INFORMATION TO USERS

This manuscript has been reproduced from the microfilm master. UMI films the text directly from the original or copy submitted. Thus, some thesis and dissertation copies are in typewriter face, while others may be from any type of computer printer.

The quality of this reproduction is dependent upon the quality of the copy submitted. Broken or indistinct print, colored or poor quality illustrations and photographs, print bleedthrough, substandard margins, and improper alignment can adversely affect reproduction.

In the unlikely event that the author did not send UMI a complete manuscript and there are missing pages, these will be noted. Also, if unauthorized copyright material had to be removed, a note will indicate the deletion.

Oversize materials (e.g., maps, drawings, charts) are reproduced by sectioning the original, beginning at the upper left-hand corner and continuing from left to right in equal sections with small overlaps.

Photographs included in the original manuscript have been reproduced xerographically in this copy. Higher quality 6" x 9" black and white photographic prints are available for any photographs or illustrations appearing in this copy for an additional charge. Contact UMI directly to order.

ProQuest Information and Learning
300 North Zeeb Road, Ann Arbor, MI 48106-1346 USA
800-521-0600

Dynamic Models of Machining Vibrations, Designed for Classification of Tool Wear

Randall K. Fish

A dissertation submitted in partial fulfillment
of the requirements for the degree of

Doctor of Philosophy

University of Washington

2001

Program Authorized to Offer Degree: Electrical Engineering

UMI Number: 3013957

UMI Microform 3013957

Bell & Howell Information and Learning Company
300 North Zeeb Road
P.O. Box 1346
Ann Arbor, MI 48106-1346

In presenting this dissertation in partial fulfillment of the requirements for the Doctorial degree at the University of Washington, I agree that the Library shall make its copies freely available for inspection. I further agree that extensive copying of this thesis is allowable only for scholarly purposes, consistent with "fair use" as prescribed in the U.S. Copyright Law. Requests for copying or reproduction of this dissertation may be referred to University Microfilms, 1490 Eisenhower Place, P.O. Box 975, Ann Arbor, MI 48106, to whom the author has granted "the right to reproduce and sell (a) copies of the manuscript in microform and/or (b) printed copies of the manuscript made from microform."

Signature Randall K. Tut

Date 6/7/01

University of Washington

Graduate School

This is to certify that I have examined this copy of a doctoral dissertation by

Randall K. Fish

and have found that it is complete and satisfactory in all respects,
and that any and all revisions required by the final
examining committee have been made.

Chair of Supervisory Committee:

Mari Ostendorf

Reading Committee:

Prof. Les Atlas

Prof. Gary Bernard

Dr. Larry Heck

Prof. Mari Ostendorf

Date: 5 June 01

University of Washington

Abstract

Dynamic Models of Machining Vibrations, Designed for Classification of Tool Wear

by Randall K. Fish

Chair of Supervisory Committee

Professor Mari Ostendorf
Electrical Engineering

The goal of this dissertation is to develop a machining tool-wear classification system which uses features drawn from accelerometers that respond to machining vibrations. Specifically, we use features from wide band accelerometer signals in a two stage dynamic classifier estimating the flank wear on end mills cutting notches in either steel or titanium work-pieces. Since no standard data set and test paradigm exists for this task, we introduce an experimental paradigm which incorporates new evaluation metrics not previously used in tool-wear monitoring.

Until recently, only static classifiers have been used for tool-wear applications. However, the process of increasing wear is *dynamic*. Individual wear events which occur at a changing rate and last for a few milliseconds gradually change a tool's cutting edge from sharp to dull. Our experiments also show that within an individual cutting pass the wear process changes as the cutter moves into and out of "regions of interest" which effect the sensor features used in classification. We select features which are sensitive to the dynamics at these various time scales. We demonstrate a single-rate dynamic classifier which models the dynamics of wear both within an individual cutting pass and also over the cutting life of the tool.

Our single-rate dynamic classifier captures the slowly varying wear phenomena by using sequential states in a hidden Markov model. To improve the modeling of the rapidly varying discrete wear events that last several milliseconds, we extend the single-rate dynamic classifier to a multi-rate classifier. The multi-rate classifier splits the task of modeling events at the two time scales into two state-coupled classifiers processing feature streams at different data rates. We demonstrate that coupling the two classifiers during classification gives better performance than combining the outputs of the separate classifiers in a second stage.

The availability of data in this application is limited. Data annotated with the correct level of wear is even more scarce. We demonstrate a method of using both labeled and *unlabeled* data to train model parameters. The broad range of cutting conditions encountered in actual industrial practice imposes the need for the classifier to generalize to cutting conditions not included in the model training. We demonstrate feature processing which allows us to generalize to a limited range of cutting conditions including the use of features drawn from accelerometers with different response characteristics.

Our classification system is not intended to be the sole arbiter of the decision of whether or not a cutter should continue to be used or be replaced. We present the information from the classifier in several different formats to assist the machinist in making an informed decision. Our system estimates the wear on the primary cutting edge at the end of each cutting pass. In addition to this estimate, we provide a measure of the confidence in the cutter wear having exceeded a predefined level considered to constitute the end of the cutter's useful life. Prior to cutting with a new tool, the useful life for the cutter is expected to be the average for this type of cutter being used under the present cutting conditions. At the end of each cutting pass, our system updates this estimate of the remaining cutter life. We incorporate the actual cutting behavior seen for the particular cutter in use resulting in a more accurate prediction than is possible with a simple average.

The accuracy of our single-rate classifier is 90% to 97% when classifying the wear on cutters milling steel. Even on the more difficult problem of classification when cutting titanium, our multi-rate classifier achieves accuracy of 94%.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

GLOSSARY

BREAKAGE: Breakage refers to the catastrophic failure of the cutter shaft.

CEPSTRA: The cepstrum is the inverse Fourier transform of the log of the spectrum of a signal.

CHATTER: Episodes of excessive cutter vibration.

CHIPPING: This type of cutter wear refers to the phenomenon where the primary cutting edge fractures locally rather then exhibiting smooth wear.

CRATER WEAR: This refers to pitting on the flute face behind the primary cutting edge.

CUTTER: The term cutter is used in the milling community to refer to the cutting bit which actually comes in contact with the metal being machined.

CUTTING PASS: A single cutting event. This includes some "air cutting" prior to entering the workpiece, the metal cutting and some final "air cutting".

CUTTING SPEED: This is the surface speed at which the cutting edge moves through the workpiece. It is determined by the spindle RPM and the diameter of the cutter.

DEPTH-OF-CUT: Both radial and axial depths-of-cut describe the cross-section of the metal chips removed from the workpiece.

FEED RATE: This is the linear speed of the cutter axis as it moves through the workpiece.

FLANK WEAR: This refers to the dulling of the primary cutting edge of a cutter through an abrasive wear mechanism.

FLUTE: One of several helical cutting edges of a milling cutter.

FRACTURE: Fracture is the term used here to describe severe chipping when there has been a bulk breakage of the primary cutting edge.

GLM: A Generalized Linear Model (GLM) defines a non-linear mapping between a desired output variable and a vector of input predictors.

HMM: A Hidden Markov Model (HMM) is a stochastic model of a process that has piecewise stationary regions, where the time evolution of the non-stationary behavior can be characterized in terms of an unobserved discrete Markov chain.

NCE: Normalized Cross Entropy (NCE) is a measure of the amount of information added by the classifier as compared to the information contained in the priors from the training data.

ROC: Receiver Operating Characteristic (ROC) plots the probability of detection vs. the probability of a false alarm for different classifier operating points.

REMAINING LIFE: The number of cutting passes between the present pass and the first for which the cutter has exceeded the WORN threshold.

TOOL: The term tool as understood in the milling community refers to the cutting bit which actually comes in contact with the metal being machined.

WORKPIECE: The bulk material being machined by the cutter.

ACKNOWLEDGMENTS

My name appears on this dissertation but if it had been left to me alone, this work would never have been completed. There are many people who played major roles in helping me to reach this point and I would like to use this opportunity to thank them.

My advisor Mari Ostendorf; whose standards of excellence have inspired me to produce work of which we can both be proud. Choosing a graduate advisor involves a large measure of luck. I consider myself very lucky indeed to have been able to work with and learn from Mari.

Gary Bernard went far beyond his role as an industry advisor. I want to thank him for his technical insights, encouragement and most of all for his friendship.

The other members of my reading committee, Les Atlas and Larry Heck, who provided insights that helped me over hurdles and opened up new avenues of investigation.

David Castanon for his guidence during the early days of this work helping me to see things from a different perspective and thus understand them better.

I want to thank the Office of Naval Research for their support of this project under the Center for Auditory and Acoustics Research (ONR-2883401).

The Boeing Commercial Airplane Group; particularly Fred Heimann for freely sharing his machining expertise, and Gary Bernard for preparing the milling data sets.

Fellow graduate students Brad Gillespie and Somsak Sukittanon for sharing the outcome of their work to enhance my own.

David Mountain and the rest of the Biomedical lab at Boston University for their insights into the human auditory system.

The members of the SSLI lab who have been with me for most of this process - Michiel Bacchiani, Becky Bates, Ivan Bulyko, Ozgur Cetin, Hari Kumar, David Palmer and Izhak Shafran. Michiel, as promised I will not mention to anyone that it is you that taught me

how to program. It will be our little secret. Thanks to Ozgur for the use of his multi-rate code. Hari for his technical assistance and friendship. Becky for innumerable favors and a sense of humor which lightened the load. Zak for the welcome into his home whenever I needed a place to stay. Special thanks to David for providing the encouragement to keep going when it was difficult to see the end and for being a sounding board when I needed to clarify my ideas.

Closer to home, I want to thank my colleagues at Eastern Nazarene College. Their prayers and encouragement made a difficult situation bearable. Special thanks to John Free who, in addition to his own impressive work load, also took on many of my responsibilities so that I would be free to pursue my research.

Finally, I want to thank my family. My parents Bill and Dorothy whose unwavering faith in me is a constant source of inspiration. My daughters Theresa, Amanda and Carissa for there patience with a dad who was too often tired, for the pride I feel whenever I think of them and for the laughter they bring into my life when I really need it. Most of all, I want to thank my wife Sue. Without her patience, support and encouragement I would never have lasted through all of this. With her, it was even fun.

Chapter 1

INTRODUCTION

Metal milling cutters suffer from various types of progressive wear which, if not controlled, degrade the quality of work produced. If cutting continues with a WORN cutting tool it will break, causing further damage to the workpiece being milled. By listening to the cutting, feeling the vibrations of the milling machine and inspecting the chips produced during cutting, master machinists are able to predict the amount of wear on the cutting edges. However, it is not possible to dedicate a master machinist to the task of constantly monitoring tool-wear. In fact, the present industrial solution is to minimize the involvement of the human operator as much as possible.

A solution commonly seen on the factory floor is to replace cutters according to a fixed schedule based on average cutter life. A conservative schedule, which avoids damage to the workpiece by replacing the cutter after a fixed time, results in an inefficient use of the cutting tool and unnecessary down-time without completely eliminating costly failures. The wide variation in usable cutter life makes this approach ineffective and operators must retain the power to override this decision. For example, our test set contains cutters which lasted for 36 minutes and ones which required replacement after only 10 minutes.

Fully automated systems attempt to reduce the problem of wear to a binary "continue use" or "replace the tool" decision. The goal is sufficient accuracy to allow unattended operation. In practice, the challenges of predicting tool-wear have resulted in system accuracy too low for acceptance on the factory floor. Improvement to a system developed under the controlled set of conditions found in a research lab will only come from new information gained from work under actual factory conditions. These potential improvements which would be made possible by the knowledge gained from fielded systems are lost because the

low reliability of most academic and many of the commercially available monitoring systems causes them to be simply ignored [1].

We propose a system which is viewed as an aid to the human operator rather than a replacement. The system described here provides an alert when close supervision of the cutting process is required and provides information about when cutter replacement is warranted. Such a system must as accurately as possible model the wear process and then translate the information into a format useful to the human operator.

The first step in accurately modeling the wear process is recognizing that cutter wear is by nature a dynamic rather than a static process. At the longest time scale, wear is not the binary process assumed by earlier tool monitoring systems. Using a binary wear model assumes that cutting tools suddenly jump from being sharp to needing replacement. In reality they move from being new to progressively greater levels of wear. Knowledge of the level of wear in a previous cutting pass can reasonably be expected to improve the accuracy of classification of features from the present pass. At a finer time scale, the behavior of a cutting tool changes as it moves through the workpiece. In some materials a progression from the "entry" to "bulk" to "exit" segments of an individual cutting pass is seen. In others the cutting tool transitions between periods of quiet and noisy cutting several times within a single pass. Accurate interpretation of the sensor signals used for indirect wear monitoring is improved if this context of the recording is understood. At the finest time scale, the rate of long term edge wear is accelerated by momentary chipping or other transient types of events. We describe in this dissertation a dynamic classifier which we have developed to model these time progressive characteristics of the wear process.

Rather than limiting ourselves to working at only one or another of the time scales where the dynamic characteristics of cutting become evident, we also describe a modification to our dynamic classifier which processes information at two different data rates. In this multi-rate classifier we not only make use of data from different time scales but model the interrelationship between events occurring at the two data rates. Slower time scale information about the estimated level of wear and the fact that a cutter is in the middle of a noisy cutting period affects the interpretation of data collected during a single cutter revolution. In the same way, repeated transient types of events such as chipping seen at the

faster time scale increase the likelihood of classification of a higher level of edge wear.

Both the single-rate and multi-rate dynamic classifier allow us to replace the binary WORN vs. NOT WORN decision with a multi-level quantized estimate of the present level of wear on the cutting tool. This progression of wear labels allows the operator to differentiate between cutters which are still almost new from those moving toward the end of their useful life. Adding a post processing stage allows us to use the information from the dynamic classifier to generate two additional outputs for the operator. In the first we generate a confidence estimate that the tool has exceeded an acceptable level of wear on the primary cutting edge. In the second we provide an estimate of the tool's remaining life.

The confidence estimate augments the quantized wear output by providing an estimate of the likelihood that the cutter is actually WORN. While the wear labels from the dynamic classifier are discrete, the confidence estimate is a continuous measure of wear. Even when the wear label remains the same, the probability of wear should increase with increased cutting. With this increased resolution on the present state of the cutting tool, it is left up to the operator to decide what level of confidence is reasonable for the particular milling operation in progress. The ongoing estimate of the tool's remaining life provides another indication of when closer supervision by the machinist is required. The intent in providing each of these types of output information is to give the machinists information which they can combine with their knowledge of the particular milling operation. The merging of information from the classifier with the knowledge of the machinist allows more timely and accurate decisions about cutter replacement.

Moving away from full automation to the manufacturing aid described here makes it possible to realize a system which is both useful and acceptable in a production machine shop. However, for such a system to be successful, practical machining characteristics must be taken into consideration. We must be able to properly classify wear when cutting conditions such as tool size and cutting speed change. Within reason, we must be able to accommodate these changes even if the resulting cutting conditions were not seen in training. Classification performance must be able to tolerate features generated by different transducers on different milling stations. In this thesis we address these issues and present the results of our attempts to deal with some of the practical aspects of a tool-wear system.

One of the major practical considerations driving system design is the availability of training data. Expanding the number of model parameters necessary to properly implement a dynamic classifier such as ours calls for more labeled data than is practically available in this type of application. To deal with the problem of sparsely-labeled training data we propose and evaluate several alternatives for using unlabeled data in estimating model parameters.

In chapter 2 we begin our discussion of monitoring tool-wear. We start with an overview of past work and a review of terminology. We present details about the processes of tool wear in general and those details particular to an understanding of wear in a metal milling application. In this work we focus on tool wear in a milling application which is acknowledged to be more demanding than the drilling and turning applications included in our review of past work. In particular, we address wear while cutting either steel and titanium. Milling steel, while challenging, is reasonably well behaved compared to the particularly challenging problems presented when working with titanium. The details of the materials and cutting conditions seen during our evaluations are presented in chapter 3. In this chapter we also present the techniques used for feature extraction and the metrics used for evaluation.

Having defined the problem, the test paradigm and the methods of evaluation, we present our single-rate dynamic classifier in chapter 4. Applying this classifier to a real world milling problem raises the need to deal with some of the practical issues encountered in tool-wear monitoring. In particular, we investigate approaches to problems of generalization (to different sensors and cutting conditions) and training in the presence of sparsely labeled data, as discussed in chapter 5. The outputs of the single-rate dynamic classifier are further processed by the second stage of our tool-wear system as detailed in chapter 6, providing additional information to the operator.

The results of classification from chapter 4 make it clear that a better model is needed for successful classification of tools used in titanium cutting. In chapter 7 the single-rate classifier discussed in chapter 4 is expanded to process information at multiple data rates. We close in chapter 8 with a review of lessons learned and thoughts about future work for this application.

Chapter 2

BACKGROUND

We begin our discussion of past efforts in tool-wear monitoring with a review of important terminology; section 2.1. This review will provide a basic understanding of the types of cutter wear, parameters of the milling process important to our research and a review of the sensors which have been used in tool-wear monitoring. Once terminology has been established, we will review previous work on feature selection and signal processing in section 2.2. We will end with a review of the types of classifiers which have been applied to this problem in section 2.3 and review their performance.

The application discussed here is tool-wear in a milling environment. The terminology discussed in sections 2.1.1 and 2.1.2 is specific to this domain. However, lessons learned from turning and drilling are also relevant to a discussion of the relationship of sensors and tool conditions. The research discussed in section 2.1.3 as well as sections 2.2.1 and 2.3 is drawn from all three machining environments.

2.1 Terminology

2.1.1 Types of wear

Tool-wear is a general term applied to machining applications such as turning, drilling and milling applications. The term **tool** as understood in the milling community refers to the **cutter** which actually comes in contact with the metal being machined, the **workpiece**. The particular type of cutter of interest to us here is an end mill. End mills have a variable number of helical cutting edges or **flutes**. The wear described in this work refers to changes in the geometry of the primary cutting edge of these flutes. Nine separate classifications of cutter wear are described in [2]. Those most often referred to in the literature are flank wear, crater wear, chipping and fracture. **Flank wear** is the dulling of the primary cutting edge

through an abrasive wear mechanism. Depending upon the nature of the milling operation, variable levels of wear are acceptable before the change in edge shape necessitates tool replacement. An automated system must either incorporate this information in its decision process or provide information which contains detail about multiple levels of wear. **Crater wear** refers to pitting on the flute face behind the primary cutting edge. Excessive crater wear changes the geometry of the edge and can deteriorate chip formation and weaken the primary cutting edge. **Chipping** occurs when the primary cutting edge fractures locally rather then exhibiting smooth wear. **Fracture** is used interchangeably to describe severe chipping when there has been a bulk breakage of the primary cutting edge and catastrophic failure when the shaft of the cutter breaks completely. Here, we will use **breakage** to mean a catastrophic failure of the cutter shaft and "fracture" to refer to severe chipping. In our work we concentrate on classification of wear rather than attempting to predict breakage. During cutting, there may be episodes of excessive vibration of the cutter known as **chatter**. It is useful to consider chatter behavior when classifying cutter wear, for chatter usually increases the rate of wear, and violent chatter can cause breakage.

In spite of the fact that most wear monitoring systems treat these conditions as individual events, it is common for more than one to be present at the same time. Systems designed to identify only one of the conditions such as flank wear, typically do not make use of the effects of another such as chipping. Chipping and chatter would be considered noise which can degrade rather than be used to enhance classifier performance. Systems which are designed to identify multiple wear conditions still treat each wear condition independently. Sensors are selected for each wear condition and separate classifiers are used. To date, no work has been done which uses knowledge of the presence of one type of wear to influence the classification of another, though clearly they are related.

2.1.2 Machining parameters

A numerically controlled milling center in a typical industrial work cell uses multiple types of cutters for a single job and may use a single cutter in several different ways. Operations may use just the lateral, radial and/or corner cutting edges. Different operations call for

changes in the amount of material removed and the rate of removal. A roughing operation requires a high rate of material removal and relaxed surface quality requirements. During a finishing operation, the required removal rate is less but the requirements for surface quality are more stringent. Therefore a higher level of wear is acceptable for a cutter used during roughing than one used in a finishing operation; even if the cutter itself is the same.

Most of these changes have an impact on the data generated for classification. We will refer to these relevant parameters, as well as pertinent characteristics of the cutter itself, as **cutting parameters**. In order for a system to be of practical use in classifying wear, it must be able to deal with these changing parameters. Since it is practically impossible to train on all cutting conditions, a research goal is to develop a classifier which can be trained under one set of conditions and generalize broadly to others unseen in training.

The cutting conditions can be grouped into three categorizes; those determined by settings on the machining center, those dictated by the cutter selected for use, and those resulting from the material being machined. In [2] the important cutting variables determined by the machining center are listed as cutting speed, feed rate and depth-of-cut. **Cutting speed** is the surface speed at which the cutting edge moves through the workpiece. It is determined by the spindle RPM and the diameter of the cutter. **Feed rate** is the linear speed of the cutter axis as it moves through the workpiece. **Radial depth-of-cut** and **axial-depth-of-cut** describe the cross-section of the metal chips removed from the workpiece.

Important parameters of the cutter itself are the diameter, number of flutes, pitch (distance between a point on one edge to the same point on the next edge), corner radius, the helix angle of the cutting flutes and the geometry of the primary cutting edge.

Characteristics of the workpiece itself constitute the final group of cutting conditions. To date, published results on the monitoring of milling cutter wear have primarily been limited to different grades of steel. However, even under identical cutting parameters, sensor features change if a different material is machined. For example, unlike steel, when machining titanium the hot workpiece material will often diffusion-bond to the cutting edges. This process of titanium from the workpiece welding to the cutter forms a "built-up edge" (BUE) which changes the geometry and cutting characteristics of the cutter. As the

BUE increases, even a fresh cutter can exhibit behavior usually associated with a WORN tool. When the forces experienced by the cutting edge exceed the material bonding forces, the BUE breaks away. If the BUE breaks away cleanly, the cutter will return to behavior associated with it's level of wear prior to BUE. However, if particles of the cutting edge are also torn away as the welded titanium breaks, there will be a more sudden increase in the level of wear [2, 3]. One cycle of build-up and release of BUE welded titanium may be as short as a second or extend over periods of as much as 30 seconds. Between these more noisy cutting periods are periods of quiet cutting with a reduced rate of wear, free of the BUE build-up and release cycles. Both noisy and quiet cutting may occur in all stages of a cutter's life.

This behavior seen when milling titanium and not when working with steel illustrates the importance of considering the material of the workpiece as one of the critical "cutting parameters". These noisy/quiet periods are investigated in the multi-rate work discussed in chapter 7.

Even when cutting the same material, changes will be encountered when the cutter passes through hard spots in the workpiece which effect the vibration signal and may damage the cutter. In this work our evaluations include the cutting of both steel and titanium. Conventional techniques that are successful in monitoring wear when cutting steel are not successful when applied to the problem of cutting titanium. We point out lessons learned which are common to both materials as well as identifying those portions of the classifier which benefit from material specific modeling strategies.

2.1.3 Types of sensors

Our interest is in *indirect* measurement of tool-wear while the tool is cutting. Sensors are used to measure wear related phenomena from which we infer the level of tool-wear. In [4], Dan and Mathew provide a review of the seven types of phenomena used to monitor tool conditions. The top five are force, power, torque, vibration and acoustic emission. Force transducers mounted on the workpiece or on the spindle bearing measure the force exerted on the cutter as the workpiece tends to push it away. If cutting speed and feed rate were

held constant during milling, changes to the primary cutting edge would be expected to show up as changes to the force on the tool. In practice, variations to both cutting speed and feed rate are expected due to hard spots in the material and variations in the milling center unrelated to wear. However, by correcting for these types of changes to cutting speed and feed rate, force and torque sensors have been used successfully.

Accelerometers and acoustic emission (AE) sensors both respond to the mechanical vibration generated by the deformation of the metal being milled, the shearing during chip formation and the breaking of the chip away from the parent material.

Power sensors typically monitor either the total power sent to the spindle or track the motor current profile during cutting. Again, assuming a constant RPM of the cutter, changes in the cutting edge often exhibit themselves as changes in the amount of power needed to maintain constant speed.

Sound, temperature and the roughness of the machined surface have also been studied but are seldom used for monitoring wear. However, microphones are useful for detecting chatter.

We evaluate wear on end mills with relatively small diameters, (1/2" and 1"). Force, torque and power are typically not sensitive or fast enough to track the small changes due to wear on 1/2" cutters; leaving vibration and acoustic emission as useful sensory modalities. Accelerometers monitor both the vibration due to chip-formation and the transient shocks caused by edge breakdown. Since they have been used successfully to monitor cutter wear [5, 6, 7], the classifier described here uses wide band features extracted from a vibration sensor.

Several researchers have shown that it is beneficial to draw features from more than one type of sensor [8]. Here we use a single accelerometer which makes the task of generalizing to different cutting conditions more difficult. We limit ourselves to a single sensor due to the availability of data. Our focus is on modeling and not on feature extraction. Rather than expanding the number of sensors providing data, we develop models which use multi-dimensional feature vectors. These models provide a straightforward extension to multiple sensor features should additional sensors become available.

2.2 Feature Selection and Signal Processing

In section 2.1.3 we described the various sensors which have been found to be effective in monitoring wear. While the primary emphasis of this thesis is the structure and initialization of the classifier, feature selection is critical to the success of the classification system. The features must contain the necessary discriminative information for the classifier to have any chance of accurate classification. We also show that some of the challenges of a practical tool-wear system can be addressed in the feature processing rather than in the classifier.

2.2.1 Feature extraction

Flank wear is caused by the microscopic removal of material from the primary cutting edge. The rate at which these wearing events occur changes throughout the life of the cutter. The duration of individual wearing events is in the millisecond time scale. As these short time scale events accumulate, the level of wear gradually increases. The useful life of a cutter extends over many minutes, or longer depending on the material hardness. We will show that it is important to model both the slowly varying wear and the shorter time scale transients when estimating cutter wear.

Emel and Kannatey-Asibu [9] identify chipping and fracture as well as the wear on tools used in turning. To capture the shorter time scale events, each sample used for classification consists of data from one millisecond of cutting. These same samples are also used to classify the slower time scale flank wear. Systems intending to track only flank wear use features from a widely varying range of time scales. The features used in [10] are the current and force from an entire drilling pass. Carolan *et al.* [11] collect features several times for every cutter rotation but then calculate the RMS energy for an entire pass. Others, such as [12], consider short time samples of cutter behavior to be indicative of slowly changing performance and sample at the same time scales used for transient detection.

Rather than choosing either a fast or slow time scale, we report on classifier performance using fast time scale features, longer time scale features and the performance of a multi-rate classifier using features from both time scales.

2.2.2 Signal processing of sensor features

Just as researchers differ on sensor selection and evaluation paradigms, (section 3.1), they differ on the number of features used and the signal processing techniques applied. One simple system [13] uses a single feature with the only processing being the calculation of RMS energy. In [11] the variation of the RMS signal is the selected feature. In an attempt to make RMS variation more robust to changes in cutting conditions, this feature is normalized by dividing each sample by the average of the standard deviation of all passes. The work by Heck and McClellen [10] on drilling is an example of a system which, while still working in the time domain, used multi-dimensional feature vectors. They used a multi-dimensional parametric representation of the shape of the motor current curve during a pass to define five features to which they added the delta in the total power.

Other systems look outside the time domain for classification features. Emel and Kannatey-Asibu [9] calculate a standard FFT and then select as features the 20 frequency components that result in the greatest distance between the means of the fault conditions in their turning experiment. Niu *et al.* [14] use manually selected wavelet coefficients as features for a neural net turning classifier. Chou and Heck [15] use wavelets in a multi-scale binary tree to generate features for a static Gaussian mixture model classifier. The wavelet coefficients are the observations at the leaves of a multi-scale tree. The correlation between the coefficients is captured by the multi-scale tree parameters; these parameters are the features used for classification. Atlas *et al.* [16] investigate other time-frequency analysis techniques in a drilling application. They show qualitatively that minimum cross entropy (MCE) distributions give sharper definition in both time and frequency compared to standard spectrograms. Gillespie and Atlas [17] have also drawn features from the ambiguity plane for use in the same milling application discussed here.

Despite the variety of features investigated, there has been little work in comparing different representations in terms of classification performance with the exception of feature reduction. We report performance using several approaches to feature selection. Since our focus is on classifier design, our intent is not to identify the optimum feature set for this application. Rather, we choose reasonable features to demonstrate the important charac-

teristics of our classifier. However, the classifier design does impact the feature selection. For example, the single and multi-rate topology of our first stage classifier requires features at different time scales. Some of the practical implementation issues are best handled by signal processing in the generation of classifier features rather than in the classifier itself. As we discuss issues relevant to feature selection, such as generalizing across changing cutting conditions, we will present our approach and demonstrate the efficacy of various techniques to deal with these feature selection issues.

2.3 Classifier Architectures

Both static and dynamic classifiers have been used in tool fault detection. In some cases, the role of the classifier was simply to evaluate the feature selection. In others, different classifier architectures were proposed to help address either the problem of identifying multiple tool conditions or to improve the classification under different cutting conditions. Until recently, all classifiers for tool fault detection have been static classifiers, as reviewed in this section. In the last few years, some researchers have attempted to model the dynamic nature of the tool-wear process with dynamic rather than static classifiers. These efforts are discussed in chapter 4.

Static classifiers applied to tool-wear monitoring have posed classification as a binary problem of determining WORN vs. NOT WORN or broken vs. not broken. When a system is intended to classify multiple types of wear, multiple binary static classifiers are combined in multi-stage or parallel topologies.

Emel and Kannatey-Asibu use a two stage classifier [9]. The first stage separates feature vectors taken during a continuous section of a turning pass from ones taken from a transient (chipping/breakage) section. The continuous vectors are fed to a classifier in the second stage to determine WORN vs. NOT WORN. The transient vectors are processed by another second stage classifier to determine whether they came from a chipping or a breakage event. The first stage differentiating continuous from transient events was correct 95-100% of the time depending upon the cutting conditions. The second stage classifiers correctly identified whether a transient was a chip or a break 88-95%, while the WORN vs. NOT WORN

performance was correct 71-75% of the time.

Parallel neural networks were used in [14] to separately classify transient or continuous types of wear on turning tools. Prior to sending a signal to one of the two classifiers, a test for wide sense stationarity was used to separate a transient signal from a continuous one. If the original sensor signal was not found to be wide sense stationary, it was said to contain transients. In this case, the wavelet coefficients of the original waveform were separated into two groups, and an inverse transform performed to obtain two distinct time waveforms, one for transients and one for the continuous components. These separated signals were then classified by two parallel neural nets; one used to track wear and the other to identify chipping and breakage. Rather than define a fixed WORN/NOT WORN threshold to evaluate the classifier assigned to track wear, it was considered successful if it recognized later passes of a tool as WORN and earlier ones as NOT WORN. The actual wear threshold varied between 11 and 20 mils depending upon the cutting conditions.

Parallel neural nets are also used in another turning application [7] when processing AE and accelerometer data. However, in this system, there is no attempt to separate the continuous waveform from the transient. Each neural network is trained for a particular tool condition (tool chatter, breakage, chatter and severe wear together, normal) and all are presented with the same feature vectors for classification. The final output of the four parallel neural networks are then evaluated by a human observer to decide tool condition. Chatter and breakage were correctly classified 100% of the time while wear was correct 87%.

The primary approach to make a classifier more robust to changing cutting conditions has been to increase the number of sensors providing classification features. The benefit of multiple sensors was shown in [8] by comparing the performance of a simple two layer neural network using three different sets of eight dimensional feature vectors drawn from a turning experiment. Each set included the cutting speed and feed rate as two of the vector dimensions. In two of the feature sets, the remaining six vector dimensions were selected from both AE and force sensors. The third set used only the AE sensor for the remaining six dimensions. The feature sets from both sensors outperformed those for the single sensor 88% to 80%. Adding a hidden layer to the neural network improved performance to 94%. This is excellent performance for a turning wear classifier working with changes to feed rate,

cutting speed and depth of cut. However, it should be pointed out that neither training nor test data for this system included samples from times when the tool was in transition between WORN and NOT WORN. Samples labeled as NOT WORN were from wear levels of 4, 5 and 10 mils. WORN training and test samples were from wear levels of 20 and 30 mils. No samples were recorded for tools with wear between 10 and 20 mils.

Sick [18] reviews the work of several researchers using multi-stage neural networks for sensor fusion in tool-wear monitoring. In these systems, the feature vectors from different sensors are processed by separate neural networks to generate a preliminary classification of tool-wear. The final classification is performed by a second stage neural network which uses the preliminary classifications from the first stage as input.

