impact on Energy and Lifetime

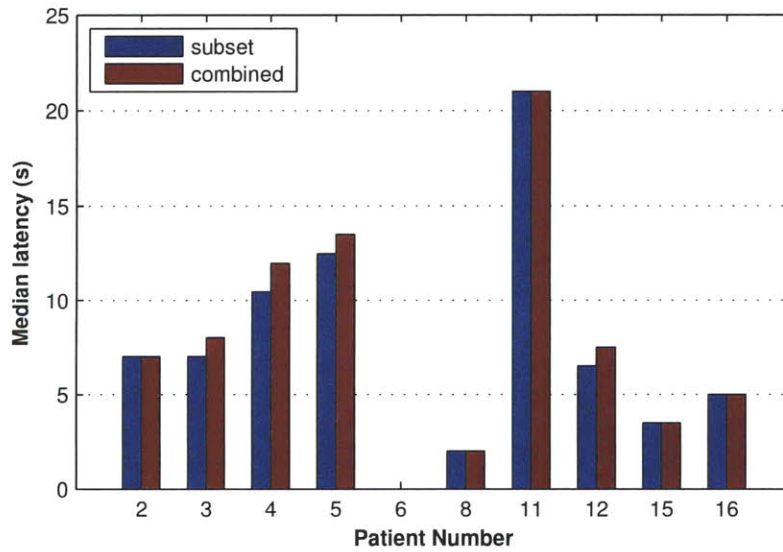
Figure 5.6 shows the average energy savings for all patients. Over the ten patients, by adding a screening detector, we reduced the energy consumed by 69% on average relative to the original 18-channel detectors. In contrast, the subset detectors used only 50% less energy.



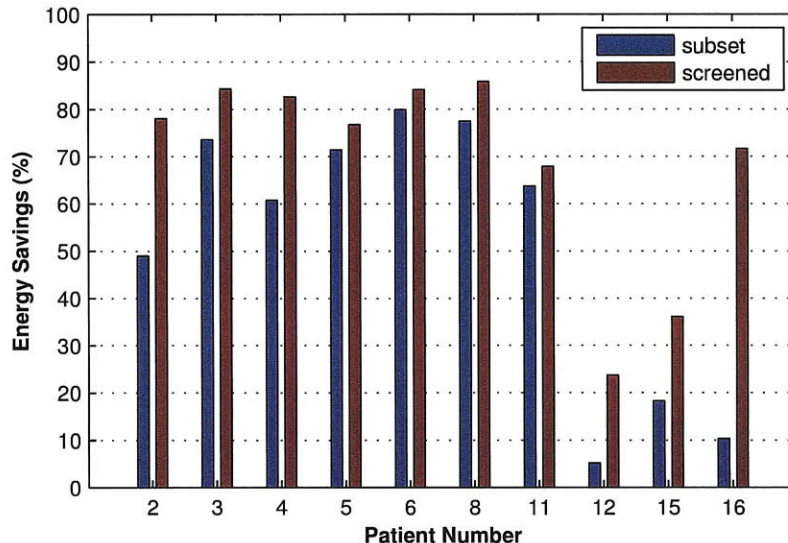
**Figure 5.3:** The average per-patient false events per hour decreased slightly from 1.2 to 1.1 when we added a screening detector.



**Figure 5.4:** The distribution of detection delays for the SUBSET and combined detectors appears similar for all patients except for Patient number 6, where the combined detector missed all the seizures.



**Figure 5.5:** The per-patient median latency for seizure onsets detected by both detectors. For Patient 6, we have no latency to report since the combined detector did not detect any seizures.



**Figure 5.6:** The energy consumed by the combined detectors was 69% less than the energy consumed by the original 18-channel detectors. In contrast, the subset detectors used only 50% less energy.

### 5.3 IMPACT OF DUTY CYCLING ON REDUCED CHANNEL DETECTORS

In an application where multiple sensors are involved, energy savings can be realized by reducing the amount of data processed in space, *e.g.*, selecting a subset of sensors, or by reducing the amount of data processed in time, *e.g.*, duty cycling. In theory, by simultaneously optimizing duty cycling and sensor selection parameters, a desired level of detection performance can be achieved using minimum energy.

For the seizure onset detection problem, we optimized the number of channels before optimize the duty cycling parameters because having fewer channels also improves patient comfort. Thus, instead of performing a joint optimization, we started with the patient-specific reduced channel detectors constructed in Chapter 4 and determined the impact of duty cycling of each detector on performance.

To duty cycle, we use the algorithm shown in Figure 5.7. The algorithm is used to determine when to sample data for and process each window. We assume that executing this algorithm is negligible compared to the other processes needed for detection. Assume that each window contains a fixed number of samples. For duty cycling the seizure onset detectors, we flip a coin that has a probability  $p$  of ending up with a “success”. If the outcome of the trial is a “success,” we wake up the device and call the function  $\text{RESET}(p, w)$ . This function is used to adapt  $p$ . For example, we can increase  $p$  to improve the odds of detecting a seizure if after analyzing the window, we know that an event will be more likely. In our case, the  $\text{RESET}$  function sets  $p$  to 1 if  $w$  is positive, *i.e.*, contains seizure activity. Otherwise, it sets  $p$  to  $p_{\text{init}}$ . After  $\text{threshold} = 4$  consecutive positive windows, a seizure is declared. If the coin toss results in

---

```

1:  $p \leftarrow p_{\text{init}}$ 
2:  $N_{\text{pos}} \leftarrow 0$  {The number of consecutive positive windows observed}
3: for all windows  $w$  do
4:   if  $\text{RANDOM}() \leq p$  {Get uniform random number in  $[0, 1]$ } then
5:      $p \leftarrow \text{RESET}(p, w)$  {Set  $p$  to 1 if  $w$  is positive}
6:     if  $w$  is positive then
7:        $N_{\text{pos}} \leftarrow N_{\text{pos}} + 1$ 
8:       if  $N_{\text{pos}} > \text{threshold}$  then
9:         Declare event
10:        Halt
11:      end if
12:    else
13:       $N_{\text{pos}} \leftarrow 0$ 
14:    end if
15:  else
16:    Skip  $w$ 
17:  end if
18: end for

```

---

**Figure 5.7:** The adaptive duty cycling algorithm. For seizure onset detection *threshold* is equal to 4.

a “failure,” then we skip analysis of the window and remain off.

