BAYESIAN MODELING OF MUSICAL EXPECTATIONS VIA MAXIMUM ENTROPY STOCHASTIC GRAMMARS

A DISSERTATION
SUBMITTED TO THE DEPARTMENT OF MUSIC
AND THE COMMITTEE ON GRADUATE STUDIES
OF STANFORD UNIVERSITY
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
DOCTOR OF PHILOSOPHY

Randal J. Leistikow

June 2006

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Jonathan Berger Principal Adviser

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Chris Chafe

I certify that I have read this dissertation and that, in my opinion, it is fully adequate in scope and quality as a dissertation for the degree of Doctor of Philosophy.

Julius O. Smith III

Approved for the University Committee on Graduate Studies.

Abstract

When presented with musical sounds, humans take advantage of prior knowledge of acoustic and musical context to accomplish an impressive array of cognitive listening tasks, such as meter tracking, transcription, style classification, instrument identification, harmonic analysis, and melody prediction. This dissertation presents a dynamic Bayesian framework for modeling listeners with differing musical expectations.

Although a simulated listener with specific experience may simply be created by learning prior distributions directly from a given musical corpus, a more interesting approach is to construct a listener whose expectations are governed by rules of music theory. Such rules are often expressed as statements involving musical tendencies, e.g., "A large upward melodic interval is typically followed by a smaller downward interval." This dissertation focuses on a novel method of transforming music-theoretic rule sets into parameterized, maximum entropy rate distributions suitable for use in dynamic Bayesian networks. Encoding rule-based expectations allows the system to infer which rules are most responsible for predicting musical attributes at each time in a piece, and to identify which rules are violated at points of musical surprise.

In addition to enabling a wide variety of interesting musical tasks to be performed using symbolic data as input, our framework can also be integrated into compatible probabilistic models that use recorded audio signals as input. The signal processing layers encode acoustic expectations by modeling the spectrotemporal evolution of instrument tones, and segment the signal into a sequence of note events. A system in which signal and symbolic layers inform one another is desirable because musical expectations can help the system compensate for corrupted signals, and the ability to

predict musical sequences suggests a future sequential Monte Carlo inference implementation in which sampling distributions concentrate on the most likely transitions, thereby avoiding the computational cost of evaluating all points in the potentially vast space of possible transitions.

Acknowledgments

I would like to begin by thanking my advisor, Jonathan Berger, whose constant support, supply of ideas, and friendship have been invaluable these many years. The other professors at CCRMA and members of my reading committee, Chris Chafe and Julius O. Smith III, have been truly wonderful in every respect, and I thank both of them for all of the knowledge they shared in their courses and seminars, and for truly caring about the welfare of their students. Much of what I learned during my coursework at Stanford was from professors in other departments, who are too numerous to list here by name, but who deserve my sincere gratitude.

My good friend and frequent research collaborator, Harvey Thornburg, deserves special thanks for the countless hours together sharing ideas, programming, and writing. Many of the ideas in this dissertation were hatched during conversations with him, and would not have been realizable without his expert assistance. I thank Craig Sapp for providing the folksong data used to construct examples in this dissertation, and for always being willing to lend a hand and share any of his incredible number of particularly useful skills. I have been honored to work and study alongside all of the other students and staff members of CCRMA and CCARH, and thank everyone for making the department an amazing place to learn and conduct research.

During the completion of this work, I have had the opportunity to work as a software engineer and researcher with two companies, Ordinate Corporation, in Menlo Park, CA, and Zenph Studios in Raleigh, NC. I thank Brent Townshend, Jarred Bernstein, and John Q. Walker, and all of the other employees of those companies for making me feel welcome and valued. I now count all of them as good friends.

My parents and wife's parents have been amazingly supportive the past several

years, as have both of my sisters. Though it might be atypical to acknowledge a member of another species, the antics and stream of dialogue from my grey parrot, Scooter, have kept me constantly laughing the past two years, so as a reward, I will let him shred my penultimate draft. My infant daughter, Keira, deserves special thanks for motivating me to finish this so I can come out of my office once in a while and enjoy her development. Last, and most of all, I thank my lovely wife, Kelly, for her encouragement and patience, especially during those times when it seemed like I might never finish this endeavor, and for all of her incredibly hard work. With love, I dedicate this dissertation to her.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Musical style is generally represented either by a set of analytic observations gleaned from a set of specific musical excerpts, or as a set of more generally-applicable rules representing a condensation of collected observations. In both cases these descriptive representations may have widely varying degrees of absoluteness and specificity. Arriving at an understanding of stylistic tendencies often evades systematization. It would, nonetheless, be both insightful and applicable to be able to encapsulate the tendencies of a particular composer or genre.

We would like to select rules from this body of knowledge and use them to inform algorithms that accomplish a wide variety of music-related tasks, but most rule statements are inherently incomplete, in the sense they assert a set of musical tendencies but fail to quantify the specific degrees to which they hold. An increasingly compelling approach is to consider both the evolution and reception of musical style from a probabilistic perspective. Leonard Meyer writes:

> The probability relationships embodied in a particular musical style together with the various modes of mental behavior involved in the perception and understanding of the materials of the style constitute the norms of the style. [54]

This dissertation addresses this fundamental problem of how to best represent musical tendencies, by presenting a framework for encoding music-theoretic rules

as conditional probability distributions designed to be used in the specification of dynamic Bayesian models. We use these dynamic systems to examine the formation, realization, and violation of musical expectations by modeling hypothetical, virtual listeners whose prior experience can be summarized by a set of music-theoretic rules.

Dynamic Bayesian networks are expressive and flexible probabilistic graphical models that allow hierarchical musical structures, however intricate, to be created by linking together several smaller local probability models describing only direct dependencies among musical attributes. A dynamic Bayesian network is fully specified by a prior distribution, quantifying what we know about all of the variables in the model just before the very first note of a piece, and a set of conditional probability distributions defining simultaneous dependencies among variables at the same time and direct dependencies on variables in the preceding time step. It is in these conditional prob- ability distributions, which we will also call *transition distributions*, that we encode rule statements about musical tendencies. Consider, for instance, the case of melodic structures and tendencies.

In modeling melodic sequences, the transition distribution corresponding to each observable note quantifies, informally, $P(N_i \,|\, Context_i, Context_{i-1})$, where N_i represents the choice of note at time i, $Context_i$ are all variables at the same time that N_i depends on, and $Context_{i-1}$ are all of the variables in the previous time step on which N_i depends. We encode any information in a rule as contraints on this transition distribution. Often, we are given the general form of a rule, but do not know exactly how to interpret the rule numerically. For example, a general post-skip-reversal assertion that "a large melodic interval tends to be followed by a smaller interval in the opposite direction," does not give us a set of numbers to assign to the distribution; we ultimately need such numbers to perform computations. The primary problem, assuming that we can determine which intervals are "large," is that we don't know the exact strength implied by the phrase "tends to." Our approach is to parameterize unknown aspects of rules, turning the above statement into the form, "the probability that a large melodic interval is followed by a smaller interval in the opposite direction in at least α times as great as it being followed by some other type of interval."

Notice, though, that the parameterized rule only explicitly deals with a few of

the variables in the transition distribution; the rule, for example, does not say anything about what happens after a small interval. Among all the possible choices of distributions that would satisfy the parameterized constraint, we choose the one that maximizes the entropy rate of the transition distribution, which is equivalent to maximizing our average uncertainty of the note N_i given its context. The most uncertain, maximially noncommittal distribution is the uniform transition distribution, because it assigns equal probability to all values of N_i regardless of its context. That would not satisfy the above-stated constraint, however, so we start from a uniform distribution and strive to maintain a distribution that is as uniform as possible given all rule constraints. Fortunately, this entropy rate maximization problem turns out to be a convex problem, so fast algorithms with guaranteed convergence properties exist to solve it. The reason that we maximize the entropy rate of transition distributions is that we want the distributions to be as widely applicable as possible. A consequence of information-theoretic asymptotic equipartition property is that any transition distribution (that admits a stationary distribution) is associated with two sets of musical pieces, a set whose probability of being generated is close to one, called the *typical set*, and a set whose probability is close to zero. For a given set of rule constraints, any distribution other than the maximum entropy rate distribution reduces the size of the typical set, which reduces the number of pieces for which inference using the model is relevant; if the probability of a piece being generated by a melodic model tends to zero, then, in a sense, all bets are off when trying to infer anything from that melody.

For various choices of the rule parameter α, we can thus obtain a whole family of maximum entropy rate transition distributions that assert the skip-reversal tendency with varying degrees of absoluteness. Given a set of musical data corresponding to a specific musical context, we use an iterative expectation-maximization algorithm to actually learn the value of α that maximizes the likelihood of the particular set of data. This rule adaptation process corresponds well to how humans interact with the world. We operate using a set of schema learned from our prior experience, then adapt those assumptions based on the particular situation at hand.

This modeling process becomes particularly interesting when multiple encoded

rules interact to determine the sequence of notes. This is accomplished by having the each note's context include a random varible that weights the contribution of multiple maximum entropy rate transition distributions depending on musical context. The choice of which rules to apply when choosing a note could depend, for example, on its beat position or whether it is the first note in a new phrase. Because we operate in the context of a Bayesian network, we can then invert the dependence, and use the probabilistic structure to infer rule activation and violation from the sequence of observed notes. Because we can model the interaction of multiple rules, we can encode and compare entire music-theoretic rule sets. The maximum entropy rate transition distibutions and model structure give us the ability to predict upcoming musical events and form musical expectations. We can observe the predictive distributions to characterize the strength and specificity of the imagined realization, then quantify the surprise associated with observing the note given our predictions.

Finally, the extensible structure of dynamic Bayesian networks make it not only relatively straightforward to add increasingly expressive musical modeling layers, but also possible to link these layers to compatible signal processing models. The resulting system can use general statements about musical tendencies to improve some aspect of the signal processing results. For example, even the simple statement that leaps larger than an octave are rare in a given type of music could significantly reduce the rate of octave assignment errors produced by an automated transcription system. In the other direction, the higher-level musical expectation layers can then include signal features that are only present in a particular musical performance and not available in any symbolic score.

This framework draws from a number of related areas of research. Maximum entropy methods similar to the one we propose are widely used in the closely-related discipline of natural language processing (NLP) to perform such tasks as sentence segmentation, part-of-speech tagging, bilingual sense disambiguation, and word reordering [4]. Tutorial introductions to maximum entropy language modeling appear in Berger *et al.* [4] Rosenfeld [76], and Ratnaparkhi [73]. A number of researchers have explicitly explored the relationship between language modeling and music modeling. Bod [12, 13, 14] applies language parsing models to music, and highlights the

commonalities between linguistic and musical perception, showing that the same parameter setting achieves maximum parsing accuracy for both a database of sentences and a collection of folksongs. Pickens [65] applies probabilistic text information retrieval techniques to the problem of monophonic music retrieval.

Dynamic probabilistic graphical models are used commonly in musical applications, because the time-evolving nature of the models naturally reflects the evolution of musical attributes over time. The most widely utilized dynamic model is a hidden Markov model (HMM), which contains two random variables, one of which is observable at each point in time (e.g., audio signal samples, notes in a score), and the other of which is hidden (e.g., unknown segmentation points in an audio signal, unknown phrase boundaries in a score). The value of the hidden state at any time i directly depends on only the value of the hidden state at the preceding time step, i–1, and the value of the observable variable at time i directly depends on only the hidden variable at that same time, i. Given a sequence of observations, the probabilistic dependence can be inverted using Bayes' rule, enabling the estimation of the optimal value of the hidden state at each point in time, or the estimation of the optimal sequence of hidden states. Representative examples of the variety of musical application areas to which hidden Markov models have been applied include: audio segmentation [68, 3], music classification [21], music composition [24], chord recognition [82], real-time interactive performance [62], melody spotting [31], harmonic modeling [66], and key identification [84].

Hidden Markov models are a special case of dynamic Bayesian networks, in which each time slice can include an arbitrarily complex hierarchical structure of observable and hidden state variables. Like an HMM, a standard dynamic Bayesian network only allows direct dependencies among the collection of variables in the same time slice and preceding time slice. A detailed definition of dynamic Bayesian networks appears in Section 2.3.1. The use of dynamic Bayesian networks in musical applications seems to be picking up momentum, with a number of researchers making considerable contributions. A probabilistic model in which one set of variables is responsible for randomly generating a set of observations is appropriately termed a *generative model*. A common thread of recent research, to which we subscribe wholeheartedly, is the use

of models in which symbolic musical layers are responsible for generating observed signal features. In these models, knowledge of musical structure encoded in symbolic layers of the model can inform states that improve aspects of the signal processing, and vice versa. Examples of research projects utilizing combined symbol/signal models include the automated transcription work of Cemgil [18, 19, 20] and Hainsworth [34, 35], a variety of score alignment, harmonic analysis, melody recognition, and transcription tasks by Raphael [71, 70, 69, 72], the quasi-harmonic signal models of Davy and Godsill [28, 33] the context-informed audio segmentation models of Thornburg [86, 88, 85], and the chord recognition model of the author *et al.* [49].

1.1 Organization of the dissertation

Chapter 2 introduces dynamic Bayesian networks, emphasizing the direct correspondence between probabilistic graphical models and the compact distributional factorizations they encode. We present a first-order Markov model of melody to demonstrate how a hypothetical listener's prior musical experience can be used in a dynamic probabilistic model to predict upcoming musical events. We briefly discuss how characteristics of predictive distributions affect the emotional response of a listener, and explain an information-theoretic metric for quantifying the surprise associated with an observed note relative to note predictions. By augmenting the first-order model's memory of previous notes and adding a hidden state representing unobserved musical attributes, we obtain a dynamic Bayesian network that forms the core of our melodic expectation models. That basic model elegantly fuses information from data-driven and schema-driven processes, providing a structure upon which a variety of more expressive musical models can be constructed.

Chapter 3 explains how standard types of Bayesian inference reflect the listening experience of a simulated lister that updates its beliefs about the current musical context as it hears each new note in real time, then retrospectively updates its earlier beliefs after additional notes are heard. We present the set of algebraic relationships necessary to recursively compute each type of inference for our basic expectation model, and apply those computational steps in an example in which a virtual listener's

objective is to determine the musical mode of a piece. The chapter concludes with a discussion relating the behavior of long-term predictions to the stationary distribution of a stochastic process, concepts which will reappear at the end of the following chapter.

Chapter 4 presents a framework for encoding musical rules as parameterized maximum entropy rate transition distributions. Maximizing entropy rate is accomplished using an iteratively convex optimization process, which is both computationally efficient and user friendly, in the sense that congruent rules can automatically be combined by simply concatenating their constraint sets. More importantly, in maximizing the entropy rate, we maximize the number of pieces the resulting stochastic process generates, making the system as widely applicable as possible given the rule constraints.

Chapter 5 explains an expectation-maximization (EM) approach to learning rule parameters from a corpus of musical data. The EM algorithm comprises a sequence of updates in which estimated parameter values converge to the values maximizing the likelihood of the corpus, assuming that an initial guess is sufficiently close. The last section of the chapter steps through an example of learning two unknown parameters.

Chapter 6 shows that if the hidden layer in the basic expectation model is a switching state that selects from several maximum entropy rate note transition distributions, we are able infer which rules are most responsible for determining any observed note, and also identify which rules are violated at points of musical surprise.

Chapter 7 demonstrates how the basic two-layer expectation model can be augmented to express increasingly complex musical relationships. Specifically, we introduce variables that represent meter, beat position, bar line crossing, harmony, and note duration. The addition of metrical context to any model of musical expectation is vital, because the musical meter contantly generates strong and specific expectations, providing a "framework against which expectations can be both realized and willfully violated." [5]

Chapter 8 demonstrates how hierarchical models of musical expectation can be incorporated into compatible dynamic Bayesian models that operate using features extracted from recorded audio signals. This chapter draws extensively from work done

in collaboration with Harvey Thornburg, and the opening sections of this chapter summarize the more detailed presentation appearing in his Ph.D. dissertation [86]. The key concept is that the signal processing layers segment the signal into discrete note events, and that the model assumes a first-order Markov relationship between notes. Each of the music expectation models in this dissertation is designed to model first-order note transitions given additional musical context, so the symbolic and signal layers can be seamlessly integrated. This means that musical context can affect the signal processing, by helping to resolve segmentation or pitch identification ambiguity, and higher layers of the combined structure can model aspects of the music that are only realized in a performance.

The final chapter summarizes the contributions of this dissertation and outlines a number of research projects that might follow directly from this work.

Chapter 2

Dynamic Bayesian Representations of Melodic Expectations

A pervasive concept in this dissertation is the utility of decomposing intricate musical relationships into more manageable components. Operating on simpler factors of a complex system often results in improved knowledge acquisition, domain modeling, computational efficiency, and overall system comprehension. These considerations, along with the desire to make statistically optimal decisions, organically lead to dynamic Bayesian network (DBN) representations of probability distributions.

This chapter provides a tutorial introduction to dynamic Bayesian networks, focusing on building intuition about the direct correspondence between graphical representations and the distributional factorizations they encode. We present a first-order Markov model to demonstrate the process of predicting upcoming musical events, describing characteristics of predictions, and measuring the surprise associated with observations relative to those predictions. By lengthening the system's memory of previous notes and adding an unobserved variable, the first-order Markov chain is extended to become a DBN that forms the core of our musical expectation models. This basic expectation model elegantly fuses information from data-driven and schema-driven processes, providing a structure upon which more expressive musical models can be constructed.

2.1 Introduction to Bayesian networks

2.1.1 Directed acyclic graphs

We define a *graph* $\mathcal{G}$ to be a pair $\mathcal{G} = (Nodes, Edges)$, where $Nodes$ is a finite set of vertices, $\{Node_1, Node_2, \ldots, Node_{|Nodes|}\}$, and $Edges$ a subset of all possible pairs of the nodes. An edge $(Node_i, Node_j)$ must connect two distinct vertices; i.e. $Node_i \neq Node_j$, and because $Edges$ is a set, at most one edge can connect any pair of nodes. An edge $(Node_i, Node_j)$ is *directed* if $(Node_i, Node_j) \in Edges$ but $(Node_j, Node_i) \notin Edges$. We indicate directivity of such an edge using $Node_i \rightarrow Node_j$, and define $Node_i$ to be the *parent* of $Node_j$, and $Node_j$ to be the *child* of $Node_i$. The set of all parents of a node $Node_k$ is denoted by Parents($Node_k$), and the set of children by Children($Node_k$). A graph $\mathcal{G}$ is a *directed graph* if all of its edges are directed. Figure 2.1 displays an example of a directed graph consisting of five nodes and five edges.

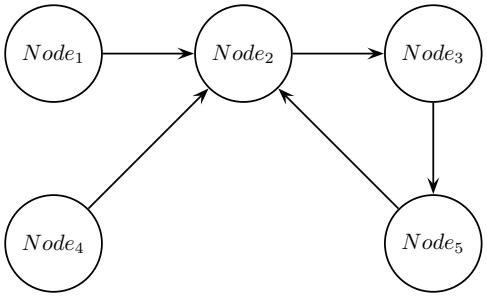


Figure 2.1: Example of a directed graph. Here, Parents($Node_2$) = $\{Node_1, Node_4, Node_5\}$, and Children($Node_2$) = $\{Node_3\}$.

A length-N *directed path* from $Node_i$ to $Node_j$ is a sequence of distinct nodes $\nu_0 : \nu_N$ where $\nu_0 = Node_i$ and $\nu_N = Node_j$, such that $(\nu_{k-1}, \nu_k) \in Edges$ for all $k = 1 : N$, and at least one of the edges is directed [26]. Visually, we can start at $Node_i$ and reach $Node_j$ by following the arrows through the sequence of nodes $Node_i \rightarrow \nu_1 \rightarrow \ldots \rightarrow \nu_{N-1} \rightarrow Node_j$. For example, Figure 2.1 contains a directed path from $Node_4$ to $Node_5$: $Node_4 \rightarrow Node_2 \rightarrow Node_3 \rightarrow Node_5$.

Modifying a directed path so that its terminating nodes are identical, $\nu_0 = \nu_N$,

produces a *directed cycle*. A *directed acyclic graph* (DAG) is a graph that contains no directed cycles; i.e., it is impossible to follow a path of directed edges from $Node_i$ through any other sequence of nodes and arrive back at $Node_i$. Figure 2.1 is not a DAG, because it contains the directed cycle $Node_2 \rightarrow Node_3 \rightarrow Node_5 \rightarrow Node_2$.

2.1.2 Bayesian network representations of probability distributions

A *Bayesian network* (BN) is a directed acyclic graph whose nodes represent a set of discrete or continuous random variables $\{X_{1:N}\}$, and whose edge structure encodes a set of conditional independence assumptions about the distribution $P(X_{1:N})$.[1] These *local Markov assumptions* are that each variable X_i is conditionally independent of its *non-descendants* given its parents:

$$(X_i \perp \text{NonDescend}(X_i) \mid \text{Parents}(X_i)) \tag{2.1}$$

where $\text{NonDescend}(X_i)$ is a set containing all variables in the domain that do not terminate a directed path from X_i [42]. See [26, 41, 42, 63, 77, 80] for detailed presentations of dependence relationships in probabilistic graphical models, and, more specifically, how knowing the value of a node's parents in a DAG blocks the flow of information from all nondescendents, including other ancestors along directed paths to the node.