The systems with published quantitative classification results are those applied to drilling and turning. For the milling application, with the exception of publications of our work [19] and work we have done in conjunction with other researchers at the University of Washington [3, 20], no quantitative evaluation of classification on independent test data sets with more than one or two tools has been published to date.

Chapter 3

TEST PARADIGM AND DATA

In this chapter we discuss the data sets used in our research and the metrics used for system evaluation. Two of our metrics are new to the problem of tool-wear monitoring. Past work in monitoring milling tool wear has been limited to a binary WORN vs. NOT WORN decision. We review the various test paradigms and evaluation metrics which have been used in the past in section 3.1. The cutting data used in our testing and the feature extraction techniques used are described in sections 3.2 and 3.3. The metrics used to evaluate system performance are described in section 3.4.

3.1 Previous Work

We will begin our discussion of the test paradigm used in this work with a review of the evaluation metrics which have been used in previous research addressing the problem of tool-wear.

There has been substantial work in tool-wear monitoring in turning, drilling and milling from which we would like to benefit and to which we would like to compare results. However, evaluating the performance of systems from previous work is difficult because of the absence of a standard test data set and the lack of any standard test paradigm and evaluation metric. The types of questions asked in the research cited here were "does this sensor track flank wear?", "what signal processing extracts good information from an accelerometer signal?", and "is a neural net better than a static Gaussian classifier?". The difficulty in comparing even those systems which ask the same question is that there is no consistent evaluation metric.

When evaluating AE as a sensor for flank wear, Hutton and Hu [13] plotted the AE RMS voltage against average flank wear during changes in cutting speed, depth-of-cut and feed rate. They showed that as flank wear increased there was a monotonic increase in AE

RMS voltage, a trend that they took as sufficient to identify the feature as a good one. Drilling force was evaluated in [21] by plotting the maximum force against the number of holes drilled. Since force increased with the increased use of the drill, it was considered a good parameter for monitoring wear. In a system using force to detect tool breakage, Lan and Naeheim [12] evaluated performance by seeing if there was a sudden change in the plot of the autoregressive force parameter which corresponded to a time when it was believed that a chip or breakage occurred. In all these studies, the evaluation tends to be qualitative or descriptive and not indicative of actual classifier performance.

In those cases where classifier performance is reported, accuracy is the criteria used but the objective of the evaluation changes. In a turning application, Emel and Kannatey-Asibu [9] used a static Gaussian classifier with the objective of evaluating how good AE features were at detecting tool-wear at two different cutting speeds and feed rates. Force and AE sensors were evaluated in another turning application [8] with a neural net to classify a tool as WORN or NOT WORN. In a drilling application, Heck and McClellan [10] evaluated power and force features with an HMM. The HMM classified the data from every hole drilled as having come from a tool which was WORN or NOT WORN. Performance was based on the percentage of times the classifier correctly labeled a pass under conditions which included changes to cutting speed and feed rate.

Either qualitative or quantitative evaluation is sufficient to choose between competing sensors, signal processing techniques or classifiers evaluated under the same test conditions. However a problem arises when trying to make comparisons when the experimental conditions chosen by different researchers are not the same. Even when accuracy is the reported performance metric, the definition of what is "correct" varies. In many cases, the level of wear corresponding to WORN is unspecified. Rather than setting a threshold for WORN prior to testing wear during turning, tools classified as WORN in [14] were measured after classification to have a wear level between 11 and 20 thousandths of an inch. Since this range was considered reasonable, these were considered to have been classified correctly. This methodology is problematic not only because it defines the wear threshold after scoring, but also because there is no assessment of missed detections.

Evaluation is typically performed on test data which is separate from that used in

training [7, 8, 9, 10]. However, some reports also include results from evaluations using the same data in both training and test [6, 10]. Since this results in performance which is optimistic, these should not be used when comparing competing systems. In most cases, the training data is drawn from all of the cutting conditions being evaluated [7, 8, 9, 10, 14], but in one case, results were reported for a system which trained on one set of cutting conditions and tested on another [6].

In the absence of a standard data set and evaluation metric, we choose to work with data representative of the milling application, apply the standard accuracy metric where our work is similar to that done previously, and propose new metrics where appropriate.

3.2 Data Sets

The accelerometer data used in our work has been provided by Boeing Commercial Airplane Group of Seattle, Washington. Data was recorded during the cutting of both steel and titanium, using cutters with 1/2" and 1" diameter. In addition to the different material used, important differences between the cutting of steel and titanium include the cutter edge geometry, spindle speeds, depths-of-cut and end-point of acceptable edge wear. The details of these data sets are described below.

3.2.1 Steel data

To gather the data necessary to evaluate classifier performance when cutting steel, a MAZAK H800 numerically controlled machining center was used to climb-cut notches in a 24-inch block of 4340 steel of Rockwell-C hardness 32. Both the 1/2" and 1" cutters were 4-flute, 30 degree helix, uncoated finishing end-mills made of M42 high-speed steel, flooded with synthetic, water-soluble coolant. The cutting parameters are listed in Table 3.1.

The vibration sensor used was a Bruel & Kjaer model 4505 accelerometer mounted radially on the front plate of the spindle housing, fed into a Bruel & Kjaer NEXUS-2692 conditioning amplifier set at 100mV/g. A SONY model PCHB244 DAT recorder set for wide-band analogue bandwidth of 25 kHz was used to record the accelerometer output.

Two channels of the DAT recorder were sampled as signed 16-bit integers at 48 kHz with

Table 3.1: *Cutting parameters for the steel and titanium data sets. Letter prefixes are added to the diameters listed for titanium cutting to distinguish between cutting with end-mills made of M42 high-speed steel (M-1/2", M-1") and end-mills made of Rex20 steel (R-1/2").*

Cutting Parameter	Steel		Titanium		
Cutter Diameter	1/2"	1"	M-1/2"	M-1"	R-1/2"
RPM	611	344	733	367	420
Feed Rate (in/min)	12.2	6.9	14.7	13.2	2.5
Axial Depth-Of-Cut	0.5"	1.0"	1.0"	0.5"	1.0"
Radial Depth-Of-Cut	0.1"	0.2"	0.2"	0.4"	0.1"
Wear Threshold	0.009"	0.010"	0.005"	0.010"	0.010"

ARIEL-ProPort/AT&T-DSP32c hardware and ENTROPIC ERS2000 software running on a SUN SPARC-4/330. The accelerometer data was recorded for about 5 sec prior to the cutter entering the workpiece, during cutting and for about 5 sec after exit. At the end of a limited number of selected cutting passes, each cutter was removed and its wear level microscopically measured and recorded by a master machinist before it was replaced and cutting continued. The labels assigned based on these measurements are the *known* labels discussed in this dissertation. The mid-point in the range of wear noted by the master machinist is used to assign the steel cutting passes to one of five wear levels designated "A - E". The range of wear in each wear level along with its mapping to a binary designation of either being WORN or NOT WORN is shown in table 3.2.

3.2.2 Titanium data

The titanium cutters are divided into a series of three data sets. Within these three series, data is recorded with different accelerometers and using different grades of titanium.

The M-1/2" data was recorded using the same MAZAK H800 machining center used when cutting steel. These titanium cutters were used to climb-cut notches in an 18-inch block of 6AL4V Titanium of Rockwell-C hardness 32-34. Data for M-1" titanium was

Table 3.2: *Steel wear levels: Wear labels W_i are defined by the midpoint of wear on the primary cutting edge measured in thousandths of an inch. The binary* NOT WORN *or* WORN *designations are determined by the specified threshold of wear. The numerical predictor used by the P(worn) GLM is $\hat{\omega}$.*

Steel Wear Levels			
W_i	Wear Midpoint	$\hat{\omega}$	Binary Label
A	Early passes	4.0	NOT WORN
B	< 6.7	6.0	NOT WORN
C	6.7 - 9.0	7.5	NOT WORN
D	9.0 - 10.25	9.5	WORN
E	> 10.3	11.0	WORN

recorded on the same MAZAK H800 during climb-cutting of a 20-inch block of 10-2-3 Titanium of Rockwell-C hardness 34-36. Both the M-1/2" cutters and M-1" cutters were 4-flute, 35 degree helix, uncoated finishing end-mills made of M42 high-speed steel, flooded with synthetic, water-soluble coolant. The cutting parameters are listed in Table 3.1.

The vibration sensor, conditioning amplifier and DAT recorder used to record the M-1/2" and M-1" titanium data were the same as used to record the steel cutting data described in section 3.2.1. The accelerometer data was recorded for about 5 sec prior to the cutter entering the workpiece, during cutting and for about 5 sec after exit. Again, microscopic measurements were made by a master machinist after a limited number of cutting passes. The mappings from the measured range of wear to a quantized wear label W_i and binary labels are shown in table 3.3.

The R-1/2" cutting data is the third of the titanium data sets. In addition to the different cutting conditions listed in table 3.1, there are significant differences between this and the M-1/2" and M-1" already described. The cutters themselves are made of Rex20 steel rather than the M42 used to cut steel and used for the M-1/2" and M-1" titanium. The titanium block being machined with the R-1/2" cutters was much harder having a Rockwell-C hardness of 38-40 compared to the 32-36 in the M-data. The larger block size,

Table 3.3: *Titanium wear levels: Wear labels W_i are defined by the midpoint of wear on the primary cutting edge measured in thousandths of an inch. The binary* NOT WORN *or* WORN *designations are determined by the specified threshold of wear. The numerical predictor used by the P(worn) GLM is $\hat{\omega}$.*

Titanium Wear Levels			
W_i	Wear Midpoint	$\hat{\omega}$	Binary Label
A	Early passes	1.0	NOT WORN
B	2.5 - 4.9	3.5	NOT WORN
C	5.0 - 7.4	6.0	NOT WORN
D	7.5 - 9.9	8.5	NOT WORN
E	10 - 12	11.0	WORN
F	> 12	14.0	WORN

24" compared to 18", and greatly reduced feed rate results in a much longer duration for a single cutting pass. The typical M-data cutting pass lasted about 74 seconds compared to approximately 580 seconds for the R-data. One impact of this increased pass duration is the increased likelihood that the edge wear will move through multiple quantized levels in a single pass. The Bruel & Kjaer model 8325 accelerometer used for the R-data has lower transverse and longitudinal resonance than the model 4505 used in all other data sets. The vibration sensor was fed into a Bruel & Kjaer 2525 conditioning amplifier set at 100mV/g and recoded with the same DAT recorder used in previous data sets.

Steel, M-1" and R-1/2" data was recorded beginning with a fresh cutter and stopped soon after a reasonable wear threshold was reached. The recording of M-1/2" data at times began after several cutting passes and stopped in the middle of a cutter's life, not running to a typical WORN level. As indicated in table 3.3, the threshold of wear for the titanium data is set to approximately 0.01". The M-1/2" cutters were stopped when wear reached approximately 0.005". Some passes in the R-data were recorded when the edge wear had extended up to twice this level. The extra wear label "F" is included to capture these passes which have excessive wear. Including them in the models for a typically WORN cutting pass

would bias the WORN model to require excessive wear before detection. The additional label also indicates that the cutter has been used beyond the point where it can be reground and used again.

3.3 Feature Extraction

Prior to use by the classifier, the raw accelerometer data described in section 3.2 is converted into a series of feature vectors intended to capture the information necessary for classification. We use features calculated at two different time scales. *Fine* rate features correspond to the rate at which the cutting flutes strike the workpiece (four times per revolution). *Coarse* rate features are determined at a period equal to forty times the period of fine rate features. We use two approaches to feature extraction. Energy features (section 3.3.1) are used on the steel test set. Cepstral features (section 3.3.2) determined at both fine and coarse rates are used in classification of both steel and titanium. Auto-ambiguity and auditory features (section 3.3.3), provided by colleagues at the University of Washington and at Boston University respectively, are also used in selected tests comparing the performance of the various feature sets.

3.3.1 Energy features

To determine the *energy* features, an FFT is computed for each time slice of accelerometer data corresponding to the time that a cutting flute is in contact with the workpiece.[1] The sum of the log of the energy in each frequency bin, log energy in the 8kHz bin and the associated derivatives of these two features are selected as features for classification, giving a total of four features. The 8kHz energy feature was suggested by initial data analysis (inspection of spectrograms) and has been found to be useful in previous work [22]. The sum of the log of the energy in each frequency bin can be thought of as a flattening of the spectral information since it acts to reduce the impact of individual frequency bins. We use this rather than the more typical log of the total energy because it gave better performance

[1] Actually, an average of FFTs from three overlapping windows within this time frame is used. We use a 512 point FFT for the steel 1/2" and 1024 for the steel 1". Therefore the 8kHz feature bin is 90Hz and 45Hz wide respectively.

in a pilot experiment. The FFT energy features with the spectrally flattened energy had an accuracy of 96% on the steel cross validation training set. Replacing it with the more typical energy features reduced accuracy to 87%. The use of feature derivatives is borrowed from the speech recognition literature [23]. The derivatives are computed using the standard linear regression formula over a five frame window.

$$\delta_t = \frac{\sum_{\theta=1}^{2} \theta(c_{t+\theta} - c_{t-\theta})}{2\sum_{\theta=1}^{2} \theta^2} \tag{3.1}$$

where c_t is the static feature value.

The total energy and 8kHz energy dimensions of the feature vectors are normalized as described in chapter 5 to make them more robust across changing cutting conditions prior to the calculation of the feature derivatives.

Coarse-rate energy features are calculated by taking the average of M consecutive fine-rate features. The derivative dimensions at the coarse rate are determined by applying the regression formula after the average of the total and 8kHz energy has been calculated. In all of our work, the ratio of fine to coarse-rate features is $M = 40$.

3.3.2 Cepstral features

The single-rate classifier described in chapter 4 makes successful use of the time-frequency information in our energy features. Cepstral features retain the time-frequency information and add the ability to generalize across changing accelerometers as will be shown in chapter 5. Cepstral coefficients are also generally decorrelated and thus are well matched to the use of diagonal covariances in our wear models.

In this work we make use of the HCopy function in the HTK tool box [23] to generate cepstral features from the sampled accelerometer data described in section 3.2. The calculation begins with the linear prediction of the system transfer function using an all pole filter:

$$H(z) = \frac{1}{\sum_{i=0}^{p} a_i z^{-i}} \tag{3.2}$$

where p is the number of poles and a_0 is defined to be one. The HTK implementation suggests a filter with more poles than desired cepstral coefficients. In our implementation we

begin with twenty cepstral coefficients using a 25th order filter. The filter coefficients a_i are chosen to minimize the mean square filter prediction error summed over the analysis window. When calculating fine-rate features we choose an analysis window which corresponds to 110% of a flute period resulting in an overlap of 10% in the data used to determine adjacent cepstral features. The cepstrum is the inverse Fourier transform of the log of the spectrum of a signal. Rather than implement the transform, log, inverse transform to obtain the cepstral coefficients, the HTK implementation computes the cepstral coefficients, c_n, from the filter coefficients, a_n, using the simple recursion:

$$c_n = -a_n + \frac{1}{n}\sum_{i=1}^{n-1}(n-1)a_i c_{n-i} \tag{3.3}$$

When calculating features at a coarse rate we simply average 40 consecutive fine-rate feature vectors.

To mitigate the effects of changing transducers, the cepstral mean subtraction discussed in chapter 5 requires a good estimate of the contribution of the transducer to the cepstral features. Our goal is to remove only the accelerometer component, i.e. without losing any long term changes due to increased wear. When the early passes of a particular cutter have been recorded, we use the first three passes to determine the cepstral mean for that cutter. For the R-data, only the first pass is used because of the long duration of these cutting passes. In the cases where the early passes are not available, we use a global cepstral mean for that accelerometer and set of cutting conditions. This global cepstral mean is determined from the first three passes of all cutters in the training data for the corresponding data series. In a fielded system, changing to a different accelerometer would require the use of a new cutter for the equivalent of three cutting passes to determine an accelerometer-specific cepstral mean vector for processing data from the new accelerometer.

After cepstral mean subtraction, the dimension of the feature vector is reduced from twenty to four. All data from passes in the training set assigned a binary label of WORN as described in section 3.4 are pooled and a global mean and variance are calculated. The same is done for those passes with a binary label of NOT WORN. The four dimensions with the greatest distance between these two data groups are chosen as wear features. We investigate

two distance metrics: the ratio of the between class scatter to the within class scatter

$$scatter\ ratio_i = \frac{(\mu_i^w - \mu_i^{nw})^2}{\frac{1}{2}\left(\sigma_{i,nw}^2 + \sigma_{i,w}^2\right)}, \tag{3.4}$$

and a modification of the squared Mahalanobis distance between corresponding dimensions of the two global models

$$r_i = \frac{1}{2}\left(\frac{|\mu_i^w - \mu_i^{nw}|}{\sigma_{i,nw}^2} + \frac{|\mu_i^{nw} - \mu_i^w|}{\sigma_{i,w}^2}\right), \tag{3.5}$$

where μ_i^w and $\sigma_{i,w}^2$ are the mean and variance of the i^{th} dimension of the WORN features and μ_i^{nw} and $\sigma_{i,nw}^2$ are the mean and variance of the i^{th} dimension of the NOT WORN features. Both metrics choose the same top three cepstral dimensions for both steel and titanium; the difference is in the dimension chosen for the fourth feature. When classifying the M-1/2" titanium cutters, the scatter ratio features had better accuracy on the CV test set: 98% vs. 93%. The performance on the test cutters went down: 71% vs. 86%. Therefore, we use the features selected by the average Mahalanobis metric in all of our classification with cepstral features. It is likely that some of the discriminative characteristics of the dimensions chosen in this way will be redundant. However, performance is good and comparable to the energy features when testing on the steel data set.

When this selection criteria is applied to the steel data set, the first four cepstral features $c_1, \ldots, c_4$ are chosen. Using the same technique on the M-1/2" titanium data selects c_1, c_4, c_5 and c_{20}. Finally the dimension is increased from four to eight with the addition of the derivative features (equation 3.1). The addition of the derivative features in testing on the steel data set had no impact on the performance of the cross-validation training set where accuracy was 94.5%. However, generalization to the 1/2" and 1" test data improved from 81% and 86% to 90% and 97%. The selected cepstral dimensions with derivative features are used for all cepstral based tests.

Our work with the FFT energy features points out the value of using spectrally flattened total energy as one dimension of our feature vectors. The energy feature available in the HTK toolkit is the typical rather than spectrally flattened total energy. To investigate its benefits when using cepstral features, we include it in a set of features for classification. Rather than choosing the first four cepstral features, we choose the first three and energy.

Substituting total energy for one of the selected cepstral features decreases performance and is not used in our evaluations.

3.3.3 Auditory and auto-ambiguity features

The research reported in this dissertation is a part of a Multi-University Research Initiative supported by the Office of Naval Research. We therefore had access to features derived from our same data sets provided by other members of the MURI. Colleagues at Boston University provided us with auditory features and colleagues at the University of Washington with auto-ambiguity features.

To generate the auditory features, a filter bank with center frequency spacing and bandwidth chosen to simulate the human auditory system was used to process the cutting data. Human perceptual experiments carried out at Boston University pointed to placing particular interest on the activity in the 4kHz and 8kHz bands. Two measures were used; the number of transients (Count) seen in each frequency band during the ten revolution window and the mean interval between transients. The transients selected occur at most once per flute, more typically less than once per revolution. Manual inspection by those performing the auditory research showed a correlation between the rate of transients and cutter wear. The 4kHz/8kHz, Count/Interval feature is added to the log total energy giving us four two dimension feature sets. We then add derivative features giving us an additional four feature sets for evaluation.

The auto-ambiguity features provided by Gillespie and Atlas [17, 24] at the University of Washington were selected using an unsupervised VQ algorithm. Auto-ambiguity features were clustered into a set of codewords and codewords found to be correlated with tool-wear were selected for use in classification. The five selected auto-ambiguity features were added to the log of the total energy resulting in a six dimensional feature vector for classification.

3.4 Test Paradigm

3.4.1 Training, Cross Validation Testing and Test

Our work includes testing on data from the cutting of both steel and titanium. However, the two different materials are never combined in a single data set for evaluation. Within these two workpiece material data sets, the available cutters are partitioned by cutter into two subsets; those used during system development (cross validation - CV) and a distinct subset held out for system evaluation (Test). The cross validation sets are used to train model parameters and evaluate different topologies and feature selection techniques, as described below. Once development is complete, models trained with all of the cutters allocated to the cross validation sets are used to classify the held out test cutters.

It is important that tuning model parameters not be done on the test data. In many statistical classification applications, system design involves two test sets, a development test set for tuning parameters and an evaluation test set for assessing performance. Because of the limited number of cutters available and the importance of including all passes from a particular cutter together in a training or test set, there was not sufficient data to have a separate development test set. Instead, model parameter tuning is done via cross validation testing on the training data. During the development phase, the N cross validation cutters are partitioned into CV sets with M cutters used for training and $N - M$ held out for test. The particular cutters assigned for train or test in each CV set are rotated until all cutters in the CV data are classified without training and testing on the same data.

Our tests with steel cutting data include fourteen 1/2" cutters and four 1" cutters (table 3.4). Six of the 1/2" cutters are allocated for cross validation testing and then training. When used for cross validation, the cutters are assigned to train/test sets in three CV partitions; each of which uses four cutters for training and tests the two held out cutters. Only 1/2" cutters are used during system development. Models trained with these six CV/Train cutters are used to test eight different 1/2" cutters and the four 1" cutters. This allows us to evaluate the ability of the models to generalize to different cutting conditions.

The titanium data is used in a series of three different tests which we refer to as Series-

Table 3.4: *The number of cutters in the CV/Train and Test sets (#); where CV is the cross validation test sets used during system development. The number of passes recorded (Rec) and hand-labeled (Lab), and the number of passes used during accuracy and confidence evaluation (*WORN *or* NOT WORN*) for the steel data set.*

Steel Cross Validation and Test Data Sets					
Train/Test Set	#	Rec	Lab	WORN	NOT WORN
Steel 1/2" CV/Train	6	87	12	6	49
Steel 1/2" Test	8	80	11	7	24
Steel 1" Test	4	53	10	4	32

Table 3.5: *The number of cutters in the CV/Train and Test sets (#); where CV is the cross validation test sets used during system development. The number of passes recorded (Rec) and hand-labeled (Lab), and the number of passes used during accuracy and confidence evaluation (*WORN *or* NOT WORN*) for the series of titanium tests.*

Titanium Cross Validation and Test Data Sets						
Train/Test Set	Data Set	#	Rec	Lab	WORN	NOT WORN
Series-A CV/Train	M-1/2"	6	70	12	9	52
Series-A Test	M-1/2"	7	75	13	11	53
Series-B CV	M-1"	7	54	15	5	40
Series-C Train	M-1/2",M-1",R-1/2"	20	148	41	10	123
Series-C Test	M-1/2",M-1",R-1/2"	14	108	27	7	87

A, Series-B and Series-C (table 3.5). Our single-rate classifier approach to training with sparsely labeled data and $P(worn)$ processing are developed using the steel data set. The Series-A tests are used to investigate the changes required when applying these to the cutting of titanium. Series-A tests use the thirteen M-1/2" cutters. Six of the cutters are allocated for cross validation testing and then training, using the same train on four, test on two CV rotation scheme used for steel. We use a different WORN or stopping threshold in Series-A tests. The M-1/2" cutters were not used until they reached the typical WORN threshold specified for titanium. Data was collected from the time a cutter was fresh until it reached a level which can be described as "mid-life" (wear level "C" in table 3.3). This is considered a point of interest because it is at this level of wear that the rate of wear is expected to accelerate and closer supervision by the operator is warranted. All of the same techniques developed to classify a cutter as WORN are used during the Series-A tests. The only change is that the threshold of interest is lowered from that usually associated with a WORN cutter to a threshold indicating mid-life.

The noisy/quiet cutting seen in titanium is expected to cause problems for our single-rate classifier. The Series-B tests are used to evaluate noisy/quiet cutting with both a single-rate and a multi-rate classifier. The Series-B tests use the seven M-1" cutters. Due to the limited numbers, all seven are assigned to cross validation testing (table 3.5). Here we use a six way cross validation with six different combinations of train on five and test on one to evaluate performance. Two of the cutters have a limited number of cutting passes and are treated as a single cutter for cross validation rotation purposes. The Series-B tests use only the M-1" cutters because the noisy/quiet cutting periods which are important in our development of the multi-rate classifier described in chapter 7 are most apparent when using these larger diameter cutters on titanium.

The Series-C tests use all of the M-1/2", M-1" and R-1/2" titanium data. The intent of these tests is to evaluate the best model performance under the changing cutting conditions and different accelerometers represented by these three data sets. As such, there is no cross validation testing because there is no further parameter tuning of the classifier designed during the Series-B tests. The twenty training cutters in the Series-C tests consist of the six M-1/2" training cutters used in Series-A, five of the M-1" cutters used in Series-B and

nine R-1/2" cutters. The fourteen test cutters consist of the seven M-1/2" test cutters, two M-1" and five R-1/2" cutters (table 3.5).

3.4.2 Assigning binary labels to test passes

Depending upon the output desired, we require either a quantized wear level or a binary WORN vs. NOT WORN label for each cutting pass to be evaluated. In chapter 5 we describe our approach for labeling the unlabeled *training* passes to expand the number of passes in the training sets with a quantized wear label. The technique described here is used to select and label unlabeled *test* passes and expand the number of cutting passes with a binary WORN vs. NOT WORN label.

For both steel and titanium data, between 15% and 38% of the cutting passes include known wear labels. Since this is not enough data for proper evaluation, we use the following criteria to assign either a binary WORN or NOT WORN label to cutting passes not inspected by the Boeing machinist. If a known label of NOT WORN is recorded at pass n, all passes prior to n are also assumed to be NOT WORN, and we assume that the first pass is NOT WORN. The passes between a known label of NOT WORN and a known WORN pass are not evaluated during testing since the pass where the transition took place is unknown. If two passes from the same tool have a known label of WORN, all passes between these two are also assumed to be WORN. Table 3.6 shows a representative example selected from the steel data set. One of the steel 1/2" cutters is assigned a known label of NOT WORN at pass 10, WORN at pass 14 and all other passes are unlabeled. Passes 1-2 are not evaluated, 3-10 are assumed to be NOT WORN, passes 11-13 are not evaluated and pass 14 is WORN. The 14 recorded passes thus result in 9 passes for evaluation. Using this approach expands the evaluation coverage of the steel cutters from 15% to 55%, M-1/2" titanium from 17% to 86%, M-1" titanium from 28% to 83% and R-1/2" from 38% to 79%.

System evaluation begins at pass three to guarantee some history for all evaluated passes and because the first passes can be trivially labeled. There are two exceptions to this. First, the longer cutting passes seen in the R-1/2" cutters led us to skip only the first pass in our evaluation. Second, several of the M-1/2" cutters do not include data for the early

Table 3.6: *Mapping from "known" to inferred test labels:* WORN *(1),* NOT WORN *(0), unlabeled (-), not used in evaluation (x).*

Cutting Pass Number	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Known Binary Labels	0	-	-	-	-	-	-	-	-	0	-	-	-	1
Inferred Test Labels	x	x	0	0	0	0	0	0	0	0	x	x	x	1

cutting passes. Since the first recorded passes are actually in the middle of the cutter's life, classification is no longer trivial. For these cutters, all passes which can be assigned an inferred test label are included in evaluation.

The number of cutters in each test set, the number of passes of recorded data, the number of passes with explicitly hand-measured wear levels, and the number of passes used in performance evaluation are shown in tables 3.4 and 3.5.

3.5 Evaluation Metrics

The three primary outputs of our system are the quantized wear labels W_i, wear confidence $P(worn)$ and a remaining life estimate for each pass. The evaluation metrics used for these different outputs are described here.

3.5.1 Evaluation of quantized wear labels

In chapter 5, we describe our approach to increasing the number of cutting passes with quantized wear labels W_i for use in training. While it is beneficial to use these estimated wear labels in training, it is not possible to use them in system evaluation. Attempting to evaluate system performance using only those passes with known wear labels would reduce the number of test cases below an acceptable level for meaningful results. Therefore, evaluation of the output wear labels begins with a mapping of the wear labels W_i to a binary WORN or NOT WORN designation. The mappings for each data set are shown in tables 3.2 and 3.3. When evaluating the accuracy of the confidence output $P(worn)$, we consider a cutter to be WORN if $P(worn) > 0.5$. In an actual installation it is up to the operator to

define the threshold of $P(worn)$ to use in a decision to retire a cutter. Although all passes recorded for a cutter are classified, the evaluation techniques described here are only applied to those passes marked for evaluation (section 3.4) after mapping the quantized wear labels W_i or the confidence estimate $P(worn)$ to the binary WORN vs. NOT WORN labels.

As in past work in tool-wear classification [8, 9, 10], we use accuracy as one metric to evaluate the performance of the binary wear labels. Since there are two types of classification errors, false alarms (Type I) and missed detections (Type II), we also give ROC curves for selected systems. In actual use, it is the missed detections which result in costly damage to the part being machined and are several orders of magnitude more expensive than false alarms. However, excessive false alarms result in the operator ignoring the wear estimates from the classifier.

3.5.2 Evaluation of the wear confidence output

In addition to using the accuracy metric described above when evaluating the $P(worn)$ output, an additional metric is needed. A system which assigns a very high $P(worn)$ to a pass which is actually NOT WORN should be considered to have lower performance than one which, while also considering the pass to be WORN, assigns a lower $P(worn)$. The normalized cross entropy metric (NCE) used in speech recognition applications [25] provides this ability to discriminate between systems. The normalized cross entropy metric evaluates the amount of information contained in the output of the classifier rather than simply the accuracy of the classifier labels:

$$NCE = \frac{H(C) - H(C|X)}{H(C)} \tag{3.6}$$

where $H(C)$ is the cross entropy inherent in the system and $H(C|X)$ is the cross entropy which remains after the information contained in the observation X is added.[2] Let $C_i = 0$ represent NOT WORN and $C_i = 1$ represent WORN. The cross entropy computed from the

[2]Note that the terms in the NCE calculation are cross entropies, since the expectation uses the empirical distribution of the test set and the probabilities P_i and $P(worn|x_i)$ are estimated based on the training data.

prior distributions of the two classes is

$$H(C) = -\frac{1}{n}\left[\sum_{i=1}^{n} C_i \log P_1 + (1 - C_i)\log P_0\right] \tag{3.7}$$

where n is the number of passes in the test set, P_1 is the relative frequency of a WORN label in the training set and $P_0 = 1 - P_1$. The cross entropy after the output of the classifier is added, $H(C|X)$, is determined using equation 3.7 after replacing P_1 with $P(worn|x_i)$ and P_0 with $P(not\ worn|x_i)$; where x_i is the predictor used by the GLM for pass i; (chapter 6).

$$H(C|X) = -\frac{1}{n}\left[\sum_{i=1}^{n} C_i \log P(worn|x_i) + (1 - C_i)\log P(notworn|x_i)\right] \tag{3.8}$$

For numerical stability, probability estimates are constrained to be always greater than ϵ and smaller than $1 - \epsilon$, where $\epsilon = 10^{-10}$.

We use the following approach to evaluate the statistical significance of the $P(worn)$ measure for competing classifiers. Let P_i^a represent the wear confidence for pass i from system "a" and P_i^b represent the wear confidence for pass i from system "b". The separation (q_i) between the estimate from systems "a" and "b" for pass i is;

$$q_i = \begin{cases} P_i^a - P_i^b & \text{for } Pass\ label = worn \\ P_i^b - P_i^a & \text{for } Pass\ label = not\ worn \end{cases}$$

We treat the q_i values as samples from a Gaussian distribution, assuming a zero mean under the null hypothesis that the two systems have the same performance. We then determine the mean and standard deviation of Q and use a standard t-test to check our hypothesis that system "a" is better than system "b".

3.5.3 Evaluation of the remaining life output

The evaluation of the remaining life prediction differs from the other system outputs in that the estimates for ALL passes are included. Since we know the total life for each cutter, we can infer the remaining life at all times in terms of the number of passes until the first time a pass is labeled as WORN. It should be noted however that training of the model for remaining life and evaluation of the results depends upon a fundamental assumption which may be in error. We assume that the first pass explicitly labeled as WORN is the actual end

Table 3.7: *The average life of the cutters in each of our data sets measured in number of cutting passes. The number of passes used in the MSE-End metric for remaining life prediction.*

Data Set	Average Life	MSE-End
Steel 1/2"	12	6
Steel 1"	8	4
Titanium M-1/2"	23	11
Titanium M-1"	8	4
Titanium R-1/2"	4	4

of the cutter life. In reality, the cutter may have reached this threshold at the previous pass or even earlier.