Using the algorithm, the probability of detecting a seizure can be computed as follows. If  $N$  consecutive positive labels are required before declaring an event, then the probability of missing a seizure with duration  $L$  is  $(1 - p)^{L-N+1}$  since the Bernoulli trials are independent and we miss the seizure if we don’t wake up for all but the last  $N - 1$  windows.

The energy consumed by a device using this algorithm is directly proportional to the on time of the device. In our algorithm, the fraction of time the device is on depends on the initial probability  $p_{\text{init}}$  and the fraction and distribution of positive seizure windows within the EEG recording. Assume we have a fixed length recording with a constant fraction of positive seizure windows. If we have a uniform distribution of positive windows, then there will likely be many short bursts of positive windows within the recording, some of which will be seizures. To detect these short seizures,  $p_{\text{init}}$  will need to be larger and thus, the device will be on more often. If on the other hand, the positive windows tend to cluster, there will be fewer events, but each event will be longer. Thus, to detect all the seizures detect, a smaller  $p_{\text{init}}$  will be needed and the device will be on less. In the seizure onset detection application, seizure windows tend to cluster. Therefore, we expect to have good detection performance with reasonably small values of  $p_{\text{init}}$ .

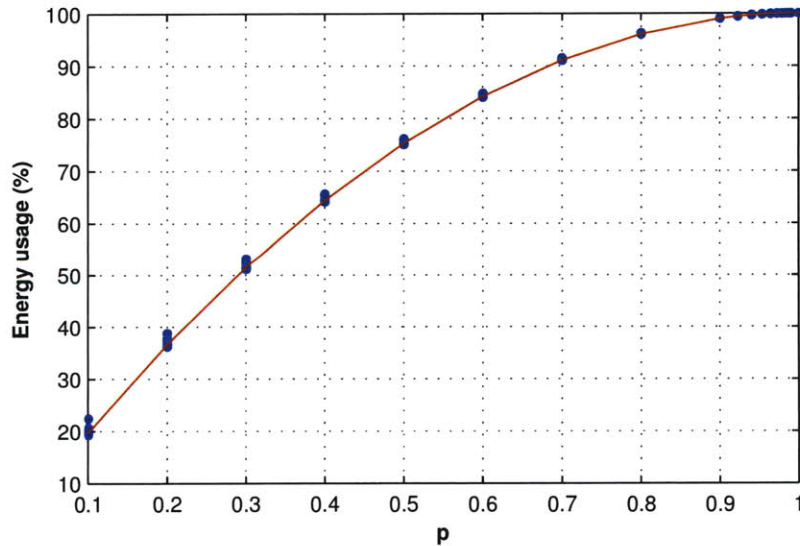
## 5.4 EVALUATION

To determine the effect of our duty cycling method on the detection of seizure onsets for a single patient, we started with the reduced channel detectors resulting from running the channel selection algorithm on the training subsets. We then duty cycled each reduced channel detector using different values of  $p$  while using the test data set as input. For each detector and value of  $p$ , we recorded usual metrics. Since our duty cycling approach is probabilistic, we performed this simulation 50 times and then took an average over the trials. In this section, we report the results of our simulations.

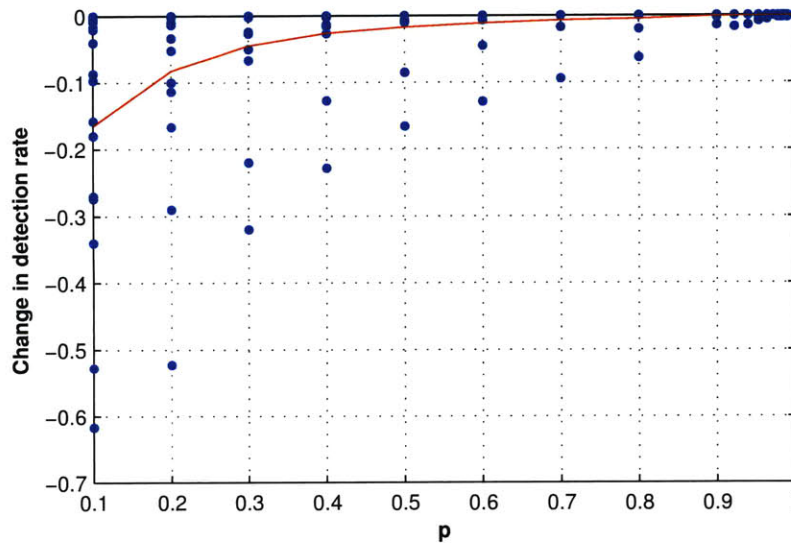
Figure 5.8 plots the energy consumed as a percentage of the energy consumed by the non-duty cycled detector as opposed to the parameter  $p$  for all patients. Each point in the plot represents the average energy usage by a detector for a particular patient for a fixed  $p$ . As expected, with a small  $p$ , less energy is consumed. However, the relationship between  $p$  and energy usage is not linear because when a positive window is detected we essentially stop duty cycling until a negative window is encountered.

Figure 5.9 plots the change in detection rate as a function of the parameter  $p$ . The change in detection rate is computed by taking the difference between the detection rate of the non-duty cycled algorithm and the detection rate of the duty cycled algorithm with parameter  $p$ . Each point in the plot represents the change in the average detection rate achieved by a patient-specific detector with  $p$  set to a different value. The red line illustrates the mean detection rate over all patients.