Under these local Markov assumptions, $P(X_{1:N})$ can be compactly factored as the product of local probabilistic models $P(X_i \mid \text{Parents}(X_i))$ quantifying how each node X_i is *directly* affected by its parents:

$$P(X_1, \ldots, X_N) = \prod_{i=1}^{N} P(X_i \mid \text{Parents}(X_i)) \tag{2.2}$$

The chain rule for Bayesian networks in (2.2) follows directly from the more general chain rule of probability (true for all distributions) if the set $\{X_{1:N}\}$ is topologically

[1] $\{X_{1:N}\}$ is shorthand notation for $\{X_1, \ldots, X_N\}$; accordingly, $P(X_{1:N}) \triangleq P(X_1, \ldots, X_N)$

ordered with parents preceding their children; i.e., $\{X_{1:i-1}\} =$ Parents$(X_i) \cup Z$, where $Z \subseteq$ NonDescend(X_i) [42]. Local Markov assumptions dictate that $(X_i \perp Z \mid \text{Parents}(X_i))$, and node ordering guarantees all parents of X_i are in $X_{1:i-1}$, so:

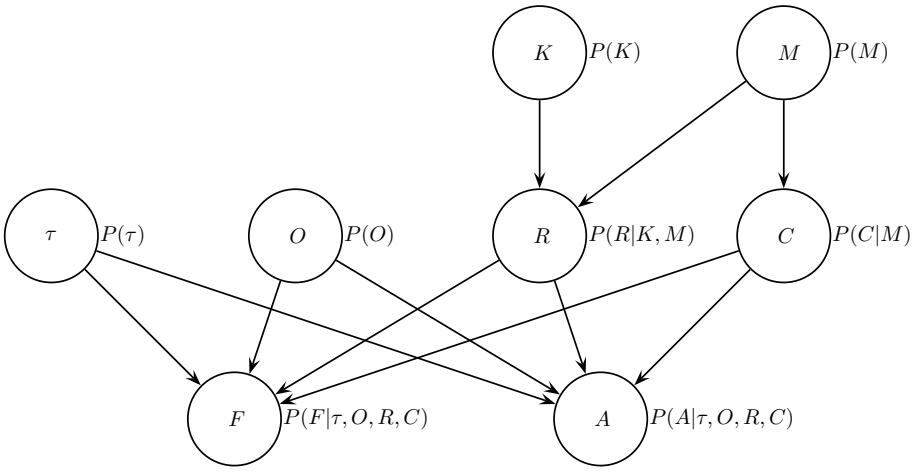
$$\begin{aligned} P(X_{1:N}) &= P(X_1)P(X_2|X_1)P(X_3|X_{1:2}) \dots P(X_N|X_{1:N-1}) \\ &= \prod_{i=1}^{N} P(X_i \mid X_{1:i-1}) \\ &= \prod_{i=1}^{N} P(X_i \mid \text{Parents}(X_i)) \end{aligned} \tag{2.3}$$

The ability to decompose a potentially complex system into smaller, interconnected local probability models offers us an extremely intuitive modeling tool. We can simply draw one node for each variable in our model, draw arrows connecting any pair of nodes where one variable directly influences the other (being careful to avoid directed cycles), and read off the resulting factorization of the joint distribution of all variables in our domain. To provide an example of the immediacy of translation between a Bayesian network's visual representation and factorization it encodes, Figure 2.2 displays a network developed by Thornburg, Smith, Berger, and the author to identify musical chords from observed short-time Fourier transform (STFT) peak frequencies and amplitudes [49]. Note that in cases where a node X_i has no parents ($\{K, M, \tau, O\}$ in Figure 2.2), $P(X_i \mid \text{Parents}(X_i)) = P(X_i)$.

It is important to temper this optimistic description of a BN-construction procedure with the fact that we are given no guarantees that all queries we might wish to perform using the resultant structure are computationally feasible. Section 2.2.1 explores relationships between compactness of representation and computational complexity.

2.1.3 Probabilistic inference interpreted as edge reversal

In Bayesian networks, the conditional distributions $P(X_i \mid \text{Parents}(X_i))$ often model causal relationships, quantifying how Parents(X_i) effect changes in X_i [64]. Bayes' rule



$$P(K, M, \tau, O, R, C, F, A) = P(K)\ P(M)\ P(\tau)\ P(O)\ P(R|\text{Parents}(R))\ P(C|\text{Parents}(C))\ P(F|\text{Parents}(F))\ P(A|\text{Parents}(A))$$
$$= P(K)\ P(M)\ P(\tau)\ P(O)\ P(R|K, M)\ P(C|M)\ P(F|\tau, O, R, C)\ P(A|\tau, O, R, C)$$

Figure 2.2: An example Bayesian network and the factorization it encodes. Each conditional probability distribution, $P(X_i|\text{Parents}(X_i))$, is displayed just to the right of the node to which it corresponds. An explanation of the variable labels, displayed here just to symbolically illustrate factorization over a DAG, appears in [49].

enables the inference of unknown causes from observed effects, in essence reversing the direction of edges in the graph.

Suppose, for example, that we observe a single note and wish to determine the probability of the musical key associated with that pitch, given just that one note. This problem corresponds to the pair of Bayesian networks depicted in Figure 2.3. Prior to observing the note, we are able to specify distributions for the two factors in Figure 2.3A: a *prior distribution* $P(Key)$ quantifying our belief about Key before observing any evidence, and a *likelihood function* $P(Note|Key)$ returning the probability of observing each note in a given key. Our goal is to obtain one of the factors in Figure 2.3B, the *posterior distribution* $P(Key|Note)$ representing our updated belief about Key once we have observed a note. Because the two networks in Figure 2.3 represent joint distributions over the same variable space, $P(Key, Note)$, Bayes' rule

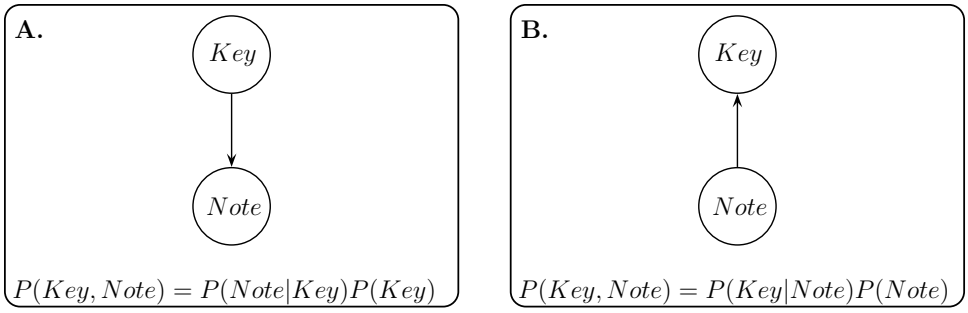


Figure 2.3: Applying Bayes' rule is equivalent to reversing the edge

follows directly from equating the factorizations of the two networks:

$$\begin{aligned} P(Note|Key)P(Key) &= P(Key|Note)P(Note) \\ P(Key|Note) &= \frac{P(Note|Key)P(Key)}{P(Note)} \end{aligned} \tag{2.4}$$

In practice, we often drop the factor $P(Note)$ from the right-hand-side of (2.4), because the process of making a statistically optimal decision typically involves maximizing the posterior, and scaling $P(Key|Note)$ by the same constant for all key values does not affect the maximization.

$$\begin{aligned} P(Key|Note) &= \frac{P(Note|Key)P(Key)}{P(Note)} \\ P(Key|Note) &\propto P(Note|Key)P(Key) \\ \text{Posterior} &\propto \text{Likelihood} \times \text{Prior} \end{aligned} \tag{2.5}$$

If the constant of proportionality must be later reconstructed, it is chosen such that $P(Key|Note)$ is a distribution (i.e., its elements sum to 1).

Up to this point, we have described Bayesian networks using only symbolic variables. To demonstrate this simple key inference example using actual numeric values, let $Key \in \{\mathrm{C_{Maj}}, \mathrm{F_{Maj}}\}$ and $Note \in$ {C,C♯/D♭,D,D♯/E♭,E,F,F♯/G♭,A,A♯/B♭,B}. Suppose we lack *a priori* knowledge about the key. Following the principle of maximum entropy

discussed in Chapter 4, we specify $P(Key)$ as uniform:

$$P(Key = \mathrm{C_{Maj}}) \;=\; P(Key = \mathrm{F_{Maj}}) \;=\; 0.5 \tag{2.6}$$

The likelihood function $P(Note|Key)$ could be obtained using a variety of approaches, such as basing it on frequency of note occurrence in a corpus of key-labeled pieces, as Temperley does in [84], or deriving it from the results of psychological experiments such as those conducted by Krumhansl and Kessler [45]. For this example, we take Krumhansl and Kessler's standardized major key profile and scale it to create the *conditional probability table* (CPT) displayed in Table 2.1.

$P(Note\|Key)$												
Key \ *Note*	C	C♯/D♭	D	D♯/E♭	E	F	F♯/G♭	G	G♯/A♭	A	A♯/B♭	B
$\mathrm{C_{Maj}}$	0.152	0.053	0.083	0.056	0.105	0.098	0.060	0.124	0.057	0.088	0.055	0.069
$\mathrm{F_{Maj}}$	0.124	0.057	0.088	0.055	0.069	0.152	0.053	0.083	0.056	0.105	0.098	0.060

Table 2.1: Likelihood $P(Note|Key)$ obtained by scaling Krumhansl and Kessler's standardized major key profile.

Because $P(Key)$ in this example is uniform, the joint distribution $P(Key, Note) = P(Note|Key)P(Key)$ is obtained by multiplying each entry in Table 2.1 by 0.5:

$P(Key, Note) = P(Note\|Key)P(Key)$												
Key \ *Note*	C	C♯/D♭	D	D♯/E♭	E	F	F♯/G♭	G	G♯/A♭	A	A♯/B♭	B
$\mathrm{C_{Maj}}$	0.076	0.027	0.042	0.028	0.053	0.049	0.030	0.062	0.029	0.044	0.028	0.035
$\mathrm{F_{Maj}}$	0.062	0.029	0.044	0.028	0.035	0.076	0.027	0.042	0.028	0.053	0.049	0.030

Table 2.2: Joint distribution $P(Key, Note) = P(Note|Key)P(Key)$

The posterior distribution $P(Key|Note)$ displayed in Table 2.3 is computed by scaling each column in Table 2.2 so that the values in the column sum to one. Normalizing the column corresponding to note n is equivalent to dividing the column's values by $P(Note = n)$, as stated in (2.4). When a full joint distribution is available, we can use it to answer any probabilistic query about the domain by

$P(Key\|Note) = P(Note\|Key)P(Key)/P(Note)$												
Key \ *Note*	C	C♯/D♭	D	D♯/E♭	E	F	F♯/G♭	G	G♯/A♭	A	A♯/B♭	B
C_{Maj}	**0.551**	0.482	0.485	**0.505**	**0.603**	0.392	**0.531**	**0.599**	**0.504**	0.456	0.360	**0.535**
F_{Maj}	0.449	**0.518**	**0.515**	0.496	0.397	**0.608**	0.469	0.401	0.496	**0.544**	**0.641**	0.465

Table 2.3: Posterior probability distribution $P(Key|Note)$. For each note, a bold number indicates $\text{argmax}_{key}\ P(key|note)$.

summing the probabilities of all atomic events that satisfy the query propositions. $P(Note = n)$ is then just the sum of the two entries in $P(Key, Note)$ for which $Note{=}n$.

$$P(Note{=}n) \ = \sum_{k \in \{C_{Maj}, F_{Maj}\}} P(Key{=}k, Note{=}n) \tag{2.7}$$

For example, to obtain the values in the leftmost column of the posterior in Table 2.3, we divide the entries in the leftmost column of Table 2.2 by their sum, $P(Note{=}\text{C}) = 0.076 + 0.062 = 0.138$.

More generally, for any joint distribution containing sets of random variables X and Y, a process called *marginalization* obtains a distribution over just X by summing out all variables in Y: $P(X) = \sum_{y \in Y} P(X, y)$. This variable elimination process is called *conditioning* if it sums conditional probabilities [77]: $P(X) = \sum_{y \in Y} P(X|y)P(y)$.[2] Exact inference in Bayesian networks involves two operations demonstrated in this above example: forming the product of conditional probabilities (as when multiplying prior and likelihood), and eliminating variables using marginalization or conditioning. Section 7.3 details a recursive algorithms for performing exact inference in a dynamic Bayesian networks designed to model musical expectation. Descriptions of several general algorithms for performing inference in Bayesian networks appear in [26, 36, 41, 56, 77].

The posterior distribution $P(Key|Note)$ in Table 2.3 exhibits behavior we might expect. The largest value in the table, $P(\text{F}_{\text{Maj}}|\text{B}^{\flat})$, corresponds to the only note

[2]Going forward, notation for marginalization and conditioning will be simplified slightly to $P(X) = \sum_Y P(X, Y)$ and $P(X) = \sum_Y P(X|Y)P(Y)$, where the sum is understood to be over the range of Y.

that appears in one key but not the other, and each key is more probable given its tonic: $P(\mathrm{C_{maj}}|\mathrm{C}) > P(\mathrm{F_{maj}}|\mathrm{C})$ and $P(\mathrm{F_{maj}}|\mathrm{F}) > P(\mathrm{C_{maj}}|\mathrm{F})$. In order to actually make a decision about the key, we would intuitively choose the one with highest posterior probability given the observed note; these choices are emphasized in Table 2.3. A well-known result in statistical decision theory [6, 10] is that if our objective is to minimize the average probability of choosing the wrong key, the decision rule maximizing the posterior is in fact statistically optimal.

2.2 A first-order observable Markov model of melody

The art of music involves the arrangement of sonic elements in time. Any meaningful model of musical expectations must therefore encode information about the emergence over time of perceptually relevant patterns and probabilities. This section demonstrates the fundamentals of probabilistic modeling of musical sequences by presenting a simple probabilistic network that models only the sequence of notes in a melody. Section 2.3 will then demonstrate how this simple model can be augmented to include other unobserved, or *hidden*, variables.

2.2.1 Modeling considerations

We begin by modeling a melody as a sequence of random variables $N_{1:K}$, where K is the length of the melody, N_i indicates that the note appears at the i^{th} position in the sequence of notes, and each N_i belongs to a set of notes $\mathcal{N}$. Such an indexed sequence of random variables is called a *stochastic process*; furthermore, because the index i advances only when the note changes, rather than at a regularly-sampled time interval, this is termed a *discrete-event* process [61, 56]. In subsequent chapters, the space of notes will be defined as the set of all semitones within a range spanning multiple octaves, but for now let $\mathcal{N}$ contain one entry per pitch class:

$\mathcal{N}$ ={C,C♯/D♭,D,D♯/E♭,E,F,F♯/G♭,A,A♯/B♭,B}.[3]

Recall that the complete joint distribution $P(N_{1:K})$ gives us sufficient information to answer any probabilistic query about our domain. If no independencies are assumed among the K variables, the complete distribution would require the specification of a K-dimensional table containing 12^K entries.[4] For example, modeling a thirty-note melody would take 12^{30} ($\approx 2.4 \cdot 10^{32}$) values. One problem with attempting to specify such a large table is that it may infeasible to acquire the necessary knowledge; to learn a distribution of that size would require an inordinately large amount of training data. A second problem is that one typically gains little insight from specifying such a huge, unstructured set of numbers. This lack of interpretable structure is one major drawback of many "black-box" approaches. In the case of neural networks, for example, after fitting training data one often has little intuition about how to interpret the information distributed in the weights [91]. A third problem is that when using such a large table for inference, marginalizing out variables is extremely computationally intensive, compared to using a factorized structure where marginalization involves nested summations over the relatively small CPTs in which each variable actually participates.

Considering possible factorization options, the joint distribution $P(N_{1:K})$ could be specified using only $12 \cdot K$ numbers if $N_{1:K}$ were assumed to be marginally independent, $P(N_{1:K}) = \prod_{i=1}^{K} P(N_i)$, but a melodic expectation model in which each note is independent of all others would be of little use. Perhaps the simplest practical assumption one could make is that the note at N_i depends on only the preceding note, N_{i-1}. In this case, our assumptions match those encoded by a first-order observable Markov model [67]:

$$P(N_i \mid N_{1:i-1}) = P(N_i \mid N_{i-1}) \tag{2.8}$$

[3]Throughout this dissertation, we collapse all inharmonic spellings of a note (e.g., C♯ and D♭ in the same octave) into a single value. If applications require the ability to discriminate among multiple spellings of the same note, the set $\mathcal{N}$ could be redefined so each spelling is a distinct entry.

[4]Because distributions sum to 1, one could omit an entry from a joint probability table and one or more values from a conditional probability table; for clarity of presentation, we include the full size of tables in variable counts (e.g., 12^K entries instead of $12^K - 1$).

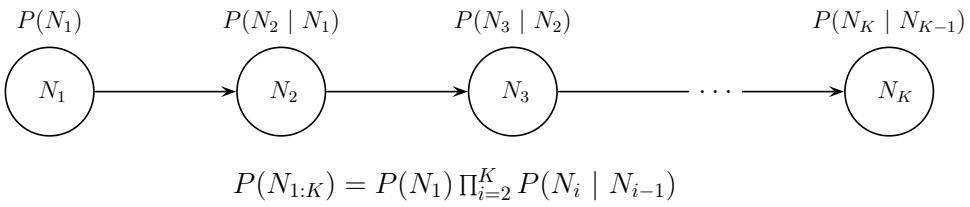


Figure 2.4: Bayesian network representation of a first-order Markov chain

Recalling the semantics of Bayesian networks in Section 2.1.2, this Markov chain can be represented as a Bayesian network simply by constructing a DAG with one node per note, such that N_{i-1} is the sole parent of N_i for all $i \in [2\!:\!K]$. Figure 2.4 displays this network and its factorization. In general, this representation would require 12 values to specify the prior, $P(N_1)$, plus $12^2 \cdot (K-1)$ values to specify the $K-1$ conditional distributions, $P(N_i \mid N_{i-1})$ for $i \in [2\!:\!K]$. However, we obtain a much more compact representation by further assuming that the Markov chain is *time invariant* or *homogeneous* — that the conditional distributions are identical regardless of the time index: $P(N_i \mid N_{i-1}) = P(N_2 \mid N_1),\ \forall\ i \in [2\!:\!K]$. This additional assumption reduces the number of necessary values to $12+12^2$. Independence and time-invariance assumptions are thus exploited to provide a more compact representation of $P(N_{1:K})$, facilitate knowledge acquisition (as it is straightforward to obtain first-order transition probabilities from a corpus of data), and provide computational savings in the inference process.

2.2.2 Model specification using distributions obtained from a corpus of folksong data

To make the above discussion concrete, this section will complete the specification of the model in Figure 2.4 using distributions obtained from the Essen folksong collection [79] using customized versions of functions in the Humdrum Toolkit [37]. The Essen folksong collection comprises approximately 8400 tonal melodies, from which we have chosen to use the 6219 that are labeled as European in origin. The average length of these European folksongs is 48 notes, and the majority of them, 5453, are in major

keys. Melodies tend to be rhythmically and tonally unembellished; for example, in major keys, only about 1% of the notes are raised or lowered. The corpus is thus well-suited for providing straightforward examples that align well with our intuition about major and minor modes.

Prior, joint, and conditional probability distributions were calculated by transposing all songs to $\mathrm{C_{Maj}}$ or $\mathrm{C_{min}}$, then counting event occurrences and normalizing the counts.[5] Because our simple Markov model does not include a variable corresponding to musical mode (major or minor), we are for now limited to working with only one mode at a time, using distributions that match the given mode. To illustrate the statistical differences between modes, Figure 2.5 displays the distribution of single notes and the joint distribution of adjacent note pairs, $P(N_{t-1}, N_t)$, for both major and minor folksongs. As expected, Figures 2.5a and 2.5b show that $P(\mathrm{E}) \gg P(\mathrm{E}^\flat)$ in major folksongs, $P(\mathrm{E}^\flat) \gg P(\mathrm{E})$ in minor songs, and the note G is the most probable in both modes, due to its dual tonic/dominant role.

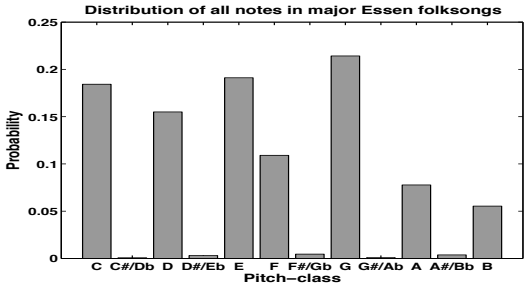
To complete the specification of the first-order Markov model in Figure 2.4 requires selecting a prior $P(N_1)$ and first-order transition distribution $P(N_i \,|\, N_{i-1})$. Two different choices of transition distribution, corresponding to major and minor modes, are obtained by normalizing the entries of the joint note distributions in Figures 2.5c and 2.5d so that all values corresponding to each choice of N_{i-1} sum to 1:

$$P(N_i \,|\, N_{i-1}) = \frac{P(N_{i-1},\ N_i)}{P(N_{i-1})} = \frac{P(N_{i-1},\ N_i)}{\sum_{N_i} P(N_{i-1}, N_i)} \tag{2.9}$$

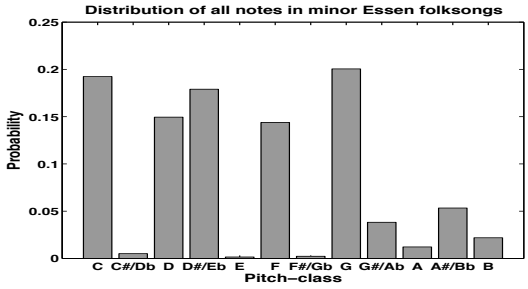
The resulting conditional distributions are displayed in Figure 2.6. Musical structure that is not apparent in the joint distributions becomes visible in these transition distributions. For example, the transition probabilities for each mode tends to concentrate along a diagonal band favoring intervals of a third or less over larger leaps.