We provide two quantitative measures of the performance of the remaining life predictor; the mean squared error (MSE) and the MSE-End. The MSE metric compares the assumed "actual" remaining life to the predicted remaining life at the end of each cutting pass. The value reported is the average error on a pass by pass basis for all cutters in the test set. While remaining life is instructive at all stages, its performance near the end of life is the most critical. During our testing, we developed remaining life predictors which had good average performance but never indicated that a cutter had reached its end of life. To highlight this performance, MSE-End reports only the average error seen during the last half of the average life for the cutters in the particular data set (table 3.7).

In addition to these quantitative measures, plots of the "actual" and "predicted" remaining life can provide insight into the performance of the remaining life predictor. Looking at plots of representative cutters is what led us to add the MSE-End metric to our tests. A flawed end-point assumption may cause an offset in the remaining life prediction which would result in poor MSE and MSE-End performance. Inspection of the remaining life plot would provide an indication of whether the predicted remaining life is decreasing as expected.

Chapter 4

SINGLE-RATE DYNAMIC CLASSIFIER

Until recently, only static classifiers have been used for tool-wear applications [6, 7, 8, 9, 14]. Feature vectors representing an entire cutting pass or drawn from some portion of a pass were collected, and classification was posed as a binary problem of determining whether these features were generated by a cutter which was WORN or NOT WORN. In reality, cutter wear is a dynamic process. Cutters move from being new to progressively greater levels of wear, and the feature vectors during each cutting event change as the cutter moves through the workpiece. Figure 4.1 shows log energy as a function of time; first for a pass associated with a tool that is NOT WORN followed by a pass from the same cutter when it is WORN. It is clear that the feature values depend upon the stage of the cutting pass, and also that the relationship between the features from different stages within a pass are an indication of wear.

Knowledge of the level of wear in a previous pass can reasonably be expected to improve the accuracy of classification of features from the present pass. Our studies and the work of others such as Thangaraj and Wright [21] have shown that flank wear increases gradually up to a point and then accelerates toward the end of the tool's life. A sudden increase in flank wear in the early stages is less likely than a gradual increase. Including this information in the decision between two wear levels can be expected to help prevent false alarms. This chapter describes our work with a dynamic classifier which exploits the information in the dynamic characteristics of milling tool wear.

4.1 Previous Work

While the majority of classifiers applied to the tool-wear problem have been the static binary systems described in section 2.3, some have begun to treat the problem as a dynamic process. Wu *et al.* [6] included three levels of wear, slight/medium/severe in their classifier,

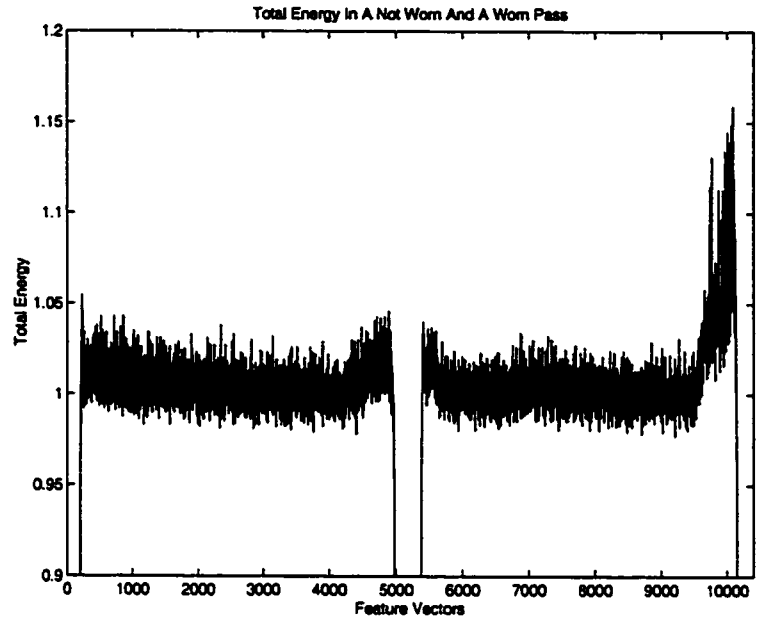

Figure 4.1: *Total log energy profile during two cutting passes. A* NOT WORN *steel cutting pass is followed by a pass from the same cutter when it is* WORN.

demonstrating that they recognized the progressive nature of wear. However, they still treated the wear levels as unrelated conditions. Heck and McClellan [10] attempted to use the progressive nature of wear to their advantage by choosing a dynamic rather than static classifier to monitor drill wear. Thirteen features drawn from power and force sensors were used in a 5 state HMM to classify progressive levels of wear. Both the feed rate and RPM of the drill were changed during testing and the system achieved an overall binary wear classification accuracy of 85%. However, there was no attempt to relate the five wear states to an actual estimate of wear and dynamics within a cutting event were not modeled. Our classifier models both the dynamics of cutter wear and the dynamics within a cutting pass. Each wear level HMM state corresponds to a range of wear on the primary cutting edge, allowing us to produce a quantized estimate of cutter wear.

Whereas Heck and McClellan used HMMs to represent the long time scale of tool-wear, HMMs can also be used to model dynamics at a finer time scale. Owsley *et al.* [26] and McLaughlin *et al.* [27] used HMMs in tool-wear monitoring to represent time variation on

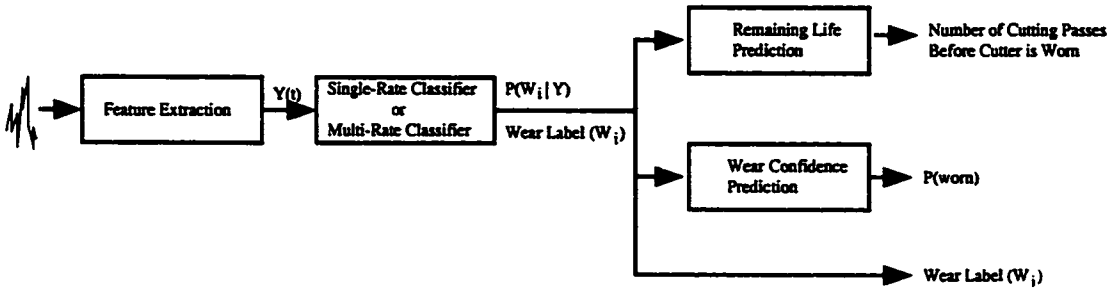

Figure 4.2: *Tool-wear system block diagram.*

the short time scale of transients observed during cutting. Once transients were identified, the vector time series for each transient was classified as having been generated by one of three HMMs representative of transient classes which appeared related to the level of tool-wear. Since training and test data was unlabeled, a WORN/NOT WORN performance evaluation was not possible. While we are interested in short time scale events, our work does not include analysis of the structure of individual transients.

4.2 System Architecture

Our classifier is divided into three serial modules; (figure 4.2). The first processes the raw accelerometer output to generate a series of feature vectors Y_i. The second stage is either the single-rate dynamic classifier described here or the multi-rate dynamic classifier described in chapter 7. Both dynamic classifiers are tasked with tracking the progressive characteristics of tool-wear. The single-rate dynamic classifier processes a series of n feature vectors, $\{Y_1, Y_2, \ldots, Y_n\}$, where $Y_i = \{y_{i1}, \ldots, y_{iT}\}$ is the length-T time series data from pass i. For each pass, the classifier provides as output the wear label, W_i and the probability of each of the wear labels given the feature data $P(W_i|Y^i)$. Note that the wear label decision and posterior probability is based on the whole history of cutter use, which we abbreviate as $Y^i = \{Y_1, \ldots, Y_i\}$. The final bank of modules consists of the generalized linear models (GLMs) described in chapter 6. The particular GLM used depends upon the desired outcome prediction.

4.3 HMMs Used to Model Wear and Pass Level Dynamics

A Hidden Markov model (HMM) is a stochastic model of a process that has piecewise stationary regions, where the time evolution of the non-stationary behavior can be characterized in terms of an unobserved discrete Markov chain. Hidden Markov models are useful, in general, for problems where there are temporal dynamics. As already pointed out, they have been used in tool monitoring applications to capture the progressive increase of wear in drilling [10] and detailed temporal dynamics of milling transients in signals from the early, middle, and late portions of a tool's life [20, 26, 27].

We model the dynamic characteristics of the metal cutting process at two different levels: the wear on the primary cutting edge, as in [10], and the different stages of a cutting pass. The models at both levels are constrained by the physical behavior of the process. We assume that cutter wear increases monotonically. We also note that with some materials, every pass has entry, bulk, and exit stages while with others, one stage of a cutting pass is indistinguishable from another.

The continuous progression of wear from sharp to dull is treated as a left-to-right Markov process. The number of states used corresponds to the number of wear levels selected for the particular experiment. Evaluations performed with data from the cutting of steel and the Series-B titanium tests are divided into five levels labeled A-E (figure 4.3). Series-C tests with titanium add an "excessive wear" state and extend the Markov process to six wear levels A-F. When classifying titanium cutters in the Series-A experiments, the reduced ending wear threshold requires only three states. The range of wear on the primary cutting edge associated with each of these wear states is listed in tables 3.2 and 3.3.

The dynamics associated with different stages of cutting or nosy/quiet periods within an individual cutting pass are modeled with an HMM. In general, HMMs [28] assume that a series of feature vectors (observations) $Y = \{y_1, \ldots, y_T\}$ can be thought of as outputs of hidden (or unobserved) states $S = \{s_1, \ldots s_T\}$. Assuming that the states are discrete and Markov and that the observations are conditionally independent given the current state,

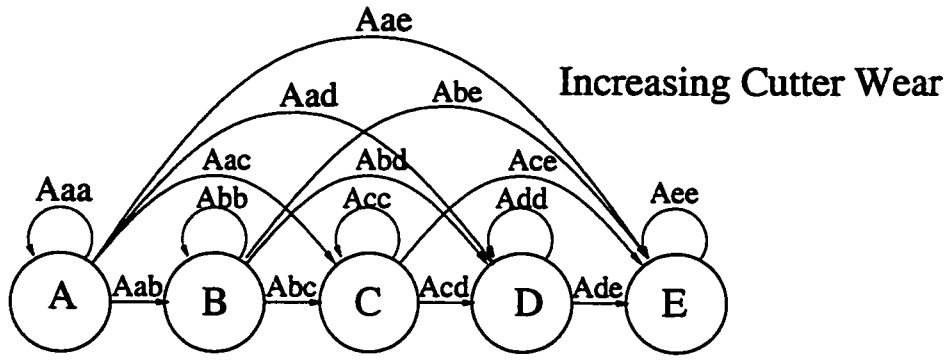

Figure 4.3: *The progressive nature of the metal cutting process is modeled as a left-to-right Markov process constrained to only allow increasing levels of wear. For materials with a* WORN *threshold of approximately 0.010" of flank wear, five or six states are used.*

then the probability of the observations is given by

$$P(Y) = \sum_{S} P(Y,S) = P(y_1|s_1)P(s_1)\prod_{t=2}^{T} P(y_t|s_t)P(s_t|s_{t-1}). \tag{4.1}$$

The HMM is characterized by the observation probability distributions $b_j(y_t) = P(y_t|s_t = j)$, the state transition probabilities $a_{ij} = P(s_t = j|s_{t-1} = i)$ and the initial state probability $\pi_j = P(s_1 = j)$. At each clock time t, corresponding to a new feature vector, the state is updated based upon the transition probability a_{ij}. Once a transition to state j is made, a feature vector y_t is produced according to the probability distribution $b_j(y_t)$. The output distributions for the HMM states are described by either a single Gaussian or by a mixture of multiple Gaussians $b_j(y) = \sum_{k=1}^{M} c_{jk}\mathcal{N}[\mu_{jk}, \Sigma_{jk}]$ where M is the number of mixtures, c_{jk} is the mixture weight and $\mathcal{N}$ is the normal distribution with mean μ_{jk} and covariance Σ_{jk} associated with state j and mixture component k. In this work, diagonal covariance matrices are used.

Once the number and topology of HMM states is chosen, the distribution parameters are learned from training data using the Baum-Welch algorithm. Here the topology is determined in part by the physical nature of the process as introduced above and in part by experimentation, as described further in section 4.4. Analogous to the use of HMMs in speech recognition, classification of wear level consists of finding the best alignment of feature vectors to HMM states via the Viterbi algorithm, which finds the most likely state

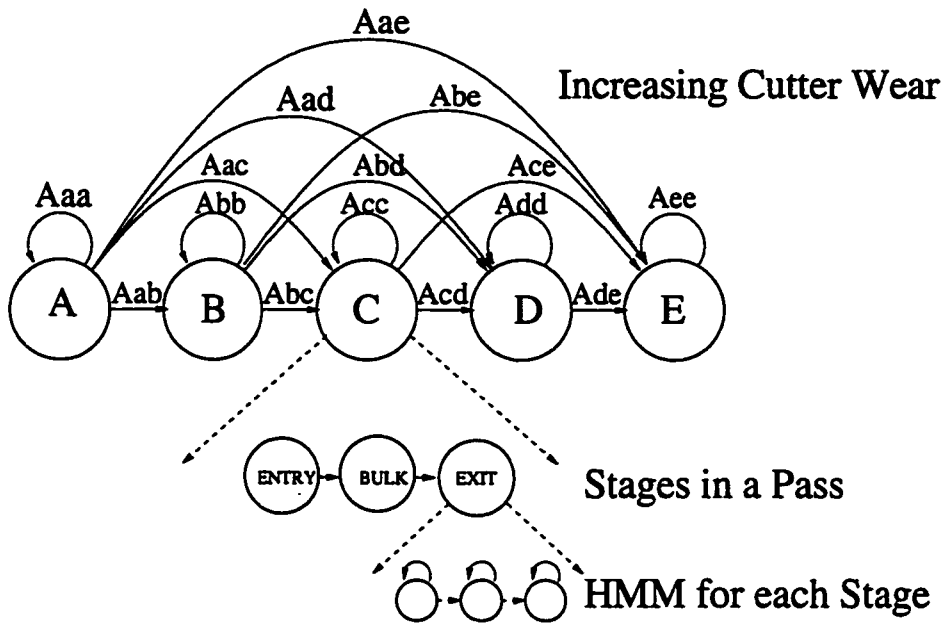

Figure 4.4: *The progressive nature of the metal cutting process is modeled at two different levels. Progressive wear is modeled as a left-to-right Markov process constrained to only allow increasing levels of wear. The progress of a cutter through a single cutting pass is also modeled as a left-to-right process, composed of sequences of HMM states.*

sequence using dynamic programming. The index of the final state points to the wear level.

4.4 HMM Topology

Prior to using the selected feature vectors to train the HMM model parameters, an HMM topology must be chosen to model the dynamics within an individual cutting pass. In our investigations we find that the optimum topology changes when the workpiece material changes from steel to titanium.

Monitoring wear when cutting steel we find that within each wear level, the feature vectors recorded when the cutter first enters the workpiece *(entry)*, behave differently than those recorded when the tool leaves the workpiece *(exit)*, which are both different from those collected during the bulk of the cutting pass *(bulk)*, as illustrated in figure 4.1. This observed ordered progress of a cutter through the workpiece is modeled as a left-to-right Markov process with each stage represented using an HMM (figure 4.4). A cutter is allowed

to remain in any of the three stages of a cut or it may transition into the next. Skipping stages or moving backwards is not allowed. The HMMs for the three stages are concatenated to form a single HMM network for a pass. While the chosen topology constrains the structure of a cutting pass there is no constraint on the duration. Cutting passes may be shorter than one second or as long as desired with no requirement that successive passes are of equal duration. When cutting titanium, this left-to-right process within a cutting pass is not present and the topology of the HMM representing a cutting pass is modified accordingly. We begin with the investigation of various topologies to capture the *entry/bulk/exit* behavior of cutting steel.

Within the different stages of a pass, it is possible that an ergodic state process (effectively a mixture distribution) may be more appropriate than a left-to-right topology to capture the irregular occurrence of events such as chipping. For that reason, we explore various HMM topologies, trading mixtures for sequential states in different positions and testing different numbers of distributions. For steel, the best performance was obtained with nine Gaussians per pass, and the four different cases of this size are shown in figure 4.5. In topology #1, only a single state with a 9-Gaussian mixture distribution is used per pass, equivalent to the topology used by Heck and McClellan [10]. In topology #2, the three stages of a pass are modeled with a single state, each having a 3-Gaussian mixture distribution. Topology #3 imposes a strict linear time progression within each of the stages, using a single Gaussian output distribution per state. Topology #4 combines these structures using the three state/single mixture topology for the *entry* and *exit* stages and the less constrained one state/three mixture topology for the *bulk* stage.

The various cutting pass topologies are evaluated on the steel training cutters using three-fold cross validation models (section 3.4). Model parameters are trained using labels assigned by the algorithm using Viterbi labeling with no rotation (chapter 5). The best performance is obtained by modeling the detailed temporal dynamics at the entry and exit stage, but imposing no ordering constraints during bulk cutting (topology #4). Since a mixture model is expected to be appropriate when modeling transient events, another way of interpreting this result is that while the model must reflect the transients during bulk cutting, the slow time scale phenomena are more important at entry and exit stages.

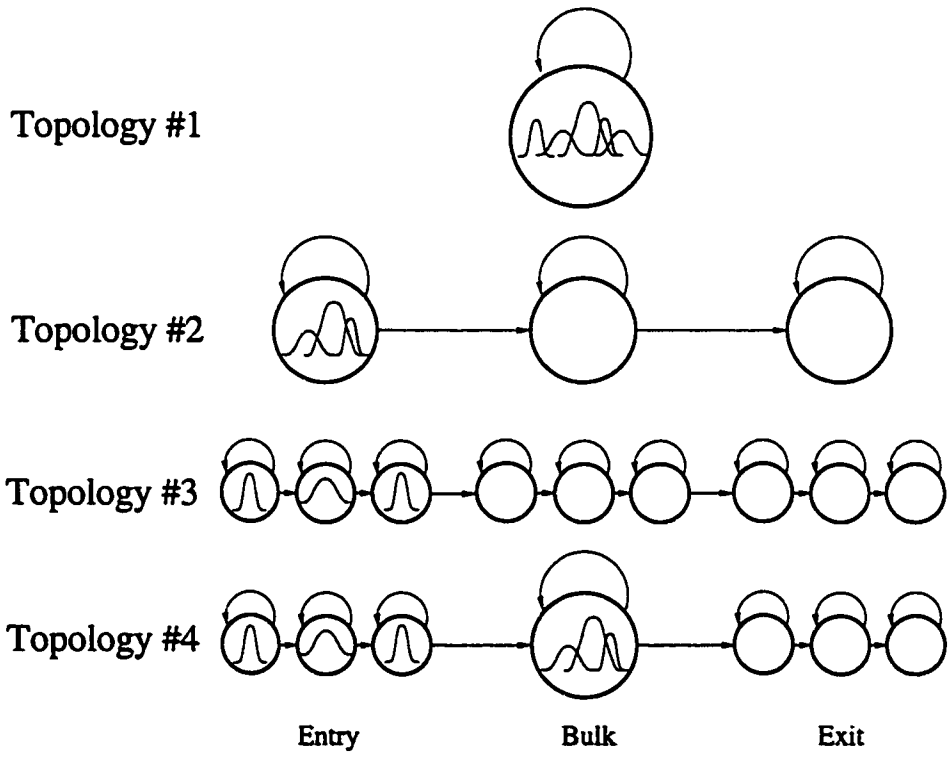

Figure 4.5: *HMM topologies investigated to model an individual steel cutting pass.*

Table 4.1: *Performance (percent correct) of four different topologies evaluated on the steel CV/Train data set. All topologies use nine Gaussian distributions per wear level.*

Topology	Cross Validation Accuracy
Chance	89.0%
#1: 1 state, 9 mixtures	89.0%
#2: 3 states, 3 mixtures each	87.3%
#3: 9 states, single Gaussian	92.7%
#4: 7 states, 3-mixtures in center state	96.4%

This balance between long and short time scales is consistent with intuitions gained from inspecting energy contours of cutting passes.

Inspection of the feature vectors for titanium suggest that it might not have the left-to-right behavior seen in steel. To test this hypothesis, we evaluate three different HMM topologies for the M-1/2" titanium cutters. The first is topology #4 found to have the best performance when used with steel. The second is topology #1 which uses the same number of free parameters but is a single state with nine mixtures. When used with steel this single state/multi-mixture topology gives performance no better than simply labeling all passes as NOT WORN (chance). Finally, we investigate a single state, four mixture model. Topologies with a single state and multiple mixtures outperform the topology intended to model a cutting pass with recognizable left-to-right progression, (table 4.2). This is the topology used in HMM testing during the Series-A experiments on the M-1/2" titanium data set. It is interesting to note that a topology with "chance" performance on steel is the optimum for titanium and the optimum topology for steel results in "chance" performance on titanium.

4.5 Training HMM Model Parameters

Learning the single-rate model parameters begins with training based on the known Boeing labels. These preliminary models are then refined using all training data and the EM

Table 4.2: *Performance (percent correct) of four different topologies evaluated during the Series-A titanium experiments. The left-to-right topology selected for steel is evaluated against topologies better suited to model transient behavior.*

Topology	CV	Test
Chance	85%	83%
7 states, 3-mixtures in center state	85%	84%
1 state, 9 mixtures	95%	94%
1 state, 4 mixtures	95%	94%

algorithm as described in chapter 5.

4.5.1 Training known label models

The specific steps used to train models using only those passes hand labeled by Boeing are outlined below. When a particular HTK tool is used for a step in the model training it is indicated in parentheses: e.g. (HCompV), (HInit), etc.

1. Specify an HMM topology which defines the number of model states but uses only a single mixture for each state.

2. Calculate a global mean and variance for the data from all known passes of the training cutters. A value of 0.01 times this global variance is used as the minimum variance allowed for any output distribution in the final model (HCompV).

3. Create data files containing only those feature vectors from the known cutting passes. Create a label file which assigns each pass its Boeing label but does not attempt to partition the data into "enter/bulk/exit"

4. When the number of states per pass (n) is greater than one, partition each pass into n equal duration segments. Combine all corresponding segments for passes with the

same wear label and calculate a mean and variance to initialize the wear level models (HInit).

5. Set the transition probabilities a_{ij} so that staying in the present state or moving to the next are equally likely. Set a_{ij} for moving backwards or skipping states within a pass to zero.

6. Use Viterbi alignment to assign each feature vector to one of the model states. Use this aligned data Y to update model parameters λ including transition probabilities and output distribution parameters for each state. Repeat Viterbi alignment and model updates N times or until the change in $P(Y|\lambda) < 0.0001$ (HInit). For our work we use $N = 20$.

7. Update the model parameters from the previous step using Baum-Welch re-estimation. Each wear level model is updated individually using only the known passes which have been assigned that label. There is no attempt to string together multiple consecutive passes. Repeat Baum-Welch re-estimation N times or until the change in $P(Y|\lambda) < 0.0001$ (HRest).

8. If a state in the HMM is expected to have more than one mixture, add one additional mixture to that state. An additional mixture is formed by splitting the mixture with the greatest mixture weight into two. Each of the new mixtures is assigned half of the original mixture weight and the new means are defined to be the mean of the unsplit mixture perturbed by plus or minus 0.2 standard deviations (MU). Repeat the re-estimation of the previous step and continue splitting mixtures until the desired number of mixtures is reached.

At this point we have wear level models in the desired topology including multiple mixture states trained on only those passes specifically labeled by Boeing.

4.5.2 Updating models using both labeled and unlabeled data

Once preliminary models have been trained as described in section 4.5.1 the model parameters are updated using *all* of the passes in the training set rather than just those passes with known labels.

i. Create feature vector files which begin and end with passes which have been explicitly labeled by Boeing. If the first cutting pass is not known, create a file which begins with the first pass and ends with a labeled pass. For example, the first pass of steel cutter s1 is labeled "A". Pass #15 is labeled "C" and pass #17 is labeled "D". Two training files are created for s1, passes #1 - #15 are put into the first and passes #15 - #17 are put in the second. The first pass for titanium cutter ti6 is unknown. The only known label is a "C" for pass #5. If ti6 were used in training, a single file would be created containing all five passes for ti6.

ii. Create a lattice defining the network of possible paths connecting the known label on the first pass in the training file to the known label on the last. Incorporate the constraint of only allowing increasing wear (section 5.2)

iii. Concatenate the models for the wear levels defined by the lattice for each training file into a single composite HMM. Run Baum-Welch re-estimation updating only the model means and variances and not the transition probabilities a_{ij} or A_{ij}. [1]

iv. Using the final models from the previous step, perform Viterbi alignment of the feature vectors from all passes for each cutter. Constrain the possible network of models to begin and end with the known Boeing labels. Using the resulting labels on all passes, determine the wear level transition probabilities A_{ij}. In the M-1/2" data set, the data for some cutters does not include early passes. The average life of the M-1/2" cutters is 23 passes. We make a manual adjustment to the A_{ij} values just determined to

[1] The version of HTK used for this step in model training is a modification of the standard HTK toolbox provided by researchers at Johns Hopkins University. While the implementation did not update transition parameters a_{ij} these are already well established with the data from the known labels and leaving them unchanged is not expected to have an impact on the final model performance.

create three sets of A_{ij} values. We use *early* A_{ij} for data beginning at pass #6 or less, *late* A_{ij} for data beginning after pass #20 and *mid* A_{ij} for the remaining cutters.

4.6 *Classification*

Our dynamic classifier can be used for real-time tool-wear monitoring, but because of constraints of our test data we choose to evaluate performance by classifying the wear-level only at the end of a pass. Since our intent is to incorporate the progressive nature of cutter wear into our classifier, our test paradigm allows the use of all previous data in the classification of the present pass. Thus, for each pass i, a decision is made on the wear state based on the full history of observations $Y^i = \{Y_1, \ldots, Y_i\}$ using the Viterbi algorithm. The classification $\hat{W}_i$ is the ending label corresponding to the most likely state sequence. An unconstrained Viterbi search is used at each pass, i.e. the decision at pass i is not constrained to have the same best wear level at pass $i-1$ as in the previous decision $\hat{W}_{i-1}$. This process of adding the data from the latest pass and finding the most likely wear label continues until all passes have been classified and evaluated.

A by-product of the Viterbi algorithm is $\delta(i,l)$

$$\delta(i,l) = \underset{S^i}{\operatorname{argmax}} \log P(Y^i, W_i = l) \tag{4.2}$$

the log likelihood of the observations for the most likely state sequence ending in wear level l. We use this likelihood to approximate the probability of observing the entire feature history for the cutter and ending in wear level l, $P(Y^i, W_i = l) \approx \delta(i,l)$. This quantity is then used to estimate the posterior probability of different levels of wear $P(W_i = l|Y^i)$ (using Bayes rule).

$$P(W_j = l|Y^i) = \frac{P(Y^i, W_i = l)}{P(Y^i)} \tag{4.3}$$

One problem with the determination of $\delta(i,l)$ is that the dimensionality of the observation sequence in a pass is so large ($T \approx 5000$) that the transition probabilities from different wear states $P(W_i = l|W_{i-1} = k) = A_{kl}$ are greatly outweighed by the observation probabilities $P(Y_i|W_i = l)$ and thus have little impact. This does not effect our constraint that wear be monotonically increasing since transitions representing decreasing wear are given a zero

transition probability. However, if not corrected, model performance is degraded since information learned about the typical progression of wear is not properly incorporated in the wear level decision. A cutter in the early stages of its life is much more likely to remain in an early wear state or move to the next higher level than it is to jump to a WORN state. Ignoring this fact when evaluating the steel cross validation data resulted in seven false alarms. A classifier which used this information had none. To prevent this mismatch between $P(Y_i|W_i = l)$ and $P(W_i = l|W_{i-1} = k)$ we modify equation 4.2 to use a weighted combination of the log probabilities in updating the Viterbi cost at the pass level.

$$\delta(i,l) = \max_{S_i:W_i=l} \log P(Y_i, S_i) + \max_{l'}[\delta(i-1,l') + \lambda \log P(W_i = l|W_{i-1} = l')], \qquad (4.4)$$

This solution is standard in speech recognition, where the wear-level transition probabilities are analogous to language model scores (or, word sequence probabilities). The scale factor λ is chosen in rotation testing on the training set to minimize the number of false alarms without any increase in the number of missed detections. It must be recognized that the choice of λ depends to some extent on the duration of a cutting pass. As the length of the observation sequence T changes, the relative weight assigned to the wear level transition probabilities will change. Since our data sets contain cutting passes of equal duration, no testing was performed to evaluate the sensitivity of λ to T.

4.7 Experiments

Experiments with our single-rate classifier include both steel and titanium features calculated at both fine and coarse data rates. Results are based upon only the dynamic classifier and do not include the post processing of the $P(worn)$ GLM. As shown in chapter 6, the addition of the $P(worn)$ GLM provides a more conservative (and more useful) confidence estimate than the HMM posterior probabilities presented here. The finer grained output resolution provided by the GLM also makes ROC plots of the classifier performance more meaningful, so these are not presented until chapter 6.

4.7.1 Steel HMM classification

The performance of the single-rate classifier on fine-rate steel features is shown in table 4.3. As a basis of comparison we include the accuracy of a classifier which simply uses the prior information that most cutting passes are NOT WORN and assigns this label to all passes (Chance). The accuracy of the dynamic classifier is quite good using either energy or cepstral features. Using a one-sided test for statistical significance on the difference in number of errors we can say with 90% confidence that the performance on the steel 1/2" cutters is significantly better than chance. (Note that because the total number of test samples is small, we use a Poisson rather than a Gaussian approximation to the binomial distribution when determining significance.) Because of the limited number of cutting passes in the steel 1" test set, making the claim that the cepstral result is significantly better than chance at a 90% confidence level requires accuracy of 100%. The troubling observation is that the NCE is negative. When the NCE is negative, the priors supposedly have more information than the predicted posterior distribution. We know this is not really the case because the HMM performance is significantly better than chance. What is happening is that the HMM is giving a biased, over-confident estimate of the posterior probability, with $P(W_i = l|Y^i) \approx 1.0$ for the most likely label and $P(W_i = l|Y^i) \approx 0.0$ for all others. Such estimates, when wrong, are severely penalized in the NCE measure. The reason for the high HMM confidence is that the large number of feature vectors ($T \approx 5000$) results in redundant data for classification, i.e. the assumption of conditional independence of observations is not valid. The second stage GLM described in chapter 6 is intended to adjust this bias.

Another possible way to reduce the impact of redundant data is to reduce the data rate. The topology chosen for the steel cutters suggests that the important activity at the beginning and end of a cutting pass may be happening at a rate significantly slower than our fine data rate. While the use of a mixture model during the bulk cutting stage indicates activity at a higher rate, our fine rate may still be excessive. We present here the results of the single-rate classifier processing data at the coarse data rate which is $\frac{1}{40}$ of the fine data rate. For this investigation we add the auditory and auto-ambiguity features described in section 3.3.3 to the energy features used at the fine data rate. The results of classification

Table 4.3: *Performance of a single-rate dynamic classifier (HMM) using energy or cepstral features to classify the steel data sets. Performance is compared to the chance performance achieved by labeling all passes as* NOT WORN. *Next to each accuracy score is listed the P-value for the hypothesis that performance is better than chance (1-statistical significance confidence level). The difference between energy and cepstra results are not statistically significant.*

	Chance	Energy			Cepstra		
Data Set	%	%	P	NCE	%	P	NCE
Steel 1/2" CV	89	96	0.10	-1.43	95	0.16	-2.64
Steel 1/2" Test	77	90	0.09	-2.78	90	0.09	-2.78
Steel 1" Test	89	94	0.46	-2.67	97	0.22	-0.83

with these coarse-rate features are listed in table 4.4. Again using the error counting test, a statistically significant difference over chance at a 90% confidence level requires accuracy $\geq$ 96% for the 1/2" CV data set, $\geq$ 89% for 1/2" test and 100% for the 1" steel cutters.

Energy refers to the same log of the spectrally flattened total energy used as one dimension of the fine-rate *energy* features whose results are reported in table 4.3. This single energy feature is added to the 8kHz energy dimension and to a single interval or count auditory feature from either the 8kHz or 4kHz band. Energy, interval and count feature sets which include delta features are also investigated. In some cases, the addition of delta features is beneficial. In others performance suffers when delta features are included. The auto-ambiguity features use the log of non-flattened total energy (AA Energy) combined with five additional features selected from the auto-ambiguity plane resulting in a six dimensional feature vector.