As expected, the average detection rate increase dramatically for as  $p$  increases. For patients where seizures can last several windows, our probabilistic duty cycling



**Figure 5.8:** The energy consumed as a percentage of the energy consumed by the non-duty cycled detector as a function of  $p$ , the duty cycling parameter.



**Figure 5.9:** Average change in detection rate as a function of  $p$ . The change in detection rate is small even for small  $p$  since some seizures are long and are easily detected. The difference rapidly converges towards 0 as  $p$  increases. At  $p = 0.5$ , the average decrease in detection rate is only 2%.

method will detect the seizure as long as enough windows are analyzed once the device wakes up. As a trade-off, the average latency of detection will be larger for small values of  $p$ , as we see in Figure 5.10. For small values of  $p$ , the change is fairly large. As  $p$  is increased, the difference gets progressively smaller. At  $p = 0.5$ , the average change in latency is 1.1 seconds, while the average detection rate degrades by only 2%. This level of performance can be achieved using 25% less energy.

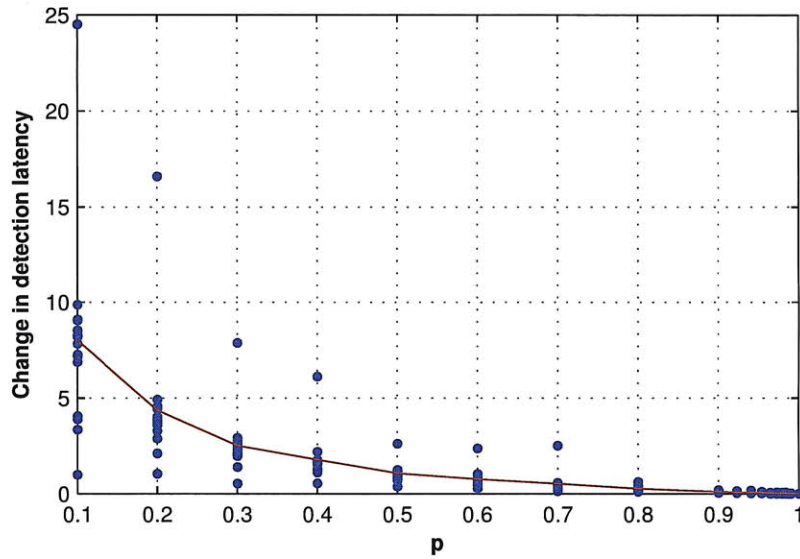
Since all windows that are skipped are labeled negative, by definition, any duty cycling we perform will lower the false events per hour. This is confirmed in Figure 5.11, where for any value of  $p$ , we see that the change in average false events per hour is negative. At  $p = 0.5$ , the average change in false events per hour is  $-0.3$ .

## 5.5 CASE STUDIES

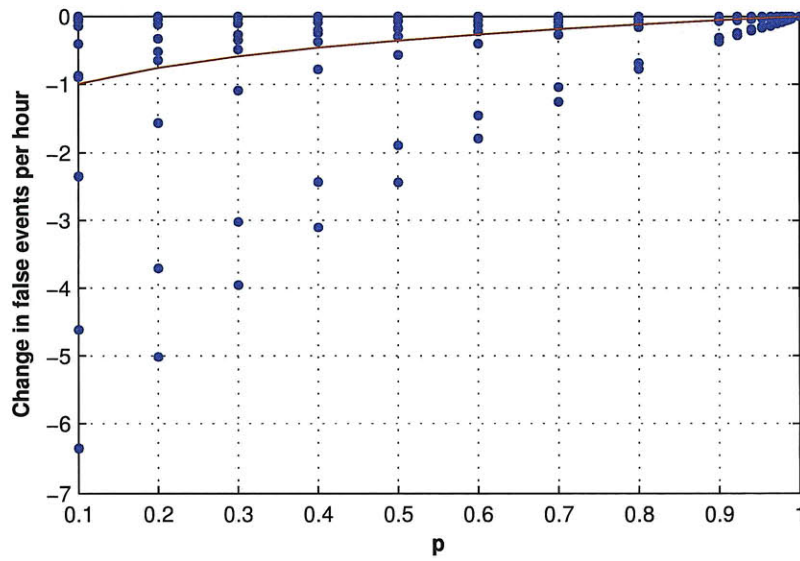
Adding duty cycling to the reduced channel detector indicates that reducing the energy consumed by 25% results in only a small degradation in the mean detection rate (2%) and mean detection latency (1.1 seconds) while *reducing* the false alarm rate by 0.3 per hour. In this section, we examine two specific patients, one where duty cycling had minimal impact on detection rate (Patient 15) and one where duty cycling had a dramatic impact on detection rate (Patient 3).

### 5.5.1 Patient 15

Figure 5.12 shows the detection rate as function of  $p$  for patient 15. Surprisingly, we see that reducing the energy consumption by more than 45% ( $p = 0.3$ ) has no



**Figure 5.10:** The average change in latency as a function of  $p$ .



**Figure 5.11:** The average change in false alarm rate as a function of  $p$ . Since any window that is skipped is considered a negative, lower false alarm rates can be achieved with lower values of  $p$ .



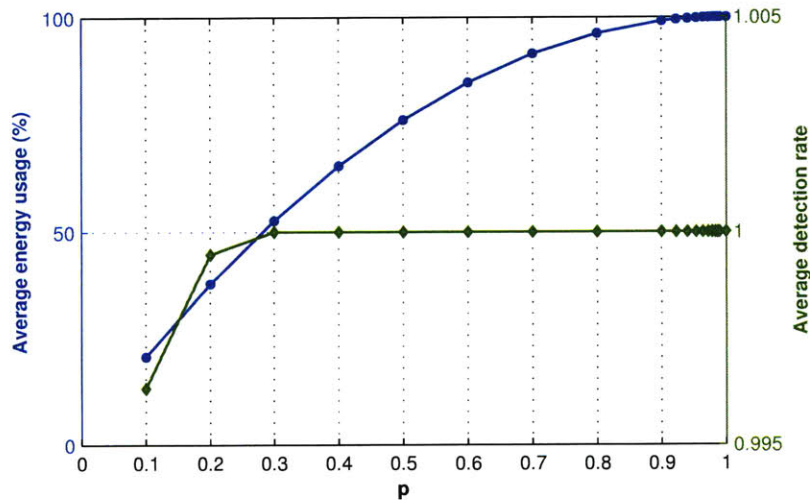
effect on the detection rate—we have a 100% detection rate. Since the median seizure duration for patient 15 was around 60 seconds, the worst case probability of detecting a seizure is about  $1 - (1 - p)^{60-3}$ . Recall that our algorithm declares a seizure after four consecutive positive windows have been observed. For  $p = 0.3$ , the probability of detection is essentially one. At  $p = 0.3$ , we are able to detect all the seizures with minimal energy consumption ( $\approx 53\%$ ).