There exist several alternatives for specifying the prior distribution. Following the principle of maximum entropy discussed in Chapter 4, if we know or wish to assert nothing about the first note of the piece, we choose a uniform distribution over all

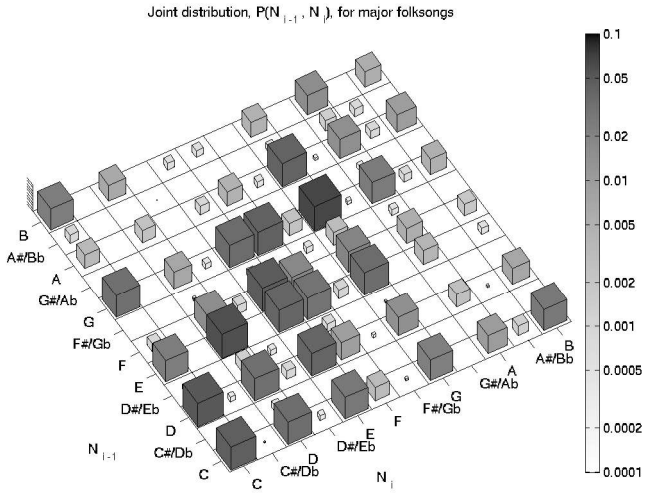
[5] For reasons discussed in Section 3.4.3, distributions were adjusted so that all entries are strictly positive; a small value ($\epsilon = 2{\cdot}10^{-16}$) was added to each event count just prior to normalizing.



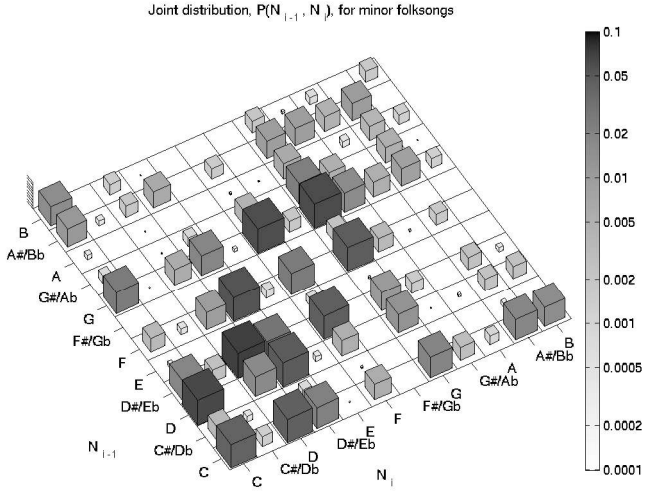
(a) Distribution of all notes in major-key folksongs



(b) Distribution of all notes in minor-key folksongs

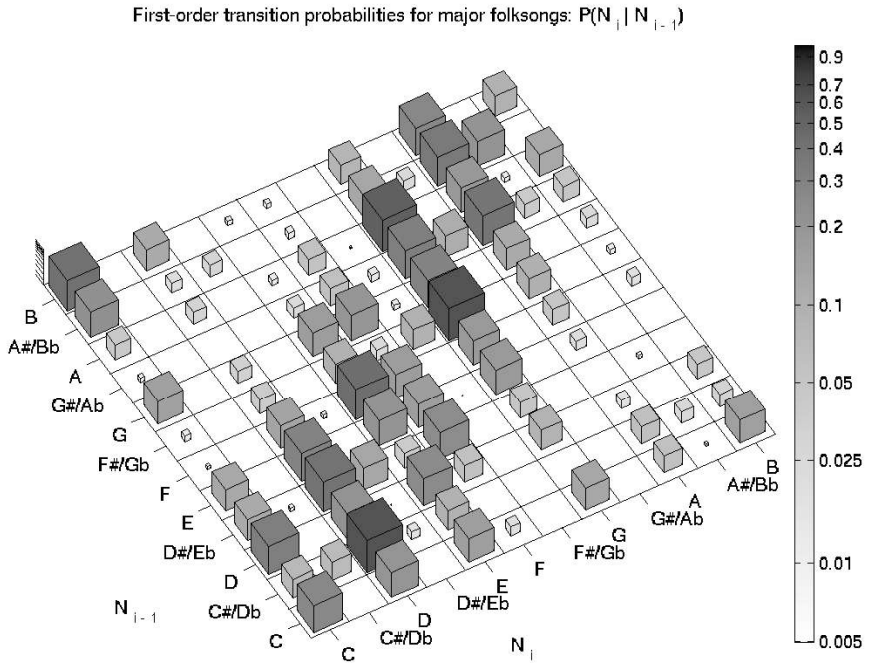


(c) Joint distribution $P(N_{i-1}, N_i)$ for major-key Essen folksongs

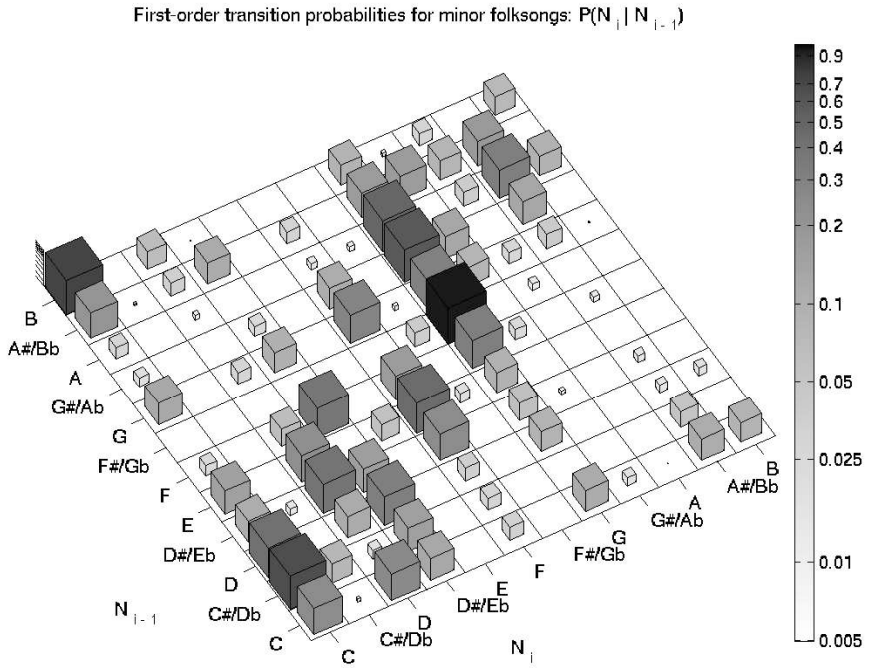


(d) Joint distribution $P(N_{i-1}, N_i)$ for minor-key Essen folksongs

Figure 2.5: Note probabilities and joint distribution of adjacent notes in Essen folksong collection. Colors and box sizes vary logarithmically according to the scales in the colorbars; probabilities less than the minimum value of each colorbar are not displayed. Numeric values are given in Tables A.1–A.6.



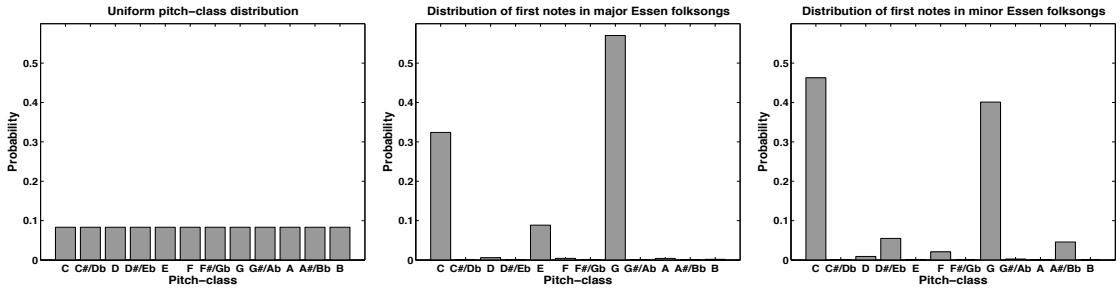
(a) First-order note transition distribution for Essen folksongs in major keys



(b) First-order note transition distribution for Essen folksongs in minor keys

Figure 2.6: First-order transition distributions are obtained by applying (2.9) to the joint distributions in Figures 2.5c and 2.5d. Colors and box sizes vary logarithmically according to the scales in the colorbars; probabilities less than the minimum value of each colorbar are not displayed. Numeric values are given in Tables A.7 and A.8.

pitch classes; Figure 2.7a displays this uniform prior. Because we have access to a corpus of data, a second candidate for the prior can be obtained by by looking at just the first notes of all of the folksongs in a given mode, and normalizing the pitch class counts to form a distribution; these first-note distributions for major and minor modes are displayed in Figures 2.7b and 2.7c.



(a) Uniform prior

(b) $P(N_1)$ calculated from major folksong data

(c) $P(N_1)$ calculated from minor folksong data

Figure 2.7: Uniform pitch-class distribution and mode-specific distributions obtained by counting first-note occurrences in the Essen folksong collection. Numeric values for (b) and (c) are given in Tables A.3 and A.4.

A third possibility is to explicitly identify the starting state of the piece by concentrating all the probability in $P(N_1)$ on a single note. Yet another alternative is to choose $P(N_1)$ to be the *stationary distribution* of the Markov chain, a property described in Section 3.4.3 that governs a Markov model's long-term behavior. The following section details the more immediate task of predicting just a single note ahead at any time in a piece of music.

2.2.3 Forming expectations and assessing realizations

Recall that our first-order Markov model actually represents the full joint distribution over all notes $N_{1:K}$. When using the system to study musical expectations, we can therefore introduce a subset of the notes into evidence, then query the system about the probability of any event given those observations. This allows us to perform arbitrary queries, such as $P(N_{16}|n_1 = \mathrm{E}, n_2 = \mathrm{D}, n_3 = \mathrm{C})$, but, more interestingly, we can construct simulated hypothetical listening situations with direct analogies to real-world music listening.

One such mode of listening is to predict the note one time step in the future, given all notes observed thus far: $P(N_{i+1}|n_{1:i})$, which in a first-order Markov chain equals $P(N_{i+1}|n_i)$. This note-to-note prediction can be implemented as an *online* listening task, because the system recursively updates its beliefs as each new note arrives, allowing the note index i to be a semi-infinite sequence. Suppose that before we hear the first note, we have no idea what its pitch will be, so we let $P(N_1)$ be a uniform distribution over all notes, as displayed in Figure 2.7a.[6] In this case, we would be equally surprised by any choice of n_1. On the other hand, if we knew the piece would be in a minor key and let $P(N_1)$ be the distribution displayed in Figure 2.7b, we would be quite surprised if $n_1 = \mathrm{E}\flat$. There exists an information-theoretic measure, appropriately termed *surprisal*, that numerically quantifies the surprise associated with an event's occurrence [90]. The surprisal of an event x depends only on its own probability $P(x)$:

$$\text{Surprisal}(x) = \frac{1}{\log_2 P(x)} \;=\; -\log_2 P(x) \;\text{ bits} \tag{2.10}$$

If we assume a uniform prior, then the surprisal associated with hearing any n_1 is $-\log_2(1/12) \approx 3.6$ bits. On the other hand, if we assume the minor-key prior, then we are much more surprised by a nondiatonic note: $\text{Surprisal}(N_1 = E\flat) = -\log_2(0.0013) \approx 9.6$ bits.

The *entropy* $H(X)$ of a discrete random variable X is a measure of uncertainty of the variable calculated as the expected value of its surprisal [25]:[7]

$$H(X) = -\sum_{x \in \mathcal{X}} P(x)\ \log_2 P(x)\ \text{ bits} \tag{2.11}$$

The entropy of the uniform note distribution is the same as the surprisal of any individual note: $-\sum_{n \in \mathcal{N}}(1/12)\log_2(1/12) = \log_2 12 \approx 3.6$ bits. Furthermore, this is the maximum entropy achieved by any distribution with 12 values [25]. The entropy

[6] When discussing probabilistic inference in the context of listening applications, we will use the terms "observe", "hear", and "listen to" interchangeably to indicate that a musical attribute was entered into evidence.

[7] As in [25], we adopt the common convention that $0 \log_2 0 = 0$, since $x \log_2 x \to 0$ as $x \to 0$.

of the nonuniform, minor-key prior is therefore less than $\log_2 12$, so knowing the piece is in minor reduces our uncertainty about the first note of the piece (values in this calculation are copied from Table A.4):

$$\begin{aligned} H(N_0) = -\big(&0.463\log_2 0.463 \;+\; 0\log_2 0 \;+\; 0.009\log_2 0.009 \;+\; 0.056\log_2 0.056 \\ &+\; 0.001\log_2 0.001 \;+\; 0.021\log_2 0.021 \;+\; 0\log_2 0 \;+\; 0.401\log_2 0.401 \\ &+\; 0.003\log_2 0.003 \;+\; 0.001\log_2 0.001 \;+\; 0.046\log_2 0.046 \;+\; 0\log_2 0\big) \\ &\approx 1.7 \;\text{ bits} \end{aligned}$$

The dynamics of the Markov chain and the information-theoretic measures of surprisal and entropy provide the tools necessary to move through a melody one note at a time, forming an expectation about the next note, quantifying the uncertainty of that expectation, observing the note, and measuring how surprising the new observation is with respect to our expectations. Each row of Figure 2.9 demonstrates the following sequence of operations by stepping one note at a time through the "Shave and a Haircut" melody displayed in Figure 2.8, assuming a uniform prior and $\mathrm{C}_{\mathrm{Maj}}$ transitions:

1. Obtain a predictive distribution for the note at time step at time $i+1$ by selecting the slice of the first-order transition distribution that corresponds to the observed note at time i: $P(N_{i+1} \mid N_i = n_i)$. For the special case of the first frame, this predictive distribution is simply the prior $P(N_1)$.

2. Compute the entropy of that distribution to quantify the uncertainty of our expectation: $H = -\sum_{n\in\mathcal{N}} P(N_{i+1}{=}n \mid N_i{=}n_i)\ \log_2 P(N_{i+1}{=}n \mid N_i{=}n_i)$ bits

3. Observe the value of the note at time $i+1$, denoted n_{i+1}.

4. Measure the surprise associated with note n_{i+1} by calculating the surprisal of the predictive distribution evaluated at the observed note value n_{i+1}: $-\log_2 P(N_{i+1} = n_{i+1} \mid N_i = n_i)$.

5. Advance to next time index, and repeat steps 1-4 until the end of the melody.

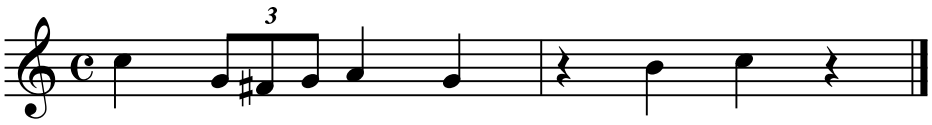
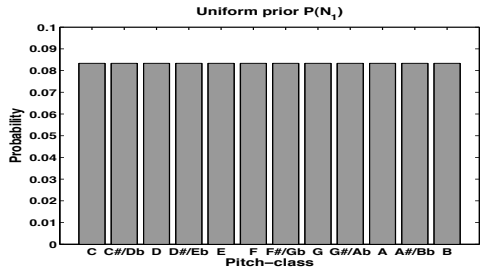


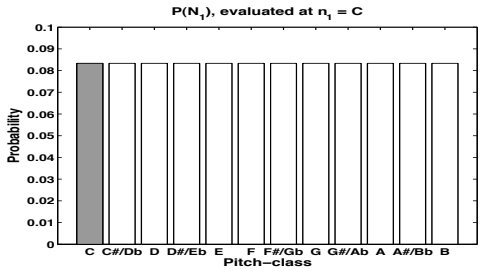
Figure 2.8: "Shave and a Haircut"

Figure 2.10 displays a summary of the entropy and surprisal associated with each time index in this example. As guaranteed by the fact that the prior $P(N_1)$ is distributed uniformly, the predictive distribution with highest entropy is that for the first note of the piece. Note that this ambiguity at the beginning of a piece does not evoke the same kind of emotional response as when a composer intentionally creates a moment of heightened uncertainty during the flow of a piece. The predictive distribution with the least entropy is $P(N_4 \,|\, N_3 = \mathrm{F}^\sharp)$. Figure 2.9g shows that we expect only a single note to follow the $\mathrm{F}^\sharp$. In addition to providing a quantitative measure of ambiguity and surprise at each position in the piece, this process allows us to qualitatively describe characteristics of the predictive distributions. Berger [5] defines a typology of expectations that seems to map well to distributional characteristics. *Specificity* describes the number of simultaneous predictions about an upcoming event, and *strength* is a rating of confidence that the predictions will be realized. The expectation induced by the distribution $P(N_4 \,|\, N_3 = \mathrm{F}^\sharp)$ is both strong and specific. This confidence leads to a firm belief that N_4 will be followed by a G. Any other realization would violate that belief and create a powerful emotional response.

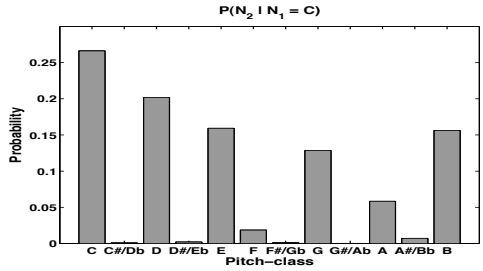
Figure 2.10b shows that the normative realization of that expectation produces the smallest surprisal value in this example. The highest surprisal value in the example is produced by the melodic sequence $N_4 = \mathrm{G} \nearrow N_5 = \mathrm{A}^\flat$, indicating that the occurrence of that pair of notes is not consistent with the system's expectations. As described in Section 2.2.2, the melodies in the Essen folksong corpus tend to be tonally unembellished; the short "Shave and a Haircut" melody used to generate the distributions in Figure 2.9, however, has two nondiatonic pitches, and of the 56908 transitions from the note G, $\mathrm{A}^\flat$ occurs only 65 times. Despite the fact that transition is assigned a high surprisal value, its emotional impact is tempered by the fact that the distribution $P(N_5 \,|\, N_4 = \mathrm{G})$ has relatively high entropy, and the corresponding expectation



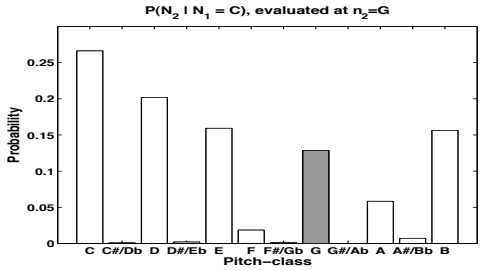
(a) Predicting n_1, steps 1-2: Before hearing the first note, we begin with a uniform prior, with entropy $\log_2 12 \approx 3.6$ bits.



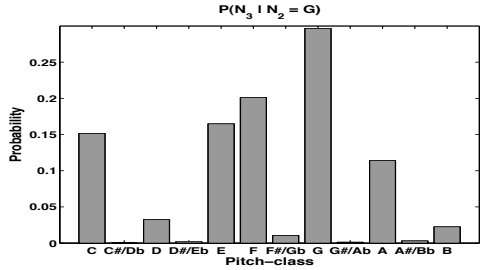
(b) steps 3-4: Look up the value in (a) for which the pitch-class equals C, then calculate the surprisal: $-\log_2 P(N_1 = \text{C}) \approx 3.6$ bits.



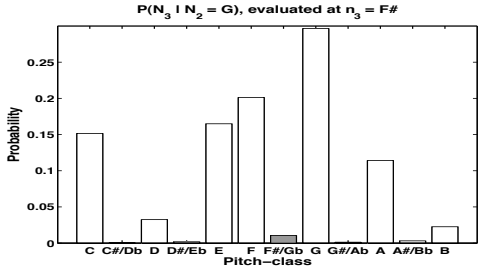
(c) Predicting n_2, steps 1-2: We obtain the appropriate slice of the conditional distribution: $P(N_2|N_1 = \text{C})$. This is equivalent to selecting the bottom row of Table A.7 or Figure 2.6a. The entropy of this distribution is ≈ 2.6 bits.



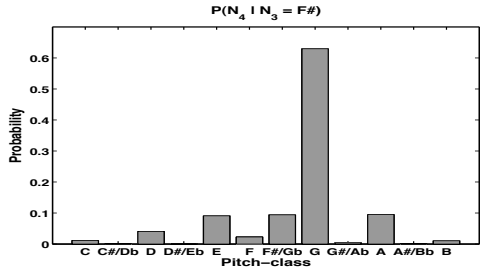
(d) steps 3-4: Look up the value in (c) for which the pitch-class equals G, then calculate the surprisal to be $-\log_2 0.129 \approx 3.0$ bits. This surprise value is very close to the average surprise for the distribution in (c).



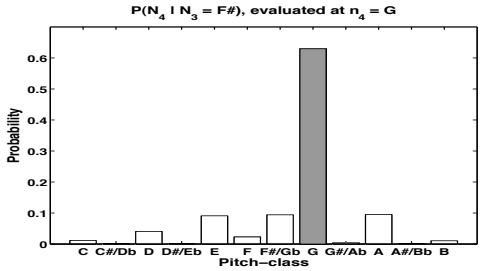
(e) Calculate entropy of $P(N_3|N_2 = \text{G})$ to be ≈ 2.6 bits.



(f) Calculate the surprisal to be $-\log_2 0.01 \approx 6.6$ bits, higher than the entropy of (e), indicating that the F♯ here might sound surprising to us.