In our fine rate experiments, performance for the 1" Test cutters defined both "D&E" to be WORN. In these results the WORN threshold is defined to be only "E". This is consistent with the wear level defined to be WORN for the 1" steel cutters and results in a significant increase in performance. Imposing the same definition on the best fine rate system using energy features reduces performance from 94% to 89% which is the same as "chance".

Table 4.4: *Performance of a coarse-rate HMM using energy, auditory (Count) and auto-ambiguity features cutting steel. The use of approximate first derivative features is indicated by* Δ.

Feature	1/2" CV	1/2" Test	1" Test
Chance	89	77	89
Energy + 8Khz Energy	93	90	56
Energy + 8Khz Count	85	94	89
Energy + 8Khz Count + Δ	98	84	53
Energy + 4Khz Count + Δ	91	84	89
AA Energy + Auto-Ambiguity	91	74	89

The results of this coarse-rate classifier are mixed. Feature sets which perform well on one set of cutters do not generalize as well to others. Delta features are helpful in some cases and not others. The only clear result is that some information which is present in the fine rate features is lost when moving to a coarse rate. Even with multiple feature sets to choose from, we are unable to reach the classification accuracy achieved with fine-rate features. The ability of the models to generalize from 1/2" to 1" is seriously degraded by the loss of fine rate details. Simply reducing the data rate is not a solution to the problem of redundant data seen at the fine rate. In chapter 7 we investigate the effect of combining both the fine and coarse rate information.

4.7.2 Titanium HMM classification

In our Series-A testing, we investigate the performance of three types of feature sets (table 4.5). The energy features are the two dimension log(*total energy*) with delta; auto-ambiguity is the five dimension log(*total energy*) plus five ambiguity plane features; and the cepstral features are the selected four cepstral features with their derivatives. The wear related activity at 8kHz seen in the steel data set is not apparent when cutting titanium. The *energy* feature set listed in the table includes only the total energy and its derivative.

Table 4.5: *Performance of three different fine rate feature sets used with the HMM classifier on the Series-A titanium test set. Performance is compared to the chance performance achieved by labeling all passes as* NOT WORN. *Results which are better than chance with confidence of at least 90% are shown in bold face.*

Data Set	Chance	Auto-Ambiguity	Energy w/ Δ	Cepstra w/ Δ
Series-A CV Test	84	**95**	**93**	**93**
Series-A Test	81	**93**	85	86

The performance of the three feature sets is similar on the cross validation testing and show a statistically significant improvement over chance performance. The auto-ambiguity features do a much better job of generalizing to the held out test cutters and are the only one of the three feature sets whose performance is significantly better than chance at a 90% confidence level. NCE results are not listed for the titanium cutters because again they are negative. While the accuracy of the wear labels W_i is quite good, the same difficulty in confidence prediction seen when evaluating steel, is seen in titanium.

Our HMM classifier trained on only 1/2" *steel* cutters is able to generalize to the unseen cutting conditions when applied to 1" steel data. When moving from the M-1/2" *titanium* data to M-1" the changes are more extreme. The "noisy/quiet" cutting periods expected to present problems for our HMM classifier are much more pronounced in the M-1" data set then they are with the M-1/2" cutters. In addition, the wear level extends up to "D" and "E" which are unseen in the M-1/2" training data. We therefore do not attempt to use the classifier trained on the M-1/2" data to classify the M-1" cutters. Rather, the M-1" data is labeled by a single-rate classifier under the train/test paradigm defined for our Series-B tests and also a multi-rate classifier. Single-rate classification with both fine and coarse-rate features is shown in table 4.6. The multi-rate results are reported in chapter 7. The accuracy of the classifier using coarse-rate features is significantly better than chance. The classifier using fine-rate features is only significant at an 80% confidence level.

The Series-C titanium tests are our most extreme case of changing cutting conditions. This data set includes changes to cutting conditions related to the cutter, machining center

Table 4.6: *Performance of the single-rate dynamic classifier (HMM) when processing cepstral features as compared to the chance performance achieved by labeling all passes as* NOT WORN. *Next to each accuracy score is listed the P-value for the hypothesis that performance is better than chance (1-statistical significance confidence level).*

Experiment Series	Chance	%	P
Series-B CV Test, fine-rate features	85	91	0.25
Series-B CV Test, coarse-rate features	85	94	0.09
Series-C Test, fine-rate features	93	93	1.00
Series-C Test, coarse-rate features	93	93	1.00

and workpiece material. In addition we use data recorded with the two different accelerometers discussed in section 5.1.2. As shown in table 4.6, the single-rate classifier performance is no better than "chance". However, the labels assigned to cutting passes reflect the expected behavior of gradually increasing wear but no cutting pass is labeled WORN. The cutting passes which are WORN are assigned wear labels which reflect wear on the verge of being WORN but none are properly classified. It appears that better generalization techniques than those discussed in chapter 5 are required when the differences in the cutters included in training and test are as extensive as those reflected in this data set. It is also possible that the cepstral mean subtraction used for the R-1/2" cutters removed information indicative of wear. The average life of the R-1/2" cutters is four passes. Determining the mean for cepstral mean subtraction with the first cutting pass is analogous to using the first six rather than the first three presently used for the M-1/2" titanium data.

Chapter 5

PRACTICAL TOOL-WEAR CLASSIFICATION ISSUES

Moving a tool-wear system from the lab to the factory floor requires consideration of a host of practical issues which impact the system design. We focus on three of these considerations in the work presented in this chapter. Systems of practical utility must be able to generalize to cutting conditions not encountered in training. Model training must accomodate the limited training data typical of this application. Finally, the system outputs must be in a form that is useful to and understandable by the human operator.

5.1 Issues in Generalization

A numerically controlled milling machine in a typical industrial work cell changes cutting parameters for different operations. Some of these changes have been shown to have an impact on the data generated for classification. In order for a system to be of practical use in classifying tool-wear, it must be able to deal with these changing parameters. While it is not necessary that a classifier function under all possible conditions to be of use, it must not be limited to a single set of milling conditions.

In this section we present our approaches to dealing with changing tool sizes and changes in transducers, focusing on feature normalization techniques.

5.1.1 Changing cutting conditions

Table 3.1 shows the range of cutting conditions in our test sets. Accommodating these changes has implications for the feature processing and the classifier model topology. Here we look at generalizing across changing cutting conditions without a change in the workpiece material. The problem of changing materials will require some changes to the basic model, addressed in chapter 7.

When working with the steel data set, features used to learn the HMM model parameters are drawn only from 1/2" cutters under one set of cutting conditions. As long as the cutting conditions in the test set remain the same as those used in training, performance is good. However, in our initial experiments, when evaluating 1" cutters working under cutting parameters unseen during training, classification accuracy dropped to 11% when even chance performance would have been 89%. All passes from the cutters not seen in training were classified as WORN. With limited processing of the features prior to use in training and test, we are able to make them more robust across changing cutting conditions. For example, when using *energy* features, we normalize the feature vectors. All feature vectors collected during the first pass of cutter j are used to calculate a mean vector μ_j for that cutter. Normalized feature vectors $\tilde{y}_i^j(t) = y_i^j(t)/\mu_j$ are then calculated for each pass and these normalized features are used in parameter training and system evaluation. With energy normalization, classification accuracy on the held out steel test data is comparable for both 1/2" and 1" cutters even though the 1" cutters are not included in the training data. Similar results are achieved with the cepstral mean subtraction described in section 5.1.2.

The generalization techniques used here allow us to deal with the limited changes to cutting conditions reflected in our data sets. More extensive changes to cutting conditions may require more extensive work with the features and/or adaptation of models. The work being done with auto-ambiguity features by Atlas *et al.* [3, 17] shows promise of providing a single feature set which will generalize across different materials and cutting conditions.

5.1.2 Changing feature transducers

Another issue of practical relevance is how to deal with the fact that different transducers are used during data collection. In the energy features described in section 3.3.1, the characteristics of the transducer are an integral part of the features calculated. Assuming a signal $x[n]$ and a transducer with characteristics $h[n]$, the actual data from the transducer will be $y[n] = x[n] * h[n]$. The data used for feature extraction after the FFT is then $Y(\omega) = X(\omega)H(\omega)$. Clearly, the use of a different transducer with characteristics $h'[n]$ will result in changed features $Y'(\omega) = X(\omega)H'(\omega)$ even if the signal under consideration is the

same.

This difficulty is common in the area of speech recognition where different microphones used to record speech are analogous to different accelerometers processing the same vibration signals. The approach in these ASR systems is to rely on cepstral rather than spectral features and use cepstral mean subtraction. The cepstrum is the inverse Fourier transform of the log of the magnitude of the spectrum of a signal[1]; as such, the product of the signal and the channel seen in the Fourier domain becomes a sum in the cepstral domain. This form makes it possible to remove the effects of the accelerometer.

To find the cepstral features,[2] we begin with the Fourier transform of the accelerometer signal for the m^{th} window of data,

$$Y_m(\omega) = \mathcal{F}\{y[n]w_m[n]\} = X_m(\omega)H(\omega), \tag{5.1}$$

where $w_m[n]$ is a moving window function. We then take the log of the magnitude of the spectral features

$$\log|Y_m(\omega)| = \log|X_m(\omega)| + \log|H(\omega)| \tag{5.2}$$

and then transform back to the time domain.

$$\tilde{y}_m[n] = \mathcal{F}^{-1}\{\log|Y_m(\omega)|\} = \mathcal{F}^{-1}\{\log|X_m(\omega)|\} + \mathcal{F}^{-1}\{\log|H(\omega)|\} \tag{5.3}$$

$$= \tilde{x}_m[n] + \tilde{h}_c[n] \tag{5.4}$$

The resulting cepstral features $\tilde{y}_m[n]$ are simply a linear combination of the accelerometer signal $\tilde{x}_m[n]$ and the accelerometer channel $h_c[n]$. Treating the transducer as a linear time-invariant channel allows us to estimate and then remove its contribution from the feature vectors. Its contribution to the final signal is estimated by finding the mean of the feature vectors.

$$\hat{h}_c[n] = \frac{1}{N}\sum_{m=1}^{N}\tilde{y}_m[n] \tag{5.5}$$

---

[1]This is actually the *Real* rather than *Complex* cepstrum. Taking the magnitude prior to the inverse Fourier transform results in the loss of phase information contained in the signal $X(\omega)$.

[2]There are several different ways of calculating the real cepstrum. The Fourier Transform derivation is simplest for explaining channel compensation; our actual implementation is described in section 3.3.2.

where N is the number of windows in the region of data used for channel estimation. The final series of cepstral feature vectors are then

$$\tilde{Y}_c = (\tilde{y}_1 - \hat{h}_c, \ldots, \tilde{y}_N - \hat{h}_c) \tag{5.6}$$

ideally removing the contribution of the linear channel.

Extending this approach to compensate for differences in cutting conditions, we calculate the $\hat{h}_c$ value on a cutter by cutter rather than transducer specific basis. The first three passes of each cutter are used to determine the cepstral mean to be subtracted from all of the data from that cutter. Models trained on only 1/2" steel data using cepstral mean subtraction are able to generalize to 1" steel cutters with no loss in performance (table 4.3). In some cases the first three passes are not available. When this is true, as in our M-1/2" titanium data, a global cepstral mean for that accelerometer and set of cutting conditions is used. This global cepstral mean is determined from the first three passes of all cutters in the training data.

The ability of cepstral mean subtraction to deal with changing accelerometers is demonstrated with two 1" cutters from the steel data set. Vibration was simultaneously recorded with two different accelerometers during cutting with these tools. These are not two instances of the same type of accelerometer but accelerometers with different response characteristics. We train four different single-rate classifiers, one using cepstral features without cepstral mean subtraction, the second using the same cepstral mean regardless of accelerometer or cutter but dependant on tool size, the third using a different cepstral mean when the accelerometer is changed, and the fourth using a different cepstral mean for each cutter and accelerometer.

The wear estimates are evaluated by measuring the "delta" between the labels assigned by the different models, i.e. how different the labels are for each cutting pass. For example if one model assigns a wear label of "A" and another assigns "C" for the same pass, that pass contributes a difference of two. There are a total of 28 cutting passes in our evaluation. If the model using the data from one accelerometer labeled every pass as wear level "A" and the model from the other accelerometer assigned a label of "E" the delta would be 112. The total delta for the models using each form of cepstral mean subtraction are shown in

Table 5.1: *Comparison of models trained with features using three different approaches to cepstral mean subtraction. The models are evaluated on data from two different accelerometers recording the same cutting events. Delta = the sum of the differences in the labels assigned to the cutting pass data from the two accelerometers.*

Cepstral Mean Estimate	Delta
Maximum Possible Delta	112
Same cepstral mean regardless of cutter or accelerometer	30
Cepstral mean specific to accelerometer	15
Cepstral mean specific to cutter and accelerometer	12

table 5.1. Using cepstral mean subtraction removes the majority of the errors introduced by the change in accelerometers. When no cepstral mean subtraction is used, all but the first pass from each cutter is labeled as WORN. These models are trained using only 1/2" cutting data. Just as with energy normalization, some type of cepstral mean subtraction is necessary to generalize from 1/2" to 1". The proper cepstral mean subtraction also allows us to deal with the changed accelerometer.

5.2 Training with Sparsely Labeled Data

When learning the parameters of our dynamic wear level models, we would ideally have a set of training cutters with a wear label assigned to every pass. In practice, this level of annotation is too costly and not practically available. For example, in the steel data set, while there are 87 different passes recorded for the six cutters in the training set, only 12 are inspected so that a wear level could be assigned (section 3.2). Limiting training to only these passes is problematic for two reasons. First, this is insufficient data to properly learn the model parameters $(b_i(y), a_{ij})$ for each wear level HMM. In addition to this, we need to define the transition probabilities A_{ij} between our quantized wear levels W_i. Missing labels in the training data forces us to make heuristic estimates of A_{ij} based on our understanding of the wear process.

Sparse data is a common problem in many classification applications and various approaches have been used to include unlabeled data in training. We present here a technique which makes use of our a-priori understanding of the wear process and unsupervised learning techniques to make use of all of the data in our training sets. The details of the approach are developed on the steel data and applied to each of the data sets described in section 3.2.

Since we define wear level A to be the first pass of each tool, we are able to add six more passes to the labeled data in the steel training set. We refer to these passes as the 18 **known** labels. To assign labels to the remaining unlabeled passes, we investigate four different approaches.

- Viterbi, no rotation
- Viterbi, non-voting rotation
- Viterbi, voting rotation
- Full EM, no rotation

The three different alternatives using Viterbi decoding to assign wear labels are compared to an algorithm which treats the missing labels as hidden states. This last approach corresponds to the Expectation-Maximization (EM) algorithm, so the implementation is a straightforward extension of the standard Baum-Welch training algorithm for HMMs and is analogous to multiple-pronunciation modeling in speech recognition.

In each of these approaches, training begins by estimating model parameters using only the 18 *known* passes, using the Baum-Welch training algorithm for HMMs initialized with a flat start.[3] The mapping of data to the entry, bulk and exit states is learned automatically via this algorithm. These preliminary models are either used to initialize the EM algorithm or to assign labels to the remaining data using Viterbi alignment with the constraints imposed by the known labels and the monotonically increasing nature of tool-wear. More specifically, the constraints are as follows.

[3]The term "flat start" is used when the initial model uses the same observation distribution for all states, which is typically estimated from the full set of data.

1. The first pass of every tool must be labeled "A" since we define this as the wear level for a fresh tool and we know that each tool in the training set is sharp initially.

2. We require that *known* passes cannot change their labels even if the change results in a more likely fit of the data to the model.

3. Labels can only reflect the same or increasing wear, (figure 5.1).

As an aside, we note that when these preliminary models are used with no constraints to label *all* passes for each of the six training cutters, some of the correctly labeled *known* passes have their labels changed and other passes are assigned labels which indicate decreasing cutter wear. When using this completely unsupervised labeling, performance degrades somewhat, as expected.

When applying this technique to data where early cutting passes are omitted, the first constraint is relaxed to allow the unlabeled cutting passes prior to the first known pass to be assigned any label reflecting wear less than or equal to the first known label. The second network in figure 5.1 shows an example of such a case.

To apply these constraints, we first partition the pass data for each cutter into files bounded by the known labels. For example, one of our steel cutters has known labels of A at pass 1, C at pass 10 and E at pass 13. The thirteen passes are partitioned into two files, one containing passes 1 - 10 and the other with passes 10 - 13. The network of HMMs chosen to label these passes is constrained to begin and end with the known labels.

When using constrained **Viterbi with no rotation**, the preliminary models trained on only the known labels are used to label all passes in the training set, including those without known labels. All passes, including the newly labeled *unknown* passes, are then used to re-estimate model parameters and the updated models are used to again assign constrained labels to the same six cutters. This process of labeling all of the training data and then using the newly labeled data to re-estimate model parameters continues until labels stop changing. This approach has the advantage of using as much training data as is available to estimate model parameters, but the disadvantage that it may converge too quickly to

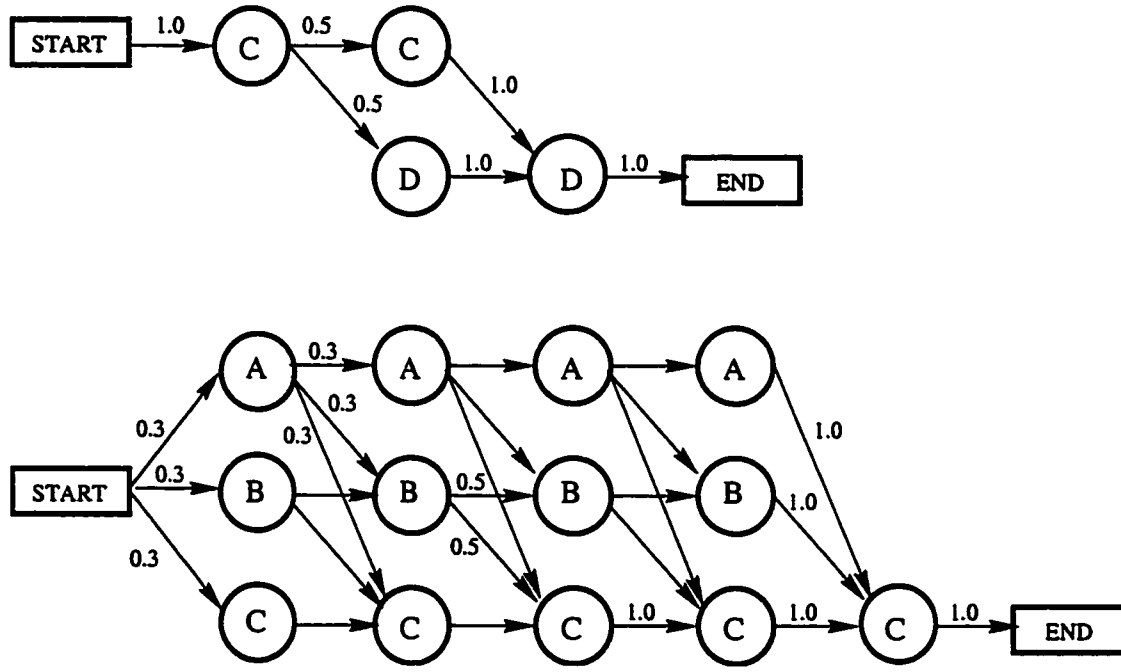

Figure 5.1: *Lattices used in the training of wear level models. The first lattice shows the network for the file containing the final three passes of cutter s1 in the steel data set which has a defined label at both the beginning and end. The second lattice shows the network for the five passes of cutter ti6 from the M-1/2" titanium data set which only has a known label for the last pass. The numbers on the arcs between wear states indicate the wear level transition probability.*

a local optimum associated with the initial labeling and is less likely to generalize to the different cutters in the test sets.

Using constrained **Viterbi with a non-voting rotation**, the six training cutters are separated into six different groups allowing us to do 6-fold cross validation labeling. Each combination uses five cutters for parameter estimation and the sixth is held out for labeling. Since the passes in the held out cutter which are being labeled are not used in the estimation of the parameters of the model used for labeling, this approach is expected to generalize better to unseen test examples.

In both the no rotation and non-voting rotation algorithms, the feature vectors from pass i are used to train parameters for the HMM corresponding to the selected most likely quantized wear level $\hat{W}_i$. This is true even if the probability of another wear state is nearly as great as the one chosen. The two remaining approaches are aimed at addressing this problem, by eliminating the data in one case and using weighted counts for alternate wear levels in the other.

In the constrained **Viterbi with a majority vote rotation**, the six training tools are separated into fifteen different train/test partitions. Each uses four cutters for parameter estimation and holds out two for labeling resulting in each pass being labeled by five different models. These multiple labels allow us to ignore passes which receive different labels from different models when re-estimating model parameters. If a pass is assigned the same label by at least three of its five rotation models, the data from that pass is considered to have a high enough likelihood of belonging to that wear state to be used in the next re-estimation of model parameters. If there is not this level of agreement, the data from that pass is not used in the subsequent re-estimation. It is expected that the slower convergence resulting from not using passes which do not receive the majority vote in the early iterations of parameter estimation will avoid local optima.

The standard way to deal with unlabeled data in HMM training would be to treat the labels as hidden variables and estimate the parameters using the **Full EM algorithm**, i.e. find the posterior probabilities of the different wear levels for unlabeled passes and update parameter estimates using weighted counts. As in the Viterbi solutions, when applying this technique constraints are imposed so that wear is non-decreasing.

Table 5.2: *Performance (percent correct) of models trained and evaluated on the steel data set with different approaches to learning of wear labels.*

	Train	1/2" Test	1" Test
Known labels	82	84	89
Viterbi with no rotation	96	94	86
Viterbi with non-voting rotation	93	94	89
Viterbi with majority vote rotation	87	87	94
Full EM	96	90	94

The performance of models trained using each of these different learning techniques is shown in table 5.2. Each approach which uses unlabeled data in training shows a performance improvement over the system trained with only the known labels. As expected, the performance improvement on the 1/2" test set increases as more cutters are used in the re-estimation of model parameters. (Note that the majority vote rotation does not use all available training data which may explain the lower performance on the 1/2" test set.) However, for the Viterbi solutions, a significant improvement on the 1/2" cutters is offset by a performance decrease when generalizing to the 1" cutters. The EM model training has both the advantage of using the maximum number of cutters for parameter re-estimation and generalizability. This technique is used to estimate model parameters for all results reported in this work.

5.3 Secondary Processing for Human Operators

tool-wear monitoring has been the focus of much promising research over the last two decades. However, to date no practical cutting tool condition monitoring system has been developed [29]. The non-linear, time-variant nature of the machining process [30] makes it difficult to model. Signals from sensors are noisy and dependent upon changes in cutting conditions. Attempts to use automated systems as the sole arbiter of the decision to continue using a cutting tool or replace it have resulted in operators turning the classifier off or

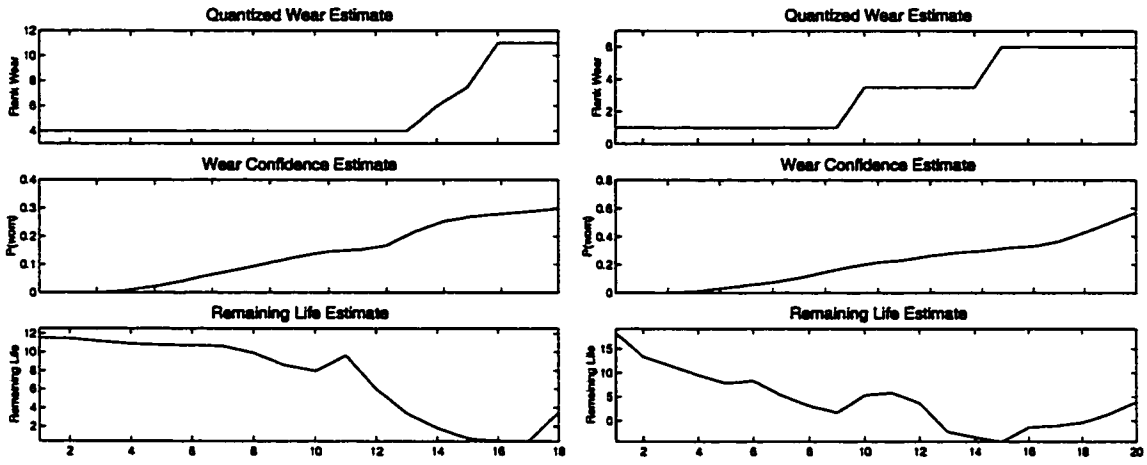

Figure 5.2: *The quantized wear estimate W_i, wear confidence estimate $P(worn)$ and the remaining life prediction for two cutters from our steel and titanium test sets. The plot on the left shows the three outputs for a steel 1" test cutter classified with a single-rate HMM. The plot on the right is from an M-1/2" test cutter classified with a single-rate HMM.*

ignoring it.

The system described here is intended not to replace the machinist on the factory floor but to become a part of the decision process by providing useful information about the state of wear on the cutting tool. A practical consideration in the design of such a system is what types of output information are useful. We provide three different outputs for the operator (figure 5.2).

At the end of each cutting pass our system generates a wear label which indicates that the average wear on the cutter is within a defined range. This is used in two ways by the operator. Cutter wear tends to progress gradually in the early stages and then accelerate toward the end. A wear label indicating that the cutter is still in one of the early stages indicates that close supervision is not necessary. As the wear label approaches the threshold of wear desired for the particular operation, the machinist can more closely monitor the cutting and decide when to stop machining with this cutter. Providing multiple wear labels prior to that designating a WORN tool allows the machinist to monitor the rate of wear as well as the latest wear estimate. If the operator notices that the wear of a particular cutter is moving through the wear levels at a greater rate than expected, this is a cue that

something unexpected is happening.

The wear label just described is the classifier's "best guess" about the wear on the cutter. Sometimes the selected label is the only reasonable fit of the data and the confidence in the label being correct is high. In other cases, multiple labels may be possible choices for the data being evaluated. Rather than simply giving the operator our "best guess" we provide a confidence in the cutter having crossed the threshold of acceptable wear, $P(worn)$. The operator may then choose to continue cutting with a tool which has been given a WORN label if the confidence in the cutter actually being WORN is low. Alternately, if other factors such as the cost of the part being machined warrant, the cutting may be stopped prior to reaching a WORN label if the confidence in the cutter being WORN has reached a sufficient level.

Finally we provide an estimate of the remaining useful life for the present cutter. As with the wear label, this indicates when close supervision is necessary and also indicates when cutter wear is not progressing as expected. Unlike the wear label which may remain the same for several cutting passes, this output indicates a decrease in the remaining useful life even if the estimated wear label remains the same.

The determination of the quantized wear estimate W_i was described in chapter 4. In chapter 6 we describe our approach to finding the $P(worn)$ wear confidence estimate and the remaining life estimate.

Chapter 6

SECONDARY PROCESSING

The three serial modules in our tool-wear system are shown in figure 4.2. Feature extraction is discussed in chapter 3. The single-rate classifier or multi-rate classifier used in the second module are described in chapter 4 and chapter 7 respectively. The final module generating either a confidence in the probability that the cutter is WORN, or a prediction of the remaining life is discussed here.

6.1 *Confidence Estimate of* WORN *Classification*

Both the single-rate and multi-rate classifiers generate a quantized wear estimate W_i. This quantized estimate can be made more continuous by using the posterior probabilities $P(W_i = l|Y^i)$ to determine the probability that the WORN threshold has been exceeded, $P(worn)$. However, the redundant data in these first stage models result in an overconfident and unrealistic confidence estimate. A second stage consisting of a generalized linear model (GLM) is added to help with the overconfident output (or, bias) in the posterior probabilities of the first stage. A GLM is chosen because the sparse training data requires a model with a limited number of parameters. A vector of GLM parameters or **predictors**, x_i is used in a logistic regression to calculate

$$P(worn|x_i) = \frac{e^{\beta^T x_i}}{1 + e^{\beta^T x_i}} \tag{6.1}$$

The selection of the predictor x_i used here is described below. Once the GLM has been trained using predictor features and known WORN vs. NOT WORN labels from the training set, the same logistic regression is applied to predict the $P(worn|x_i)$ for test passes.

Our GLM is implemented with the S-Plus statistical analysis software [31]. Specifying the GLM requires selection of the features which will be used as predictors x_i and training the regression coefficients β. The vector of features used in the GLM may contain both

numeric and factor entries. A factor is a discrete variable which represents values from some specified set of possible levels. The wear label $W_i \in (A,\ B,\ C,\ D,\ E)$ is an example of a factor. When a factor is used, the single entry is expanded into a set of binary variables each acting as an indicator function representing the presence or absence of one of the possible levels. During training, a regression coefficient is determined for each of these binary factor variables.

Using the steel data set, we investigate four different predictors tried individually and in combination. The first uses only the categorical label selected to be the most likely for each pass, $\hat{W}_i$. Since this is a five level factor feature, its model uses the regression

$$\beta^T x = \beta_1 f_A + \beta_2 f_B + \beta_3 f_C + \beta_4 f_D + \beta_5 f_E \tag{6.2}$$

where the factors $f_l = 1$ if $\hat{W} = l, l \in (A, B, C, D, E)$ and 0 if $\hat{W} \neq l$. The second predictor is a vector $\bar{P}_i$ of the probabilities $P(W_i = l|Y^i)$ for each of the five wear labels l. The third predictor is again the most likely wear level but using a numeric representation $\hat{\omega}_i$ (Table 3.2), which requires only a single regression coefficient. The fourth predictor is the ratio $\mathcal{L}_i$:

$$\mathcal{L}_i = \log \frac{\max_{l \in \{\mathcal{W}\}} P(W_i = l, Y^i)}{\max_{l \in \{\bar{\mathcal{W}}\}} P(W_i = l, Y^i)}, \tag{6.3}$$

i.e. the likelihood of the most likely WORN class over the most likely NOT WORN class. The set of labels for the WORN case depends on the particular test: $\mathcal{W}_{steel} = \{D, E\}$ $\mathcal{W}_{TiSeriesA} = \{C\}$, $\mathcal{W}_{TiSeriesB} = \{E\}$, and $\mathcal{W}_{TiSeriesC} = \{E, F\}$. $\mathcal{L}_i$ changes sign when the most likely label changes from WORN to NOT WORN, and its magnitude increases with increased certainty of the WORN vs. NOT WORN decision.

Table 6.1 shows the performance of these GLM predictors. Using $\hat{W}$ alone or with $\bar{P}$ has a negative impact on accuracy even though it results in an improvement in NCE. Using either $\hat{\omega}$ or $\mathcal{L}$ restores the pre-GLM accuracy and gives the desired improvement in NCE. The difficulty with $\hat{W}$ and $\bar{P}$ is probably due to the larger number of free parameters which must be trained for these factors rather than numeric predictors. For this data, where the HMM posterior probability estimate is near 1 or 0, $\bar{P}$ is essentially an indicator vector that is redundant with $\hat{W}$ and introduces too many degrees of freedom for robust generalization. NCE performance using $\mathcal{L}$ is better than that with $\hat{\omega}$ for both diameter cutters in the steel

Table 6.1: *Performance of different predictor variables used to train regression coefficients in the P(worn) GLM: $\hat{W}$ is the categorical wear label, $\hat{\omega}$ is the numeric estimate of wear, $\bar{P}$ is the vector of wear probabilities and $\mathcal{L}$ is the likelihood ratio.*

	Steel 1/2" Test		Steel 1" Test	
Features	%	NCE	%	NCE
no GLM	90	-2.78	94	-2.67
$\hat{W}$	71	-0.33	72	-0.89
$\hat{W}$, $\bar{P}$	71	-0.33	72	-0.89
$\hat{\omega}$	90	+0.58	94	+0.25
$\hat{\omega}$, $\bar{P}$	90	+0.50	94	-0.28
$\mathcal{L}$	90	+0.63	94	+0.62
$\hat{\omega}$, $\mathcal{L}$	90	+0.65	94	+0.57

test set. While using both $\mathcal{L}$ and $\hat{\omega}$ together give a slight NCE improvement on the 1/2" test set, we choose the simplest predictor set giving comparable performance. $\mathcal{L}$ is the single predictor for our $P(worn)$ GLM.

6.1.1 Interpreting NCE

A major difficulty in applying statistical solutions to the problem of milling tool-wear is the limited amount of labeled data for training and system evaluation. Past work in this area has often been limited to using data from only one or two different cutters. The data provided by Boeing is extensive compared to what is generally discussed. However for a solution based upon statistical modeling, the amount of data is very small. In addition to the problems caused for training, the limited number of examples in our evaluation data sets raises questions of statistical significance when we are reporting accuracy results. Since we make use of the NCE metric to ameliorate some of these difficulties, we need to understand its strengths and also its shortcomings.