Another benefit of duty cycling is that the number of false alarms per hour will drop as  $p$  is decreased. This is evident in Figure 5.13. The trade-off is that the detection latency will increase.

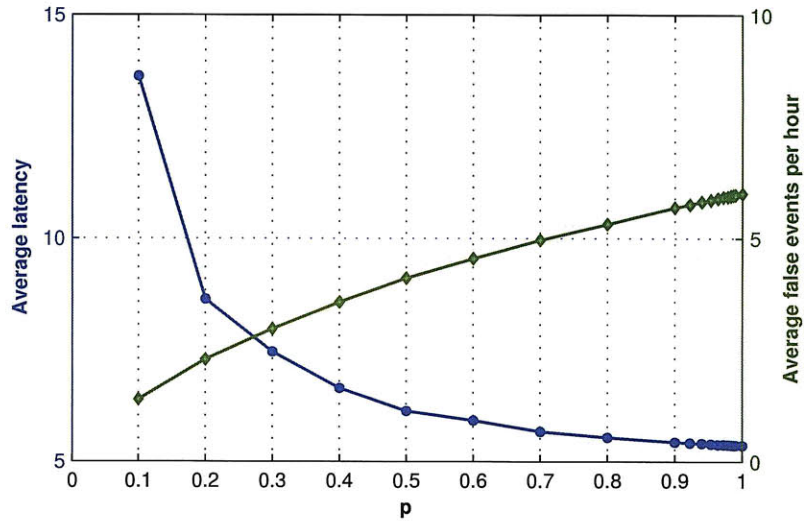
### 5.5.2 Patient 3

Figure 5.14 shows the detection rate as a function of  $p$  for patient 3. In contrast to our results for patient 15, here, we see that there is no value of  $p$  for which we are able to detect all the seizures that the original non-duty cycled algorithm was able to detect. The reason is that the median seizure duration for patient 15 was 5 s. Thus, detection rates even for high values of  $p$  will still be low. Using our model, we computed the probability of detecting a typical seizure as  $1 - (1 - p)^2$ . For  $p = 0.5$ , the probability of detection is about 0.75.

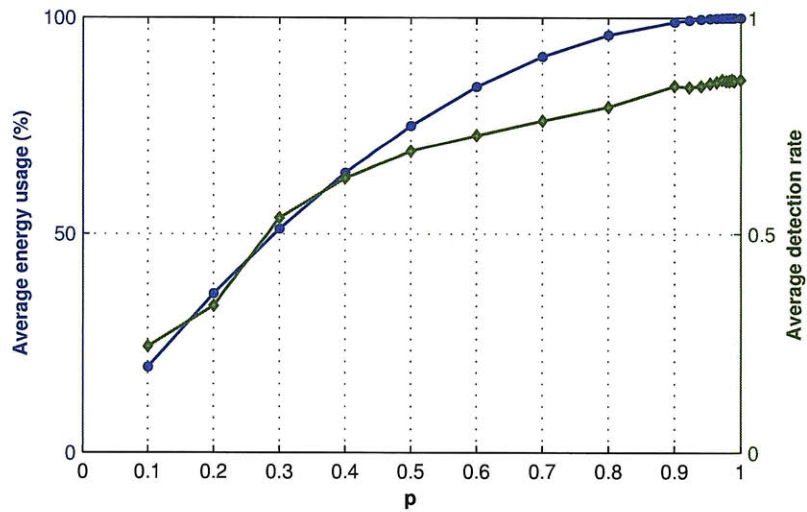
As with patient 15, increasing the off time through more duty cycling decreases the false alarms per hour (see Figure 5.15).



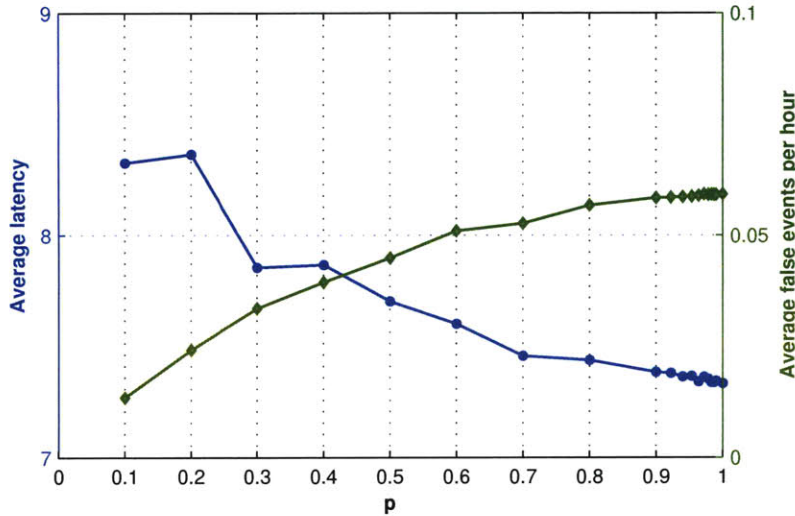
**Figure 5.12:** The average energy consumed as a percentage of the energy consumed by the non-duty cycled reduced channel detector and the detection rate plotted as a function of  $p$  for patient 15. As expected, increasing  $p$  improves the detection rate and also increases the amount of energy consumed.