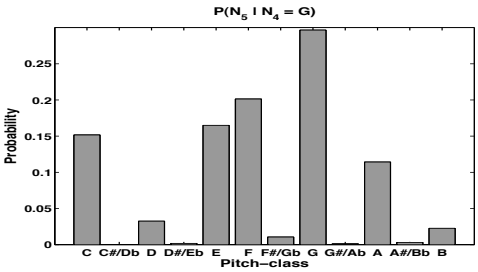


(g) Entropy of $P(N_4|N_3 = \text{F}\sharp)$ is only ≈ 1.9 bits, due to the distribution's concentration on G

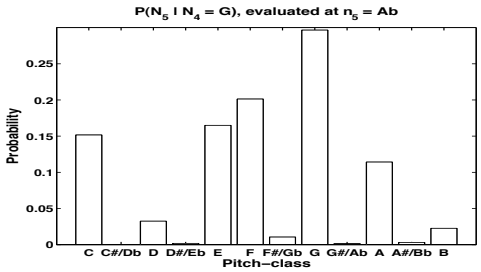


(h) Observing G fulfills our expectations, resulting in a small surprisal of $-\log_2 0.63 \approx 0.7$ bits.

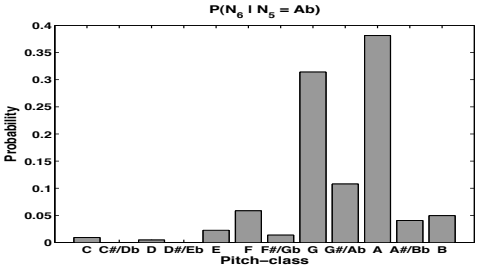
Figure 2.9: Stepping through the note-to-note prediction and realization process



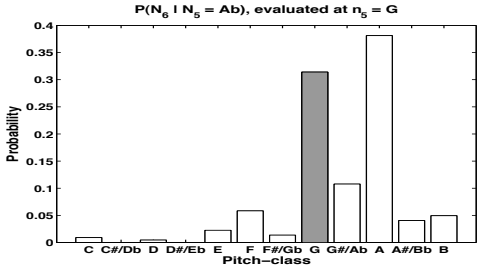
(i) $P(N_5|N_4 = \text{G})$ is the same as in (e), with entropy ≈ 2.6 bits.



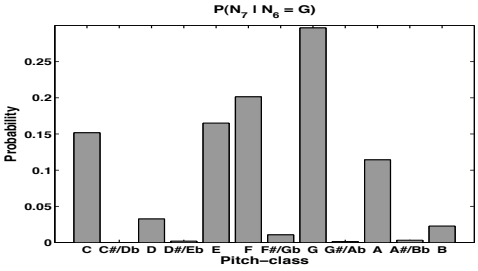
(j) Choice of note $\text{A}^\flat$ is a surprising departure from G: $-\log_2 0.001 \approx 9.8$ bits.



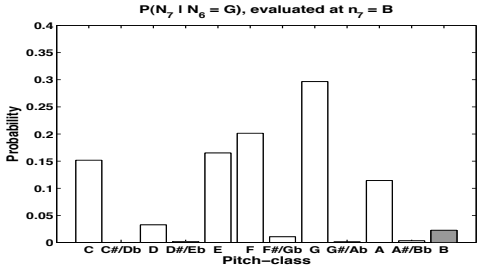
(k) $P(N_6|N_5 = \text{A}^\flat)$ has entropy ≈ 2.3 bits.



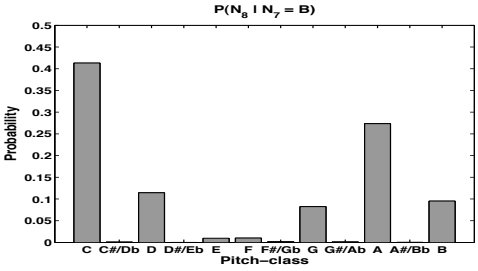
(l) G is the second most probable note given $\text{A}^\flat$, and the surprise associated with it is a relatively modest $-\log_2 0.314 \approx 1.7$ bits.



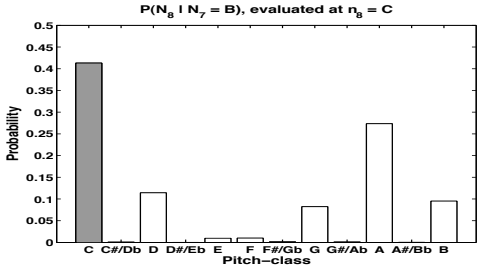
(m) $P(N_7|N_6 = \text{G})$ is again identical to (e) and (i), with entropy ≈ 2.6 bits.



(n) A surprisal value of ≈ 5.5 bits is higher than the entropy of (g).

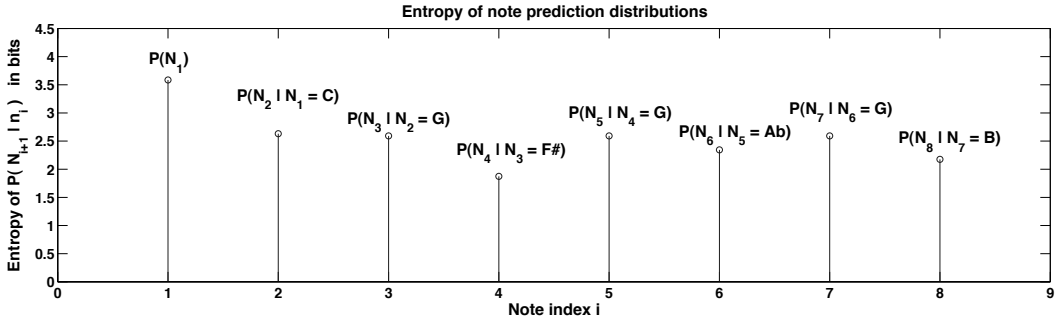


(o) $P(N_8|N_7 = \text{B})$ has entropy of ≈ 2.2 bits.

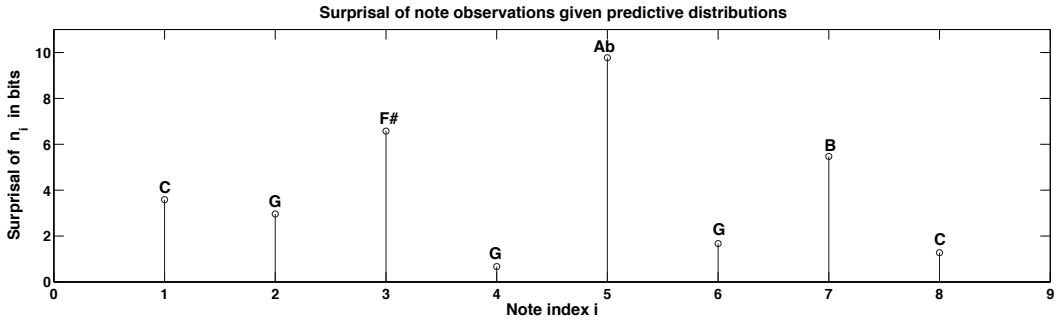


(p) Resolving to the most probable note results in a low surprisal value of ≈ 1.3 bits

Figure 2.9: Stepping through the note-to-note prediction and realization process (cont.)



(a) Entropy values



(b) Surprisal values

Figure 2.10: Summary of entropy and surprisal values for the "Shave and a Haircut" example

is neither strong nor specific (Figure 2.9i). Future research will involve the development of more refined measures of surprise that take into account the strength and specificity properties of predictive distributions.

2.3 Modeling musical context by tiling Bayesian networks over time

While a first-order Markov chain can be used to perform some musically interesting listening tasks, such as predicting an upcoming note and measuring the surprisal of actual observations, music immerses human listeners in a much richer context. The prediction query $P(N_i|N_{i-1})$ is a special case of a general inference query $P(Attribute_i|Context_i)$, where attributes and context involve variables such as note

history, harmony, meter, instrument, composer, lyrics, or any music-related quantity of interest. Similarly, a first-order Markov chain is a special case of a Dynamic Bayesian Network (DBN), a probabilistic structure capable of compactly encoding intricately-related hierarchical musical relationships.

2.3.1 Dynamic Bayesian networks

Consider a discrete-event stochastic process in which Z_i is the set of all variables at the i^{th} event index, with $i \in [1:K]$, and in which Z_i^j is the j^{th} of J such variables at index i. The following definition is taken directly from Murphy [56]. A DBN is defined to be a pair, $(B_1, B_{\rightarrow})$, where B_1 is a BN which defines the prior $P(Z_1)$, and $B_{\rightarrow}$ is a two-slice temporal Bayes net (2TBN) which defines $P(Z_i|Z_{i-1})$ by means of a DAG that factors as follows:

$$P(Z_i|Z_{i-1}) = \prod_{i=1}^{K} P(Z_i^j|\text{Parents}(Z_i^j)) \tag{2.12}$$

The nodes in the first slice of a 2TBN do not have any parameters associated with them, but each node in the second slice has an associated CPD which defines $P(Z_i^j|\text{Parents}(Z_i^j))$, Parents can be in either slice of the 2TBN, and the edge topology is arbitrary, so long as the overall DBN is a DAG. We assume that the model is first-order Markov and time-invariant; if parameters of the CPDs can change over time, we add those parameters to the state space as random variables. The semantics of a DBN can thus be defined by "tiling" or "unrolling" the 2TBN until we have K time-slices. To add a new slice at index $i+1$, a copy of the 2TBN is appended to the DBN in such a way that the nodes of its first slice correspond to the DBN's nodes at time i. This recursive overlapping is the reason that nodes in the first slice of the 2TBN do not have associated CPDs; they merely serve as placeholders used to define dependencies of variables on parents in the previous time index. The resulting joint distribution of the overall DBN is given by

$$P(Z_{1:K}) = \prod_{i=1}^{K} \prod_{J=1}^{J} P(Z_i^j|\text{Parents}(Z_i^j)) \tag{2.13}$$

In the case of the first-order Markov chain in Figure 2.4, there is only one variable, N_i, per time slice, so B_1 contains a single node specifying $P(N_1)$, and $B_{\rightarrow}$ comprises two nodes with the single directed edge $N_{i-1} \rightarrow N_i$ and associated CPD, $P(N_i|N_{i-1})$. Figure 2.11 displays the components B_1 and $B_{\rightarrow}$ unrolled to create a DBN with K time slices.

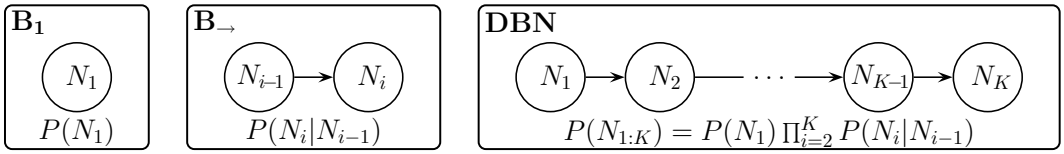


Figure 2.11: Construction of a simple DBN by beginning with the slice for the prior, B_1, then tiling a two-slice temporal Bayes net, $B_{\rightarrow}$, over time.

2.3.2 Adding memory of more than one note in the past

We will now begin to build on the simple first-order Markov chain of notes to construct more expressive musical models. As stated in the preceding section, a DBN is assumed to be a first-order Markov chain of arbitrarily complex Bayesian networks. Many of the music-theoretic rules discussed in Chapter 4, however, require that our model retains a history of more than a single note. Figure 2.12 shows a third-order Markov model in which N_i is probabilistically dependent on three previous notes: N_{i-3}, N_{i-2}, and N_{i-1}. Such a dependence model is commonly termed an *N-gram*, and in this case, the four notes $N_{i-3:i}$ form a 4–gram. The use of N-grams is popular in the fields of natural language processing and speech recognition, where the following state augmentation technique is often employed.

To satisfy the first-order Markov assumption of a DBN while capturing a history of length N_{hist}, we augment the state at each time index to include $N_{hist}-1$ additional variables, and include deterministic connections to propagate the history among them. In the case of a third-order model, $N_{hist} = 3$, so we augment the state N_i to be the set of variables $\{N_i^{(0)}, N_i^{(1)}, N_i^{(2)}\}$. Each variable $N_i^{(j)}$ memorizes the note j time-slices

earlier, $N_{i-j}^{(0)}$, via the following recursion for $j \in [1\!:\!N_{hist}-1]$:

$$P(N_i^{(j)} = n_i^{(j)} | N_{i-1}^{(j-1)} = n_{i-1}^{(j-1)}) = \delta(n_i^{(j)}, n_{i-1}^{(j-1)}) \triangleq \begin{cases} 1 & \text{if } n_i^{(j)} = n_{i-1}^{(j-1)}, \\ 0 & \text{otherwise.} \end{cases} \tag{2.14}$$

This is most easily understood by looking at Figure 2.12, which displays a third-order Markov chain and equivalent state-augmented DBN. The set of note history variables functions as a deterministic delay line, with each time-slice transition shifting notes one position down the delay line. The first-order conditional distribution in the DBN, $P(N_i^{(0)}|N_{i-1}^{(0)}, N_{i-1}^{(1)}, N_{i-1}^{(2)})$ is thus identical to the CPD in the third-order Markov model, $P(N_i|N_{i-1}, N_{i-2}, N_{i-3})$.

2.3.3 Adding hidden states to form a basic autoregressive DBN for modeling musical expectations

Thus far, we have focused only on modeling a sequence of observed notes. We now extend the model to include an unobserved hidden state that in some way affects the note observation sequence. Whereas Section 2.2 assumed that major or minor mode was known, we can add a hidden state, S_i, to our DBN to represent that mode as an unknown random variable. We use the variable X_i to indicate the observed note state, which can comprise either a single note variable or augmented multi-variable note history, as described above. Figure 2.13 displays the resulting DAG, which factors as follows:

$$P(X_{1:K}, S_{1:K}) = P(S_1)P(X_1 \,|\, S_1) \prod_{i=2}^{K} P(S_i \,|\, S_{i-1})P(X_i \,|\, X_{i-1}, S_i) \tag{2.15}$$

At each event index, S_i probabilistically selects which note transition CPD to apply, $P(X_i|X_{i-1}, S_i = \mathrm{C_{Maj}})$ or $P(X_i|X_{i-1}, S_i = \mathrm{C_{min}})$. A state that selects from a finite number of parameters is called a *switching state*. The model in Figure 2.13 is an *autoregressive hidden Markov model* (AR-HMM) [56]; it is termed autoregressive

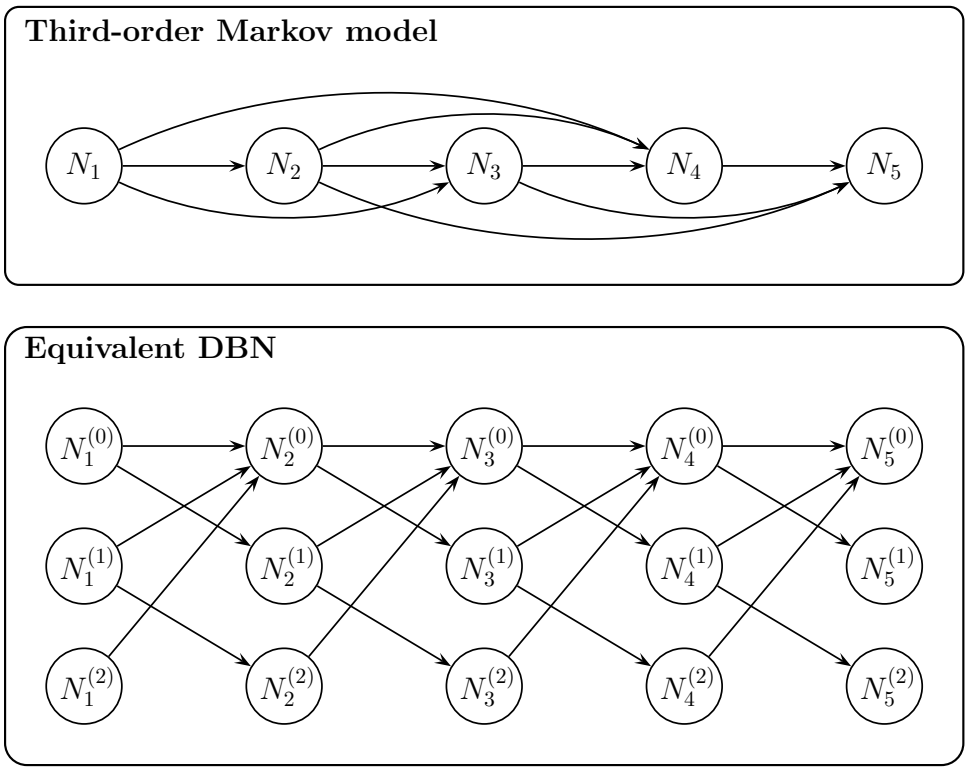


Figure 2.12: DBN representation of a third-order Markov chain. Diagonal edges moving from top-left to bottom-right of the DBN, $N_{i-1}^{(j-1)} \rightarrow N_i^{(j)}$, act as a delay line, deterministically copying note history.

because observations depend on previous observations, and hidden Markov because of hidden states and Markov constraints. A standard hidden Markov model (HMM) [67], displayed in Figure 2.14, is obtained by removing all edges connecting note observations.

In contrast to a standard HMM, in which observations are conditionally independent given hidden states, an AR-HMM naturally facilitates the simultaneous modeling of both bottom-up/data-driven and top-down/schema-driven processes. The relationship between these two types of processes has been studied extensively by researchers of musical expectancy. Notably, the implication-realization model of Narmour [57, 58],

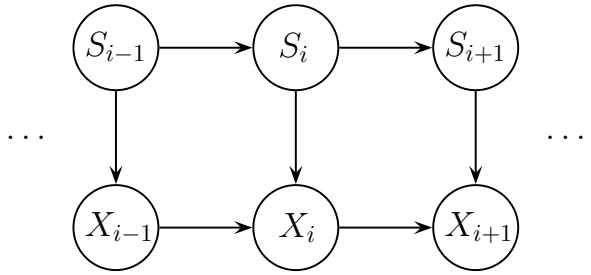


Figure 2.13: Basic autoregressive HMM for modeling musical expectations.

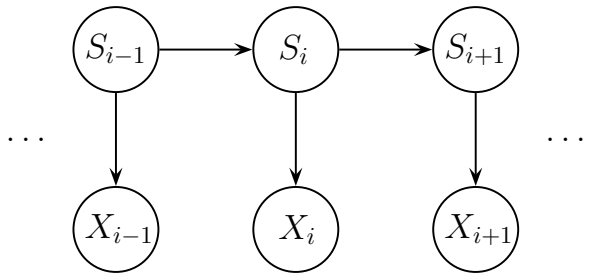


Figure 2.14: Standard HMM, in which observations are conditionally independent given hidden states.

reviewed by Cross [27], defines a set of archetypal melodic patterns that can be used to describe our "innate", "reflexive", "hard-wired" reaction to primitive note-to-note contours. Aarden [1] examines human sensitivity to these local note-to-note transitions using a series of probe tone experiments. In Narmour's model, musical style, which is learned from experience, is applied as a top-down process influences and sometimes dominates the implications of those contours. A cross-cultural study by Krumhansl *et al.* [46] shows that the responses of non-expert listeners tend to reflect a dependence on bottom-up processes, such as Narmour's implication-realization model or simple note-to-note frequency models like the one described in Section 2.2.2, whereas the response of expert listeners additionally demonstrates a reliance on top-down, style-specific models, which shape the contribution of the low-level psychological principles.

The basic model in Figure 2.13 provides a template for building increasingly expressive models of musical expectation capable of fusing information from bottom-up and top-down processes. Subsequent chapters show how to replace the musical mode

variable, S_i, with an arbitrarily-complex hierarchal musical structure, and how to join the note variable, X_i, to layers operating on audio signal features. The following chapter explores how the addition of a hidden layer to the first-order Markov chain enables the modeling of a variety of modes of listening, and details the computations necessary to perform the corresponding types of Bayesian inference.

Chapter 3

Inference Types and Modes of Listening

In addition to modeling both data-driven and schema-driven processes, the structure of the basic musical model in Figure 3.2 lends itself to modeling several human listening tasks. Section 2.2.3 examines the process of forming expectations and assessing realizations in a first-order Markov process containing a single observable note at each time slice. The addition hidden state variables, $S_{1:K}$, enables new types of listening in which the objective is to estimate the value of the hidden state at time index i, given the benefit of observing some time range of notes. Each of these types of listening corresponds to a standard type of Bayesian inference, for which computationally-efficient, recursive algorithms exist.

As in Section 2.2.3, we present the probabilistic algebra necessary to step through a piece one note at a time, predicting both observable and hidden states at future time slices, examining qualities of our predictions, and calculating the surprisal of observations given those predictions. In addition, we show how the system is able to go back and reassess its predictions, updating its belief about a hidden state once it observes additional notes. This step-by-step process is demonstrated using a musical mode identification system whose local probability distributions are specified using the familiar Essen folksong note transitions from Chapter 2.