The NCE metric is expected to fall between zero and one. When $P(worn|x_i) = 1$ for

the WORN passes and $P(worn|x_i) = 0$ for passes which are NOT WORN, $NCE = 1$. If we were using entropy in our calculation, the conditioned entropy $H(C|X)$ could never be greater than the entropy $H(C)$ and the lower bound of NCE would be zero. However, since we are using a *cross* entropy, it is possible to have the NCE take on negative values. The $P(worn|Y^i)$ taken directly from the classifier output or the $P(worn|x_i)$ from the output GLM is not bounded by the $P(worn)$ calculated from the labeled training data. A bias such as that seen in the output of the single-rate HMM results in $P(worn|Y^i)$ estimates of $approx1$ or $approx0$. Siu and Gish [25] have demonstrated that the NCE metric is very sensitive to estimates such as these which are very near the tails of the distribution. Using these $\approx 1/0$ values for $P(worn)$ on our steel CV data set, we get $NCE = 1.0$ when all of the labels are correct. Inserting a single error changes the score to $NCE = -0.215$. The solution proposed by Siu and Gish is to throw out these instances with low probability which they considered "outliers". Since these "outliers" constitute virtually all of our data, this is clearly not an option.

The wear confidence GLM pulls the overconfident estimates of the first stage HMMs and MHMMs away from the tails and makes the NCE metric more indicative of performance. This metric is useful but must be interpreted cautiously. In our experiments with the titanium Series-C cutters, our classifier accuracy was no better than chance. However the output of the wear confidence GLM gave $NCE = 0.88$. This unexpectedly good performance was because all cutting passes were assigned a low $P(worn)$. Since there are many more NOT WORN passes, the bad performance on the WORN passes is not sufficient to impact the NCE. In all of our reporting of NCE results, we also make frequent use of ROC curves and histograms to demonstrate the performance of competing classifiers.

6.1.2 Steel single-rate experiments

The results of the single-rate classifier using energy features reported in section 4.7.1 are repeated in table 6.2. Two additional columns have been added to indicate the accuracy and NCE performance after processing by the GLM. As can be seen, the accuracy is unchanged but the NCE has been dramatically improved. As seen in figure 6.1, the $P(worn|Y^i)$ prior

Table 6.2: *Performance of a single-rate HMM using energy features to classify the steel data set with and without the use of the second stage $P(worn)$ GLM.*

	Chance	HMM		HMM & GLM	
Test Set	%	%	NCE	%	NCE
Steel 1/2" CV	89	96	-1.43	96	+0.49
Steel 1/2" Test	77	90	-2.78	90	+0.63
Steel 1" Test	89	94	-2.67	94	+0.62

to the second stage GLM is approximately 1 or 0 for each cutter. One effect of the GLM is to make this overconfident $P(worn|Y^i)$ from the first stage more conservative which results in the improved NCE.

The range of values for $P(worn|x_i)$ introduced also makes it possible for the operator to effect system performance by adjusting the threshold corresponding to a WORN cutter. As indicated in figure 6.2, changing the WORN threshold for steel 1" test cutters from 0.5 to 0.74 would reduce the false alarm rate without a change in missed detections and result in an improvement in accuracy from 94% to 97%. Providing $P(worn|x_i)$ as an output of the system gives the operator the information to make such a threshold adjustment and improve performance.

The ROCs plotted for the steel test set, figure 6.3, are another way of showing that providing $P(worn|x_i)$ gives the operator the ability to choose the system operating point. In figure 6.3 we show each of the individual data points in the test sets. The ROC plots in the remainder of this dissertation will show only the line connecting the individual data points. Including each data point in plots comparing two competing systems makes the plot too cluttered for proper interpretation. The limited number of test passes in each data set make it inappropriate to plot the typical convex hull for the ROC implying the ability to choose a continuum of operating points. Our ROC plots are shown as discrete steps reflecting actual system performance. In figure 6.3 we do not show the ROC for the $P(worn|Y^i)$ directly from the classifier prior to the GLM. The difference in the wear confidence prediction between

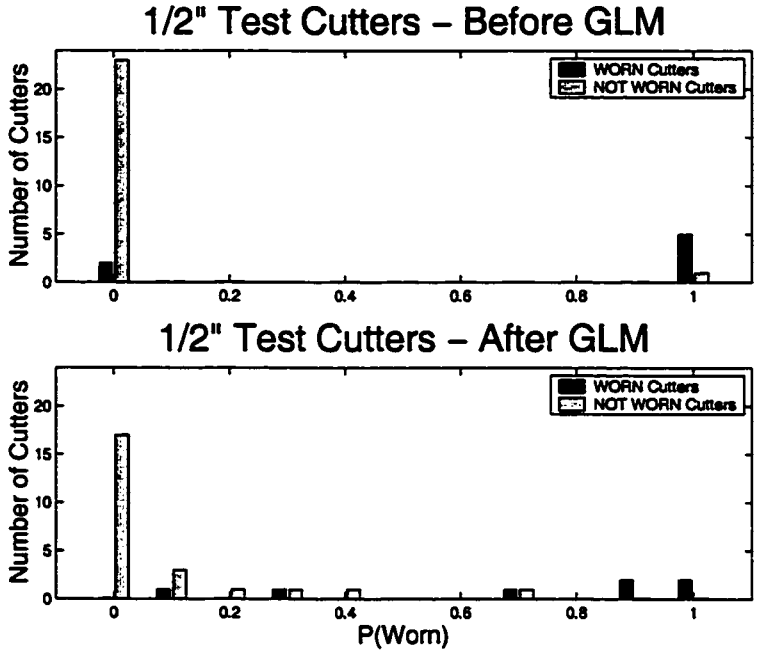

Figure 6.1: *The number of steel 1/2" test cutters at different levels of P(worn) before and after the P(worn) GLM.*

the individual data points is beyond the precision of our processor making it impossible to sweep the operating point and generate a meaningful ROC.

6.1.3 Titanium single-rate experiments

The results of the single-rate classifier using cepstral features reported in section 4.7.2 are repeated in table 6.3. An additional column has been added for each feature set to include the NCE performance after processing by the *P(worn)* GLM. When our only metric was accuracy, the auto-ambiguity features appeared to give clearly better performance than the cepstral features (Test = 93% AA vs. 86% cepstral). However, comparing the NCE score for the two cases shows the cepstral features giving the better performance (Test = 0.188 AA vs. 0.323 cepstral). Looking at the histogram in figure 6.4 adds some insight. Both classifiers assign one pass which is NOT WORN a *P(worn)* which is clearly in the WORN region. Using the auto-ambiguity features, it is possible to reduce the threshold of *P(worn)*

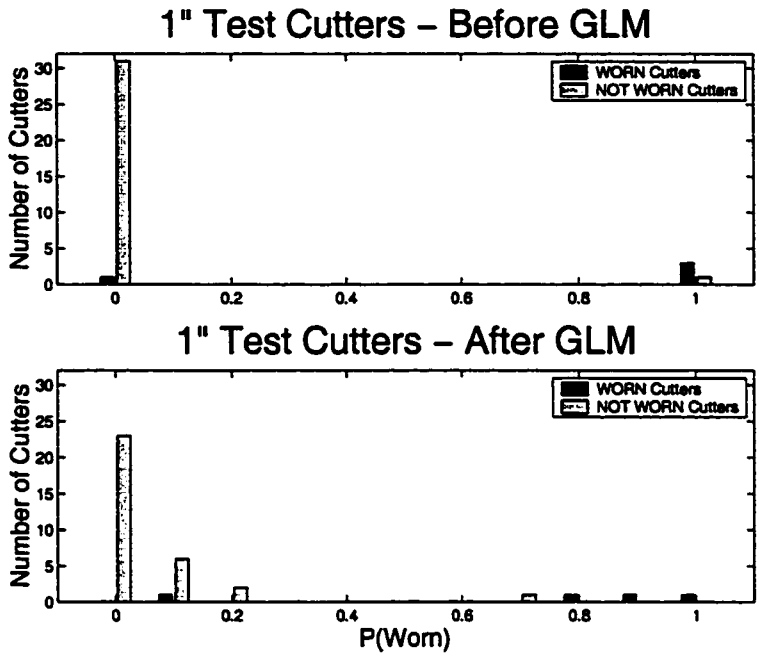

Figure 6.2: *The number of steel 1" test cutters at different levels of P(worn) before and after the P(worn) GLM.*

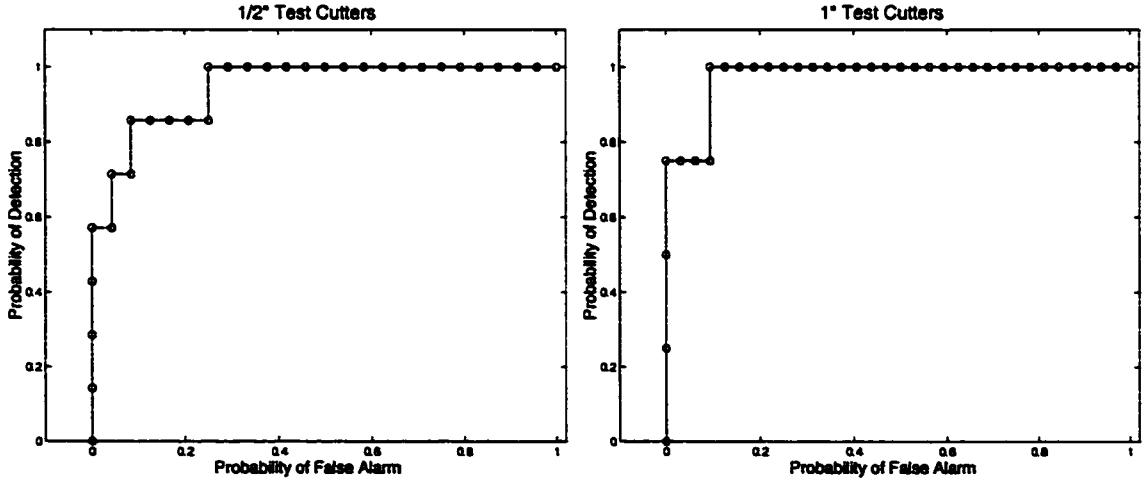

Figure 6.3: *ROC curves for the steel 1/2" and 1" test sets.*

Table 6.3: *Performance of three different feature sets used with the single-rate dynamic classifier (HMM) on the Series-A titanium data set. Performance is compared to the chance performance achieved by labeling all passes as* NOT WORN.

	Chance	Auto-Ambiguity		Energy w/ Δ		Cepstra w/ Δ	
Data Set	%	%	NCE	%	NCE	%	NCE
Series-A CV Test	84	95	0.11	93	0.17	93	0.45
Series-A Test	81	93	0.19	85	0.18	86	0.32

indicative of a WORN cutter enough to detect six of the WORN passes without adding another false alarm. The overlap between WORN and NOT WORN passes in the cepstral classification does not permit this and thus the accuracy is less. However, the cepstral features assign a higher $P(worn)$ to cutting passes which are WORN, and a lower $P(worn)$ to NOT WORN passes than is typical with the auto-ambiguity classifier. This behavior is rewarded by the NCE metric and accounts for the superior performance. Looking at the ROCs in figure 6.5 we see that the cepstral features give better performance on the CV test cutters but the choice of one system over the other when classifying the Test cutters depends upon the desired operating point. Using NCE, the cepstral features give superior performance. Using accuracy, auto ambiguity features are the better choice. Since channel normalization is easy for cepstra, we choose to work with cepstral features. Since auto ambiguity features show promise of generalizing across changing workpiece materials, efforts to deal with changing accelerometers should be pursued.

6.2 Remaining Life Prediction

The primary goal within the reach of present research in tool-wear monitoring is answering the question, "Is it time to replace the present cutter". One way of formulating this question is to ask either for a decision about whether the present cutter has exceeded some defined threshold of wear or to provide a probability of the cutter having passed this threshold. These options correspond to the quantized wear estimate W_i and $P(worn)$ already discussed.

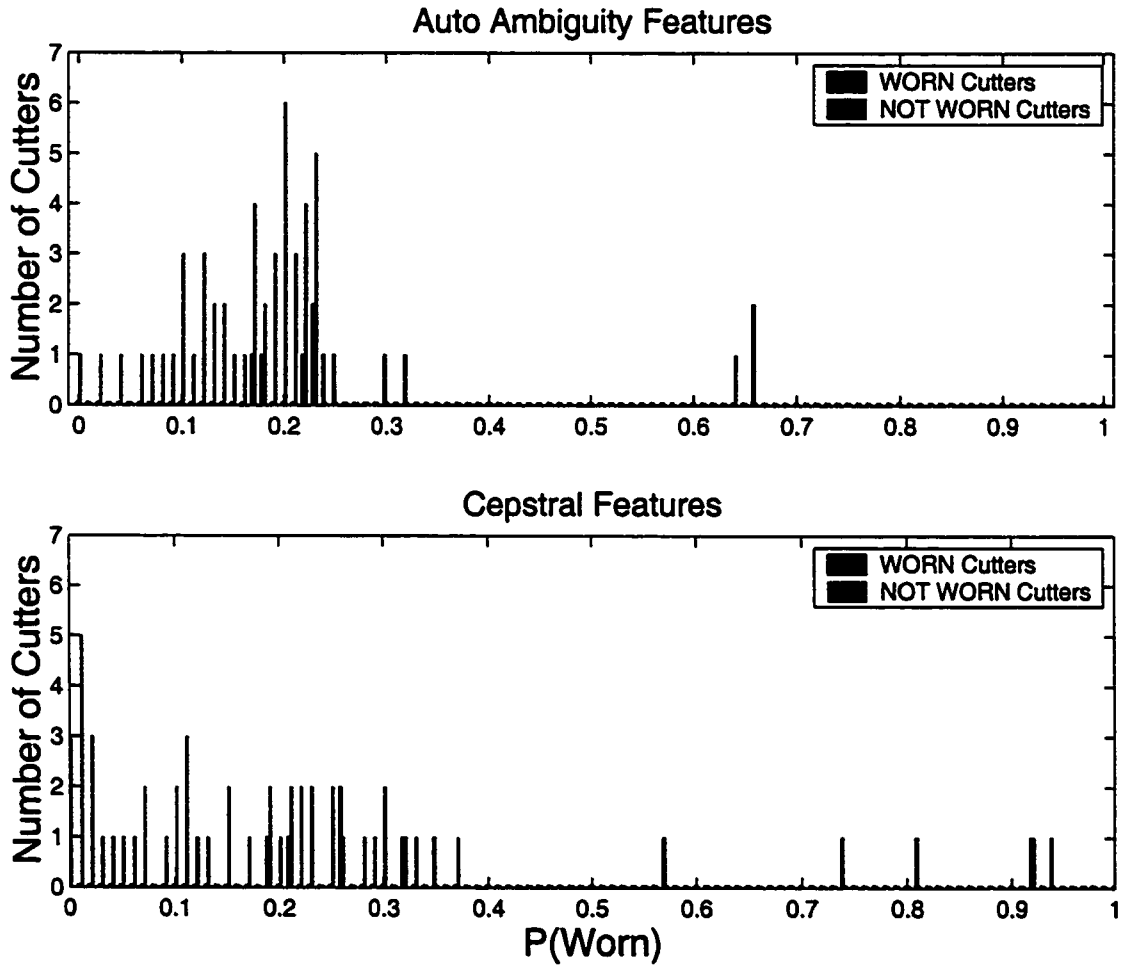

Figure 6.4: *The number of Series-A 1/2" test cutters at different levels of $P(worn|x_i)$. The top plot shows the performance of a classifier using auto-ambiguity features and the bottom indicates the performance of the same classifier using cepstral features.*

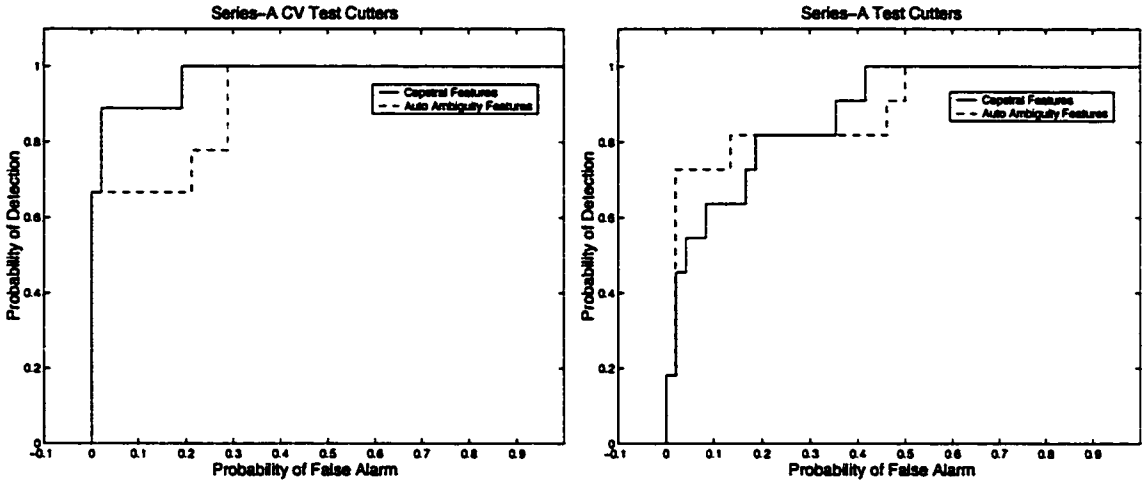

Figure 6.5: *ROC performance for a single-rate classifier using either auto-ambiguity or cepstral features on the M-1/2" titanium cutters. The plot on the left shows performance on the CV Test data set and the plot on the right shows performance on the Test data set.*

Another way to pose the same question is to ask: "How much more life remains on this present cutter?" This is the question addressed by remaining life prediction.

6.2.1 Prediction based on average life

One baseline for remaining life prediction assumes the same average life for all cutters under a particular set of cutting conditions. The remaining life estimate for a new cutter would be set to this average, and the remaining life would be updated by removing the duration of the actual cutting time experienced. Figure 6.6 shows this type of remaining life estimate for three of the M-1/2" titanium cutters. In cases where the cutter behavior is "average", such as the center cutter in the figure, the prediction is quite good. The average life prediction for the first cutter in the figure significantly underestimated its useful life. We define the end of life to be the first cutting pass with a WORN label. Since the first cutter shown in the figure was used for an additional pass after the first WORN label, the actual remaining life is negative. The prediction for the third, estimated that the cutter which was actually WORN still had ten cutting passes remaining. Clearly a better approach is necessary.

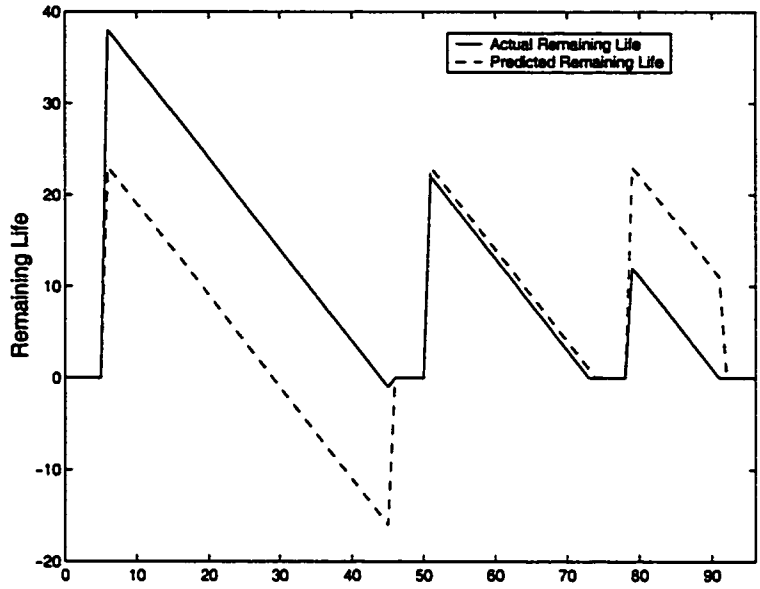

Figure 6.6: *The estimate of remaining life vs. the actual remaining life based upon a constant average life for all cutters used under a particular set of cutting conditions. Performance is shown for three of the cutters in the M-1/2" titanium data set.*

6.2.2 Geometric prediction of remaining life

Each cutting pass can be viewed as a binary variable which can take on the value of being WORN or NOT WORN. The question of remaining life then becomes, "how many passes before we encounter the first WORN"? The geometric distribution describes the probability of the first success after a number of trials. The mean of the geometric distribution then gives the average number of trials before the first success; in our case the average number of passes at one wear level before the first pass at a higher wear level is encountered.

Assigning quantized wear labels to each cutting pass allows us to estimate the wear level transition probabilities used in classification, section 4.6. These wear level transition probabilities include the probability of remaining at the same level of wear A_{ii} and the probability of moving to a different wear state A_{ij}. For each wear level we can estimate the probability of moving to a higher level of wear as $1 - A_{ii}$. We then use the mean of the

geometric distribution to estimate the average number of passes at each wear level.

$$Geometric\ Average\ Wear\ Duration(i) = \frac{1}{1 - A_{ii}} \tag{6.4}$$

The geometric estimate of remaining life for a cutter classified as wear level W_i is then simply the sum of the average duration for wear level i and all higher wear level.

$$Geometric\ Remaining\ Life(i) = \sum_{j=i}^{k-1} \frac{1}{1 - A_{jj}} \tag{6.5}$$

where k is the first WORN label.

The geometric estimate of the same three M-1/2" titanium cutters is shown in figure 6.7. This is an improvement over an average life prediction. However, the Markov assumption at the heart of an HMM is one drawback of using the geometric distribution for a remaining life prediction. The probability of remaining at an assigned wear level A_{ii} is the same each time a pass is classified as wear level W_i. For a remaining life prediction this can be interpreted as saying that multiple cutting passes at the same wear level do not reduce the cutter's remaining life. We know that our wear levels are only a quantized estimate of cutter wear and that with each successive cutting pass its remaining life is diminished. To get a more finely tuned estimate of remaining life, we need to use additional information provided by the dynamic classifier. Just as a GLM was used to estimate the $P(worn)$ in section 6.1, we use a GLM to combine various predictors to determine an estimate of remaining life.

6.2.3 GLM prediction of remaining life

In section 6.1 we saw how information available from the dynamic classifier could be used to estimate $P(worn)$ with a GLM. Combining available information into an estimate of remaining life can again use a GLM or a standard linear model. The classic linear model $y = \beta^T x + \epsilon$ assumes that ϵ is normally distributed with zero mean and constant variance. Design of such a linear model consists of finding the best predictors x and learning the regression coefficients β from training data.

A GLM introduces more flexibility, requiring two functions for its definition. A *link* function which describes how the mean depends on the predictors, $g(\mu) = \beta^T x$ and a *variance* function which describes how the variance of predicted value y depends upon the

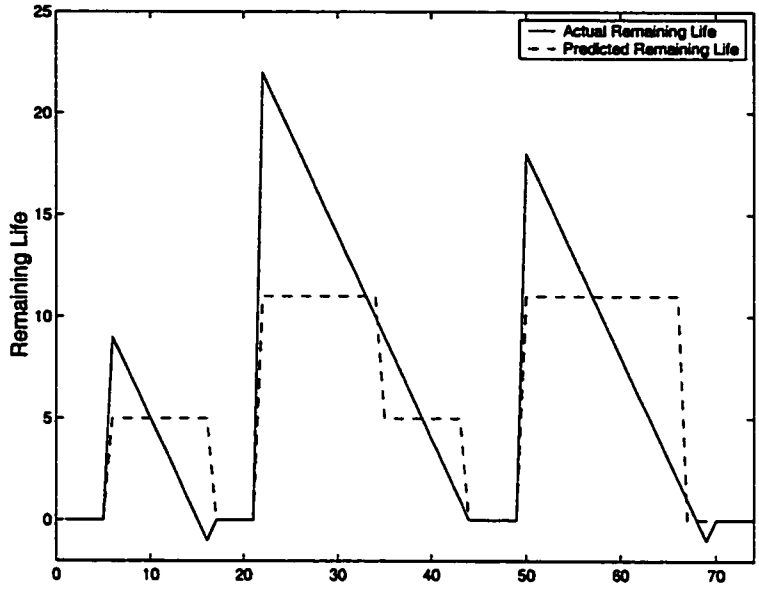

Figure 6.7: The geometric estimate of remaining life vs. the actual remaining life for three of the cutters in the 1/2" Series-A titanium training set.

mean, $var(y) = \phi V(\mu)$ with ϕ constant. The binomial nature of the $P(worn)$ prediction in section 6.1 led us to choose a GLM with a binomial link function (equation 6.1). Remaining life is not a binary process. Design of the remaining life predictor must include the selection of the best link function as well as the requirement to identify the best predictors and learn the regression coefficients [31].

We evaluate performance of both a linear model and various GLMs using a preliminary set of predictors. The linear model is implemented as a GLM by specifying the appropriate link and variance functions. During selection of the GLM link function we use the *Ratio* and *Pass Number* predictors discussed below. In place of the estimate of the probability of the wear label used in our final GLM, $P(W_i = l|x^i)$, we manually define $P(W_i = l) = 1$ when the *known* label is l and $P(W_i = l) = 0$ for all other wear labels. When the label on the cutting pass is not known, we assign a $P(W_i = l) = 1$ to all wear labels which are greater than or equal to the last known label or are less than or equal to the next known label. Regression coefficients are learned from the cutters in the M-1/2" titanium training

Table 6.4: Performance (MSE = mean squared error) of a linear model and various GLMs used to predict remaining life of the M-1/2" training cutters.

Remaining Life Model	Link Function	Variance Function	MSE
Linear	$\mu = \beta^T x$	$var(y) = \phi$	2.72
GLM Inverse Gaussian	$\mu = (\frac{1}{\beta^T x})^{1/2}$	$var(y) = \mu^3 \phi$	3.37
GLM Gamma	$\mu = \frac{1}{\beta^T x}$	$var(y) = \mu^2 \phi$	3.12
GLM Poisson	$\mu = e^{\beta^T x}$	$var(y) = \mu$	2.78

set labeled under the cross validation paradigm described in section 3.4.1. Table 6.4 shows the performance of the linear model and the GLMs from this evaluation. The performance listed is that achieved by each model using the same set of preliminary predictors on the CV training M-1/2" cutters . As can be seen, the linear model and the Poisson GLM model have nearly identical performance. While the Poisson link function is slightly worse when evaluated with an MSE score, it is a bit better than the linear model at tracking the actual remaining life near the end of life. However, the Poisson link function does not allow a prediction of negative remaining life. Using the Poisson GLM would require that we either floor the minimum remaining life at 0 or remove passes which go beyond the first recorded WORN pass from the train and test sets. Flooring the remaining life at zero would include predictor values with more wear than necessary to get to a zero remaining life. This would tend to make it harder to get to a prediction of the end of life. In actual practice, there will be times when the cutter is used beyond the WORN threshold and these cases should be included in our work. We therefore choose to use the GLM which implements a linear model and proceed to find the best predictor features.

The relative success of the geometric prediction of remaining life makes it a prime candidate as a predictor feature in the remaining life GLM. In addition to the geometric prediction we investigate the efficacy of using it along with the following list of possible predictors.

- Geometric: The remaining life based on wear transition probabilities and quantized wear level estimate described in section 6.2.2.

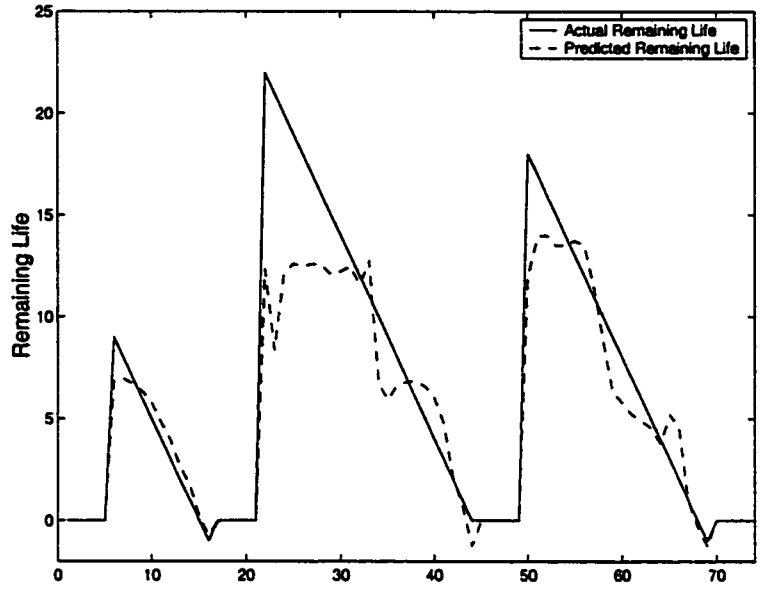

Figure 6.8: The estimate of remaining life vs. the actual remaining life for three of the cutters in the M-1/2" titanium data set using the geometric prediction, $P(W_i = l|Y^i)$ and Viterbi Wear Label Ratios.

- Viterbi Wear Label Ratio (Ratio): The ratio of the log likelihood of the Viterbi path ending in $W_i = J$ to the sum of the log likelihood of the Viterbi paths ending in $W_i = K$ where wear level K represents all labels with greater wear than J. For M-1/2" titanium, for example, this results in two predictors $\frac{\mathcal{L}_A}{\mathcal{L}_B+\mathcal{L}_C}$ and $\frac{\mathcal{L}_B}{\mathcal{L}_C}$. (Note: this is different from the $\mathcal{L}$ feature used for confidence prediction since here we are interested in a continuous range of lifetimes rather than binary categories.)

- Pass Number (Pass): The number of milling passes experienced by the present cutter since it was new. Since evaluation is at the end of a pass, the minimum number for this predictor is 1.

- $P(W_i = l|x^i)$: The probability of the present pass being wear label l for all $l \in$ (*Quantized Wear Labels*)

Each of the predictors listed above except for $P(W_i = l|x^i)$ are available outputs of

the dynamic classifier. Additional GLMs are trained to estimate $P(W_i = l|x^i)$. When our objective is the probability of a WORN label, we choose predictors which are the ratio of the likelihood of the WORN label to the likelihood of the NOT WORN. Here we are interested in the probability of a particular label W_i. The predictors used in these wear label GLMs are the ratios of the Viterbi path likelihood value for each wear label to the sum of the likelihoods of all other wear levels.

$$\mathcal{L}_{i,l} = \log \frac{P(W_i = l, Y^i)}{\sum P(W_i \neq l, Y^i)} \tag{6.6}$$

determined with the data only from those passes with known labels. A separate GLM is trained for each wear label $l \in$ (*Quantized Wear Labels*). These GLMs are used to determine the $P(W_i = l|x^i)$ for all cutting passes, which are then used as predictors in the remaining life GLM. No attempt is made to normalize the wear level probabilities so that they sum to one, since these are used as predictors in another GLM and not directly as probabilities.

Plotting the remaining life using each of these predictors gives a qualitative assessment of performance. The present pass number when used alone never predicts a remaining life of less than five passes. The Viterbi wear label ratio also has trouble getting below five passes and tends to predict life in plateaus rather than ramps. $P(W_i = l|x^i)$ does well at getting down to zero life but underestimates remaining life during early cutting. As already seen, the geometric predictor results in discrete steps rather than a continuous slope. Table 6.5 lists a quantitative assessment of the performance of these predictors used separately and in combination.

The best performance on the M-1/2" titanium cutters is achieved using a six dimension predictor feature vector consisting of the two Viterbi Wear Level ratios, $\frac{\mathcal{L}_A}{\mathcal{L}_B+\mathcal{L}_C}$ and $\frac{\mathcal{L}_B}{\mathcal{L}_C}$, the three $P(W_i = l|x^i)$ values and the Geometric prediction of remaining life. Because of the additional wear labels W_i in the M-1" data set, the remaining life GLM in the Series-B tests uses a ten dimension feature vector.

Table 6.5: Performance (MSE = mean squared error) of various combinations of predictors in a GLM used to predict remaining life on the M-1/2" training cutters.

Pass	Ratio	$P(W_i = l\|x^i)$	Geometric	MSE
X				3.84
	X			3.98
		X		2.83
			X	3.75
X	X			3.53
X		X		2.86
	X	X		2.72
		X	X	2.78
	X	X	X	2.66
X	X	X	X	2.72

6.3 Experimental Results of the Remaining Life GLM

The performance of the remaining life GLM is demonstrated both by the quantitative metrics, mean squared error over the entire cutter life (MSE), and mean squared error over the last half of average cutter life (MSE-End) described in section 3.5, and by plots of actual vs. predicted remaining life. When a plot is shown, all cutters in the data set being evaluated are included.