**Figure 5.13:** Latency and false alarm rate per hour plotted versus  $p$  for patient 15. For small  $p$ , the false alarm rate per hour is reduced dramatically. The cost of having such a low false alarm rate is a high detection latency. At  $p = 0.3$ , we are able to detect all the seizures with minimal energy consumption ( $\approx 53\%$ ).



**Figure 5.14:** The average energy consumed as a percentage of the energy consumed by the non-duty cycled reduced channel detector and the detection rate plotted as a function of  $p$  for patient 3.



**Figure 5.15:** Latency and false alarm rate per hour plotted versus  $p$  for patient 12.

### 5.5.3 Summary

In this chapter, we explored the usefulness of two duty cycling techniques: screening and duty cycling.

### 5.5.4 Screening Approach

In screening, the on-off state of the detector is controlled using the output of a less accurate, but lower energy detector instead. Lower energy detection is achieved on a per-window basis by analyzing fewer channels. The original detector, that uses more channels, is consulted only when the less accurate detector suspects an event. If the number of windows that the screener does not know how to label is small enough such that the cost of processing the window twice does not dominate the savings on windows where only the screener is executed.

For our patient population, the use of a screening detector was not always possible or helpful. For the remaining population, on average, 69% additional energy savings was realized. Moreover, the screening detector used 2.8 channels. In terms of detection performance, 74 out of the 76 seizures detected by the subset detectors were detected when screening was added. The median latency increased slightly from 7.3 to 7.8 seconds, but the false alarm per hour decreased from 0.77 to 0.73.

Overall, the screening detectors we constructed did not dramatically improve energy consumption and had marginal impact on the other metrics. The high inpatient variability in the number of channels suggests there is no consistent subset of channels that can be derived from the remaining channels. Moreover, from the results, one could conclude that by reducing the number of channels from 18 to 6.4 using channel selection, the additional use of screening offers diminishing returns.

### **5.5.5 Duty Cycling**

Intuitively, when duty cycling is applied to any detection algorithm, the detection rate will suffer. However, the degree of degradation will vary depending on the nature of the events being detected. For patients with long lasting seizures, missing an event is unlikely even using the duty cycling approach we describe—though detection latency will suffer. On average, using 25% less energy yielded only a 2% drop in detection rate.

---

## Summary and Conclusions

In this thesis, we examined the impact of using less data on the performance and energy consumption of online data processing algorithms, specifically, online medical detection or assessment algorithms. Ignoring data unequivocally reduces energy consumption since data reduction allows electronic components, such as sensors, processors and radios, to be turned off or used less frequently. As a trade-off, however, we expect the output of the algorithm to degrade in quality. However, using medical monitoring algorithms as examples, we showed that while throwing away data can have an impact on performance, the impact can be small compared to the achieved energy savings.

We explored the impact of our data reduction methods in the context of two specific online medical monitoring applications: post-ACS risk stratification and seizure onset detection. The primary methods that we investigated were random duty cycling and sensor selection.

### 6.1 DATA REDUCTION THROUGH RANDOM DUTY CYCLING

For the problem of post-ACS risk stratification, we first transformed a batch algorithm that analyzed an entire 24 hour recording of ECG samples into an *online* algorithm that processed the data in five minute windows and then applied random duty cycling to the algorithm in order to reduce energy consumption.

The original algorithm for post-ACS risk stratification was designed to determine the risk of cardiovascular death for patients who experienced an acute coronary syndrome. We converted this batch algorithm to an online algorithm to provide continuous risk assessment. In both algorithms, the level of risk is derived from the MV-DF measure of morphologic variability.

Our online version of the algorithm changed the average MV-DF value by 4% with an average coefficient of variation of 0.14 for a test population of 753 patients. Thus, the online algorithm was a good approximation of the original algorithm. Moreover, in dividing the population into low and high risk, only 38 or 5% of the patients were reclassified relative to the original algorithm.

Since elements of the algorithm are computationally and energy intensive, we ap-

plied random duty cycling to reduce energy consumption. Given the structure of the online algorithm, a straightforward way to implement duty cycling was to randomly select a subset of the five minute windows for processing. Since the online algorithm derived its risk assessment by treating the five minute windows as samples of a distribution, we showed that random duty cycling could effectively approximate the full distribution if enough samples have been obtained. For example, while sampling 60% of the windows and thus reducing energy consumption by approximately 40%, only 6% or 45 out of the 753 patients were reclassified compared to the original algorithm.

## 6.2 DATA REDUCTION THROUGH SENSOR SELECTION

Many medical monitoring or detection algorithms collect and process data from multiple sensors. To reduce the energy consumed by these algorithms, we can select and process the data from a subset of sensors.

In Chapter 4, we applied the idea of sensor selection in the context of online seizure onset detection. To detect seizures, we started with a seizure onset detection algorithm that processed data from several EEG channels. Using the wrapper machine learning feature selection technique in a patient-specific manner, we were able to reduce the average number of channels needed to perform seizure detection for 16 patients from 18 to 6.4. This corresponds to an average energy savings of 60%. At the same time, detection performance was slightly degraded. Of the 143 total seizure onsets, the reduced channel detectors were able to find 136 seizure onsets compared to 139 detected by the full 18-channel detectors. The median detection latency increased slightly from 6.0 to 7.0 seconds, while the false alarm rate per hour increased from 0.07 to 0.11.

Compared to other feature selection methods, the wrapper approach, which allows users to define the objective function to compare channel subsets, did comparatively well in finding good channel subsets.

In Chapter 4, we explored the utility of the wrapper method for selecting and constructing reduced channel detectors using pre-recorded data. If we wish to select channels for deploying a reduced channel detector customized to a new patient, there are a number of approaches. In Section 4.11, we describe one possible selection strategy. To evaluate and compare channel selection strategies for deployment purposes, further investigation will be required.