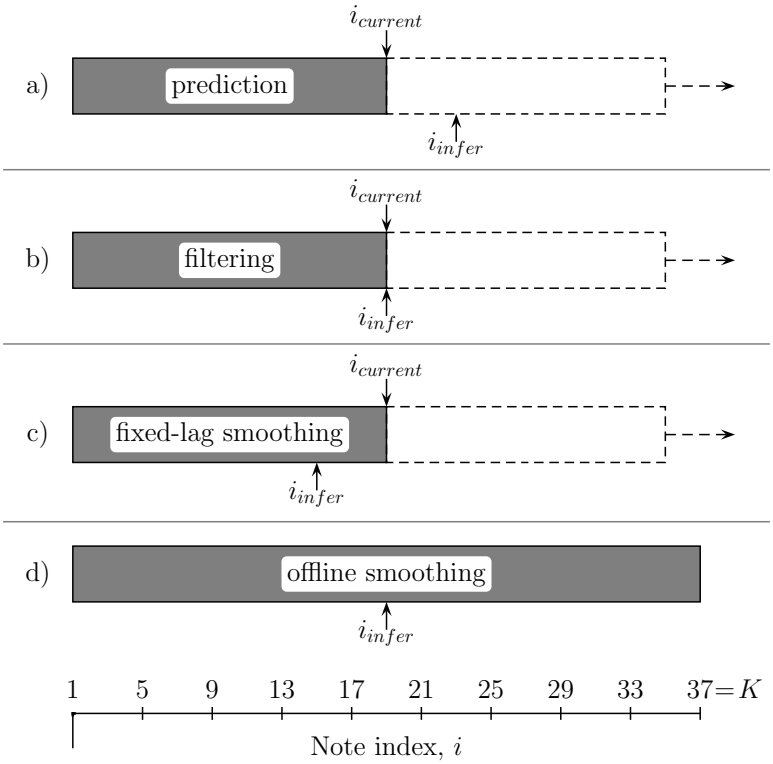


Figure 3.1: Summary of standard Bayesian inference types. Filled-in rectangles indicate the range of observed notes, $i_{current}$ indicates the current listening position, and i_{infer} indicates the note index whose hidden state to be inferred. Because the bottommost subfigure displays ranges for offline smoothing, i_{curent} is not displayed; $i_{current}$ is effectively the last note of the piece, which is in this example indicated as index 37. A similar figure appears in the introductory chapter of Murphy [56].

3.1 Standard Bayesian inference types

Figure 3.1 summarizes the time ranges of a musical piece involved in standard types of Bayesian inference. A recursive algorithm accomplishing these inference types is described in Section 7.3.

Prediction

Figure 3.1a corresponds to a mode of listening in which a person attempts to predict the value of an observable or hidden state at some time in the future. The solid rectangle indicates the observed range of notes, $x_{1:i_{current}}$, where observations are

denoted using a lowercase x. The listener wants to predict ahead to i_{infer}; i.e., $P(S_{i_{infer}} \mid x_{1:i_{current}})$, where $i_{infer} > i_{current}$. The process of predicting more than one time step in the future, when $i_{infer} - i_{current} > 1$, can be accomplished using repeated single-step predictions.

Filtering

Figure 3.1b corresponds to an active, real-time mode of listening in which a person estimates the value of hidden states at the current time, given all notes up to and including the current time; i.e., $P(S_{i_{current}} \mid x_{1:i_{current}})$. Filtering is accomplished by stepping forward through the piece one note at a time, using two phases per time step. First, a *time update* is performed, propagating the internal state one time step from index i to $i+1$, using only the local probability models to predict S_{i+1}. Then a *measurement update* is performed, using the observation x_{i+1} to further adjust the system's belief about S_{i+1}.

Smoothing

Figures 3.1c and 3.1d correspond to retrospective listening in which filtered estimates are revised after hearing additional notes; i.e., $P(S_{i_{infer}} \mid x_{1:i_{current}})$, where $i_{infer} < i_{current}$. A musical example illustrating the benefit of this type of retrospective listing is a pivot chord, which can only be identified in hindsight. Suppose that the hidden state were to contain a pair of variables for chord and musical key. As a listener stepped forward through a piece he or she would likely label the chord as belonging to the estimated key at time $i_{current}$; the filtered estimate would concentrate on that single key/chord combination. After hearing additional notes, however, the listener's belief about chord and key at that earlier pivot point would be adjusted to concentrate on the two corresponding key/chord combinations. Because of musical constructs like the pivot chord, one might want to let several notes "soak in", before making a judgment about hidden states.

Fixed-lag smoothing is a type of *online* inference in which the inference delay, $i_{current} - i_{infer}$, is held constant. One benefit of online inference strategies, such as

fixed-lag smoothing, filtering, and prediction, is that they can be applied to signals of potentially infinite length, because they recursively propagate quantities one step at a time. The downside of fixed-lag smoothing is that computation time increases linearly with the number of lag steps [56]. The *offline* smoothing process depicted in 3.1d, in which the system has access to all observations, does not incur such a computational penalty. During the forward filtering pass, the system stores quantities that are used by a faster smoothing pass that runs backward through the piece. The following section details this forwards-backwards algorithm.

3.2 Prediction, filtering, and smoothing computations for basic expectation model

The basic AR-HMM model displayed in Figure 2.13 is copied here for convenient reference.

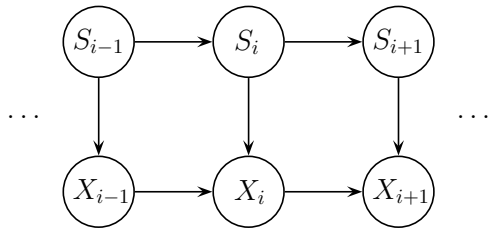


Figure 3.2: Basic autoregressive HMM for modeling musical expectations.

Using the DAG-interpretation skills developed in Section 2.1.2, we read off the model factorization as:

$$P(X_{1:K}, S_{1:K}) = P(X_1 \mid S_1)P(S_1)\prod_{i=2}^{K} P(X_i \mid S_i, X_{i-1})P(S_i \mid S_{i-1}) \tag{3.1}$$

where the DAG has been unrolled to model a melody K notes long.

3.2.1 Forward filtering pass

At the start of each filtering iteration, we assume we have access to $P(S_i \mid x_{1:i})$. This is initialized at the start of the piece by the prior $P(S_1)$. At the end of each iteration, we obtain $P(S_{i+1} \mid x_{1:i+1})$, satisfying the assumption for the next time step. Generally speaking, the filtering process works by starting with $P(S_i \mid x_{1:i})$, then bringing in variables at time $i+1$ and marginalizing out the influence of variables at time i. Specifically, the steps are:

1. Start the time update by bringing in the hidden state at time $i+1$:

$$P(S_i, S_{i+1} \mid x_{1:i}) = P(S_{i+1} \mid S_i) P(S_i \mid x_{1:i}) \tag{3.2}$$

2. Marginalize out S_i to produce a one-step state prediction:

$$P(S_{i+1} \mid x_{1:i}) = \sum_{S_i} P(S_i, S_{i+1} \mid x_{1:i}) \tag{3.3}$$

3. Start the measurement update by introducing the observation distribution (no actual observation at time $i+1$ yet):

$$P(S_i, S_{i+1}, X_{i+1} \mid x_{1:i}) = P(S_i, S_{i+1} \mid x_{1:i}) P(X_{i+1} \mid x_i, S_{i+1}) \tag{3.4}$$

4. Compute a one-step observation prediction by marginalizing out the hidden states:

$$P(X_{i+1} \mid x_{1:i}) = \sum_{S_i, S_{i+1}} P(S_i, S_{i+1}, X_{i+1} \mid x_{1:i}) \tag{3.5}$$

5. Observe x_{i+1}

6. Compute the surprisal of that observation by evaluating the one-step observation prediction $P(X_{i+1} \mid x_{1:i})$ at the observed note x_{i+1}:

$$\text{Surprisal}(x_{i+1} \mid x_{1:i}) = -\log_2 P(X_{i+1} = x_{i+1} \mid x_{1:i}) \tag{3.6}$$

7. Store the following factor, a smoothed two-slice estimate, for use later the backward smoothing pass:[1]

$$P(S_i, S_{i+1} \,|\, x_{1:i+1}) = \frac{P(S_i, S_{i+1}, x_{i+1} \,|\, x_{1:i})}{P(x_{i+1} \,|\, x_{1:i})} \tag{3.7}$$

8. Obtain the filtered posterior, and store it for use later in smoothing:

$$P(S_{i+1} \,|\, x_{1:i+1}) = \sum_{S_i} P(S_i, S_{i+1} \,|\, x_{1:i+1}) \tag{3.8}$$

3.2.2 Backward, offline smoothing pass

To compute the smoothed posterior, we assume that we have access to $P(S_{i+1} \,|\, x_{1:K})$ at the start of each iteration, and use the factors stored during the filtering phase, (3.7) and (3.8), to compute $P(S_i \,|\, x_{1:K})$. We initialize the process with the filtered posterior from the last time index in the piece, $P(S_K \,|\, x_{1:K})$, then step backward note-by-note using the following recursion:

$$P(S_i \,|\, x_{1:K}) = \sum_{S_{i+1}} P(S_{i+1} \,|\, x_{1:K}) \frac{P(S_i, S_{i+1} \,|\, x_{1:i+1})}{P(S_{i+1} \,|\, x_{1:i+1})} \tag{3.9}$$

3.3 Mode identification example

We demonstrate the correspondence between standard types of Bayesian inference and types of listening by creating a hypothetical listener whose goal is to identify the musical mode (Major or minor) of an observed melody. Operating within the context of the basic AR-HMM, we let the hidden state S_i represent the mode at time i, which takes values from the set $\mathcal{S} = \{\text{Maj}, \text{min}\}$. Note observations x_i take values from a set of pitch classes $\mathcal{X} = \{\text{C, C}^\sharp/\text{D}^\flat\text{, D, ..., B}\}$.

In order to complete the model specification, we must quantify the two factors for the prior, $P(X_1 \,|\, S_1)$ and $P(S_1)$, and the two transition factors $P(S_i \,|\, S_{i-1})$ and

[1] This smoothed two-slice estimate is also necessary to perform learning, a process discussed in Chapter 5.

$P(X_i \,|\, X_{i-1}, S_i)$. We assume, for this example, that there is no reason to favor one key over the other at the start of the piece, so we distribute $P(S_1)$ uniformly. We let the other prior factor, $P(X_1 \,|\, S_1)$, be the distributions of first notes in the Essen folksong collection; these first-note distributions are displayed in Figures 2.7b and 2.7b. For the hidden state transition distribution, we assume that at each time index, the probability of changing mode is equal to α_{mode_change}:

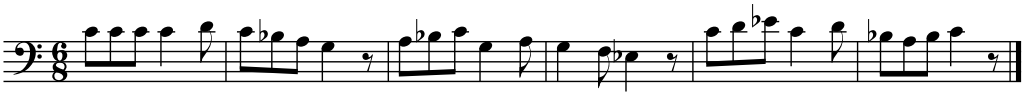
$$P(S_i = s_i \,|\, S_{i-1} = s_{i-1}) = \begin{cases} 1 - \alpha_{mode\_change} & \text{if } s_i = s_{i-1} \\ \alpha_{mode\_change} & \text{otherwise} \end{cases} \tag{3.10}$$

We let $\alpha_{mode\_change} = 0$, constraining the key to be constant over the course of any single melody. The mode state, S_i, is a switching state, determining whether to apply the major-key or minor-key note transition distribution, either $P(X_i \,|\, X_{i-1}, S_i = \text{Maj})$, which is displayed in Figure 2.6a, or $P(X_i \,|\, X_{i-1}, S_i = \text{min})$, which is displayed in Figure 2.6b.

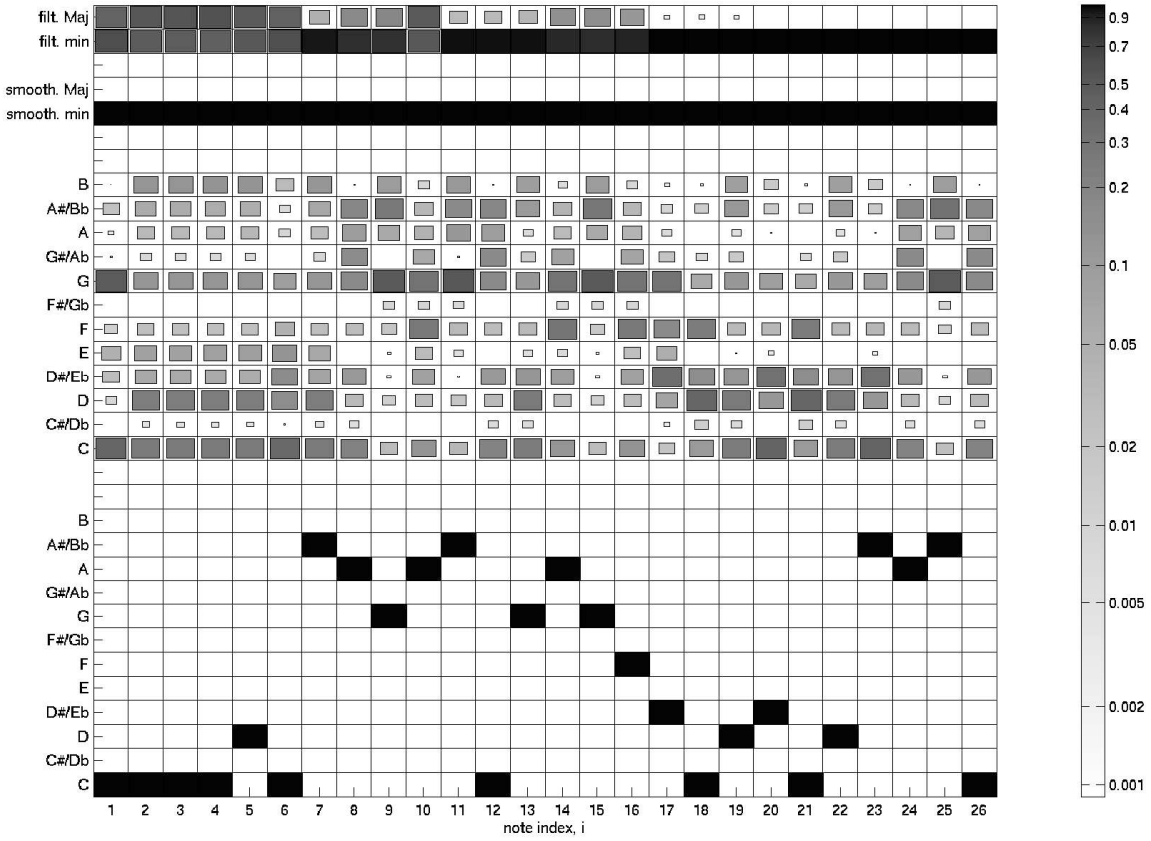
To identify the mode of a melody K notes long, we use the inference steps in (3.2)–(3.9) to compute the smoothed posterior $P(S_i \,|\, x_{1:K})$, which is a sufficient statistic for for the the decision problem which minimizes the expected probability of mode identification errors. Because we've constrained the mode to be constant over the course of the piece, we identify the mode of the piece, $\hat{S}_{1:K}$, by looking at any time slice (we choose the first) and selecting the mode that maximizes the smoothed posterior:

$$\hat{S}_{1:K} = \underset{s \in \mathcal{S}}{\text{argmax}} \;\; P(S_1 = s \,|\, x_{1:k}) \tag{3.11}$$

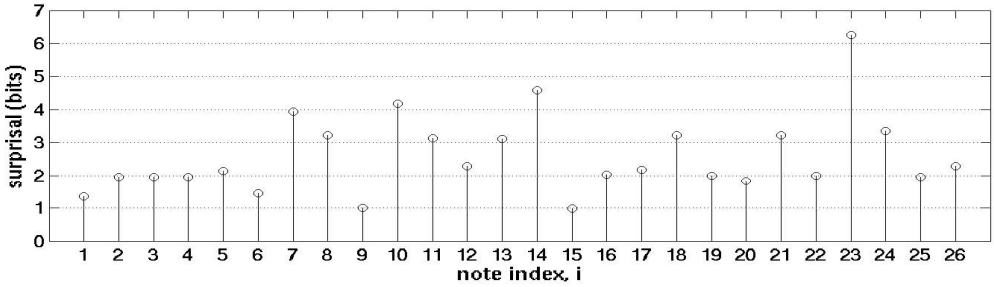
Using this decision rule and the above distributional specifications, the system correctly identifies the mode of 97.6% of the 6219 folksongs melodies. Figure 3.3 uses one of the 149 pieces for which the mode was incorrectly identified to demonstrate how Bayesian inference results reflect the cognitive processes of a hypothetical listener whose objective is to identify the musical mode of a melody. The top subfigure, Figure 3.3a, displays a score of the folksong, transposed down a major-second from its original key, because the header information in the Essen data labels it as being in the



(a) One of the Czech folksongs in the Essen collection, transposed to C_{Maj}.



(b) From top to bottom: filtered and smoothed posteriors (2 rows each), note predictions (12 rows), and piano-roll representation of the song (12 rows)



(c) Surprisal associated with observing note i, given the predictive distribution at time i

Figure 3.3: Example demonstrating the listening experience of a virtual listener whose objective is to identify the musical mode of a melody.

key of $\mathrm{D_{Maj}}$. From top to bottom, Figure 3.3b displays filtered posterior $P(S_i \mid x_{1:i})$, smoothed posterior $P(S_i \mid x_{1:K})$, note prediction distribution $P(X_{i+1} \mid x_{1:i})$, and piano-roll representation of the score. Notes are equally-spaced in time in that representation, because the musical model does not account for note durations; Chapter 7 extends this model to incorporate duration, meter, and beat position. The bottom subplot, Figure 3.3c, displays the surprisal associated with observing the note at time i, given the predictive distribution displayed for that time.

Figure 3.3b and 3.3c summarize a scenario in which a hypothetical listener updates its beliefs about the current musical mode as each new note in the song is heard, then which at the end of the piece uses all of the notes in the melody to reassesses its earlier beliefs. At the start of the piece, the listener's prediction $P(X_1)$ is determined by the prior, which equally weights major-key and minor-key first-note probabilities. Because this prediction most favors the notes C and G, when x_1 is in fact the note C, it is not surprising, as indicated by the first value in Figure 3.3c. Having observed that note, the listener believes that song has a somewhat greater probability of being in a minor key, $P(S_1 = \min \mid x_1 = \mathrm{C}) = 0.59$, because the note C begins minor-key pieces more frequently than it begins major-key pieces.

Observing the following five notes, $x_{2:6}$ does little to confirm the mode of the piece, because the filtered posterior is about uniform, with the maximum value trading back and forth between modes. The observation $x_7 = \mathrm{B}^\flat$, however, is surprising (3.9 bits), and immediately shifts the listener's belief about mode to minor, with $P(S_7 = \min \mid x_{1:7}) = 0.95$. Thinking that the piece is now in minor, the listener is again somewhat surprised that $x_8 = \mathrm{A}^\natural$, and that note reduces the listener's certainty about minor mode to $P(S_8 = \min \mid x_{1:8}) = 0.83$. After observing $x_9 = \mathrm{G}$ the listener is very surprised (4.6 bits) that $x_{10} = \mathrm{A}^\natural$; the note prediction distribution has $P(X_{10} = \mathrm{A} \mid x_9 = \mathrm{G}) = 0.07$. This causes the listener to back off its certainty about minor mode, so that its belief is split again almost exactly uniformly between the two modes.

As earlier in the piece, hearing $x_{11} = \mathrm{B}^\flat$ causes the listener to believe strongly that the piece is in a minor key, and then the note $x_{14} = \mathrm{A}^\natural$ again backs the listener's certainty about minor mode back down to $P(S_{14} = min \mid x_{1:14} = \mathrm{C}) = 0.86$. Three notes later, hearing $x_{17} = \mathrm{E}^\flat$ increases the listener's certainty of minor all the way to 0.997,

and then $x_{20} = \mathrm{E}^{\flat}$ further increases the certainty to 0.99996. The most surprising note of the piece is $x_{23} = \mathrm{B}^{\flat}$, because $P(X_{23} = \mathrm{B}^{\flat} \mid x_{22} = \mathrm{D}) = 0.014$. Although surprising in a minor key, that transition is even more surprising in a major key, so x_{23} reinforces the listener's belief that the song is in a minor key. The listener is so confident, in fact, that that the motion back to $\mathrm{A}^{\natural}$ at x_{24} only very slightly decreases its belief, and by the end of the piece, $P(S_{26} = \min \mid x_{1:26}) = 0.9999996$.

With the benefit of observing the entire song, and knowledge that the mode is constant across the entire song, the listener revises its belief about earlier hidden states, assinging probability very near one to minor mode at all time indices: $P(S_i = \min \mid X_{1:26}) = 0.9999996$, $\forall i \in 1:26$. Figure 3.3b shows that the smoothed posterior concentrates on only minor mode for all time indices. Using this smoothed posterior, we choose the mode of the entire song to be minor: $\hat{S}_{1:26} = \mathrm{argmax}_{S_1} P(S_1 \mid X_{1:26}) = \min$.

Looking at the score in Figure 3.3a, we see that between major or minor mode, minor seems to be the correct choice, because the song contains no occurrences of either $\mathrm{E}^{\natural}$ or $\mathrm{B}^{\natural}$. The song is actually in dorian mode, a scale with semitone steps numbering 2-1-2-2-2-1-2. For modes other than major or minor, the editors of the Essen collection seem to consistently choose the key signature corresponding to the parallel major mode and then add all accidentals. By inspection, most of the 149 songs for which the system incorrectly identified mode are not actually in either major or minor mode. Any future research we do using the Essen folksong collection will take this into account.

3.4 Predictions and stationarity

In order to predict the value of the hidden state more then one time index into the future, one simply executes the corresponding number time updates, repeating the sequence of steps (3.2) and (3.2). This section takes a diversion to discuss properties of long-term predictions and define terms that are necessary prerequisites for understanding our justification of the maximum-entropy rule-encoding framework

presented in Chapter 4. Readers familiar with the conditions under which a discrete-valued Markov chain admits a stationary distribution may wish to scan or skip over this section.