The cutters in our data sets included average life ranging from four to twenty-three cutting passes (table 3.7). As a basis of comparison, we show the error of a prediction based on this average life when reporting the performance of our GLM remaining life prediction.

The remaining life prediction for the same Series-A titanium cutters used to estimate the GLM regression coefficient β (figure 6.9), shows good performance. However, the plots in figure 6.10 show that the remaining life GLM has difficulties generalizing to the cutters in the test set. However, the test set includes cutters with a wider variation in cutting life

Table 6.6: *Performance (MSE) of the remaining life GLM for the Titanium Series-A and Steel data sets compared to performance using the average life for each test set. The P-value shown is for the hypothesis that the remaining life prediction is significantly better than the estimate based upon the average cutter life. The reduction in the error rate using the GLM remaining life over a prediction based on average life is shown in the column on the right.*

	Average Prediction	GLM Prediction		Reduction in Error
Test Set	MSE	MSE	P	MSE
Titanium Series-A Test	6.28	4.45	< 0.3	29%
Steel 1/2" Test	3.41	3.10	< 0.1	9%
Steel 1" Test	7.30	3.36	< 0.0005	54%

than seen during training. The remaining life GLM still improves MSE performance by about 29% (table 6.6) and MSE-End by 34% (table 6.7).

The same features selected when evaluating the M-1/2" titanium cutters are used as predictors in the remaining life GLM for our steel cutters. When the data set is changed, training cutters from the new data set are used to train a new remaining life GLM. In chapter 4 we presented results of a single-rate classifier using several feature sets. The results presented here use the outputs of the single-rate classifier processing cepstral features. Looking at figure 6.11, we again see a problem when attempting to generalize from the 1/2" training cutters to the cutters in the 1/2" Test set. The 1/2" test set contains several cutters with life much shorter than the average for this data set. The GLM predicts excessive remaining life for these cutters and only shows a 9% improvement over an average life prediction. The steel remaining life GLM actually generalizes better to the cutters in the 1" test set (figure 6.12) showing an improvement over the average prediction of 54% for the entire cutter life (table 6.6) and 29% over the last half of the average cutter life (table 6.7).

Table 6.7: *Performance (MSE-End)of the remaining life GLM for the Titanium Series-A and Steel data sets compared to performance using the average life for each test set. The P-value shown is for the hypothesis that the remaining life prediction is significantly better than the estimate based upon the average cutter life. The reduction in the error rate using the GLM remaining life over a prediction based on average life is shown in the column on the right.*

	Average Prediction	GLM Prediction		Reduction in Error
Test Set	MSE-End	MSE-End	P	MSE-End
Titanium Series-A Test	6.84	4.51	< 0.3	34%
Steel 1/2" Test	3.71	3.54	< 0.3	5%
Steel 1" Test	5.75	4.11	< 0.05	29%

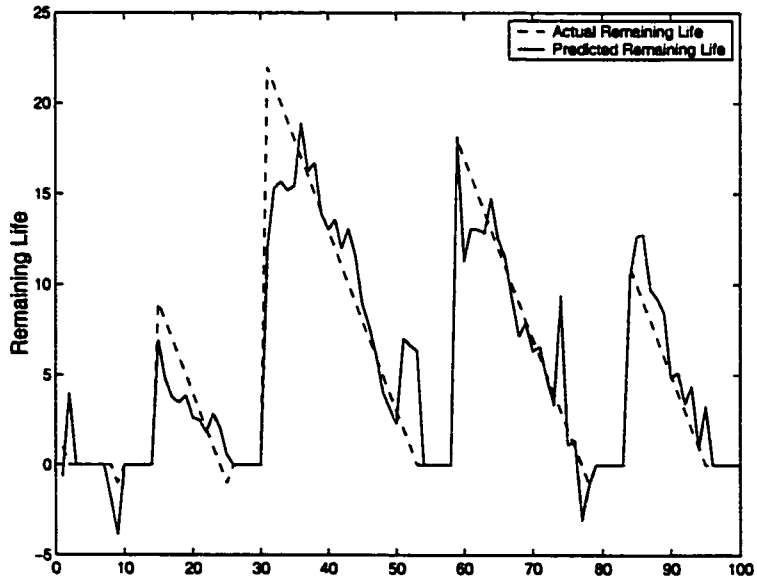

Figure 6.9: *Series-A titanium CV test cutting. Actual remaining life vs. the remaining life predicted by our remaining life GLM is shown for the six cutters in the CV test set.*

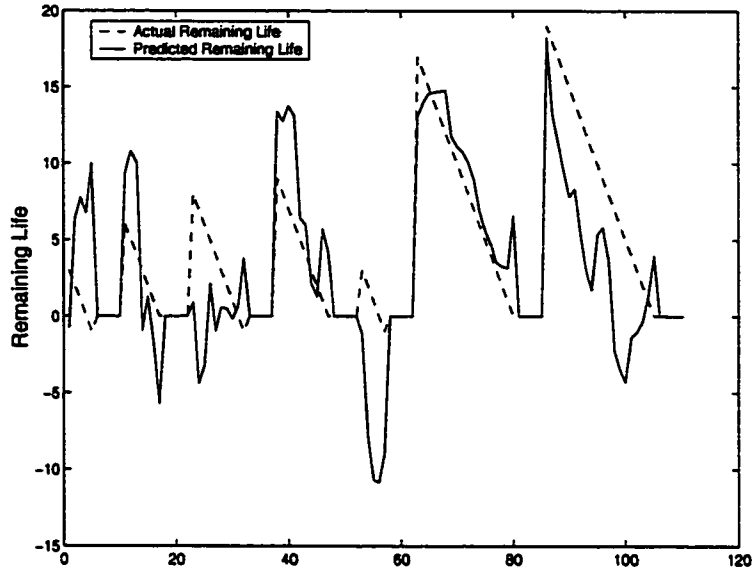

Figure 6.10: *Series-A titanium Test cutting. Actual remaining life vs. the remaining life predicted by our remaining life GLM is shown for the seven cutters in the Series-A test set.*

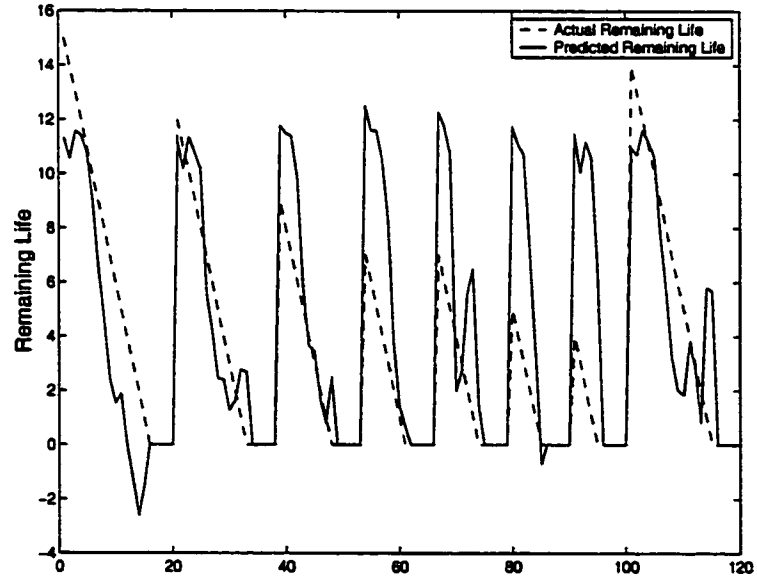

Figure 6.11: *Steel Test cutting classified by a single-rate HMM processing cepstral features. Actual remaining life vs. the remaining life predicted by our remaining life GLM is shown for the eight cutters in the steel test set.*

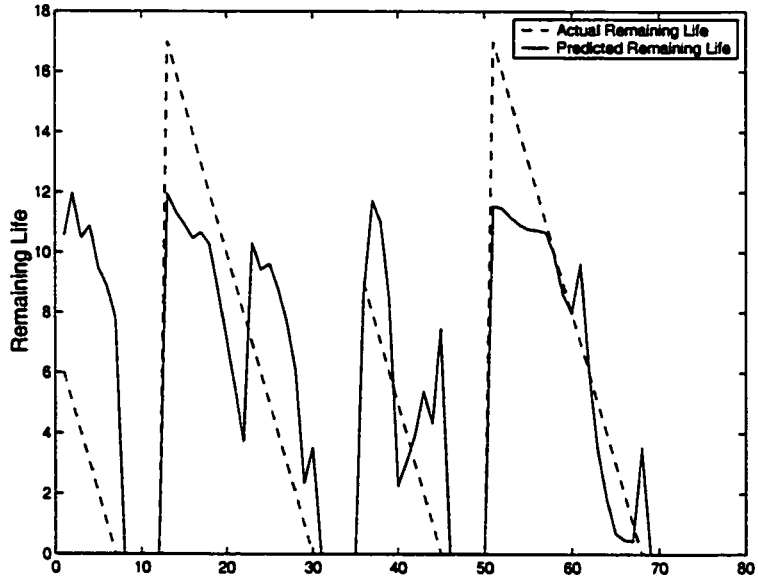

Figure 6.12: *Steel 1" Test cutting classified by a single-rate HMM processing cepstral features. Actual remaining life vs. the remaining life predicted by our remaining life GLM is shown for the four cutters in the steel 1" test set.*

Chapter 7

MULTI-RATE HMM

As we have said several times, wear events occur at different time scales. Rather than choose to work at one scale or the other, we present here a classifier working at multiple data rates. This multi-rate dynamic classifier processes two separate channels of data. Each channel is expected to capture a type of wear phenomenon. The lower rate, or *coarse*, channel is expected to track more slowly varying features indicative of the flank wear, noisy/quiet periods due to BUE or the profile of a cutting pass; *enter/bulk/exit*. The second higher rate, or *fine* channel captures the chipping which occurs at all wear levels but is believed to occur more frequently in higher wear states. We refer to this multi-rate HMM as an MHMM.

7.1 Models for Multi-Rate Processes

HMMs have long been used to model parallel streams of single-rate feature vectors by assuming independent observations in a single HMM. More recently, work has been done using multiband [32, 33] or factorial HMM [34, 35, 36, 37] systems but the focus is on multiple feature streams at the same rate. In this section, we consider different variations of HMMs that can be used to characterize **multi-rate** processes, specifically loosely coupled and state-coupled models.

The approach described here is applicable to more than two simultaneous data rates. However, we have limited ourselves to two to simplify the implementation and because we believe this to be reasonable for the application. In the coarse-rate HMM, the length N sequence of coarse-rate feature vectors $Y^c = \{y_1^c, \ldots, y_N^c\}$ from each cutting pass is assumed to have been generated by one of a variable number of wear level HMMs indicative of progressively greater levels of cutter wear. The number of quantized wear levels depends upon the material under test. The length T sequence of fine-rate feature vectors $Y^f =$

$\{y_1^f, \ldots, y_T^f\}$ is assumed to have been generated by states in a fine-rate HMM which are either wear-level dependent or shared across all wear levels. Being in a particular fine state is intended to represent the presence or absence of a transient. Since our hypothesis is that groupings of transients are related to wear level, the activity in the fine rate HMM is also indirectly related to the average cutter land wear.

In a *loosely coupled* model, parallel classifiers processes features from D distinct data streams $\{Y^1, \ldots, Y^D\}$. Each classifier determines $P(Y^i, S^i)$, where S^i represents the state sequence associated with the i^{th} HMM, independently of the other data streams. A second stage such as a neural network or generalized linear model, F, is used to combine the outputs of the parallel classifiers into a final classification or class probability:

$$P(Y, S) = F\left\{P(Y^1, S^1), \ldots, P(Y^D, S^D)\right\}. \tag{7.1}$$

Figure 7.1 shows a loosely coupled model processing two data streams with feature vectors at different data rates using a graphical model representation. The graphical model shows the statistical independence between the random variables in the coarse and fine HMMs. Figure 7.2 illustrates a hypothetical state topology for such a system. Dupont and Bourlard [38] demonstrated a performance improvement when a similar loosely coupled multi-rate topology was used to capture both phoneme and syllable level information in a speech recognition system. In tool wear monitoring, loosely coupled systems have been used primarily for combining features at the same rate for different sensors [18].

A second *state-coupled* model for a multi-rate classifier couples the parallel HMMs *during* the calculation of $P(Y, S)$ rather than combining the independent probabilities *after* determination of $P(Y, S)$ for each HMM. In this model, the present state S_t^i in data stream i is dependent not only on its own feature and state sequence $\{y_1^i, \ldots, y_t^i, s_1^i, \ldots, s_t^i\}$ but also upon the features and states in one or more parallel HMMs. Figure 7.3 modifies the graphical model of figure 7.1 to show the dependence of the classifier processing features at a higher rate on the classifier processing features at a slower data rate. Considering these to be the *fine* and *coarse* data rates of our multi-rate classifier, we can write the probability of the present wear state S as,

$$P(S) = P(S^c)P(S^f|S^c). \tag{7.2}$$

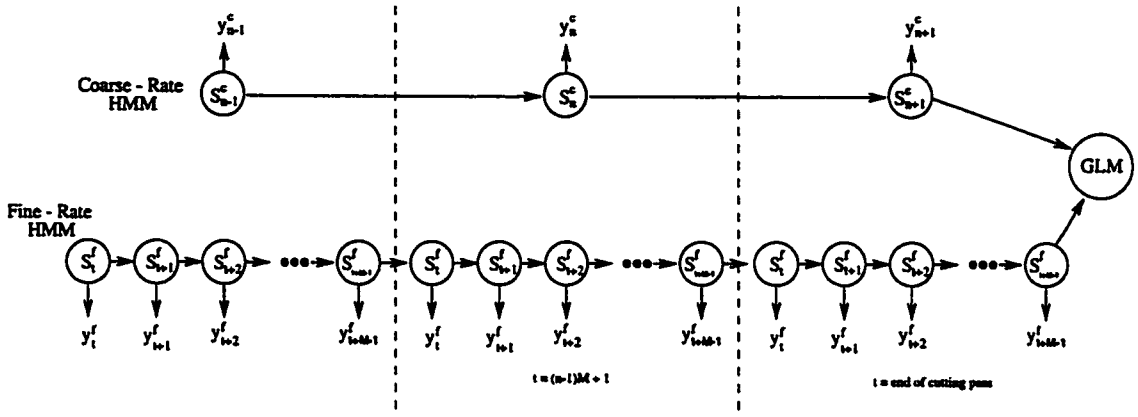

Figure 7.1: *Graphical model indicating the independence of the classifiers processing the two data streams. The final state shows the dependence of the final output on the two parallel classifiers.*

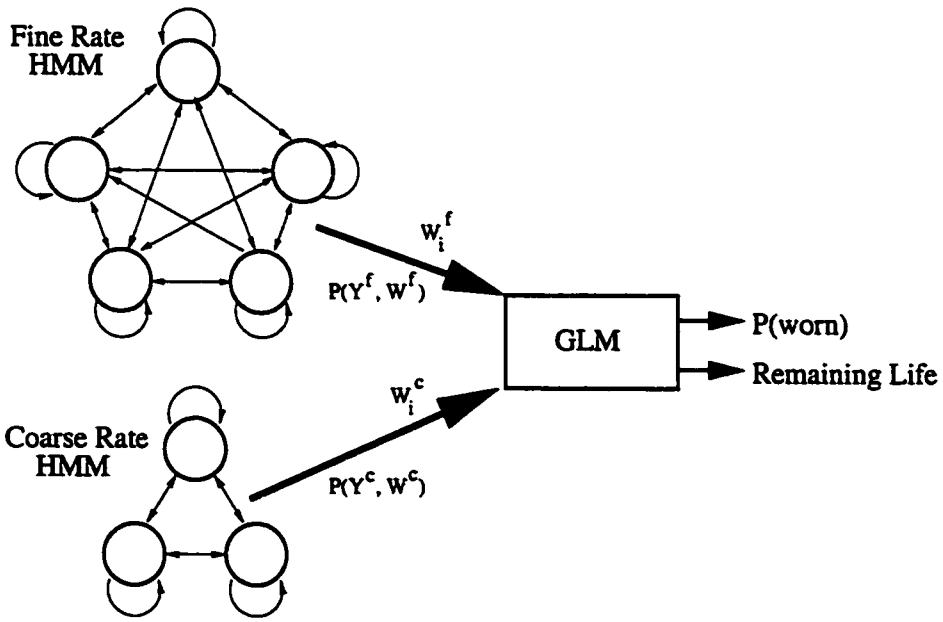

Figure 7.2: *An example of a loosely coupled MHMM combining two independent HMMs via a secondary GLM.*

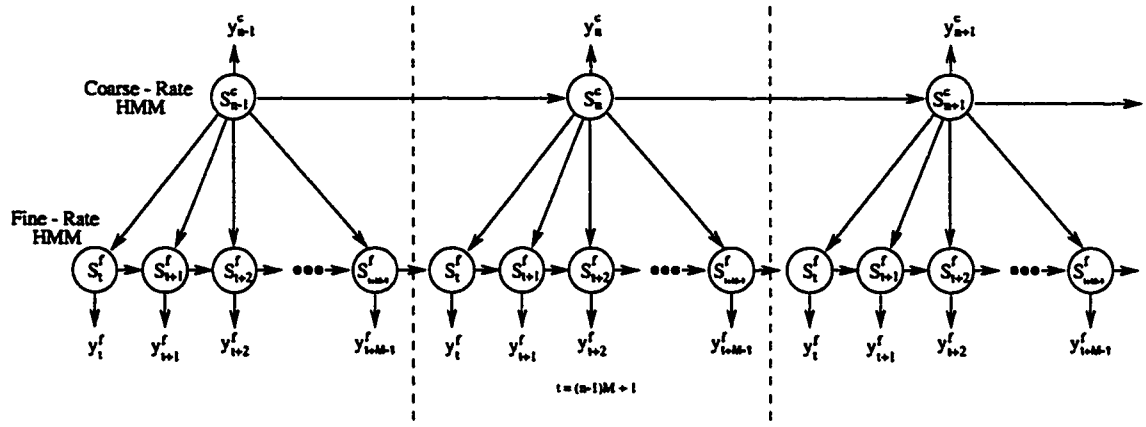

Figure 7.3: *Graphical model indicating the dependence of the fine-rate state sequence on the present coarse state.*

It is possible to make both the transition parameters and output distributions of one classifier conditionally dependent upon another. In our state-coupled MHMM, the fine-rate output distributions are dependent only upon the present fine-rate state. However, the transition parameters are conditioned on both the fine and coarse state. Figure 7.4 is analogous to figure 7.2 in illustrating the state topologies. The coarse-rate HMM in figure 7.2 represents one of the coarse-rate wear-level dependent HMMs shown in figure 7.4.

Making the assumption that states are discrete and Markov and observations are conditionally independent given the current state (within each feature stream), we write the probability of the observations as

$$\begin{aligned} P(Y) &= \sum_{S^c} P(Y^c, S^c) \sum_{S^f} P(Y^f, S^f | S^c) \\ &= \sum_{S^c} \left(P(y_1^c | s_1^c) P(s_1^c) \prod_{n=2}^{N} P(y_n^c | s_n^c) P(s_n^c | s_{n-1}^c) \right) \bullet \\ &\quad \sum_{S^f} \left(P(y_1^f | s_1^f) P(s_1^f | s_1^c) \prod_{t=2}^{T} P(y_t^f | s_t^f) P(s_t^f | s_{t-1}^f, s_n^c) \right). \end{aligned} \tag{7.3}$$

Li *et al.* [39] describe a similar system for image classification. In their work, the two feature streams are at the same data rate. The transition probabilities of the HMMs used to

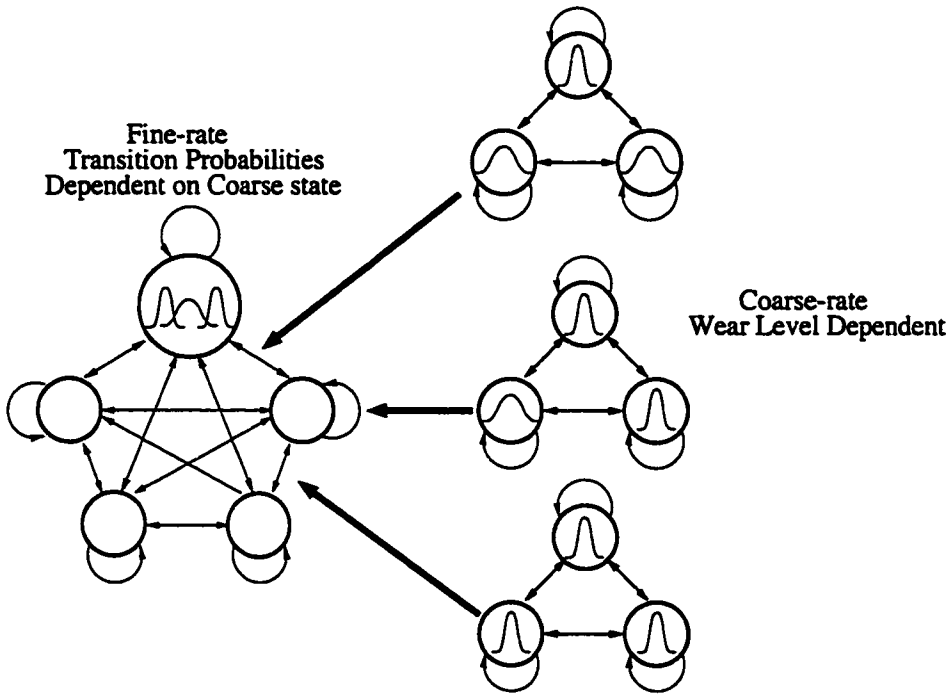

Figure 7.4: *MHMM coupled via the dependence of the fine-rate transition probabilities on the coarse-rate state.*

classify one block of an image are dependent upon the state of HMMs classifying adjacent image blocks.

In our state-coupled multi-rate classifier we model the dependence between the two feature streams explicitly via the dependence of the fine-rate state on the state of the coarse-rate HMM. Since classification involves finding the maximum likelihood wear state S which includes both the fine and coarse rate features, both data rates influence the selection of the most likely wear label W_i. Suppose that transient behavior is captured by $s_t^f = i$ and during training it is seen that transients are more likely to occur at higher levels of wear. When finding the most likely state sequence for the fine rate features, the probability of transitioning to $s_t^f = i$ will be greater when the most likely coarse-level state represents a higher wear level. In this way, the coarse features effect the state sequence in the fine-rate HMM. A sequence of fine-rate features indicative of transient behavior will lead to the selection of a coarse-rate state representative of a higher wear level to increase the likelihood of the fine-rate data. Thus the features at both rates influence the state sequence in both HMMs.

7.2 Multi-Rate Topology

In the loosely coupled MHMM presented here, the outputs of the fine-rate and coarse-rate classifiers discussed in chapter 4 are combined in the $P(worn)$ or Remaining Life GLMs described in chapter 6. For our loosely coupled multi-rate topology, we use a three state ergodic coarse-rate HMM and a fourteen state ergodic fine-rate HMM. The choice of the number of states in the fine-rate classifier is based on our desire to have a similar number of free parameters for comparison with our state-coupled multi-rate classifier, described next. The topology for the state-coupled MHMM takes one of two forms. Just as in the loosely coupled MHMM, the coarse-rate HMM models each wear level with a different three-state, single-mixture ergodic HMM. In the first of the state-coupled MHMM topologies, we assume that the energy or frequency content of transients changes with changing wear level. The fine-rate HMM is a five state single-mixture ergodic HMM whose model parameters are wear-level dependent.

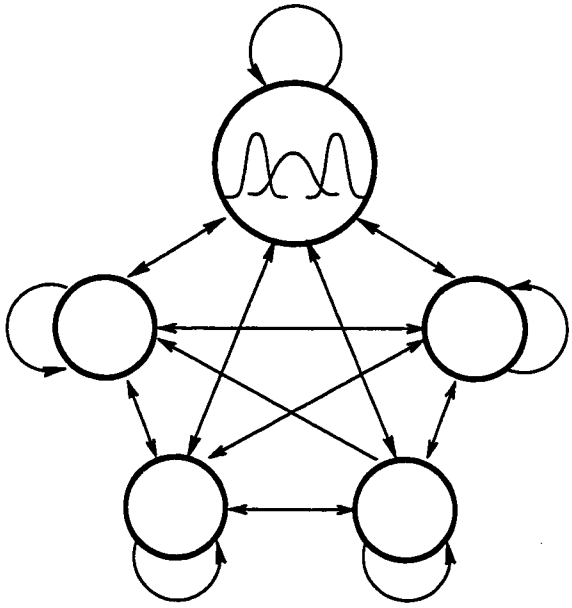

Figure 7.5: *Fine rate topology shared across all wear levels* W_i.

In the second state-coupled MHMM, we assume that the differences in transient behavior over the life of a cutter are limited to the rate of transients rather than changes in the energy or frequency content. In this case, we use the same five state HMM to represent transients at all wear levels W_i. Four of the five states in the fine rate HMM use single mixture output distributions intended to capture transient activity. The fifth state is a multi-mixture model with the number of mixtures equal to the number of wear levels (figure 7.5). This multi-mixture state is intended to model the "normal cutting" or "non-transient" activity which is expected to be wear level dependent. In both models of the fine-rate features, the transition probabilities between states are dependent upon the coarse-rate state. In the topology which ties "transient" states across wear levels, the mixture weights for the observation distributions within the "normal cutting" state are not dependent upon the coarse-rate state.

7.3 *Multi-Rate Decoding with State-Coupled Models*

Classification of cutter wear requires that we determine $P(W_j = l|Y^j)$ for each of the wear labels $l \in$ (*Quantized Wear Levels*) (equation 4.3). We determine this recursively for each additional pass of data. Here $Y^j = (Y_1, \ldots, Y_j)$ represents all passes up to and including the pass being classified. $Y_k = (y_1^f, \ldots, y_T^f, y_1^c, \ldots, y_N^c)$ denotes both the fine and coarse observation sequence from a single pass consisting of T fine rate and N coarse rate observations. The forward algorithm is used to determine $P(Y_k|W_k = l)$ for each wear level l. The wear level transition probabilities A_{ij} learned during training determine $P(W_j = l)$.

What is needed at this point is the details of the forward algorithm for our MHMM. We introduce the quantity

$$\alpha_t(j, J) = P(y_1^f, \ldots, y_t^f, y_1^c, \ldots, y_n^c, s_t^f = j, s_n^c = J) \tag{7.4}$$

which is the joint probability of both the fine and coarse observations up to time t, the fine state s^f at time t being j and the coarse state s^c being J. The fine rate features which are indexed $t = 1, 2, 3, \ldots$ and the coarse features indexed $n = 1, 2, 3, \ldots$ are synchronous so that $n = 1$ aligns with $t = 1$ and $n = 2$ aligns with $t = M + 1$. In general, the two data streams synchronize at $t = (n-1)M + 1$. We use lower case to denote the fine state and upper case to denote the coarse. The joint probability $\alpha_t(j, J)$ can be updated recursively; however, the update is different depending upon whether or not the value of t corresponds to integer multiples of the coarse rate index n. At these times t, the coarse state is updated and a coarse feature is generated. At all other times, only the fine rate changes state and emits an observation. We develop the forward algorithm in detail for the more complex case of changing t and n and then present the simplified expression which is valid when only the fine-rate state is changing. When $t = (n-1)M + 1, n = 1, 2, 3, \ldots$

$$\begin{aligned} \alpha_t(j, J) &= \textstyle\sum_i \sum_I \; P(y_1^f, \ldots, y_t^f, y_1^c, \ldots, y_n^c, s_{t-1}^f = i, s_{n-1}^c = I, s_t^f = j, s_n^c = J) \qquad (7.5) \\ &= \textstyle\sum_i \sum_I \; P(y_1^f, \ldots, y_{t-1}^f, y_1^c, \ldots, y_{n-1}^c, s_{t-1}^f = i, s_{n-1}^c = I) \bullet \\ &\qquad P(s_t^f = j, s_n^c = J | y_1^f, \ldots, y_{t-1}^f, y_1^c, \ldots, y_{n-1}^c, s_{t-1}^f = i, s_{n-1}^c = I) \bullet \\ &\qquad P(y_t^f, y_n^c | y_1^f, \ldots, y_{t-1}^f, y_1^c, \ldots, y_{n-1}^c, s_{t-1}^f = i, s_{n-1}^c = I, s_t^f = j, s_n^c = J) \end{aligned}$$

(7.6)

The first term in equation 7.6 is simply $\alpha_{t-1}(i,I)$. The second term describes transition probabilities. Using the Markov state assumption that the next state in both the coarse and fine HMMs is independent of the observation sequence, we can simplify the second term in equation 7.6

$$P(s_t^f = j, s_n^c = J | y_1^f, \ldots, y_{t-1}^f, y_1^c, \ldots, y_{n-1}^c, s_{t-1}^f = i, s_{n-1}^c = I) = P(s_t^f = j, s_n^c = J | s_{t-1}^f = i, s_{n-1}^c = I). \quad (7.7)$$

In our state-coupled implementation, the present state of the coarse HMM is only dependent upon its previous state and may be expressed as a_{IJ}^c. The fine state is conditioned both on the previous fine state and upon the present coarse state. We will represent the fine state transition probability as $a_{ij}^f(J)$. Therefore, equation 7.7 is simply $a_{ij}^f(J)a_{IJ}^c$.

If we assume that both the fine and coarse observations are conditionally independent, given the fine and coarse states respectively, the third term in equation 7.6 becomes

$$P(y_t^f, y_n^c | y_1^f, \ldots, y_{t-1}^f, y_1^c, \ldots, y_{n-1}^c, s_{t-1}^f = i, s_{n-1}^c = I, s_t^f = j, s_n^c = J) = P(y_t^f | s_t^f = j) P(y_n^c | s_n^c = J) = b_j^f(y_t^f) b_J^c(y_n^c). \quad (7.8)$$

Using these terms in equation 7.6 and adding the case where there is no coarse state change gives

$$\alpha_t(j,J) = \begin{cases} \sum_i \sum_I \alpha_{t-1}(i,I) a_{ij}^f(J) a_{IJ}^c b_j^f(y_t^f) b_J^c(y_n^c) & \text{for } t = (n-1)M+1, n = 1,2,3,\ldots \\ \sum_i \alpha_{t-1}(i,J) a_{ij}^f(J) b_j^f(y_t^f) & \text{for all other } t \end{cases}$$

The final terms required in the determination of the forward algorithm for the multi-rate HMM are $P(s_1^c = J) = \pi_J^c$ and $P(s_1^f = j | s_1^c = J) = \pi_j^f(J)$. Using these terms the initial value of α is

$$\alpha_1(j,J) = \pi_j^f(J) \pi_J^c b_j^f(y_1^f) b_J^c(y_1^c) \quad (7.9)$$

7.4 Multi-Rate Parameter Estimation for State-Coupled Models

Just as in the single-rate classifier, the fine-rate output distribution $b_j^f(y_t^f)$ and coarse-rate output distribution $b_J^c(y_n^c)$ may be described by a single Gaussian or by a mixture of multiple

Gaussians. The model parameters listed below must be determined during model training in order to implement the decoding described in section 7.3.

μ_J^c	The mean of the coarse-rate output distributions
Σ_J^c	The covariance of the coarse-rate output distributions
μ_j^f	The mean of the fine-rate output distributions
Σ_j^f	The covariance of the fine-rate output distributions
π_J^c	The coarse-rate initial state distribution
$\pi_j^f(J)$	The fine-rate initial state distribution conditioned on the coarse state
a_{IJ}^c	The coarse-rate state transition probability
$a_{ij}^f(J)$	The fine-rate state transition probability conditioned on the coarse state

The desired model parameters θ, are those which maximize the probability of the state sequence and the observations, $\log P(S,Y|\theta)$. Since the actual state sequence for each pass of training data is hidden, we must deal with all possible sequences and maximize $E[\log P(S,Y|\theta)]$. This is an iterative process using the Expectation-Maximization (EM) algorithm. During the E-step we compute $Q(\theta|\theta^{(p)}) = E[\log P(S,Y|\theta)|Y,\theta^{(p)}]$ where $\theta^{(p)}$ is the estimate of the model parameters at the p^{th} iteration of the EM algorithm. In the M-step we update the model parameters, $\theta^{(p+1)} = argmax_\theta Q(\theta|\theta^{(p)})$.