## 6.3 DATA REDUCTION THROUGH DUTY CYCLING AND SENSOR SELECTION

In Chapter 5, we investigated the impact of a hybrid approach, *i.e.*, the impact of combining duty cycling and sensor selection. Two hybrid methods were explored. In the first method, we applied the concept of screening to the detector. In screening, we periodically control the operation of the detector using the output of less accurate, but lower energy “screening” detector as the control input. Lower energy detection is achieved on a per-window basis by analyzing fewer channels. The original detector, which uses more channels, is consulted only when the less accurate detector cannot classify an event.

For a given input signal, the total energy consumed by the combination of the two detectors will be less than the energy consumed by using just the original detector if the number of windows that the screener cannot label contains an event is small.

In this thesis, we applied the screening technique to 10 out of the 16 patients since the remaining patients had reduced channel detectors consisting of one channel. By applying screening to these patients, we reduced the energy consumed by the 18-channel detectors by 69%. On average, each screening detector used 2.8 out of the 8.0 channels used by the reduced channel detector. In terms of detection performance, 78 out of the 80 seizures detected by the subset detectors were detected when screening was added. The median latency increased slightly from 6.2 to 6.3 seconds and the false alarm per hour decreased slightly from 1.2 to 1.1.

As an alternative to using a screening detector to gate the operation of the reduced channel detector, we investigated duty cycling the reduced channel detector on a window by window basis. For those patients with seizure events with durations spanning tens to hundreds of seconds, duty cycling does not affect the seizure detection rate. On average, our random duty cycling scheme was able to reduce energy consumption by about 25% while degrading detection performance by only 2%. The latency did increase, but only by 1.1 seconds. Moreover, the false alarm rate per hour decreased by 0.3 events per hour.

#### 6.4 FUTURE WORK

In this thesis, we applied data reduction methods to reduce the energy consumption of two online medical monitoring algorithms. Though our focus is on medical applications, the methods we described should have general applicability beyond the medical domain. Future work should apply these methods to other domains that involve real-time data collection and processing from one or more sensors in energy-constrained environments.

One assertion of this thesis is that aspects of medical practice will move from clinical environments into the homes and offices of individuals. This thesis has focused on the problem of extending the battery lifetime of systems that are designed to perform real-time processing of medical data collected from sick patients. With computing resources becoming more pervasive, in the form of better cell phones and better connectivity, collecting data from healthy people to *prevent* disease is both compelling and realizable. However, carefully managing the power of these devices will be critical since healthy patients will have lower tolerance for poor battery lifetime. Since we will not know in advance what data is important and what is not, determining what data to collect and process will be a difficult challenge.





## Bibliography

- [1] Y. Agarwal, R. Chandra, A. Wolman, P. Bahl, K. Chin, and R. Gupta. Wireless Wakeups Revisited: Energy Management For VoIP over Wi-Fi Smartphones. In *MobiSys '07*, pages 179–191, 2007.
- [2] P. W. Armstrong, Y. Fu, W.-C. Chang, E. J. Topol, C. B. Granger, A. Betriu, F. V. de Werf, K. L. Lee, and R. M. Califf. Acute Coronary Syndromes in the GUSTO-IIb Trial: Prognostic Insights and Impact of Recurrent Ischemia. *Circulation*, 98:1860–1868, 1998.
- [3] K. Barr and K. Asanovic. Energy Aware Lossless Data Compression. In *Proceedings of the First International Conference on Mobile Systems, Applications, and Services (MobiSys 2003)*. USENIX, May 2003.
- [4] A. Bauer, J. W. Kantelhardt, P. Barthel, R. Schneider, T. Mäkikallio, K. Ulm, K. Hnatkova, A. Schömig, H. Huikuri, A. Bunde, M. Malik, and G. Schmidt. Deceleration capacity of heart rate as a predictor of mortality after myocardial infarction: cohort study. *Lancet*, 367:1674–1681, 2006.
- [5] A. Y. Benbasat and J. Paradiso. A Framework for the Automated Generation of Power-Efficient Classifiers for Embedded Sensor Nodes. In *Proceedings of the Fifth ACM Conference on Embedded Networked Sensor Systems*, pages 219–232, November 2007.
- [6] C. M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.
- [7] J. E. Calvin, L. W. Klein, B. J. VandenBerg, P. Meyer, J. V. Condon, R. J. Snell, L. M. Ramirez-Morgen, and J. E. Parrillo. Risk stratification in unstable angina. Prospective validation of the Braunwald classification. *Journal of the American Medical Association*, 273(2):136–141, January 1995.
- [8] E. J. Candés and M. B. Wakin. An Introduction To Compressive Sampling. *IEEE Signal Processing Magazine*, pages 21–30, March 2008.
- [9] Low-Cost Low-Power 2.4 GHz RF Transceiver (Rev. B), September 2007. Texas Instruments Datasheet.
- [10] C.-C. Chang and C.-J. Lin. *LIBSVM: a library for support vector machines*, 2001. Software available at <http://www.csie.ntu.edu.tw/~cjlin/libsvm>.