3.4.1 Longer-term predictions in a first-order Markov chain

To simplify this discussion, we return to the simple first-order Markov chain of single notes, displayed in Figure 2.4, operating only on a single key at a time, and characterized by the factorization:

$$P(N_{1:K}) = P(N_1) \prod_{i=2}^{K} P(N_i \mid N_{i-1}) \tag{3.12}$$

Section 2.2.3 discussed the process of predicting a single note ahead in this Markov chain, obtaining $P(N_{i+1} \,|\, n_i)$. We now consider how to predict a note at any time in the future, $P(N_{i+h} \,|\, n_{1:i})$, where $h \in \mathbb{Z}^+$. Whereas a one-step prediction simply involves selecting a row from the note transition CPT, the lack observations at times $i{+}1, i{+}2, \ldots, i{+}h{-}1$ leads to an iterative prediction process that steps forward from index i to $i{+}h$. The following equations derive the necessary recursion:

$$\begin{aligned}
P(N_{i+h} \,|\, n_{1:i}) &= P(N_{i+h} \,|\, n_i) && \text{Markov property} \\
&= \sum_{n \in \mathcal{N}} P(N_{i+h-1} {=} n, N_{i+h} \,|\, n_i) && \text{marginalization} \\
&= \sum_{n \in \mathcal{N}} P(N_{i+h-1} {=} n \,|\, n_i) P(N_{i+h} \,|\, N_{i+h-1} {=} n, n_i) && \text{chain rule} \\
&= \sum_{n \in \mathcal{N}} P(N_{i+h-1} {=} n \,|\, n_i) P(N_{i+h} \,|\, N_{i+h-1} {=} n) && \text{Markov property}
\end{aligned} \tag{3.13}$$

To explicitly trace this process forward through three prediction steps, starting at time i with the observations $n_{1:i}$:

1. predict one note ahead by simply selecting $P(N_{i+1} \,|\, n_i)$ as in Section 2.2.3

2. use that CPD in the calculation that predicts two steps ahead:
   $P(N_{i+2} \,|\, n_i) = \sum_{n \in \mathcal{N}} P(N_{i+1} = n \,|\, n_i) P(N_{i+2} \,|\, N_{i+1} = n)$

3. reuse $P(N_{i+2} \,|\, n_i)$ to predict three steps ahead:
   $P(N_{i+3} \,|\, n_i) = \sum_{n \in \mathcal{N}} P(N_{i+2} = n \,|\, n_i) P(N_{i+3} \,|\, N_{i+2} = n)$

To predict from time i to $i+h$, we thus perform h successive time updates, each of which is the combined act of multiplying the current state of the chain by the transition distribution to obtain the joint distribution $P(N_{i+h-1}, N_{i+h})$, then marginalizing out N_{i+h-1} to advance indices one step forward in the chain.

3.4.2 Stochastic matrix representations and operations

It turns out that if we represent $P(N_{i+h-1} \,|\, n_i)$ as a vector, and the note transition CPT as a *right stochastic matrix*, a square matrix in which each row sums to one, then each time update can be carried out using standard matrix multiplication. If elements of the note space $\mathcal{N}$ are indexed using $\nu_1, \nu_2, \ldots, \nu_{|\mathcal{N}|}$, then $P(N_{i+h-1} \,|\, n_i)$ can be represented as a length-$|\mathcal{N}|$ column vector p_{i+h-1} in which the k^{th} element is $P(N_{i+h-1} = \nu_k \,|\, n_i)$. Similarly, $P(N_{i+h} \,|\, N_{i+h-1})$ can be represented as the $|\mathcal{N}| \times |\mathcal{N}|$ matrix $T_{i+h \,|\, i+h-1}$, in which the element at (k, l) is $P(N_{i+h} = \nu_l \,|\, N_{i+h-1} = \nu_k)$. To create a matrix of this form from the CPDs in Tables A.7 and A.8, each table must be flipped top-to-bottom, because they are presented with the highest pitch on top, matching our sense of pitch height, but opposite the indexing of the columns. A time update then becomes:

$$\begin{aligned} P(N_{i+h} \,|\, n_i) &= \sum_{n \in \mathcal{N}} P(N_{i+h-1} = n \,|\, n_i) P(N_{i+h} \,|\, N_{i+h-1} = n) \\ {p_{i+h}}^{\mathrm{T}} &= {p_{i+h-1}}^{\mathrm{T}} T_{i+h \,|\, i+h-1} \end{aligned} \tag{3.14}$$

where ${p_{i+h}}^{\mathrm{T}}$ denotes the transpose of p_{i+h}, an $|\mathcal{N}|$-element row vector. Furthermore, because of time stationarity, ($T_{i+h \,|\, i+h-1} = T_{2|1}$), the general prediction process involves multiplication by powers of $T_{2|1}$:

$$
\begin{aligned}
{p_{i+1}}^{\mathrm{T}} &= {p_i}^{\mathrm{T}} T_{2|1} \\
{p_{i+2}}^{\mathrm{T}} &= {p_{i+1}}^{\mathrm{T}} T_{2|1} = {p_i}^{\mathrm{T}} T_{2|1}^2 \\
&\cdots \\
{p_{i+h}}^{\mathrm{T}} &= {p_{i+h-1}}^{\mathrm{T}} T_{2|1} = {p_i}^{\mathrm{T}} T_{2|1}^h
\end{aligned} \tag{3.15}
$$

Returning to the "Shave and a Haircut" example, suppose that we observe the first note of the melody, i.e., $P(N_1 = \mathrm{C}) = 1$. Then, if $\mathcal{N}$ is indexed in the order $\mathrm{C}, \mathrm{C}^\sharp/\mathrm{D}^\flat, \ldots, \mathrm{B}$, then ${p_1}^{\mathrm{T}} = [1\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0\,0]$. Let $T_{2|1}$ be the major-mode transition distribution in Table A.7, flipped top-to-bottom so its pitch indexing is consistent. Figure 3.4 displays successive predictions $P(N_{1+h} \,|\, n_1) = p_1 T^h$, as the value of h increases from one to ten notes in the future.

Notice that as h increases, the predictive distributions p_{1+h} start to look identical at every time step. Figure 3.5 shows that as h increases, the process converges to the same predictive distribution even if we do not observe the first note, instead initializing $P(N_1)$ with a uniform prior. In fact, any choice of prior distribution in this example would result in the same type of convergence. The following section explains the conditions under which long-term predictive distributions converge in this way.

3.4.3 Stationary distributions

The *stationary distribution* of a first-order, discrete-valued, time-invariant Markov chain with note transition matrix $T_{2|1}$, if one exists, is a unique distribution p for which:

$$
{p}^{\mathrm{T}} = {p}^{\mathrm{T}} T_{2|1} \tag{3.16}
$$

In other words, ${p}^{\mathrm{T}}$ is a left eigenvector of $T_{2|1}$ associated with the eigenvalue 1, and as $h \to \infty$, $T_{2|1}^h$ converges to a rank one matrix in which all rows equal ${p}^{\mathrm{T}}$.

The distribution is termed a "stationary distribution" because if $P(N_1)$ is chosen to be this special distribution, the resulting stochastic process is a *stationary process*,

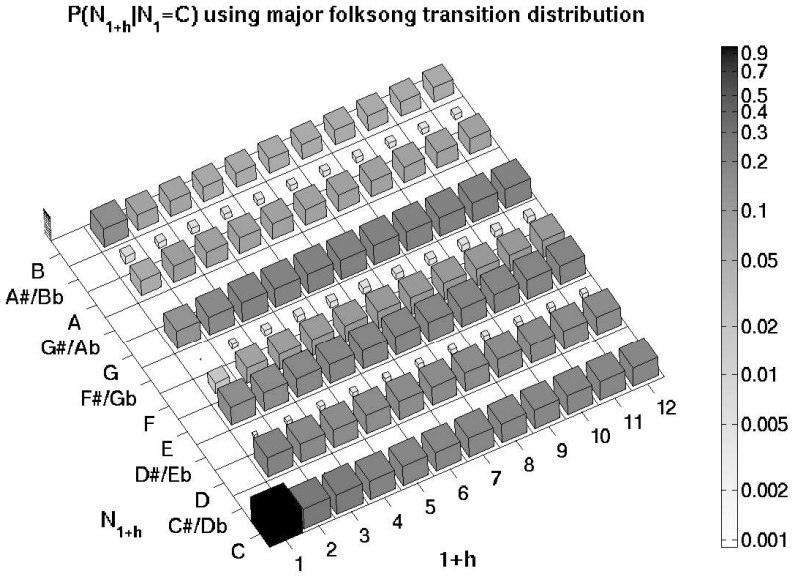


Figure 3.4: Longer-term predictions using major-key Essen folksong transition distribution. We observe the first note, $P(N_1 = \mathrm{C}) = 1$, then move through time, obtaining ${p_2}^{\mathrm{T}} = {p_1}^{\mathrm{T}} T_{2|1}$, ${p_3}^{\mathrm{T}} = {p_2}^{\mathrm{T}} T_{2|1}$, $\ldots$, ${p_{10}}^{\mathrm{T}} = {p_9}^{\mathrm{T}} T_{2|1}$. Note that the predictive distribution at index $h+1$ begins to look identical to the one at index h after only a few time updates.

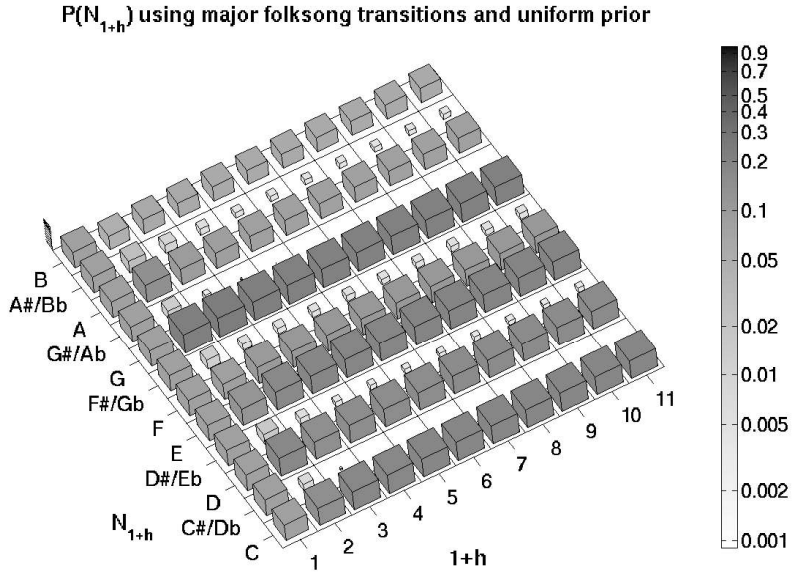


Figure 3.5: Longer-term predictions using major-key Essen folksong transition distribution, starting the process with a uniform prior, $P(N_1)$. As the time interval h increases, predictions converge to the same distribution as when h increases in Figure 3.4

one in which the joint distribution of any subset of the random variables is invariant with respect to shifts in the time index [25]. In this example, stationarity of the process means that:

$$P(n_1, n_2, \ldots, n_K) = P(n_{1+j}, n_{2+j}, \ldots, n_{K+j}) \tag{3.17}$$

for every time index shift j and all combinations of note values $n_{1:K}$.

A Markov chain with transition distribution $T_{2|1}$ admits a stationary distribution if and only if the following two requirements are satisfied:

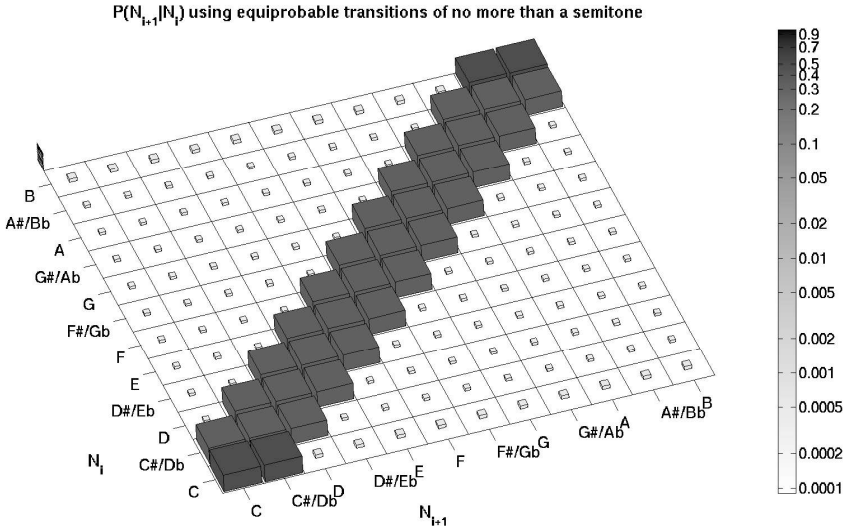
1. The chain is *irreducible*, allowing every note to eventually be reached from every other note with nonzero probability. If we index elements of the transition matrix as above, using subscripts (k, l), then a chain is irreducible if for each pair of indices (k, l) there exists a time index n such that $P([T^n_{2|1}]_{(k,l)}) > 0$.

2. The chain is *aperiodic*, so that note transitions do not cycle in a periodic pattern. A chain is aperiodic if there exists an $N < \infty$ such that $P([T^n_{2|1}]_{(k,k)}) > 0$ for all k and all $n \geq N$.

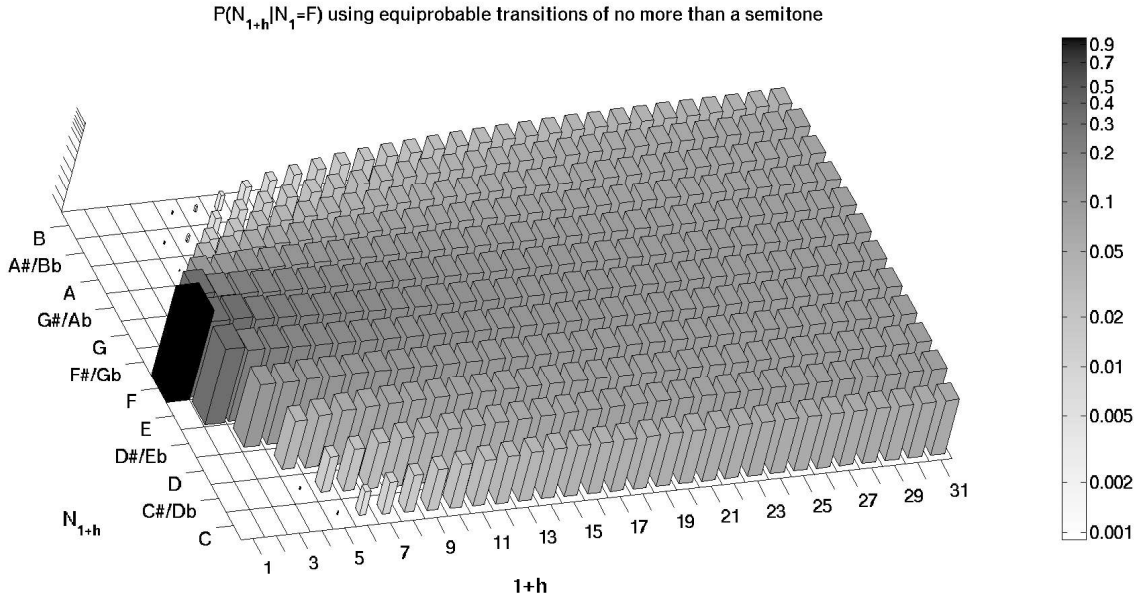
A finite-valued stochastic process having a unique stationary distribution is also an *ergodic* process [55]. An especially simple method of ensuring that the process is ergodic is to specify a transition matrix in which all entries are nonzero [30]. For example, to ensure the ergodicity of processes specified using note transition distributions obtatined from the Essen folksong collection, we added a small nonzero value to each raw transition count just prior to distributional normalization.

We conclude this chapter with one final example demonstrating the stochastic mixing process that occurs as a predictive distribution converges to a stationary distribution. Figure 3.6a displays a note transition matrix $T_{2|1}$ in which notes no greater than a semitone away are chosen with equal probability, and all other entries are chosen with some small probability. Because no entries are zero, the Markov chain is guaranteed to have a unique stationary distribution. Figure 3.6b displays convergence to that stationary distribution, initializing the process with a prior that concentrates on a single note. A stopping rule stating that the convergence process is

complete when no element of the predictive distribution differs from its value in the preceding time slice by more than 0.001 is activated at note index 31. The concept of a stationary distribution and its relationship to hand-coded rules describing musical tendencies will figure prominently in the following chapter.



(a) Transition matrix with equiprobable choices of steps no greater than one semitone. Note that no values in the transition matrix are zero, ensuring that the process admits a stationary distribution.



(b) Convergence to a stationary distribution. Iterations stop when no element of the predictive distribution differs from its value in the preceiding slice by more than 0.001.

Figure 3.6: Convergence of equiprobable stepwise transitions to a stationary distribution.

Chapter 4

Parameterized Maximum Entropy Rate Transition Distributions

The examples in Chapters 2-3 demonstrate that conditional probability distributions learned directly from data can be used in dynamic Bayesian networks to infer unknown musical attributes. We can even look at such a transition distribution and explain how certain musical tendencies are manifested in it. For example, concentration along the diagonal in 2.6a indicates that small melodic intervals tend to be favored over large intervals, and the vertical stripes indicate that diatonic pitches are favored over nondiatonic ones.

Rather than starting with an unstructured CPD and attempting to describe aspects of it, this dissertation develops a novel framework in which we describe a set of individual musical tendencies, then automatically combine them to form structured transition distributions containing all of the intricacy of their interaction. Because this constructive approach supplies us with handles to individual tendencies, we can create a system capable of inferring activation or violation of specific musical rules. The rule combination framework also gives us a way to easily create virtual listeners aware of different collections of musical tendencies, and to then compare the predictions of each listener. As with an unstructured CPD, we can determine the moments in a piece that are surprising, but the rule structure allows us establish a metric to quantify the degree to which each particular musical assumption has been violated.

Most music theories are inherently incomplete from the standpoint of computational statistics, in that they do not attempt to quantify transition probabilities or assign numerical scores that might easily be interpreted probabilistically. Rather, they typically comprise a set of more general statements about musical tendencies, or a set of prescriptive or prohibitive rules. To give an example, each of the thirteen rules of voice leading reviewed by Huron [38] includes either the words "should" or "should not"; e.g., "7. *Conjunct Motion Rule.* When a part must change pitch, the preferred pitch motion should be by diatonic step. Sometimes this rule is expressed in reverse: 8. *Avoid Leaps Rule.* Large melodic intervals should be avoided." Human composers naturally understand how to combine such a set of unquantified rules, and are able to artistically balance, adhere to, or violate them. To be useful in the context of a probabilistic graphical model, however, we ultimately have to quantify exactly what "should" and "should not" actually mean in terms of musical transition probabilities.

To handle such inherently incomplete rule statements, the framework presented in this chapter gives users the ability to encode the general form of a rule, using parameters to control unquantified aspects of the rule. For example, the *Avoid Leaps Rule* might be restated as, "The probability of making a large melodic leap is less than or equal to α_{leap}." We can either assign a value to α_{leap} by hand, if we have a good sense about the strength of the phrase "should be avoided" in the original statement of the rule, or we can use the algorithm presented in Chapter 5 to actually learn the value of α_{leap} from a corpus of musical data.

To encode such parameterized rules, we adopt a maximum entropy philosophy in which we assert everything that we know about a rule, while carefully avoiding asserting anything else. The term "carefully avoiding" is adopted from Jaynes [40], who states:

> the fact that a certain probability distribution maximizes entropy subject to certain constraints representing our incomplete information, is the fundamental property which justifies use of that distribution for inference; it agrees with everything that is known, but carefully avoids assuming anything that is not known.[1]

[1]This passage is quoted in a maximum-entropy natural language processing paper of Berger *et*

Prior to presenting a formal information-theoretic justification for this principle in Section 4.6, we introduce the method using a set of simple examples demonstrating how parameterized rules are encoded as a set of distributional constraints, and how the desire to avoid extraneous assertions leads to conditional probability distributions which are as uniform as possible given those constraints.

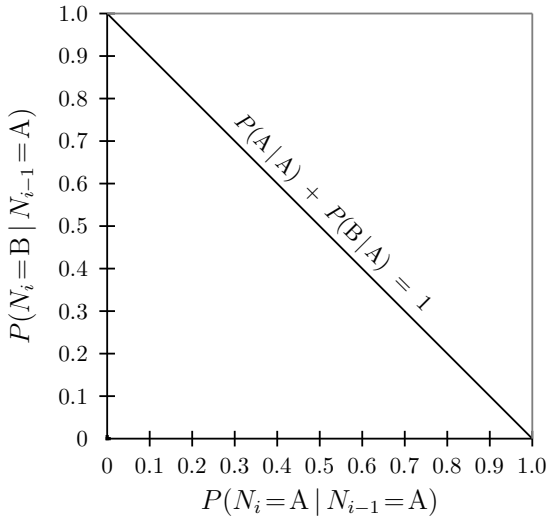
4.1 Representing a parameterized rule as a set of linear constraints

In order to present a simple example demonstrating how a parameterized musical rules can be represented as a set of distributional constraints, this section will again operate within the context of a simple first-order Markov chain (Figure 2.4), using a reduced variable space containing only two notes, $\mathcal{N} = \{\mathrm{A}, \mathrm{B}\}$. Recall that in order to completely define the model, we must specify $P(N_1)$ and $P(N_i \,|\, N_{i-1})$. Suppose that we want to model a virtual listener without the benefit of any musical experience. Such a listener only knows the constraints imposed by the definition of a probability distribution; i.e., that all probabilities are between zero and one, and that distributions sum to one. This set of *simplex constraints* is displayed in Figure 4.1.