The calculations for the E-step and M-step require $\alpha_t(j,J)$ introduced in the forward algorithm developed in section 7.3, and three additional terms, $\beta_t(j,J)$, $\gamma_t(j,J)$ and $\xi_t(j,J;k,K)$. We use the backward algorithm to determine $\beta_t(j,J)$, the probability of the observation sequence from $t+1$ to the end $t=T$ given that $S_t = j,J$. This quantity is also calculated recursively initializing $\beta_T(j,J) = 1\ \forall j,J$; which is just the probability of a sequence finishing. The steps used in the determination of $\beta(j,J)$ depended upon whether or not $t=(n-1)M+1, n=1,2,3,...$, just as we saw with $\alpha(j,J)$.

$$\beta_t(j,J) = \begin{cases} \sum_i \sum_I \beta_{t+1}(i,I) a_{ji}^f(I) a_{JI}^c b_i^f(y_{t+1}^f) b_I^c(y_{n+1}^c) & \text{for } t=(n-1)M+1, n=1,2,3,... \\ \sum_i \beta_{t+1}(i,J) a_{ji}^f(J) b_i^f(y_{t+1}^f) & \text{for all other } t \end{cases}$$

The state occupancy, $\gamma_t(j,J)$, which describes the probability of the system being in a particular state (j,J) at time t given the entire sequence of observations, is calculated using

both the forward and backward results:

$$\begin{aligned}\gamma_t(j,J) &= \frac{P(y_1^f,\ldots,y_T^f,y_1^c,\ldots,y_N^c,s_t^f=j,s_n^c=J)}{P(y_1^f,\ldots,y_T^f,y_1^c,\ldots,y_N^c)} \\ &= \frac{\alpha_t(j,J)\beta_t(j,J)}{\sum_i\sum_I \alpha_T(i,I)}.\end{aligned} \tag{7.10}$$

Finally, we need the probability of seeing a particular state transition at time t:

$$\xi_t(j,J;k,K) = P(s_{t-1}^f=j,s_{n-1}^c=J,s_t^f=k,s_n^c=K|Y_f^T,Y_c^N,\theta^{(p)}) \tag{7.11}$$

where Y_f^T is the fine rate sequence of length T and Y_c^N is the coarse rate sequence of length N. For the two cases of t described for previous terms we can write this as:

$$\xi_t(j,J;k,K) = \begin{cases} \frac{\alpha_{t-1}(j,J)a_{jk}^f(K)a_{JK}^c b_k^f(y_t^f)b_K^c(y_n^c)\beta_t(k,K)}{\sum_i\sum_I \alpha_T(i,I)} & \text{for } t=(n-1)M+1, n=1,2,3,\ldots \\ \frac{\alpha_{t-1}(j,K)a_{jk}^f(K)b_k^f(y_t^f)\beta_t(k,K)}{\sum_i\sum_I \alpha_T(i,I)} & \text{for all other } t \end{cases}$$

7.4.1 The E-step

The observations Y used in parameter estimation consist of multiple (r) sequences of fine and coarse-rate features drawn from the cutting passes in the training data set. To calculate $Q(\theta|\theta^{(p)})$, we assume that the training sequences are independent, and calculate $E[\log P(S^i,Y^i|\theta)|Y^i,\theta^{(p)}]$ where S^i and Y^i are the state sequence and observation sequence for a single training pass i. We then use these individual results to determine the update for this iteration:

$$Q(\theta|\theta^{(p)}) = \sum_{i=1}^{r} E[\log P(S^i,Y^i|\theta)|Y^i,\theta^{(p)}] \tag{7.12}$$

$$\begin{aligned} &= \sum_{i=1}^{r}\sum_j\sum_J \gamma_1^i(j,J)(\log(\pi_j^f(J)) + \log(\pi_J^c)) \\ &+ \sum_{i=1}^{r}\sum_{t=2}^{T}\sum_j\sum_J\sum_k\sum_K \xi_t^i(j,J;k,K)(\log a_{jk}^f(K)) \\ &+ \sum_{i=1}^{r}\sum_{t'=2}\sum_J\sum_K \xi_{t'}^i(j,J;k,K)(\log a_{JK}^c) \\ &+ \sum_{i=1}^{r}\sum_{t=1}^{T}\sum_j\sum_J \gamma_t^i(j,J)(\log b_j^f(y_t^f)) \\ &+ \sum_{i=1}^{r}\sum_{t'=1}\sum_J \gamma_{t'}^i(j,J)(\log b_J^c(y_n^c)) \end{aligned} \tag{7.13}$$

where t' indicates those values of t when there is a change in the coarse state, $t' = (n-1)M+1$ for $n = 1, 2, \ldots$. S^i is the state sequence corresponding to the *i-th* cutting pass and Y^i are the observations from this pass. The probability of state occupancy, $\gamma_t^i(j, J)$ and the state transition probability, $\xi_t^i(j, J; k, K)$ now have the superscript i to indicate that they are conditioned only on the observations from the *i-th* pass.

7.4.2 The M-step

Looking at equation 7.13 we see the three primary model parameters: the initial state probabilities ($\pi_j^f(J)$, π_J^c), transition probabilities ($a_{jk}^f(K)$, a_{JK}^c) and output distributions ($b_j^f(y_t^f)$, $b_J^c(y_n^c)$) in separate summations. We can therefore treat the maximization of each case separately.

The ML estimates of the initial state probabilities are found by imposing the constraints that $\sum_j \pi_j^f(J) = 1$ and $\sum_J \pi_J^c = 1$, taking the derivative of the Q function and setting it equal to zero. Solving these equations we get

$$\hat{\pi}_k^f(K) = \frac{\sum_{i=1}^r \gamma_1^i(k, K)}{\sum_{i=1}^r \sum_j \gamma_1^i(j, K)} \tag{7.14}$$

and

$$\hat{\pi}_K^c = \frac{\sum_{i=1}^r \sum_k \gamma_1^i(k, K)}{\sum_{i=1}^r \sum_j \sum_J \gamma_1^i(j, J)} \tag{7.15}$$

Prior to taking the derivative of the Q function with respect to the transition probabilities, we impose the constraints that $\sum_k a_{jk}^f(K) = 1 \forall j$ and that $\sum_K a_{JK}^c = 1 \forall J$. Solving we get

$$\hat{a}_{JM}^c = \frac{\sum_{i=1}^r \sum_{t'=2} \sum_j \sum_m \xi_{t'}^i(j, J; m, M)}{\sum_{i=1}^r \sum_{t'=2} \sum_j \sum_m \sum_K \xi_{t'}^i(j, J; m, K)} \tag{7.16}$$

where t' indicates those values of t when there may be a change in the coarse state; and

$$\hat{a}_{jm}^f(M) = \frac{\sum_{i=1}^r \sum_{t=2}^{L_i} \sum_J \xi_t^i(j, J; m, M)}{\sum_{i=1}^r \sum_{t=2}^{L_i} \sum_J \sum_k \xi_t^i(j, J; k, M)} \tag{7.17}$$

where L_i is the length of the fine observation sequence for the *ith* cutting pass. For time $t \neq t'$ where there is no change in the coarse state, $\xi_t^i(j, M; m, M)$ is used in place of $\xi_t^i(j, J; m, M)$.

For our case of Gaussian output distributions we determine the values for $\mu^{(p+1)}$ to be

$$\hat{\mu}_j^c = \frac{\sum_{i=1}^{r} \sum_{t'=1} \sum_j \gamma_{t'}^i(j,J) y_{t'}^{c_i}}{\sum_{i=1}^{r} \sum_{t'=1} \sum_j \gamma_{t'}^i(j,J)} \tag{7.18}$$

and

$$\hat{\mu}_j^f = \frac{\sum_{i=1}^{r} \sum_{t=1}^{L_i} \sum_J \gamma_t^i(j,J) y_t^{f_i}}{\sum_{i=1}^{r} \sum_{t=1}^{L_i} \sum_J \gamma_t^i(j,J)} \tag{7.19}$$

Finally, the estimates of the covariances for the coarse and fine states are determined using the latest estimate of $\hat{\mu}_j^f$ and $\hat{\mu}_j^c$

$$\hat{\Sigma}_j^c = \frac{\sum_{i=1}^{r} \sum_{t'=1} \sum_j \gamma_{t'}^i(j,J)(y_{t'}^{c_i} - \hat{\mu}_j^c)(y_{t'}^{c_i} - \hat{\mu}_j^c)^T}{\sum_{i=1}^{r} \sum_{t'=1} \sum_j \gamma_{t'}^i(j,J)} \tag{7.20}$$

and

$$\hat{\Sigma}_j^f = \frac{\sum_{i=1}^{r} \sum_{t=1}^{L_i} \sum_J \gamma_t^i(j,J)(y_t^{f_i} - \hat{\mu}_j^f)(y_t^{f_i} - \hat{\mu}_j^f)^T}{\sum_{i=1}^{r} \sum_{t=1}^{L_i} \sum_J \gamma_t^i(j,J)} \tag{7.21}$$

7.5 Multi-Rate Model Initialization

Accurate classification requires wear level models able to discriminate between the quantized wear levels W_i. Information about the wear level is contained in the feature vectors capturing transient activity, in the relationship between feature vectors from different portions of a single cutting pass and in the relationship between feature vectors drawn from cutting passes at different wear levels. In our single-rate classifier we model the dynamics of the feature vectors within a cutting pass and across wear levels. In our multi-rate HMM we expand the model to capture the information contained in the changing rate of transients.

When training our single-rate models, we make the assumption that all feature vectors are descriptive of the wear level and/or the stage in a cutting pass being modeled and all are used in training. In our multi-rate models, we seek to discriminate between different wear states, changes in the stage of a cutting pass (which may include the difference between noisy and quiet cutting) and between times when transients are occurring and when they are absent. When training the coarse-rate models of our MHMM, we again assume that *all* of the feature vectors are descriptive of the phenomena we wish to model. Each coarse-rate feature vector is considered to be equally important in our discrimination between wear levels, stages of a pass and noisy/quiet cutting. Therefore, all feature vectors have an equal

contribution to the model parameters. Initialization is carried out as has already been described for a single rate classifier. We will refer to this as **Impartial Initialization.**

When training the fine-rate HMMs in the MHMM our hypothesis is that the feature vectors descriptive of the transient behavior we wish to model comprise only a small portion of the complete set of fine-rate features. In their work analyzing the transient behavior of milling, Gillespie and Atlas [17] pointed out that the majority of their ambiguity plane feature vectors were not indicative of wear. They concluded that measuring wear would require the detection of "exceptional" feature vectors. To explore this further, they clustered their feature vectors using a VQ codebook which minimized a mean square error distortion criterion. A single code word accounted for more than 90% of the feature vectors regardless of the level of wear on the cutter. The frequency of occurrence of the remaining codewords was related to the level of tool-wear; further supporting the supposition that the "exceptional" feature vectors were the ones important in classification. Work done with discriminative training of HMMs is beyond the scope of this dissertation. In the absence of labeled transient data, we rely on an ad hoc initialization of the transient state output distributions $b_j^f(y^f)$ with only "exceptional" feature vectors to capture the desired transient behavior. We refer to this as **Scarce Initialization.**

Once both the coarse and fine-rate HMMs have been initialized, the output distributions are re-estimated and the coupling between the coarse state and the fine transition probabilities is learned using five iterations of the constrained EM described in section 5.2. We investigate two approaches to scarce initialization of the fine-rate HMM output distributions and compare their performance to models trained using impartial initialization.

7.5.1 Scarce initialization of fine-rate models

One of the five states in the fine-rate HMM is intended to capture the "normal cutting" behavior at each wear level. When the same HMM is shared across all wear levels, this single state contains multiple mixtures, one for each wear level. When all of the fine-rate states are wear-level dependent, this state contains a single Gaussian representative of "normal cutting" at that wear level. To find the initial output distribution parameters for this state

we train a single-state, single-mixture HMM for each wear level W_i using the labeled training data. For this one state, training uses impartial initialization.

The remaining four fine rate HMM states are intended to capture transient behavior. We will refer to the first implementation of scarce initialization for these four states as the **distant mixture** method. All of the training data regardless of its wear label is used to train a single state, nine mixture HMM. These mixtures are compared to the wear-level dependent "normal cutting" models. The four transient states are initialized with the $b_j^f(y^f)$ parameters of the four mixtures which have the largest distance from the "normal cutting" models. The distance metric used is:

$$r^2 = \frac{1}{2W}\sum_{i=1}^{W}(\mu_t - \mu_i)^t\Sigma_i^{-1}(\mu_t - \mu_i) + (\mu_i - \mu_t)^t\Sigma_t^{-1}(\mu_i - \mu_t) \tag{7.22}$$

where W is the number of wear-dependent mixtures, μ_t and Σ_t are the mean and variance of the mixture being evaluated as a potential transient model and μ_i and Σ_i are the mean and variance of the i^{th} wear-dependent mixture model trained with only the labeled training data.

We will refer to the second implementation of scarce initialization of the four transient states of the fine-rate HMM as the **outlier clustering** method. Here all of the training data, regardless of wear label, is combined into a single cluster. Divisive clustering is used to partition the data into K clusters. As the value of K increases, the average number of features in each cluster and the average distance from each feature vector to its nearest cluster both decrease. If our intent was to find the four clusters which best represented all of the feature vectors, we would set $K = 4$ and use the cluster parameters to initialize our fine-rate models. However, our intent is to find the four clusters which best cover the feature space. In fact, we are looking for clusters other than those which contain the majority of the fine-rate feature vectors. Once K clusters have been defined, agglomerative clustering is used to reduce the number of clusters. The steps in agglomeration are as follows:

1. Calculate all pair-wise combinations of the Euclidean distance between the means of the K clusters.

2. Combine the feature vectors assigned to the two "closest" clusters and recalculate the

parameters of the new cluster.

3. Do NOT reassign feature vectors in other clusters even if the newly defined cluster is now "closer" than its original cluster.

4. Repeat until the desired number of clusters is reached.

Clusters combine with those which are close in feature space regardless of the number of members in the cluster. The final set of clusters covers the extent of the feature space rather than attempting to include the majority of the features.

In our work, we continue divisive clustering up to 50 clusters and agglomerate down to 9; just as we choose 9 mixtures for the distant mixture method. Using this combination of divisive to 50, agglomerate down to 9, we are able to satisfy the additional constraint imposed on our final outlier clusters that they each contain at least 1% of the total number of feature vectors in the training data. Preliminary divisive clustering allowed up to 1000 clusters. After agglomeration we were left with several clusters containing less than 0.05% of the feature vectors. We considered these to be too "exceptional" to be of use.

Once the nine clusters are defined, we apply the same metric used in distant mixture to choose the parameters of the four clusters which have the largest distance from the "normal cutting" models. Figure 7.6 shows the initial fine-rate model means for the four transient states and the five wear level mixture means of the "normal cutting" state derived from the three approaches to initialization. The outlier initialized states cover the greatest extent of the feature space. The states initialized by distant mixture are different from those from impartial initialization but it is difficult to draw any conclusions from the difference.

If our assumption that the energy or frequency content of the transients is not wear-level dependent is correct, we would expect to see the models for the transient states stay close together for all wear levels after the EM parameter re-estimation step. Figure 7.7 shows the fine rate models after five iterations of EM. The four initial transient states and the three wear-level dependent models for the least worn state "A" and the two most worn states "D&E" are shown as filled circles. After parameter re-estimation, the five fine-rate states are again shown for wear labels "A","D" and "E". In some cases we do see pairs of transient

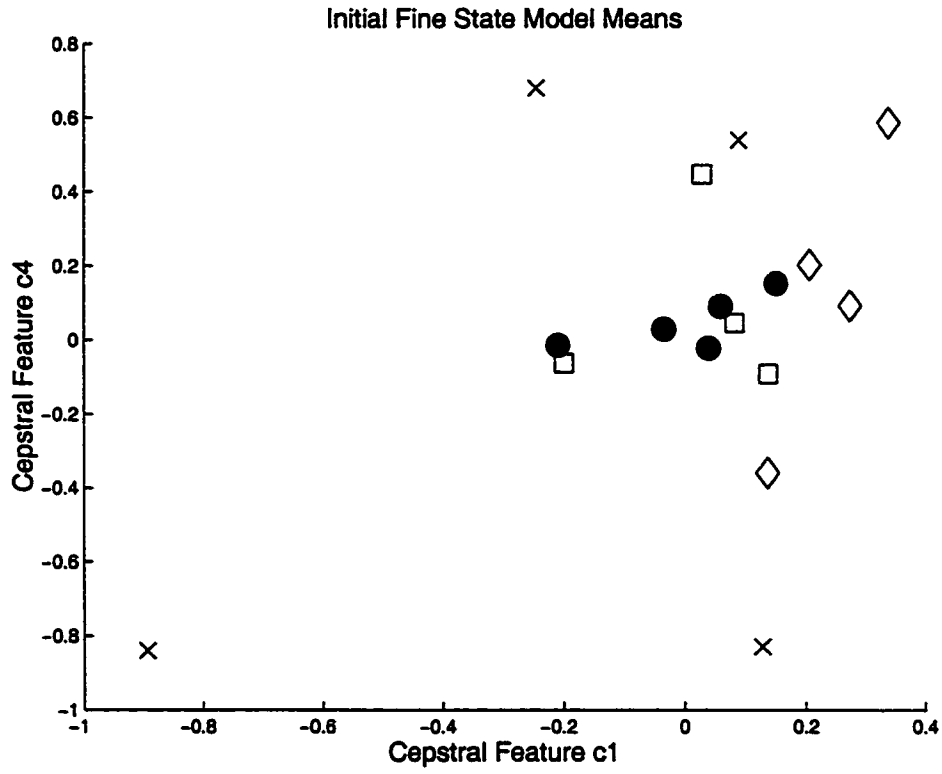

Figure 7.6: *Initial model means for two dimensions of the fine-rate transient states. The three approaches to model initialization are included: impartial initialization (square), "distant mixture" (diamond) and "outlier clustering" (x). The filled circles indicate the means for the five wear-dependent states.*

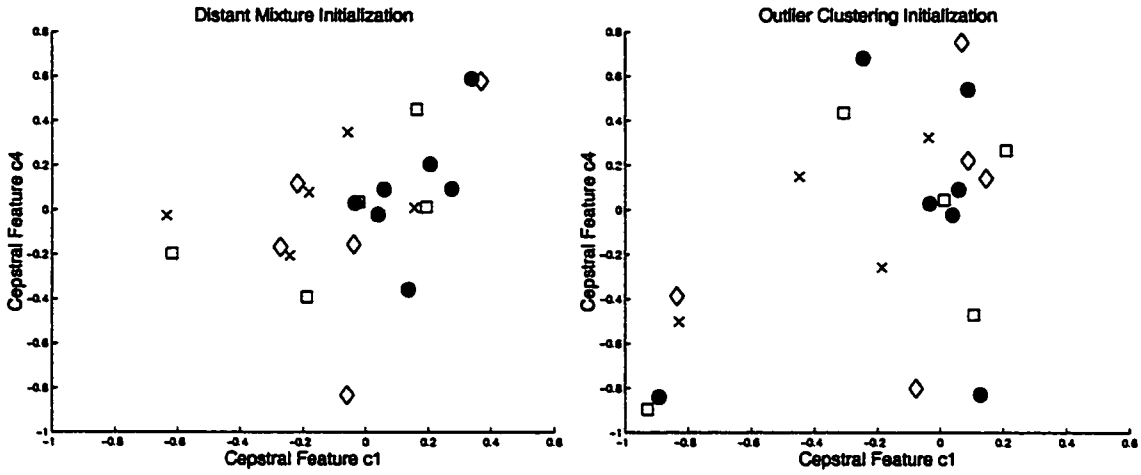

Figure 7.7: *Fine-rate transient states after re-estimation of parameters using all training data. The plot on the left shows the models initialized with "distant mixture". The plot on the right shows models initialized with "outlier clustering". The shaded circles indicate the initial model means for the four transient states and the three wear-level dependent states corresponding to wear levels A,D,E. The five states after parameter re-estimation are shown for wear level A (square), D(x) and E(diamond).*

states representing different wear levels. However, the clustering of all wear levels is not apparent. This indicates that the energy or frequency content of transients may indeed be wear-level dependent. Regardless of the initialization technique, the models move to better cover the feature space. In many cases, a particular region of the feature space is occupied by models from each of the wear levels. We cannot conclude from the plots that one region of the feature space is more indicative of a particular wear level than another. However, not allowing any change in the output distributions from their initialized values is expected to degrade performance.

7.6 Multi-Rate Model Experiments

In all of the results reported in this chapter, accuracy performance is based on the binary mapping of the wear labels W_i and the NCE score is calculated using the $P(worn)$ output of the GLM in the last stage of our system. All multi-rate development experiments are on the Series-B titanium data set since this exhibits the noisy/quiet cutting which we expect

will benefit from the multi-rate model. The Series-B data uses only a CV test set. The cross validation makes the accuracy performance reported from the wear labels W_i a "fair" test since the evaluated passes are not included in the model training. However, the regression coefficients for the $P(worn)$ GLM are determined using all of the CV cutters. As such, the $P(worn)$ GLM used for the NCE scores is evaluating the same cutters used in its training. While this is still an acceptable means to compare the performance of competing classifier topologies and initialization techniques, performance is expected to degrade when these systems are applied to held out test cutters. In our multi-rate experiments we review the following questions:

1. Is it better to use a single-rate or multi-rate architecture?

2. Is multi-rate performance better when the two parallel HMMs are treated independently and coupled by a GLM (loosely coupled) or by using the state-coupled model directly in classification?

3. Which approach to initialization of fine-rate models has superior performance?

4. Are the wear-level-dependent changes in transient behavior limited to rate or are the energy and frequency content also wear level dependent?

In our work with cutting steel, single-rate classifiers processing *fine-rate* data outperformed those using *coarse-rate* features. Changing the workpiece material to titanium, we see that the single-rate classifiers trained and tested under the cross validation paradigm defined for our Series-B tests show superior performance using *coarse-rate* features (table 7.1). For the single-rate classifiers, only the accuracy of the coarse-rate features are statistically better than chance at a 90% confidence level. The NCE score for the fine-rate classifier indicates that it has over-confidence problems.

When these two classifiers, operating at different data rates, are combined with a $P(worn)$ GLM in our loosely coupled MHMM architecture, the accuracy drops below the level required to claim a significant difference from chance at the 90% confidence level based on error counting. It should be noted that the difference between the accuracy of 91% and

Table 7.1: *Performance of single-rate and multi-rate classifiers applied to the titanium data in the Series-B experiments. Next to each accuracy score is listed the P-value for the hypothesis that performance is better than chance (1-statistical significance confidence level). The difference between the accuracy score of the various classifiers is not statistically significant. NCE performance is determined with a GLM which is trained on the same CV data set.*

Classifier	% Accuracy	P	NCE
Chance	85	-	-
Single-rate Fine	91	0.25	+0.00
Single-rate Coarse	94	0.09	+0.11
Multi-rate Loosely Coupled	91	0.25	+0.12
Multi-rate State-Coupled	94	0.09	+0.31

Table 7.2: *The P-value for the hypothesis that the wear confidence estimate performance of one classifier is different than another (1-statistical significance confidence level). The confidence performance (NCE) being compared is listed in table 7.1. SR Fine = Single-rate Fine, SR Coarse = Single-rate Coarse, MR Loosely = Multi=rate Loosely Coupled and MR State = Multi-rate State-Coupled. Results which are not statistically significant at a confidence level of at least 85% are indicated by N.S.*

Confidence Differences				
Classifier	SR Fine	SR Coarse	MR Loosely	MR State
Single-rate Fine	-			
Single-rate Coarse	< 0.15	-		
Multi-rate Loosely Coupled	0.10	N.S.	-	
Multi-rate State-Coupled	0.0005	0.001	0.001	-

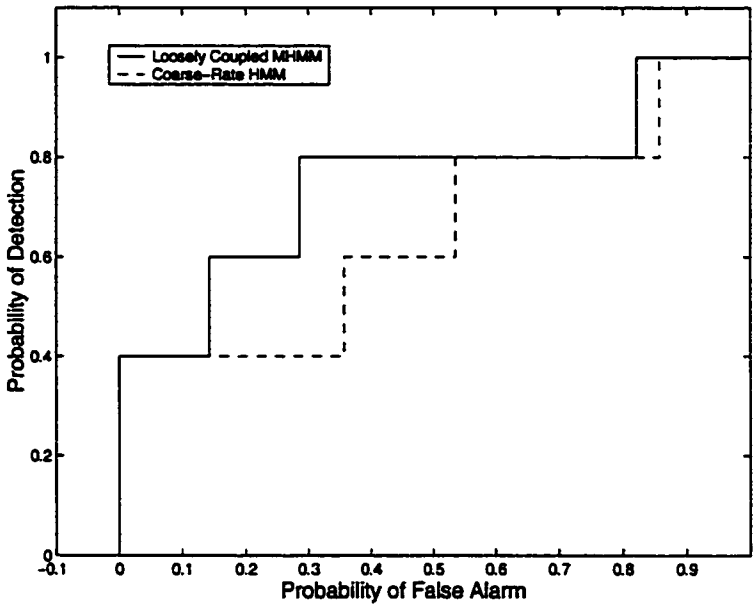

Figure 7.8: *Performance of a loosely coupled MHMM compared to a single-rate HMM processing coarse-rate features.*

94% is one additional missed detection. Looking at the ROC in figure 7.8 we see that over a range of operating points, the loosely coupled multi-rate classifier does give improved performance over the best single-rate HMM. When the feature streams at the two data rates are used in our state-coupled multi-rate architecture, the accuracy remains at the higher coarse-rate level and there is a noticeable increase in NCE. Using the approach to evaluate the statistical significance of the wear confidence estimate described in section 3.5 we can say with confidence at the 99.9% level that the state-coupled architecture has better performance than either the single-rate coarse HMM or the loosely coupled MHMM. Here the statistical significance is based on a Gaussian model of the differences between confidence predictions. The P-values for the hypothesis that the wear confidence estimate of one classifier is different from another with statistical significance are shown in table 7.2.

The state-coupled architecture apparently does a better job of using the information in the fine-rate data than the loosely coupled MHMM (figure 7.9). In our choice of a multiplier on the wear level transition probabilities discussed in chapter 4, we opted to minimize the

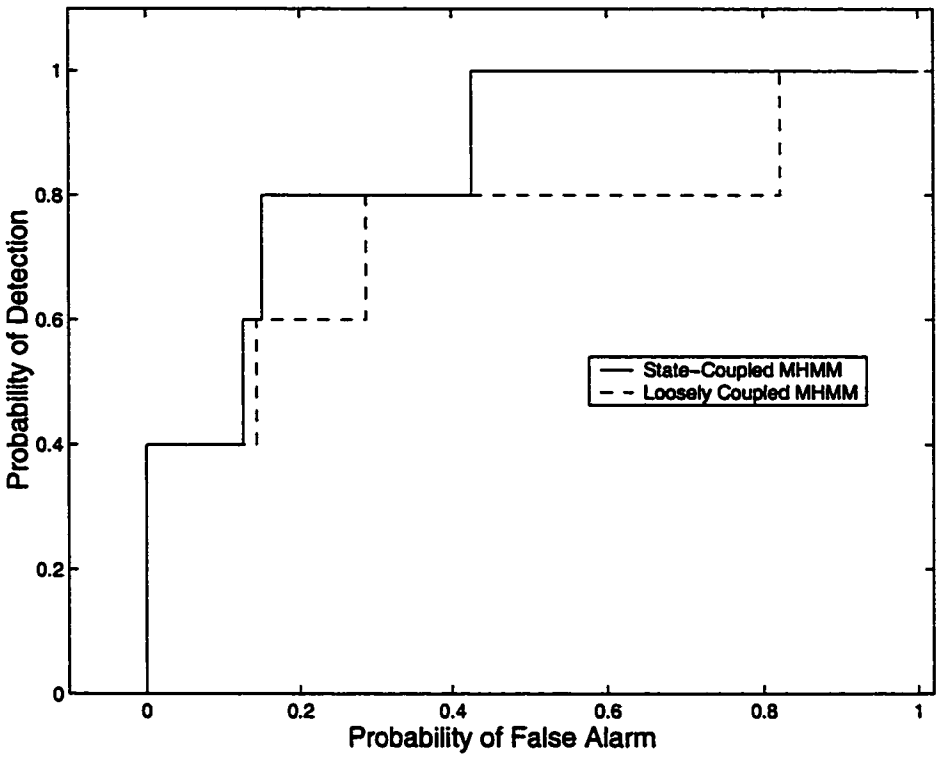

Figure 7.9: *ROC comparing the performance of two MHMMs. One uses a loosely coupled topology and the other the state-coupled.*

number of missed detections without any increase in the number of false alarms. Looking at figure 7.10 we see that setting the $P(worn)$ threshold to 0.5 results in two errors (two missed detections) for the state-coupled topology. Lowering the threshold to 0.3 lowers the accuracy since there are three errors (one missed detection and two false alarms). However, it may be more important to increase the WORN detection from 60% to 80% even at the cost of the two false alarms. Attempting to increase the WORN detection to the same level for the loosely coupled MHMM requires us to drop the $P(worn)$ threshold below 0.2 resulting in an unacceptably high twenty false alarms. The state-coupled MHMM does a much better job of separating the WORN passes from those which are NOT WORN.

In section 7.5 we discuss strategies for initializing our fine-rate models so that they will capture the transient behavior which we wish to model. As shown in table 7.3, when

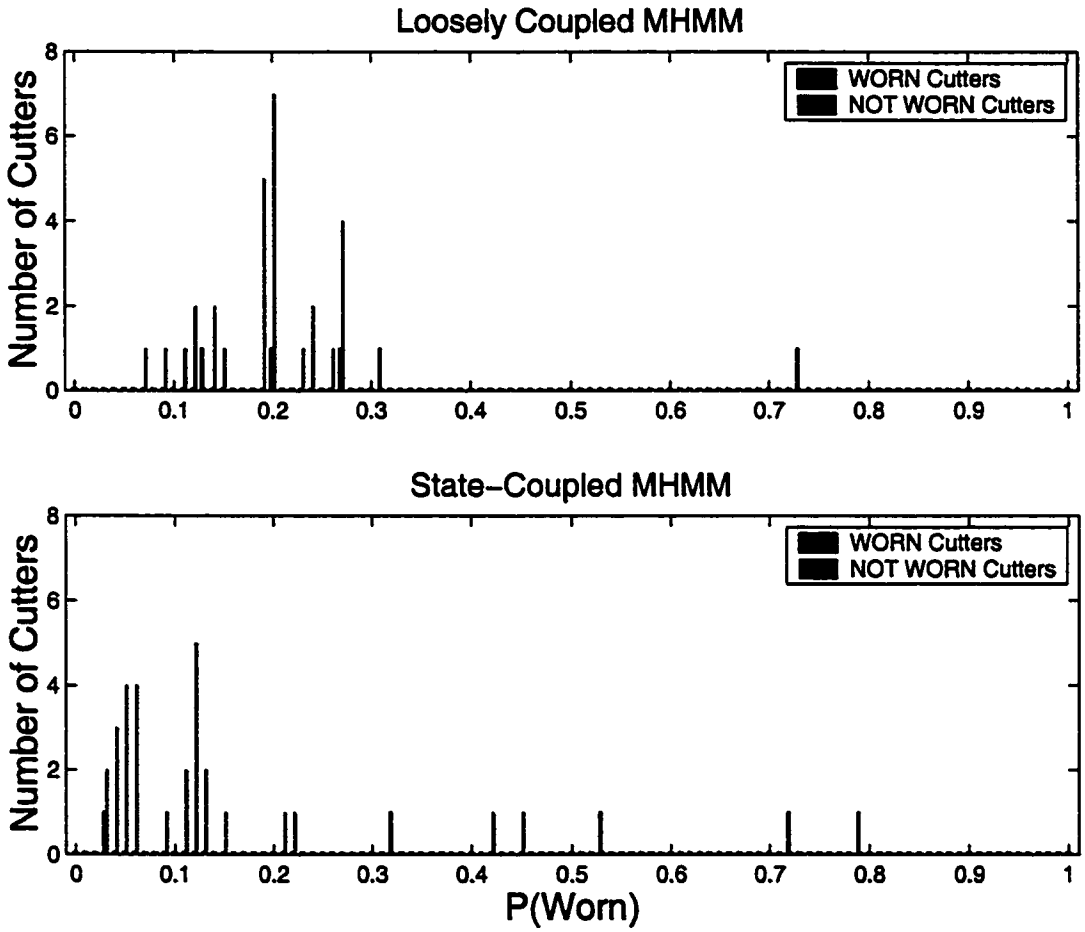

Figure 7.10: *The number of Series-B test cutters at different levels of $P(worn|x_i)$. The top plot shows the performance of an MHMM coupled via a second stage GLM (loosely coupled). The bottom plot indicates the performance of a state-coupled MHMM.*

Table 7.3: *Performance of state-coupled multi-rate classifiers using three different approaches for initialization of the fine-rate models.*

Initialization	% Accuracy	NCE
Chance	85	-
Impartial	91	+0.06
Distant Mixture	94	+0.31
Outlier Clustering	91	-0.01

the fine-rate models are initialized with all fine-rate data treated equally, performance of our state-coupled MHMM is actually worse than seen in the loosely coupled architecture which uses the same "impartial" initialization. Both distant mixture and outlier clustering initialization attempt to use only "exceptional" features for the fine-rate models. As shown in figure 7.6, outlier clustering tends to select cluster means which are more exceptional than those used in distant mixture initialization. Accuracy performance is similar for the two techniques but the NCE metric shows, with confidence at the 99% level, that distant mixture initialization is superior to both impartial and outlier clustering initialization (table 7.3).