- [11] T. H. Cormen, C. E. Leiserson, R. R. Rivest, and C. Stein. *Introduction to Algorithms*, chapter 10. MIT Press, September 2001.
- [12] D. E. Culler, J. Hill, P. Buonadonna, R. Szewczyk, and A. Woo. A Network-Centric Approach to Embedded Software for Tiny Devices. In *EMSOFT 2001: First International Workshop on Embedded Software*, pages 114–130, October 2001.
- [13] P. de Chazal, M. O’Dwyer, and R. Reilly. Automatic Classification of Heartbeats Using ECG Morphology and Heartbeat Interval Features. *Biomedical Engineering, IEEE Transactions on*, 51(7):1196–1206, July 2004.
- [14] D. L. Donoho. De-Noising by Soft-Thresholding. *IEEE Transactions on Information Theory*, 41:613–627, May 1995.
- [15] P. Dutta and D. Culler. System Software Techniques for Low-Power Operation in Wireless Sensor Networks. In *Proceedings of the 2005 IEEE/ACM International Conference on Computer-Aided Design*, pages 925–932, 2005.
- [16] P. Dutta, M. Grimmer, A. Arora, S. Bibyk, and D. Culler. Design of a Wireless Sensor Network Platform for Detecting Rare, Random, and Ephemeral Events. In *IPSN '05: Proceedings of the 4th International Symposium on Information Processing in Sensor Networks*, pages 497–502, 2005.
- [17] J. S. Ebersole and T. A. Pedley. *Current Practice of Clinical Electroencephalography*. Lippincott Williams and Wilkins, 3rd edition, 2003.
- [18] E. Foundation. Epilepsy and seizure statistics. <http://www.epilepsyfoundation.org/about/statistics.cfm>, December 2009.
- [19] E. Glassman and J. V. Guttag. Reducing the Number of Channels for an Ambulatory Patient-Specific EEG-based Epileptic Seizure Detector by Applying Recursive Feature Elimination. In *Proc. of the 28th IEEE EMBS Annual International Conference*, pages 2175–2178, August 2006.
- [20] J. Gotman. Automatic recognition of epileptic seizures in the EEG. *Electroencephalography and Clinical Neurophysiology*, 54(5):530–540, November 1982.
- [21] I. Guyon and A. Elisseeff. An Introduction to Variable and Feature Selection. *Journal of Machine Learning Research*, 3:1157–1182, March 2003.
- [22] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik. Gene Selection for Cancer Classification using Support Vector Machines. *Machine Learning*, 46:389–422, 2002.
- [23] P. S. Hamilton and W. J. Tompkins. Quantitative investigation of QRS detection rules using the MIT/BIH arrhythmia database. *IEEE Trans Biomed Eng*, 33(12):1157–1165, December 1986.

- [24] C. Kitchin and L. Counts. *A Designer's Guide to Instrumentation Amplifiers*, chapter Chapter I—In-Amp Basics. Analog Devices, third edition, 2006.
- [25] R. Kohavi and G. H. John. Wrappers for Feature Subset Selection. *Artificial Intelligence*, 97(1-2):273–324, 1997.
- [26] P. Laguna, G. B. Moody, and R. G. Mark. Power Spectral Density of Unevenly Sampled Data by Least-Square Analysis: Performance and Application to Heart Rate Signals. *IEEE Transactions on Biomedical Engineering*, 45(6):698–715, June 1998.
- [27] T. Lal, M. Schroder, T. Hinterberger, J. Weston, M. Bogdan, N. Birbaumer, and B. Scholkopf. Support Vector Channel Selection in BCI. *IEEE Transactions on Biomedical Engineering*, 51(6):1003–1010, June 2004.
- [28] Q. Li, R. G. Mark, and G. D. Clifford. Robust heart rate estimate fusion using signal quality indices and a Kalman filter. *IOP Physiol. Meas.*, 29:15–32, 2008.
- [29] D. Lloyd-Jones, R. Adams, M. Carnethon, G. D. Simone, T. B. Ferguson, K. Flegal, E. Ford, K. Furie, A. Go, K. Greenlund, N. Haase, S. Hailpern, M. Ho, V. Howard, B. Kissela, S. Kittner, D. Lackland, L. Lisabeth, A. Marelli, M. McDermott, J. Meigs, D. Mozaffarian, G. Nichol, C. O'Donnell, V. Roger, W. Rosamond, R. Sacco, P. Sorlie, R. Stafford, J. Steinberger, T. Thom, S. Wasserthiel-Smoller, N. Wong, J. Wylie-Rosett, and Y. Hong. Heart disease and stroke statistics—2009 update: a report from the American Heart Association Statistics Committee and Stroke Statistics Subcommittee. *Circulation*, 119(3):e21–e181, January 2009.
- [30] M. Malik, J. Camm, J. T. Bigger, et al. Heart rate variability: standards of measurement, physiological interpretation, and clinical use. *Circulation*, 93(5):1043–1065, 1996.
- [31] G. Mathur, P. Desnoyers, D. Ganesan, and P. Shenoy. Capsule: An Energy-Optimized Object Storage System for Memory-Constrained Sensor Devices. In *Proceedings of the Fourth ACM Conference on Embedded Networked Sensor Systems*, November 2006.
- [32] D. A. Morrow, M. Benjamin M. Scirica, MD, E. Karwatowska-Prokopczuk, S. A. Murphy, A. Budaj, S. Varshavsky, A. A. Wolff, A. Skene, C. H. McCabe, and E. Braunwald. Effects of Ranolazine on Recurrent Cardiovascular Events in Patients With Non–ST-Elevation Acute Coronary Syndromes: the MERLIN-TIMI 36 Randomized Trial. *Journal of the American Medical Association*, 297(16):1775–1783, April 2007.
- [33] MSP430x15x, MSP430x16x, MSP430X161x, Mixed Signal Microcontroller. <http://focus.ti.com/lit/ds/symlink/msp430f1611.pdf>, August 2006. Datasheet.