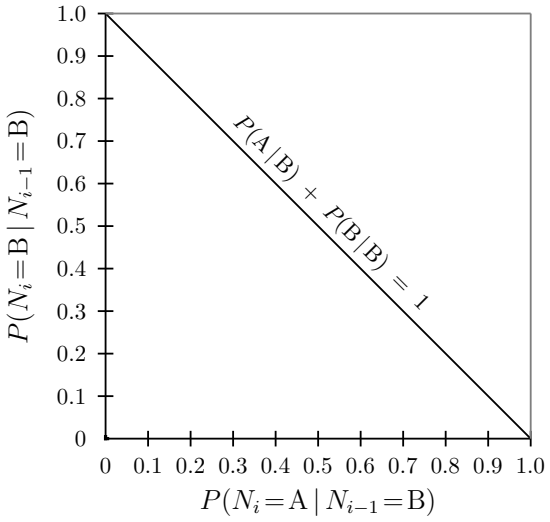
Looking at Figure 4.1, we see that a valid conditional probability table could be generated by choosing any point on each of the two diagonal lines. The logical choice, however, is to assign all transitions equal probability; i.e., $P(N_i \,|\, N_{i-1}) = 0.5$ for all values of N_i and N_{i-1}. Any other choice clearly favors one musical transition over another, violating the assumption that the listener has no knowledge of musical tendencies.

Now suppose that the same listener, through experience, becomes aware of the following musical tendency: "The probability of repeating a note is less than or equal to α_{repeat}." This rule statement can be translated into two linear inequality constraints on the variable spaces depicted in Figure 4.1: $P(N_i = \mathrm{A} \,|\, N_{i-1} = \mathrm{A}) \leq \alpha_{repeat}$ and $P(N_i = \mathrm{B} \,|\, N_{i-1} = \mathrm{B}) \leq \alpha_{repeat}$. For example, if $\alpha_{repeat} = 0.25$, the labeled gray

al. [4].



(a) Simplex constraints ensuring that $P(N_i \mid N_{i-1} = \mathrm{A})$ is a distribution: $0 \leq P(N_i = \mathrm{A} | N_{i-1} = \mathrm{A}) \leq 1$, $0 \leq P(N_i = \mathrm{B} | N_{i-1} = \mathrm{A}) \leq 1$, $\sum_{N_i} P(N_i \mid N_{i-1} = \mathrm{A}) = 1$



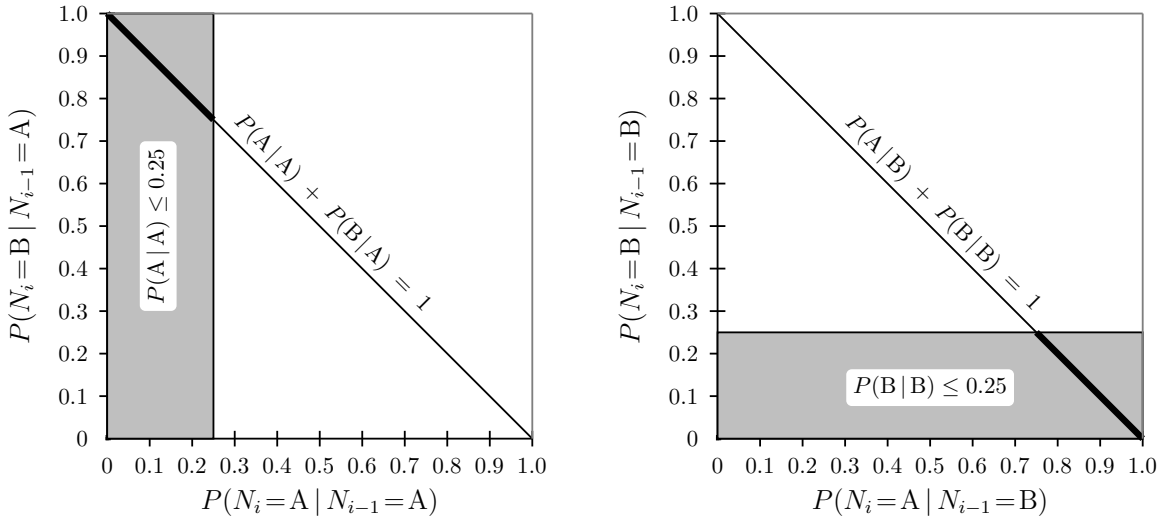
(b) Simplex constraints ensuring that $P(N_i \mid N_{i-1} = \mathrm{B})$ is a distribution: $0 \leq P(N_i = \mathrm{A} | N_{i-1} = \mathrm{B}) \leq 1$, $0 \leq P(N_i = \mathrm{B} | N_{i-1} = \mathrm{B}) \leq 1$, $\sum_{N_i} P(N_i \mid N_{i-1} = \mathrm{B}) = 1$

Figure 4.1: A completely naive listener is only aware of constraints ensuring that values form valid conditional probability distributions. Though each subfigure displays a small window of the space of $\mathbb{R}^2$, the actual probability space lies only along the diagonal line; we show this window to establish a direct correspondence with the three-dimensional plots in Figure 4.3.

rectangles in Figure 4.2 correspond to the regions satisfying the repeat constraints. Valid transition probabilities now lie on the segments of the simplex constraint lines that intersect these gray regions. Choosing a specific point along each of these bold line segments no longer seems as clear-cut of a decision as when any non-uniformity clearly violated assumptions about the listener's complete naiveté. Our decision rule is to choose the values that are as "uniform as possible" given the rule constraints.

4.2 Entropy rate

In the same way that a uniform distribution over a single variable achieves maximum entropy, a uniform transition distribution in the context of a stochastic process achieves the maximum *entropy rate*. Cover and Thomas [25] define the entropy rate



(a) Intersection of simplex constraint and repeat constraint, $P(N_i = \mathrm{A} \mid N_{i-1} = \mathrm{A}) \leq 0.25$

(b) Intersection of simplex constraint and repeat constraint, $P(N_i = \mathrm{B} \mid N_{i-1} = \mathrm{B}) \leq 0.25$

Figure 4.2: Intersection of repeat and simplex constraints, displayed as bold line segments.

of a stochastic process $\{X_i\}$ as:

$$H_r(\mathscr{X}) = \lim_{m \to \infty} \frac{1}{m} H(X_1, X_2, \ldots, X_m) \tag{4.1}$$

where the limit exists. This is the per symbol entropy of the m variables. For a stationary stochastic process, that entropy rate is equal to the conditional entropy of the last random variable given the past:

$$H_r(\mathscr{X}) = \lim_{m \to \infty} H(X_m \mid X_{m-1}, X_{m-2}, \ldots, X_1) \tag{4.2}$$

The entropy rate of a first-order, stationary Markov chain operating on an indexed, discrete variable space $\mathcal{X} = \{x_1, x_2, \ldots, x_{|\mathcal{X}|}\}$, with transition matrix containing entries $T_{k,l} = P(X_i = x_l \mid X_{i-1} = x_k)$, and stationary distribution vector μ containing

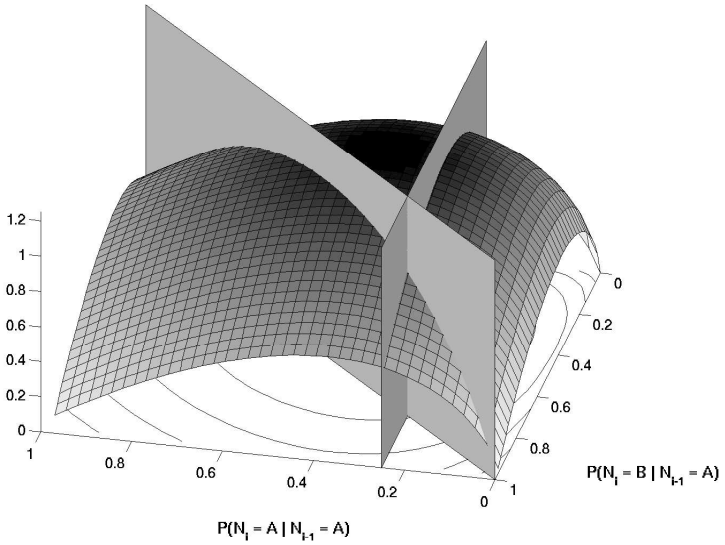
entries $\mu_k = P_{stationary}(x_k) = [T^N]_{1,l}$ for sufficiently large N, is given by:

$$\begin{aligned} H_r(\mathscr{X}) &= \lim_{m\to\infty} H(X_m \,|\, X_{m-1}, X_{m-2}, \ldots, X_1) && (4.3)\\ &= \lim_{m\to\infty} H(X_m \,|\, X_{m-1}) && (4.4)\\ &= H(X_2 \,|\, X_1) && (4.5)\\ &= \sum_k \mu_k \sum_l -T_{k,l} \log_2 T_{k,l} && (4.6)\\ &= -\sum_{k,l} \mu_k T_{k,l} \log_2 T_{k,l} && (4.7) \end{aligned}$$

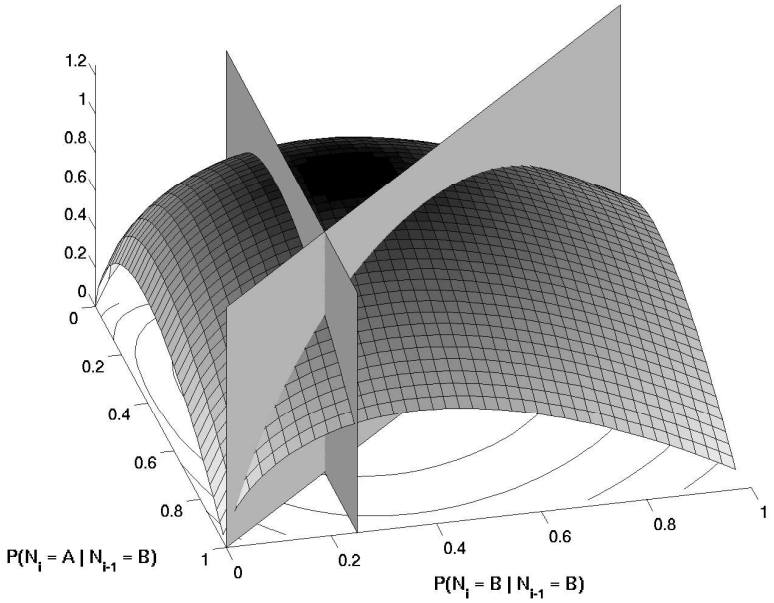
We can use these definitions to revisit the repeat constraints example in Figure 4.2 and choose the points along the bold lines that maximize the entropy rate of the process. The version of the entropy rate calculation in (4.6) matches the format of Figure 4.2 well, in that the outer index k corresponds to the conditional distribution given a single fixed value of N_{i-1}, and each subfigure displays constraints on the conditional distribution given one value of N_{i-1}. This correspondence allows us to directly overlay the entropy function in the inner sum of (4.6) onto the existing constraint plots. Figure 4.3a displays $-\sum_{n\in\{\mathrm{A,B}\}} P(N_i = n \,|\, N_{i-1} = \mathrm{A}) \log_2 P(N_i = n \,|\, N_{i-1} = \mathrm{A})$ as a function of $P(N_i = \mathrm{A} \,|\, N_{i-1} = \mathrm{A})$ and $P(N_i = \mathrm{B} \,|\, N_{i-1} = \mathrm{A})$. Figure 4.3b is a plot of the same form with $N_{i-1} = \mathrm{B}$. In Figure 4.3a, points satisfying the constraints are those on the segment of the diagonal simplex line that are right of the stepwise line, $P(N_i = A|N_{i-1} = A) = 0.25$. Constrained to that segment, the function achieves its maximum value at the intersection of the two constraint lines. Figure 4.3b shows that when $N_{i-1} = B$, the function also achieves its maximal value at the intersection of the two constraint lines.

4.3 Maximizing entropy rate via convex optimization

In the simple example presented in Figure 4.3, there are only four total constraints, and only four entries in the conditional probability table. Suppose that the Markov



(a) $-\sum_{n\in\{\mathrm{A,B}\}} P(N_i = n \,|\, N_{i-1} = \mathrm{A}) \log_2 P(N_i = n \,|\, N_{i-1} = \mathrm{A})$, cut by planes representing simplex and repeat constraints.



(b) $-\sum_{n\in\{\mathrm{A,B}\}} P(N_i = n \,|\, N_{i-1} = \mathrm{B}) \log_2 P(N_i = n \,|\, N_{i-1} = \mathrm{B})$, cut by planes representing simplex and repeat constraints

Figure 4.3: Entropy rate calculations over space of conditional note probabilities, with $\alpha_{repeat} = 0.25$. Plots have been rotated to look from the angle best showing the intersection of constraint regions. Recall that the actual probability space only consists of the points along the diagonal simplex constraint line; we show an extended window of $\mathbb{R}^2$ to stress the concavity of the entropy function in multiple dimensions.

chain instead used an augmented note state memorizing three notes in the past, as discussed in Section 2.3.2: $P(N_i^{(0)} \mid N_{i-1}^{(0)}, N_{i-1}^{(1)}, N_{i-1}^{(2)})$. If the space of notes spanned three octaves, the number of entries in the transition CPT would equal $(12 \cdot 3)^4 = 1679616$. In order to guarantee that the table entries corresponding to $P(N_i^{(0)} \mid N_{i-1}^{(0)}, N_{i-1}^{(1)}, N_{i-1}^{(2)})$ form distributions, we must specify one simplex constraint for every possible combination of $N_{i-1}^{(0)}$, $N_{i-1}^{(1)}$, and $N_{i-1}^{(2)}$; i.e., we need $36^3 = 46656$ simplex constraints. Add to that number of simplex constraints all other known melodic constraints, and we quickly have a seemingly daunting problem involving millions of values subject to hundreds of thousands of constraints. Fortunately, our choice of optimization objective and constraints leads to fast numeric solutions.

Observe in Figure 4.3 that the entropy function is concave. Because all linear functions are convex, the constraints are also convex, so the constrained maximization of the entropy function is a convex problem. Even extremely large convex optimization problems can be solved by one of many available software packages. For example, an article by Fang and Tsao [32] presents, "An efficient computational procedure for solving entropy optimization problems with infinitely many linear constraints." All examples in this dissertation were computed using the primal-dual interior method for convex objectives (PDCO) by Saunders and Tomlin [78]. Boyd and Vandenberghe [15] provide in-depth coverage of convex optimization problems and algorithms. In addition to computational efficiency, another particularly appealing aspect of convex optimization problems is that if any single feasible solution can be found that satisfies the constraints, the optimal solution is guaranteed to be found.

Unfortunately, while maximization of the entropy function in the inner sum of (4.6) is handled extremely efficiently using convex optimization, the stationary distribution μ in the outer sum of (4.6) depends on the transition matrix itself. Due to this circular dependence, it is not clear at this time whether it is possible to cast the overall entropy rate maximization as a convex problem; however, the following iterative approach seems to produce excellent results:

1. Begin by initializing the stationary distribution, μ, as uniform.

2. Use a convex optimization approach to obtain the maximum entropy rate distribution for the current μ.

3. Replace μ with the stationary distribution of the transition distribution obtained in step 2.

4. Repeat steps 2 and 3 until the solution converges.

4.4 Musical examples of constraint types

Whereas the repeat rule constrained only a single value of N_i for each value of N_{i-1}, constraints may in general involve linear combinations of several variables. This section describes several types of musical constraints, and presents a number of simple musical examples demonstrating how maximizing entropy rate produces transition distributions that are uniform as possible given the constraints. This maximal uniformity reflects our guiding principle that we should encode everything we know about musical tendencies without introducing any unintended bias.

4.4.1 Constraining individual entries of the transition matrix

The repeated note rule stated in Section is an example of a rule that can be encoded using several constraints that individually constrain only a single entry in the transition matrix. For each value of $n \in \mathcal{N}$, the encoding of the repeated note rule constrains one entry of the transition table, such that $P(N_i = n \mid N_{i-1} = n) \leq \alpha_{repeat}$.

We expand on that repeated note example by examining optimization results when the space of notes is extended from two notes to the thirteen notes in the range C_4 to C_5. As before, the model for a completely naive listener corresponds to the uniform transition matrix displayed in Figure 4.4a. Figures 4.4b-4.4d display the maximum entropy rate distribution corresponding to three different values of α_{repeat}. As shown in Figure 4.3, the optimization solutions lie at the intersection of the simplex and repeat constraints, so the values along the diagonal in Figure 4.4b equal 0.01, and values along the diagonal in Figure 4.4c equal 0.05. Figure 4.4d is different, however,

in that it shows that the constraint $\alpha_{repeat}=0.1$ does not affect the optimal transition matrix; the maximum entropy rate solution with $\alpha_{repeat} = 0.1$ is identical to the unconstrained solution, because $0.1 > 1/13$, so the uniform transition table with all entries equal to $1/13$ satisfies the inequality constraints. While it is difficult to display a function of thirteen variables, we can display a similar example for the two-note space; Figure 4.5 shows that a setting of $\alpha_{repeat} = 0.6$ has no effect, because the constraint satisfaction region in each plot contains the maximum-entropy, uniform point at which all transition probabilities equal 0.5. Section 4.5 demonstrates that while adding a specific constraint by itself might not alter the optimization solution, that constraint may effect the solution when combined with other constraints.

Given the form of the repeat rule, the stationary distribution is uniform regardless of the choice of α_{repeat}, so the iterative optimization approach converges after a single iteration. For example, in the two-note case, where diagonal elements of the optimal solution equal $\alpha^*_{repeat} = \min(0.5, \alpha_{repeat})$:

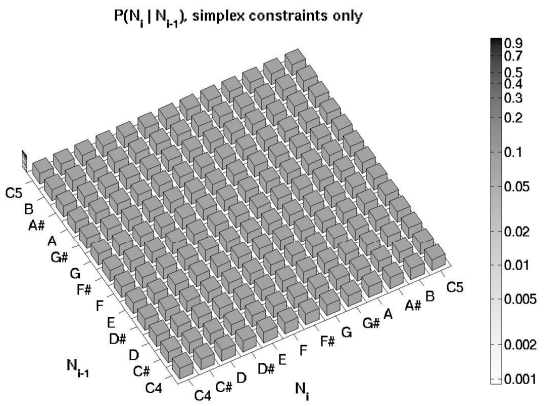
$$[0.5 \;\; 0.5] \begin{bmatrix} P(\mathrm{A}|\mathrm{A}) = \alpha^*_{repeat} & P(\mathrm{B}|\mathrm{A}) = 1 - \alpha^*_{repeat} \\ P(\mathrm{A}|\mathrm{B}) = 1 - \alpha^*_{repeat} & P(\mathrm{B}|\mathrm{B}) = \alpha^*_{repeat} \end{bmatrix} = [0.5 \;\; 0.5] \tag{4.8}$$

4.4.2 Constraining the sum of entries relative to a value

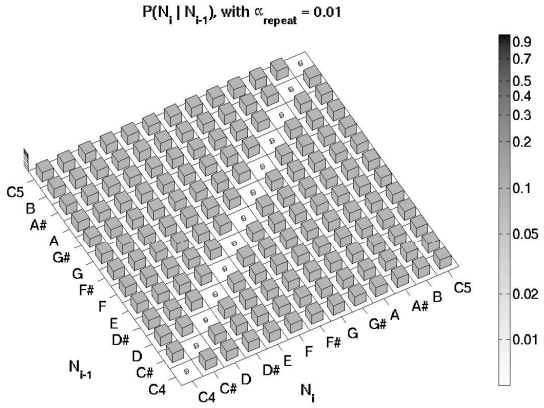
A second type of constraint is to constrain the sum of entries in the transition matrix relative to some value. For each transition matrix, we encode $|\mathcal{N}|$ simplex constraints, ensuring that $\sum_{N_i} P(N_i \,|\, N_{i-1}) = 1$, for each value of N_{i-1}.

To present another example of this constraint type, we encode a rule governing the probability of nondiatonic notes. Assume that the key is fixed as $\mathrm{C_{Maj}}$, and consider the rule statement, "Regardless of the value of N_{i-1}, the probability that N_i is nondiatonic is less than or equal to $\alpha_{nondiatonic}$." That is, if the set $\mathcal{A}(\mathrm{C_{maj}})$ contains all nondiatonic notes in $\mathrm{C_{Maj}}$, then the nondiatonic rule can be interpreted as the set of $|\mathcal{N}|$ constraints, one for each value of N_{i-1}:

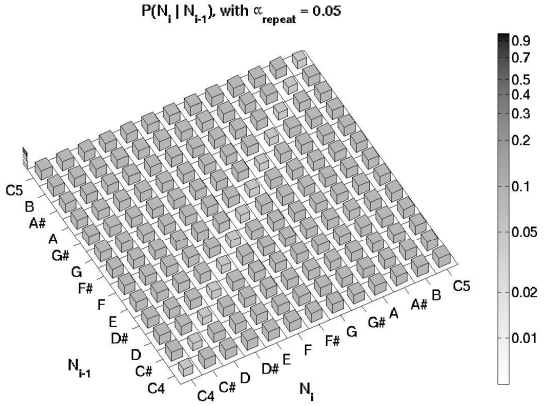
$$\sum_{n \in \mathcal{A}(\mathrm{C_{maj}})} P(N_i = n \,|\, N_{i-1}) \leq \alpha_{nondiatonic} \tag{4.9}$$



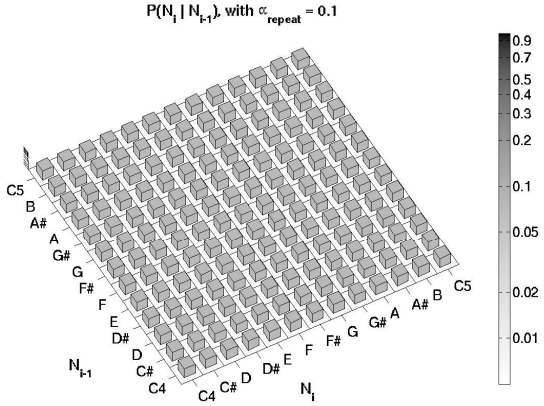
(a) Only simplex constraints result in completely uniform note transition matrix



(b) Note transition distribution with $\alpha_{repeat} = 0.01$



(c) Note transition distribution with $\alpha_{repeat} = 0.05$



(d) Note transition distribution with $\alpha_{repeat} = 0.10$ is completely uniform, because $0.10 > 1/13$.