The histograms in figure 7.11 show that models initialized with distant mixture assign lower $P(worn)$ to cutting passes which are NOT WORN than is typical with impartial initialization models. This greater separation between the WORN and NOT WORN passes is reflected in the higher NCE score in table 7.3. The superior performance of the distant mixture initialization is also see in the ROC of figure 7.12.

Once fine-rate models are initialized, five iterations of EM training with all training data are used to update model parameters. In all cases, it is at this stage of training that the coupling between the coarse state and the fine-rate transition probabilities is estimated. If only the rate of transients are wear-level dependent, the wear-dependent transition probabilities should be sufficient to model transient activity. If the energy and frequency content of the transients are also wear level dependent, we need to allow the output distributions, initialized with the same μ and Σ, to change during training. Figure 7.7 shows the fine-rate

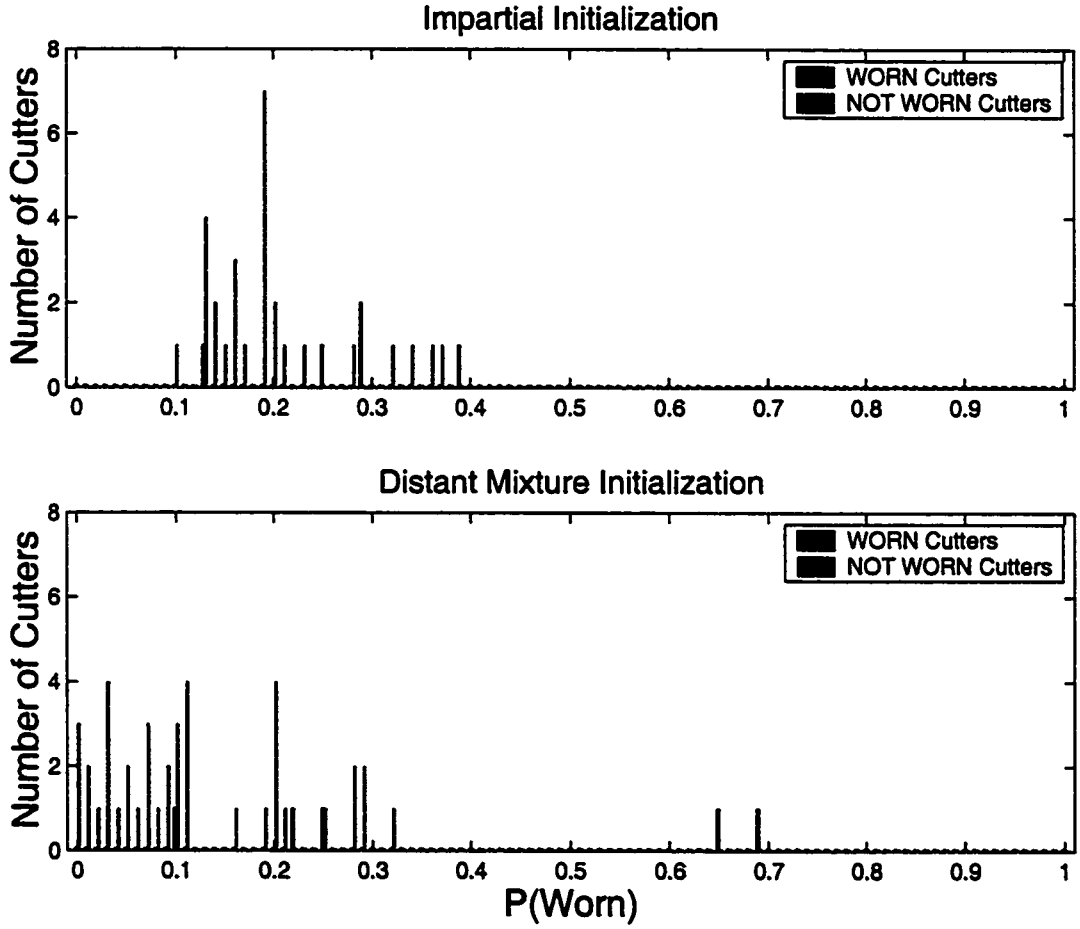

Figure 7.11: *The number of Series-B test cutters at different levels of $P(worn|x_i)$. The top plot shows the performance of an MHMM whose fine-rate models are initialized using impartial initialization. The plot on the bottom shows the performance when initialization uses distant mixture.*

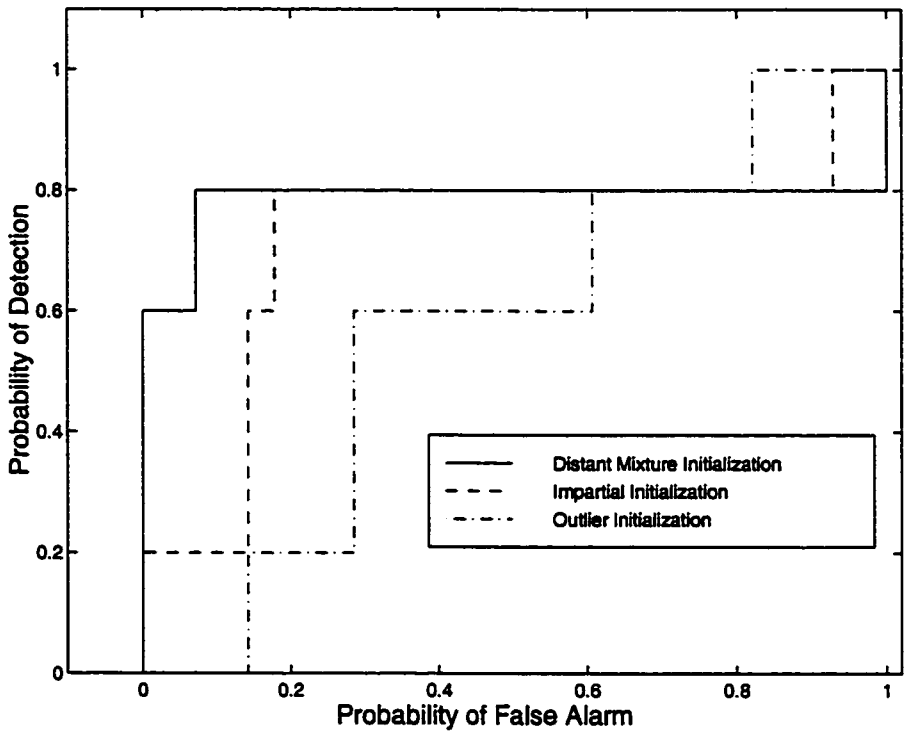

Figure 7.12: *Performance of a state-coupled MHMM whose fine-rate models used each of the three initialization options.*

Table 7.4: *Performance of coupled multi-rate classifiers using three different approaches for initialization of the fine-rate models.*

	$b_i(t)$ Wear-Level Dependent		$b_i(t)$ Shared Across Wear Levels	
Initialization	% Accuracy	NCE	% Accuracy	NCE
Distant Mixture	94	+0.31	94	+0.09
Outlier Clustering	91	-0.01	82	+0.03

model means at three wear levels after five iterations of EM. As noted earlier, because of the separation in the wear-dependent model means,we expect worse performance if the output distributions of our fine-rate models are tied across wear levels. Table 7.4 shows that the NCE performance of the distant mixture state-coupled MHMM exhibits a statistically significant degradation when the transients are required to share output distributions across all wear levels.

In our presentation of remaining life prediction in chapter 6 we report performance only on the cutters in our test sets. The test paradigm for the Series-B multi-rate testing only has cross validation cutters. We include a plot of remaining life for the multi-rate classifier results (figure 7.13) for information only. No MSE or MSE-End performance is reported because we are training and testing our remaining life GLM on the same data.

The limited data in our Series-B testing makes it impossible to state conclusions to our research questions with conviction using only the accuracy performance metric. However, with the histograms, ROCs and NCE scores we are able to conclude that a state-coupled MHMM using distant mixture initialization gives the best performance on the M-1" titanium data. We can also see that modeling transient behavior requires both transition probabilities and output distributions which are wear-level dependent.

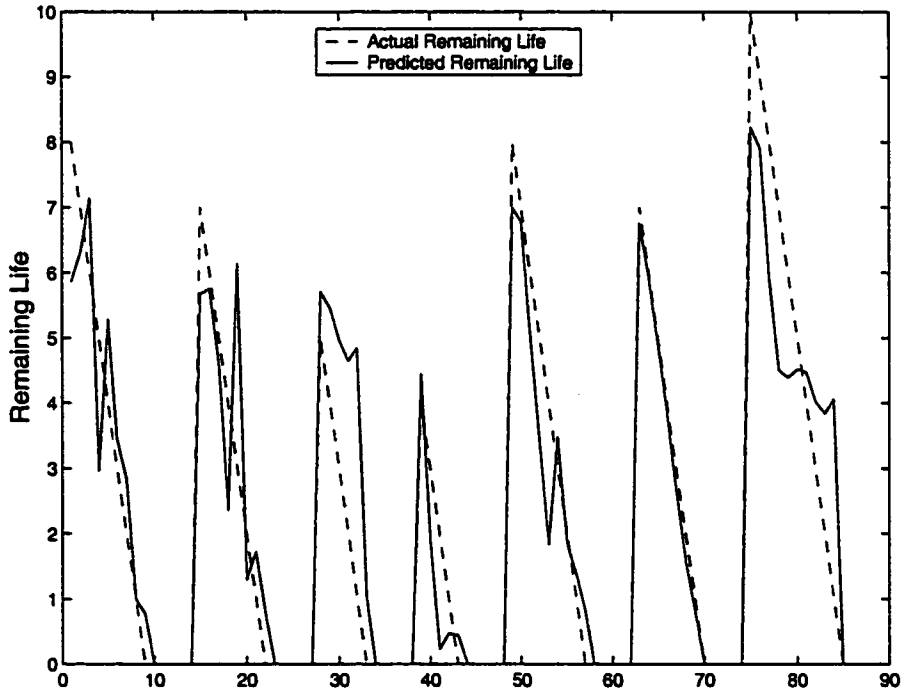

Figure 7.13: *Series-B titanium CV training cutting classified with the Multi-rate HMM. Actual remaining life vs. the remaining life predicted by our remaining life GLM.*

Chapter 8

SUMMARY AND FUTURE WORK

The premise behind the work in this dissertation is that the problem of classifying milling tool-wear can be viewed as a problem in speech recognition. We borrowed acoustic modeling techniques developed for the task of automatic speech recognition and adapted them to our machining application. Different cutters were treated as different speakers. Levels of wear were viewed as words in a limited vocabulary. Changing accelerometers were treated like changing microphones. The progression of wear was described by a finite-state "grammar" (left-to-right Markov process).

In making this connection we opened the door to a two-way flow of information. The work of this dissertation demonstrated the benefits of applying the powerful techniques developed for speech recognition to the tool-wear application. Having developed a "speech type" classifier for tool-wear, we have also set the stage for investigations targeted for speech systems to be first tried on the tool-wear application. Speech recognition system are complex enough and training corpora large enough that simple testing requires days of compute time. The similarity of the tool-wear application and the reduced complexity allows a quick look at new techniques with experiments that can be run in minutes rather than days. It is possible that insights learned in the fast turn around application may be useful.

In section 8.1 of this chapter we review the main contributions of our work and in section 8.2 we discuss some possible extensions.

8.1 Review of Main Contributions

Evaluation Paradigm At the beginning of our work, there was no standard data corpus, no agreed upon test methodology or common set of evaluation metrics available for work in milling tool-wear. In this thesis we have developed a test paradigm which was used here

Table 8.1: *Summary of single-rate HMM classifier performance for both steel and titanium data sets. The first column indicates the accuracy performance of the "chance" system using only the prior information from the training data.*

Test Set	Chance %	%
Steel 1/2" CV Test	89	96
Steel 1/2" Test	77	90
Steel 1" Test	89	94
Titanium Series-A CV Test	84	93
Titanium Series-A Test	81	86
Titanium Series-B CV Test	85	94

and by colleagues working on this application task. If approval is obtained to release the Boeing data to the public, we and our colleagues at Boeing will have established the first common test corpus for research in milling tool-wear.

A significant part of defining the evaluation paradigm was showing the benefits of using multiple performance metrics. The results of the experiments reported in chapters 6 and 7 make it clear that the typical accuracy metric used in the past is not sufficient to characterize the relative benefits of competing systems. The NCE metric, while it also has its limitations, allows greater insights during classifier development.

Dynamic single-rate classifier

During the development of our single-rate classifier we experimented with various topologies and training mechanisms. In the process, we identified good design principles for HMMs used in a tool-wear application.

Using our single-rate HMM classifier, we were able to demonstrate accuracy greater than 90% and good NCE performance on almost all of the steel and titanium test sets (table 8.1). The data in the Series-B tests includes the noisy/quiet cutting periods expected at the outset of our research to be a very challenging problem. The 94% accuracy achieved with our multi-rate classifier on this data set is particularly encouraging.

Table 8.2: *Performance of single-rate and multi-rate classifiers applied to the titanium data in the Series-B experiments. The performance of both a loosely coupled and state-coupled MHMM is reported.*

Test Set	% Accuracy	NCE
Single-rate Coarse	94	+0.11
Multi-rate Loosely Coupled	91	+0.12
Multi-rate State-Coupled	94	+0.31

In our work we used our classifier as a common test bed to compare frequency domain energy features, cepstral features, auto-ambiguity features and features motivated by human auditory testing. Those working on other feature sets for this application would benefit from an existing classifier for feature evaluation. By basing our system on the HTK and S-Plus software which has seen widespread application, it is possible for other researchers to easily use our design procedures.

Dynamic multi-rate classifier

During our work on this application we discovered that proper modeling of tool-wear requires that we process features at different data rates. We were able to demonstrate that performance improved when a multi-rate rather than a single-rate system was used (table 8.2). We also showed that a state-coupled multi-rate model gave better performance than one using a loosely coupled topology.

As with the single-rate classifier, our work identified design principles relating to the topology and training of a multi-rate classifier. Of particular interest was our work on model initialization. We showed that the initialization of the models intended to capture transient phenomena was important, and that the models of transients require output distributions which are wear-level dependent.

Multiple presentations of output information

The information provided by our system goes beyond that typically provided by a tool-wear classifier. The typical output of such a system is usually limited to binary WORN or

Table 8.3: *Reduction in the accuracy error on the steel data set when using unlabeled data in training rather than just those passes explicitly labeled in the training data set.*

Test Set	Only Known Labels	Constrained Training	Reduction in Error
Steel 1/2" CV Test	82%	96%	78%
Steel 1/2" Test	84%	90%	37%
Steel 1/2" CV Test	89%	94%	45%

NOT WORN labels. Some have extended this recently to indicate levels of wear but do not connect these levels with an actual range of wear on the cutter as we do with our quantized wear label W_i.

The confidence estimate $P(worn)$ was shown to give a finer grained resolution of tool-wear. Using the NCE evaluation mechanism with this additional output we are able to draw conclusions about the efficacy of some of our modeling approaches that would not have been possible with accuracy alone because of the small amount of data available.

The variability of the milling process makes it surprising that a prediction of future life would have any success. The demonstrated ability of our remaining life predictor to use a cutter's past behavior to update prediction of remaining life is very encouraging.

Practical challenges

Other tool-wear applications such as drilling and turning report on data sets with hundreds of examples for training and test. Until now, the cost associated with collecting labeled test data for a milling application has hampered the use of statistical inference systems. Our constrained EM training provides a way to use very sparsely labeled data during training with an increase in performance as shown in table 8.3.

It is not practically possible to gather the data necessary to train a classifier under all combinations of cutting conditions expected to be seen in practice. We were able to show that using feature normalization or cepstral mean subtraction, our single-rate classifier was able to properly classify cutters being used under cutting conditions not seen during training. We were also able to expand the challenge of changing cutting conditions to include changes

to the accelerometers. We showed that cepstral mean subtraction allowed us to use data from different accelerometers in our training or to classify data from a different accelerometer than the one which was used to train model parameters.

8.2 Future Work

Future work with our system falls naturally into three main categories: factory floor research, algorithm and design research, and connections to speech recognition.

Factory floor research

The next step for the system described in this work is for it to be installed in the research facilities at Boeing Commercial Aircraft. This is expected to serve as a transition between the very controlled environment of the research lab and the less controlled environment of the factory floor. Past experience leads us to believe that our system will require important modifications before it is ready to be a standard manufacturing tool. It is difficult to anticipate the areas which will most benefit from extensions to the present system because of the high accuracy achieved on the data presently available. However, we are able to speculate about areas which are likely to cause problems.

In our work, the duration of each cutting pass was a constant within each data set. The multiplier for the wear level transition probabilities was learned during training and we were able to treat it as a constant. Moving to variable duration cutting will require a dynamic treatment of this model parameter and an implementation that allows wear transitions within a pass.

All of our cutting data consisted of climb-cutting of notches in the workpiece material. It remains to be seen how well our present pass topology will model changes in the types of milling operations. We found that changing workpiece material necessitated a change in HMM topology. If changing cutting operations also calls for a change in topology we will need a means of selecting the proper model for the desired operation.

In our desire to continually expand the amount of labeled data available for continued research, we propose the use of the present system in the data collection process. The wear on a cutter is low, (level "A" or "B") for most of the usable life. Taking the time to remove

and inspect cutters for annotation during these portions of their life is not as important as during the final minutes when wear is more rapid. If the present system were used to alert the operator when the end of life is near, measurements of wear which are temporally close together could be done without a significant increase in the time required by evenly spaced inspections. More labeled data near the end of a cutter's life where rapid wear begins will allow better identification of this critical time in the cutter life.

Algorithm and design research

Both the single-rate and multi-rate classifier achieved a 94% accuracy on the M-1" titanium cutters. The multi-rate showed improved performance using the NCE metric. However, table 8.4 shows that even the multi-rate classifier has difficulty properly modeling the titanium cutting. The table shows three of the cutters from the M-1" data set which exemplify the types of problems seen. The quantized wear labels assigned to cutter ti101 move to a wear level on the verge of being WORN (D) much too soon. The wear labels for ti103 do a good job of tracking wear up through the middle of life but stop at mid-life even for a cutter which is WORN. Finally we see a label (A) normally associated with a fresh cutter in the middle of the life of ti106. The features from the first four passes appear to be classified properly. However, when pass five is added to the entire feature sequence Y^i, the features from this new pass look so much like a fresh cutter that the maximum likelihood path through the wear lattice changes its previous decision and labels all five passes as "A". We assume that this is the result of a poorly modeled "quiet" cutting period. It appears that we were not truly successful in modeling the transient activity which we sought to capture in the fine-rate HMM and perhaps not the pass-level phenomenon we desired at the coarse. Our approach used an ad hoc technique for model initialization but then used a training approach based on a maximum likelihood criteria. Work being done with discriminative training which uses maximum mutual information or minimum classification error may be more appropriate for this application.

The edge wear we seek to model is actually the accumulation of small wear events. Adding the ability to model the *rate* of wear over a specified time increment is likely to provide improved performance for our system.

It is likely that the features selected to model dynamics within a cutting pass and across

Table 8.4: *Comparison of the known label to the label assigned by the best multi-rate classifier for three cutters in the M-1" titanium data set. Unknown labels are indicated by (x).*

ti101 - known	A	x	B	x	C	x	x	D	D	D
	A	B	D	D	D	D	D	D	D	D
ti103 - known	A	x	x	x	C	C	C	x	E	
	A	A	B	C	C	C	C	C	C	
ti106 - known	A	x	x	x	C	x	x	E		
	B	B	C	C	A	E	E	E		

wear levels are not the optimum features to model transient behavior. Work should be done to investigate feature selection for the different time domains.

Connections to speech recognition

Finally, in keeping with our interest in applying speech recognition techniques to the tool-wear problem, it would probably be useful to apply approaches developed for speaker adaptation to changing cutting conditions beyond the scope of the techniques already explored in this dissertation. Conversely, it would be interesting to apply the state-coupled multi-rate model to speech recognition applications, where the slow time scale might capture phenomena such as speaker identity.

One of the most challenging problems in tool wear monitoring is generalizing to changed cutting conditions. While our classifier and generalization techniques show promise, we expect more work will need to be done. Preliminary work with auto ambiguity features indicate that they will be able to generalize to different workpiece material. Work should be done to address channel normalization with these features.

There is a strong connection between the work here and speaker verification. Speaker verification models are trained with a limited amount of enrollment data and using a single type of microphone. The models must then be improved with additional data obtained during use and across multiple types of microphones. Future work in tool wear modeling

should use advances in speaker verification. Specifically, tool wear monitoring would benefit from speaker verification in answering the following questions:

- How should data recorded on the factory floor after the tool wear system has been installed be used to update classifier models?
- How can model transformation techniques developed to handle cell phone to office phone mismatch be used to transform models trained under one set of cutting conditions to another?
- What channel normalization techniques can be applied beyond cepstral mean subtraction?

BIBLIOGRAPHY

[1] G. Byrne, "The status of R&D in tool-condition monitoring," in *Proceedings of the CIRP/VDI Conference, VDI Berichte*, 1995, vol. 1179, pp. 17–45.

[2] Technical Editorial dept. Sandvik Coromant, *Modern Metal Cutting - A Practical Handbook*, Sandvik Coromant, Fair Lawn, NJ, 1996.

[3] M. Ostendorf, L. Atlas, R. Fish, O. Cetin, S. Sukittanon, and G.D. Bernard, "Joint use of dynamical classifiers and ambiguity plane features," in *Proceedings of ICASSP*, 2001, vol. to appear.

[4] L. Dan and J. Mathew, "Tool wear and failure monitoring techniques for turning - a review," *Int. J. Mach. Tools Manufact.*, vol. 30, pp. 579–598, 1990.

[5] S. Kim and B.E. Klamecki, "Milling cutter wear monitoring using spindle shaft vibration," *ASME Journal of Manufacturing Science and Engineering*, vol. 119, pp. 118–119, February 1997.

[6] R. Du, M.A. Elbestawi, and S.M. Wu, "Automated monitoring of manufacturing processes, part 2: Applications," *ASME Journal of Manufacturing Science and Engineering*, vol. 117, pp. 133–141, May 1995.

[7] X. Q. Li, Y. S. Wong, and A. Y. C. Nee, "A comprehensive identification of tool failure and chatter using a parallel multi-ART2 neural network," *ASME Journal of Manufacturing Science and Engineering*, vol. 120, pp. 433–442, May 1998.

[8] S. Rangwala and D. Dornfeld, "Sensor integration using neural networks for intelligent tool condition monitoring," *ASME Journal of Manufacturing Science and Engineering*, vol. 112, pp. 219–228, August 1990.

[9] E. Emel and E. Kannatey-Asibu, Jr., "Tool failure monitoring in turning by pattern recognition analysis of AE signals," *ASME Journal of Engineering for Industry*, vol. 110, pp. 137–145, May 1988.

[10] L.P. Heck and J.H. McClellan, "Mechanical system monitoring using hidden markov models," in *Proceedings of ICASSP*, 1991, vol. 3, pp. 1697–1700.

[11] T. A. Carolan, D. P. Hand, J. S. Barton, J.D.C. Jones, P. Wilkinson, and R. L. Reuben, "Assessment of tool wear in milling using acoustic emission detected by a fiber-optic interferometer," *ASME Journal of Manufacturing Science and Engineering*, vol. 118, pp. 428–433, August 1996.

[12] M. Lan and Y. Naeheim, "In-process detection of tool breakage in milling," *ASME Journal of Engineering for Industry*, vol. 108, pp. 191–197, August 1986.

[13] D.V. Hutton and F. Hu, "Acoustic emission monitoring of tool wear in end-milling using time-domain averaging," *ASME Journal of Manufacturing Science and Engineering*, vol. 121, pp. 8–12, February 1999.

[14] Y.M. Niu, Y.S. Wong, G.S. Hong, and T.I. Liu, "Multi-category classification of tool conditions using wavelet packets and ART2 network," *ASME Journal of Manufacturing Science and Engineering*, vol. 120, pp. 807–816, November 1998.

[15] K. C. Chou and L.P. Heck, "A multiscale stochastic modeling approach to the monitoring of mechanical systems," in *IEEE-SP International Symposium on Time-frequency and Time-scale Analysis*, 1994, pp. 25–27.

[16] L.E. Atlas, G.D. Bernard, and S.B. Narayanan, "Applications of time-frequency analysis to signals from manufacturing and machine monitoring sensors," *Proceedings of the IEEE*, vol. 84, pp. 1319–1329, September 1996.

[17] B.W. Gillespie and L. Atlas, "Data-driven time-frequency classification techniques applied to tool-wear monitoring," in *Proceedings of ICASSP*, 2000, pp. 649–652.

[18] B. Sick, *Online and indirect tool wear monitoring in turning with artificial neural networks: a review of more than a decade of research*, Academic Press, unpublished.

[19] R. Fish, M. Ostendorf, G.D. Bernard, D. Castanon, and H. Shivakumar, "Modeling the progressive nature of milling tool wear," in *Proceedings of the ASME, Manufacturing Engineering Division*, 2000, vol. 11, pp. 111–117.

[20] L. Atlas, M. Ostendorf, and G. Bernard, "Hidden markov models for monitoring machining tool-wear," in *Proceedings of ICASSP*, 2000, vol. 6, pp. 3887–3890.

[21] A. Thangaraj and P.K. Wright, "Computer-assisted prediction of drill-failure using in-process measurements of thrust force," *ASME Journal of Engineering for Industry*, vol. 110, pp. 192–200, May 1988.

[22] D.A. Anderson, "Tool sensing for milling," in *Society of Manufacturing Engineers (SME) conference, Machine Monitoring Sensors Clinic*, 1989, pp. MS89–460.

[23] J. Odell, D. Ollason, V. Valtchev, and P. Woodland, *The HTK Book*, Cambridge University, 1997.

[24] B. Gillespie and L. Atlas, "Optimizing Time-Frequency Kernels for Classification," *IEEE Transactions on Signal Processing*, p. in press, 2001.

[25] M. Siu and H. Gish, "Evaluation of word confidence for speech recognition systems," *Computer Speech & Language*, vol. 13, pp. 299–319, 1999.

[26] M.D. Owsley, L.E. Atlas, and G.D. Bernard, "Self-Organizing feature maps and hidden markov models for machine-tool monitoring," *IEEE Transactions on Signal Processing*, vol. 45, pp. 2787–2798, 1997.

[27] J. McLaughlin, L. Owsley, L.E. Atlas, and G.D. Bernard, "Advances in real-time monitoring of acoustic emissions," in *Proceedings of the SAE Aerospace Meeting*, 1997.

[28] L.R. Rabiner, "A tutorial on hidden markov models and selected applications in speech recognition," in *Proceeding of the IEEE*, 1989, vol. 77, pp. 257–285.

[29] R.G. Silva, K.J. Baker, S.J. Wilcox, and R.L. Reuben, "The adaptability of a tool wear monitoring system under changing cutting conditions," in *Mechanical Systems and Signal Processing*, 2000, vol. 14, pp. 287–298.

[30] R.G. Silva, R.L. Reuben, K.J. Baker, and S.J. Wilcox, "Tool wear monitoring of turning operations by neural network and expert system classification of a feature set generated from multiple sensors," in *Mechanical Systems and Signal Processing*, 1998, vol. 12, pp. 319–332.

[31] J. M. Chambers and T. J. Hastie, *Statistical Models in S*, Wadsworth & Brooks, 1992.

[32] H. Bourlard, S. Dupont, and C. Ris, "Multi-stream speech recognition," Tech. Rep. IDIAP-RR 96-07, IDIAP, 1996.

[33] N. Mirghafori, *A Multi-Band Approach to Automatic Speech Recognition*, Ph.D. thesis, ICSI, UC Berkeley, CA, USA, 1999.

[34] Z. Ghahramani and M.I. Jordan, "Factorial Hidden Markov Models," *Machine Learning*, vol. 29, pp. 245–273, 1997.

[35] L.K. Saul and M.I. Jordan, "Mixed Memory Markov Models," *Machine Learning*, vol. 37, pp. 75–87, 1999.

[36] B.T. Logan and P.J. Moreno, "Factorial HMMs for Acoustic Modeling," in *Proceedings of ICASSP*, 1998, pp. 813–816.

[37] H.J. Nock and S.J. Young, "Loosely-Coupled HMMs for ASR," in *Proceedings of ICSLP*, 2000, pp. III:143–146.

[38] S. Dupont and H. Bourlard, "Using multiple time scales in a multi-stream speech recognition system," in *Proceedings of Eurospeech*, 1997, vol. 1, pp. 3–6.

[39] J. Li, A. Najmi, and R. M. Gray, "Image classification by a two dimensional hidden markov model," in *Proceedings of ICASSP*, 1999, vol. 6, pp. 3313–3316.

[40] L.P. Heck, "Signal processing research in automatic tool wear monitoring," in *Proceedings of ICASSP*, 1993, vol. 1, pp. 55–58.

[41] L.P. Heck and K. C. Chou, "Gaussian-mixture model classifiers for machine monitoring," in *Proceedings of ICASSP*, 1994, vol. 6, pp. 133–136.

[42] S. S. Cho and K. Komvopoulos, "Correlation between acoustic emission and wear of multi-layer ceramic coated carbide tools," *ASME Journal of Manufacturing Science and Engineering*, vol. 119, pp. 238–245, May 1997.

[43] Y. Fan and R. Du, "Monitoring rotating tools using laser diffraction," *ASME Journal of Manufacturing Science and Engineering*, vol. 118, pp. 664–667, November 1996.

[44] C.S. Leem, D.A. Dornfeld, and S. E. Dreyfus, "A customized neural network for sensor fusion in on-line monitoring of cutting tool wear," *ASME Journal of Manufacturing Science and Engineering*, vol. 117, pp. 152–159, May 1995.

[45] B. Sick, "Monitoring the wear of cutting tools in cnc-lathes with artificial neural networks," in *Proceedings of ICASSP*, 1999, vol. 4, pp. 3381–3384.

[46] M. Bacchiani, M. Ostendorf, Y. Sagisaka, and K. Paliwal, "Design of a speech recognition system based on non-uniform segmental units," in *Proceedings of ICASSP*, 1996, vol. 1, pp. 443–446.

[47] M.I. Jordan, Z. Ghahramani, and L.K. Saul, "Hidden markov decision trees," Tech. Rep. 9606, MIT Computational Cognitive Science, June 1996.

[48] T. Svendsen and F. Soong, "On the automatic segmentation of speech signals," in *Proceedings of ICASSP*, 1987, pp. 77–80.

[49] Y. Linde, A. Buzo, and R.M. Gray, "An algorithm for vector quantizer design," *IEEE Transactions on Communications*, vol. 28, pp. 84–95, 1980.

[50] H. Shivakumar, "Prediction of tool wear likelihood for milling applications," M.S. thesis, Boston University, 1999.

[51] L.R. Rabiner and B.H. Juang, "An introduction to hidden markov models," *IEEE ASSP Magazine*, pp. 4–16, 1986.

VITA

Randall K. Fish grew up in Bolton, Connecticut, and received a B.S. in Physics with High Distinction, from Eastern Nazarene College in 1979. He received a B.S. in Electrical Engineering, with Highest Distinction, from Boston University in 1980. In 1982 he received an M.S. in Electrical Engineering from Boston University. Between 1980 and 1994 he held positions in several engineering corporations including Director of Engineering at DataProducts and Vice President of Engineering at Artisan Development and Saber Equipment. Since 1994 he has been an Associate Professor of engineering in the EngineeringPhysics department of Eastern Nazarene College. He is the author of three conference papers.