- [34] C. S. Myers and L. R. Rabiner. A Comparative Study of Several Dynamic Time-Warping Algorithms for Connected Word Recognition. *The Bell Systems Technical Journal*, 60(7):1389–1409, 1981.
- [35] E. M. Ohman, C. B. Granger, R. A. Harrington, and K. L. Lee. Risk Stratification and Therapeutic Decision Making in Acute Coronary Syndromes. *Journal of the American Medical Association*, 284(7):876–878, August 2000.
- [36] W. H. Olson. *Medical Instrumentation: Application and Design*, chapter Basic Concepts of Medical Instrumentation. Wiley, third edition, 1997.
- [37] Y.-T. Peng and D. Sow. Data Scaling in Remote Health Monitoring Systems. In *IEEE International Symposium on Circuits and Systems*, 2008.
- [38] T. Pering, Y. Agarwal, R. Gupta, and R. Want. CoolSpots: Reducing the Power Consumption of Wireless Mobile Devices with Multiple Radio Interfaces. In *MobiSys '06*, pages 220–232, 2006.
- [39] J. Polastre, J. Hill, and D. Culler. "versatile low power media access for wireless sensor networks". In *Proceedings of the Second International Conference on Embedded Networked Sensor Systems*, pages 95–107, 2004.
- [40] H. Qu and J. Gotman. A Patient-Specific Algorithm for the Detection of Seizure Onset in Long-Term EEG Monitoring: Possible Use as a Warning Device. *IEEE Transactions on Biomedical Engineering*, 44(2), February 1997.
- [41] C. Sadler and M. Martonosi. Data Compression Algorithms for Energy-Constrained Devices in Delay-Tolerant Networks. In *Proceedings of the Fourth ACM Conference on Embedded Networked Sensor Systems*, pages 265–278, November 2006.
- [42] E. Shih, P. Bahl, and M. J. Sinclair. Wake on Wireless: An Event Driven Energy Saving Strategy for Battery Operated Devices. In *MobiCom '02*, pages 160–171, 2002.
- [43] A. Shoeb, B. Bourgeois, S. Treves, S. Schachter, and J. Guttag. Impact of Patient-Specificity on Seizure Onset Detection Performance. In *IEEE EMBS 2007*, pages 4110–4114, September 2007.
- [44] A. Shoeb, H. Edwards, J. Connolly, B. Bourgeois, S. T. Treves, and J. Guttag. Patient-Specific Seizure Onset Detection. *Epilepsy & Behavior*, 5(4):483–498, 2004.
- [45] A. Shoeb, T. Pang, J. Guttag, and S. Schachter. Non-invasive computerized system for automatically initiating vagus nerve stimulation following patient-specific detection of seizures or epileptiform discharges. *Int J Neural Sys*, 2009.
- [46] A. Shoeb, S. Schachter, D. Schomer, B. Bourgeois, S. Treves, and J. Guttag. Detecting Seizure Onset in the Ambulatory Setting: Demonstrating Feasibility. In *Proc. of the 27th IEEE EMBS Annual International Conference*, pages 3546–3550, September 2005.

- [47] J. M. Smith, E. A. Clancy, C. R. Valeri, J. N. Ruskin, and R. J. Cohen. Electrical alternans and cardiac electrical instability. *Circulation*, 77:110–121, 1988.
- [48] M. B. Srivastava, A. P. Chandrakasan, and R. W. Brodersen. Predictive System Shutdown and Other Architectural Techniques for Energy Efficient Programmable Computation. *IEEE Transactions on Very Large Scale Integration (VLSI) System*, 4(1):42–55, March 1996.
- [49] M. Stone. Cross-validatory choice and assessment of statistical predictions. *Journal of the Royal Statistical Society. Series B (Methodological)*, 36(2):111–147, 1974.
- [50] P. Sung. Risk stratification by analysis of electrocardiographic morphology following acute coronary syndromes. Master’s thesis, Massachusetts Institute of Technology, February 2009.
- [51] Z. Syed, B. Scirica, C. M. Stultz, and J. V. Guttag. Risk-stratification Following Acute Coronary Syndromes Using a Novel Electrocardiographic Technique to Measure Variability in Morphology. *Computers in Cardiology*, 2008.
- [52] Z. Syed, B. M. Scirica, S. Mohanavelu, P. Sung, C. P. Cannon, P. H. Stone, C. M. Stultz, and J. V. Guttag. Relation of Death Within 90 Days of Non-ST-Elevation Acute Coronary Syndromes to Variability in Electrocardiographic Morphology. *American Journal of Cardiology*, 103(3):307–311, February 2009.
- [53] Z. Syed, C. M. Stultz, B. M. Scirica, C. P. Cannon, K. Attia, I. O. Stebletsova, S. Mohanavelu, P. H. Stone, and J. V. Guttag. Morphological Variability: A New Electrocardiographic Technique for Risk Stratification After NSTEMI. *Circulation*, 116:II\_634, October 2007.
- [54] Z. Syed, P. Sung, B. M. Scirica, D. A. Morrow, C. M. Stultz, and J. V. Guttag. Spectral Energy of ECG Morphologic Differences to Predict Death. *Cardiovascular Engineering*, 2009.
- [55] T. van Dam and K. Langendoen. An Adaptive Energy-Efficient MAC Protocol for Wireless Sensor Networks. In *Processings of the First ACM Conference on Embedded Networked Sensor Systems*, pages 171–180, November 2003.
- [56] M. van den Broek and E. Beghi. Accidents in Patients with Epilepsy: Types, Circumstances, and Complications: A European Cohort Study. *Epilepsia*, 45(6):667–672, May 2004.
- [57] N. Verma, A. Shoeb, J. Guttag, and A. Chandrakasan. A Micro-power EEG Acquisition SoC with Integrated Seizure Detection Processor for Continuous Patient Monitoring. In *Proceedings of the 2009 Symposium on VLSI Circuits*, June 2009.
- [58] E. Waterhouse. New horizons in ambulatory electroencephalography. *IEEE Engineering in Medicine and Biology Magazine*, 22(3):74–80, May-June 2003.

- [59] S. B. Wilson, M. L. Scheuer, R. G. Emerson, and A. J. Gabor. Seizure detection: evaluation of the Reveal algorithm. *Clinical Neurophysiology*, 115(10):2280–2291, October 2004.
- [60] S. M. Zaacks, P. R. Liebson, J. E. Calvin, J. E. Parrillo, and L. W. Klein. Unstable angina and non-Q wave myocardial infarction: does the clinical diagnosis have therapeutic implications? *Journal of the American College of Cardiology*, 33(1):107–118, January 1999.
- [61] W. Zong, G. B. Moody, and D. Jiang. A robust open-source algorithm to detect onset and duration of QRS complexes. *Computers in Cardiology*, 30:737–740, 2003.