Figure 4.4: Maximum entropy rate transition distributions corresponding to three different values of α_{repeat}.

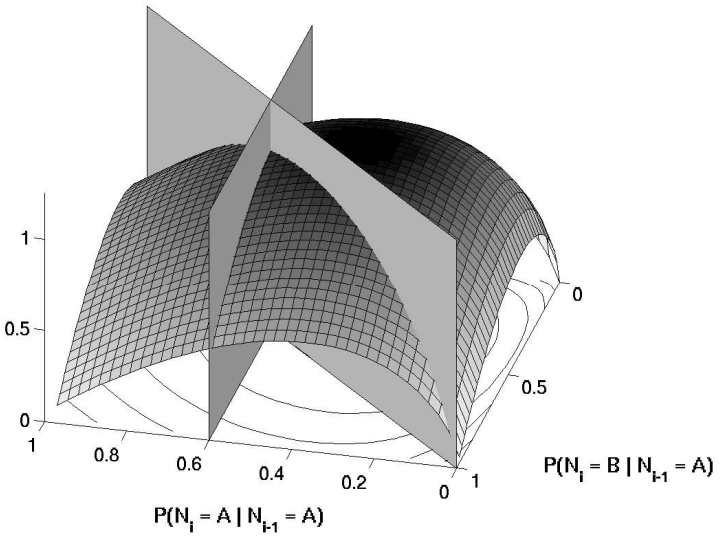
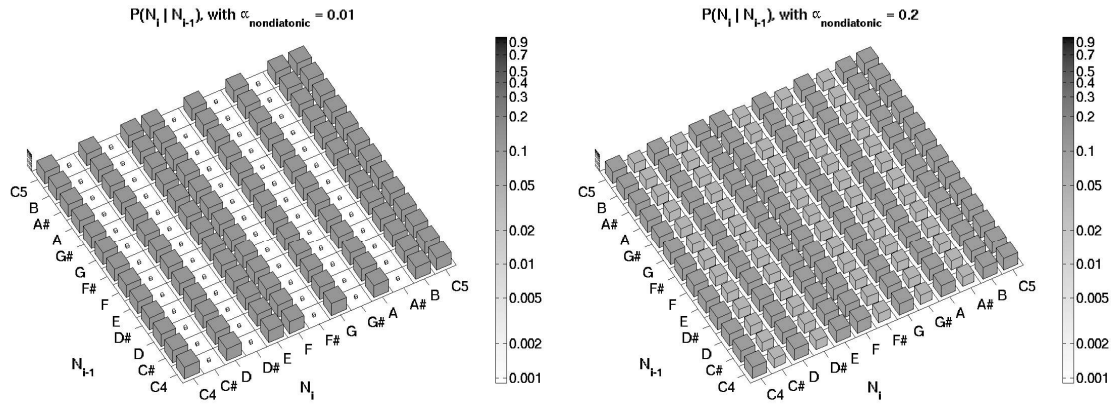


Figure 4.5: A constraint having no effect on the optimization solution. With $\alpha_{repeat} = 0.6$, all points on the simplex line to the right of $P(N_i = \mathrm{A} \mid N_{i-1} = \mathrm{A})$ satisfy the constraints. Along this segment, the function achieves its maximum value at $(0.5, 0.5)$, which is the same point as without the repeat constraint.

Figure 4.6 displays the maximum entropy rate distributions obtained using two different values of $\alpha_{nondiatonic}$.

4.4.3 Constraining the sum of entries relative to the sum of other entries

A third type of constraint involves comparisons between two sets of entries in the transition matrix. For example, the *Conjunct Motion Rule* stated in the introduction to this chapter, "When a part must change pitch, the preferred pitch motion should be by diatonic step", might be restated, "If the pitch changes, the probability of moving by diatonic step is greater than or equal to α_{step} times the probability of moving by all other intervals." We define $\mathcal{D}_{step}(k, n)$ to be the set of all notes one diatonic step away from the note n in the key k, and $\mathcal{D}_{non\_step}(k, n)$ to be all other notes where the pitch changes; i.e., $\mathcal{D}_{non\_step}(k, n) = \mathcal{N} \setminus (\{n\} \cup \mathcal{D}_{step}(k, n))$. For example, $\mathcal{D}_{step}(\mathrm{C_{Maj}}, \mathrm{F}_4^\sharp) =$



(a) Optimization results for $\alpha_{nondiatonic}=0.01$ (b) Optimization results for $\alpha_{nondiatonic}=0.2$

Figure 4.6: Maximum entropy rate transition distributions given nondiatonic constraints relative the key $\mathrm{C_{Maj}}$.

$\{\mathrm{F}_4, \mathrm{G}_4\}$, and $\mathcal{D}_{non\_step}(\mathrm{C_{Maj}}, \mathrm{F}_4^\sharp) = \{\mathrm{C}_4, \mathrm{C}_4^\sharp, \mathrm{D}_4, \mathrm{D}_4^\sharp, \mathrm{E}_4, \mathrm{G}_4^\sharp, \mathrm{A}_4, \mathrm{A}_4^\sharp, \mathrm{B}_4, \mathrm{C}_5\}$. The conjunct motion rule can then be expressed as a set of $|\mathcal{N}|$ constraints, for each value of N_{i-1}, denoted n_{i-1}:

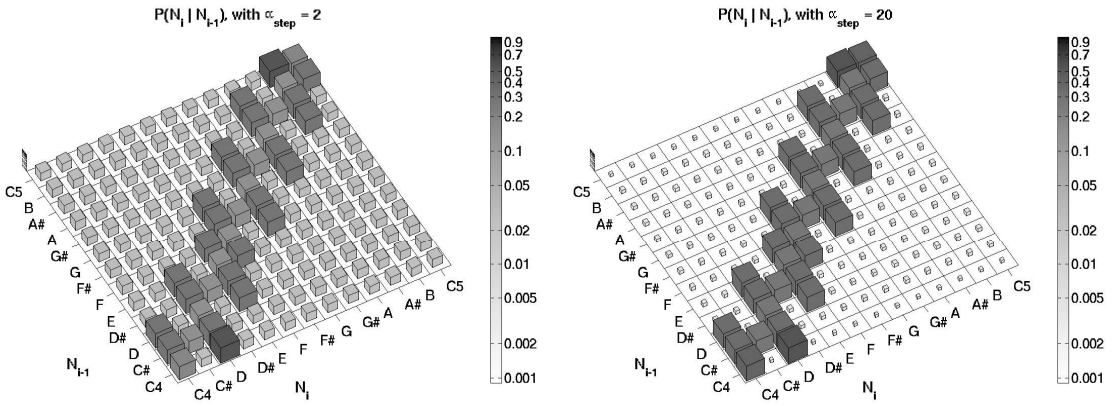
$$\sum_{n \in \mathcal{D}_{step}(\mathrm{C_{Maj}}, n_{i-1})} P(N_i = n \,|\, N_{i-1} = n_{i-1}) \geq \alpha_{step} \quad \cdot \sum_{n \in \mathcal{D}_{non\_step}(\mathrm{C_{Maj}}, n_{i-1})} P(N_i = n \,|\, N_{i-1} = n_{i-1}) \quad (4.10)$$

Figure 4.7 displays the maximum entropy rate distributions obtained using two different values of α_{step}.

4.5 Combining rules

A particularly powerful and elegant aspect of our rule encoding framework is that multiple rules can be combined simply by including all of their constraints in a convex optimization routine that maximizes the entropy rate of the transition distribution given the combined set of constraints. The resulting transition distribution automatically contains all of the intricate interactions among musical tendencies specified by the individual rules.

In our implementation, the constraints associated with each rule are represented



(a) Optimization results for $\alpha_{step} = 2$

(b) Optimization results for $\alpha_{step} = 20$

Figure 4.7: Maximum entropy rate transition distributions given diatonic stepwise constraints relative the key $\mathrm{C_{Maj}}$. Notice that repeated note values along the diagonal of the transition matrix are not constrained.

using two matrix equations:

$$\mathrm{Cx} = \mathrm{d} \qquad \text{simplex and other equality constraints} \tag{4.11}$$

$$\mathrm{Wx} \leq \mathrm{z} \qquad \text{inequality constraints} \tag{4.12}$$

where x is a vector representing the optimal solution, $P(N_i \,|\, N_{i-1})$, and the indexing of each row of C and W matches the indexing of x. For example, the repeated note constraints for a space of two notes with $\alpha_{repeat} = 0.25$, displayed in Figure 4.3, can be represented using the following two matrix equations:

$$\begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} P(A \,|\, A) \\ P(B \,|\, A) \\ P(A \,|\, B) \\ P(B \,|\, B) \end{bmatrix} = \begin{bmatrix} 1 \\ 1 \end{bmatrix} \qquad \text{simplex constraints} \tag{4.13}$$

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} P(A \,|\, A) \\ P(B \,|\, A) \\ P(A \,|\, B) \\ P(B \,|\, B) \end{bmatrix} \leq \begin{bmatrix} 0.25 \\ 0.25 \end{bmatrix} \qquad \text{repeat constraints} \tag{4.14}$$

Combining constraints from multiple rules is thus as straightforward as concatenating the matrix equations associated with each individual rule to create a new pair of matrix equations.[2] As discussed in Section 4.3, as model complexity increases to account for additional musical context, vector x can easily contain millions of elements. Equations (4.14) and (4.14) show, however, that the matrices C and W tend to be very sparse, with nonzero elements producing a linear combination of only a few elements of x. Many convex optimization software packages explicitly take advantage of constraint sparsity to greatly improve computational efficiency.

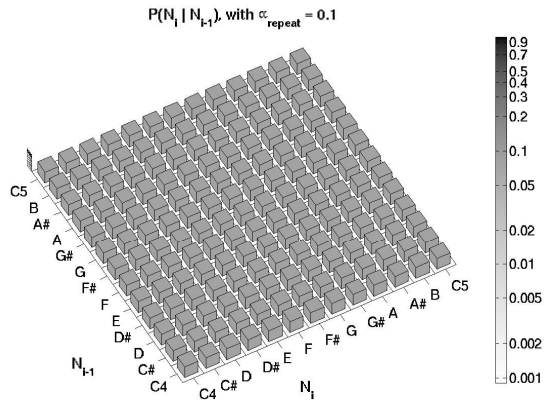
The blocked matrix equations for combining repeat, diatonic stepwise, and nondiatonic rules are:

$$[\mathrm{C}_{simplex}]\,[\mathrm{x}] = [\mathrm{d}_{simplex}] \tag{4.15}$$

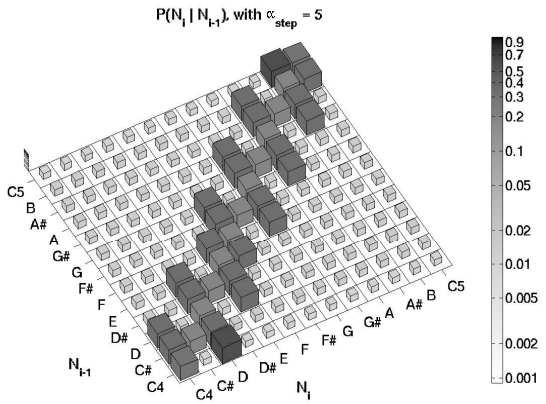
$$\begin{bmatrix} \mathrm{W}_{repeat} \\ \mathrm{W}_{step} \\ \mathrm{W}_{nondiatonic} \end{bmatrix} \begin{bmatrix} \\ \mathrm{x} \\ \\ \end{bmatrix} \leq \begin{bmatrix} \mathrm{z}_{repeat} \\ \mathrm{z}_{step} \\ \mathrm{z}_{nondiatonic} \end{bmatrix} \tag{4.16}$$

Figure 4.8 displays the results of combining these rule constraints. As demonstrated in Section 4.4.1, the repeat constraints by themselves have no effect when $\alpha_{repeat} = 0.1$. Figure 4.8c shows, however, that the same constraints do affect the optimization solution when they are applied in conjunction with another set of stepwise constraints; values along the diagonal representing repeated notes are less than corresponding values in Figure 4.8b, which displays the results for stepwise constraints alone. The participation of all the rules and intricate interactions among tendencies they describe is evident in Figure 4.8d.

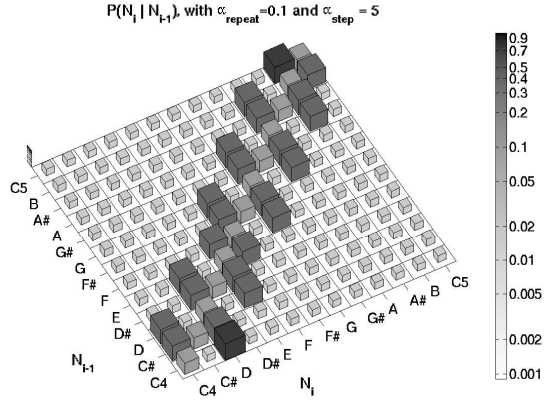
[2] When concatenating equality matrices for multiple rules, simplex constraints are included only once to avoid redundancy.



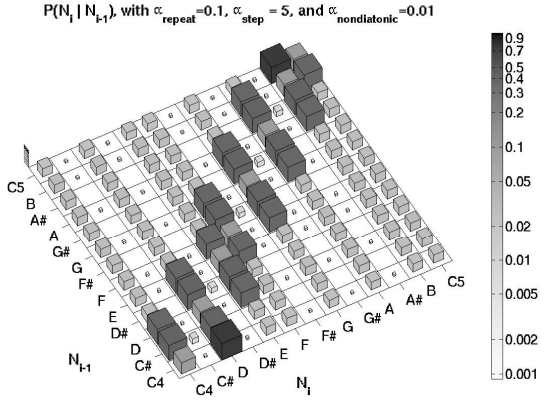
(a) Simplex and repeat constraints



(b) Simplex, and stepwise constraints



(c) Simplex, repeat, and stepwise constraints



(d) Simplex, repeat, stepwise, and nondiatonic constraints

Figure 4.8: Combined rules resulting from concatenating the constraint matrices from individual repeat, diatonic stepwise, and nondiatonic constraint rules.

4.6 Implications of the asymptotic equipartition property

The preceding sections present examples showing how maximizing entropy rate subject to a set of convex constraints obtains distributions that are as uniform as possible given those constraints. We have argued that a transition matrix that is as uniform as possible is the correct one to use for inference, because any other choice would inadvertently favor certain transitions, overstepping the set of assumptions to which we are willing to commit. This section presents a another, more formal information-theoretic justification for using maximum entropy rate distributions.

If we consider a stationary ergodic process $\{N_1, N_2, \ldots\}$ with values taken from a countable set of notes $\mathcal{N}$, the number of unique melodic sequences containing k notes equals $|\mathcal{N}|^k$. Calculating the likelihood of observing a specific k-note sequence, $n_{1:k}$, distributed according to $P(N_{1:k})$, is as straightforward as taking the product of the corresponding entries in the prior and conditional distribution tables:

$$P(n_{1:k}) = P(n_1) \sum_{i=2}^{k} P(n_i \,|\, n_{i-1}) \tag{4.17}$$

The Shannon-McMillan-Breiman theorem, detailed in Cover and Thomas [25], relates that likelihood, $P(n_{1:k})$, to the entropy rate of the process, $H_r(\mathscr{N})$:

$$-\frac{1}{k} \log_2 P(n_{1:k}) = -\frac{1}{k} \sum_{i=1}^{k} \log_2 P(n_i \,|\, n_{i-1}, \ldots, n_1) \tag{4.18}$$

$$\rightarrow H_r(\mathscr{N}), \text{with probability } 1 \tag{4.19}$$

That is, $P(n_{1:k})$ is close to $2^{-kH_r(\mathscr{N})}$ with high probability. This asymptotic equipartition property (AEP) for stationary ergodic processes was first stated in Shannon [81], then proven in McMillan [53] and Breiman [16]. Algoet and Cover [2] present a particularly clear sandwich argument reducing the proof to direct applications of the ergodic theorem.

In order to prove several implications of the AEP, Cover and Thomas [25] define the *typical set* $A_\epsilon^{(k)}$ with respect to $P(N_{1:k})$ as as the set of sequences $n_{1:k}$ with the following property:

$$2^{-k(H_r(\mathscr{N})+\epsilon)} \leq P(n_{1:k}) \leq 2^{-k(H_r(\mathscr{N})-\epsilon)} \tag{4.20}$$

Cover and Thomas [25] then prove the following implications:[3]

[3]Cover and Thomas actually prove these statements in the case where $N_{1:k}$ are i.i.d., but claim that their arguments can be applied for a general stationary ergodic process.

1. The number of elements in the typical set is about $2^{kH_r(\mathcal{N})}$:

$$|A_\epsilon^{(k)}| \leq 2^{k(H_r(\mathcal{N})+\epsilon)} \tag{4.21}$$

$$|A_\epsilon^{(k)}| \geq (1-\epsilon)2^{k(H_r(\mathcal{N})-\epsilon)} \quad \text{for } k \text{ sufficiently large} \tag{4.22}$$

2. The typical set has probability nearly 1:

$$\sum_i P(A_{\epsilon,i}^{(k)}) > 1-\epsilon \quad \text{for } k \text{ sufficiently large} \tag{4.23}$$

   where $A_{\epsilon,i}^{(k)}$ is the i^{th} element of the typical set.

3. All elements of the typical set are nearly equiprobable, each with probability about $2^{-kH_r(\mathcal{N})}$. If $n_{1:k} \in A_\epsilon^{(k)}$,

$$H_r(\mathcal{N}) - \epsilon \leq -\frac{1}{k}\log_2 P(n_{1:k}) \leq H_r(\mathcal{N}) + \epsilon \tag{4.24}$$

To provide musical examples, suppose again that the space $\mathcal{N}$ contains the thirteen notes from C_4 to C_5. In Figure 4.8a, the transition matrix with simplex and (inactive) repeat constraints is completely uniform. The entropy rate of the corresponding process can be calculated using (4.7) to be $\log_2(13) \approx 3.70$ bits. The number of typical sequences of length k is thus (nearly) equal to the number of possible sequences of that length, 13^k, and each individual sequence has probability close to $1/13^k$. Figure 4.8c adds stepwise constraints, reducing the entropy rate to approximately 2.20 bits. The distribution displayed in Figure 4.8d, with all rule constraints active, has a corresponding entropy rate of about 2.08 bits. This demonstrates the fundamental principle that adding a constraint can never increase a distribution's entropy rate; the rate can only remain the same, if the additional constraint is inactive, or decrease, if it is active.

Although many transition distributions may satisfy the constraints specified for a given musical rule, any distribution other than one with maximal entropy rate effectively contains additional, unspecified constraints that drive down the entropy rate

and decrease the number of musical pieces in the typical set. Because the probability of the stochastic process producing melodies not in the typical set tends to zero, every unspecified constraint limits the relevance of the resulting model. In addition to providing a flexible and computationally efficient system for encoding and combining inherently incomplete descriptions of musical tendencies, our maximum entropy framework thus leads to dynamic probabilistic models that are as widely applicable as possible given a set of musical constraints.

Chapter 5

Learning Rule Parameters from Data

In the examples presented in Chapter 4, the collection of α-parameters, such as α_{repeat} and $\alpha_{nondiatonic}$, were all set by hand to values that would produce easily interpretable figures demonstrating rule encoding results. This chapter describes an iterative algorithm for learning unknown rule parameters from musical data.

The ability to specify parameter values by hand is useful in several contexts, including:

- *Composing music*: If the user of the system wishes to compose music using the resulting stochastic process, the set of α parameters can be changed to control and color aspects of the generated music. Music was generated probabilistically on computers as early as 1957 by Brooks, *et al.* [17], and the use of probabilistic models in automated music composition is today extremely common. A good overview of statistical music generation techniques and probabilistic sampling methods appears in Conklin [24].

- *Replicating results*: A user may wish to replicate prior research results that quantify transition distributions or otherwise weight a set of musical tendencies in ways that can be interpreted probabilistically.

- *Guessing parameter values*: In some cases, a user may desire to encode a rule

describing music for which it is not immediately feasible to obtain a sufficient collection of relevant training pieces from which to learn the parameter values. In such a situation, a user may want to test the general performance of a rule, using an educated guess about reasonable parameter values, prior to investing the time and effort necessary to obtain such a data set.

In cases where a suitable corpus of musical data is available, however, it may be preferable to fit the set of α parameters to that data. Although satisfactory inference results can often be obtained by choosing conservative values for each the α parameters, making the parameters as restrictive as possible, while remaining true to the data, reduces the variance of resulting inference estimates.

5.1 Expectation-maximization algorithm

Working within the context of the basic AR-HMM depicted in Figure 2.13, musical data may consist of either a single observation sequence $X_{1:K}$ or a corpus of pieces. The observation sequence for the n^{th} of N pieces in a corpus has $K^{(n)}$ observations, and is denoted $X^{(n)}_{1:K^{(n)}}$; the corresponding hidden state sequence is denoted $S^{(n)}_{1:K^{(n)}}$. When the set of pieces is grouped together, the sets of all observed and hidden state sequences are labeled $X^{(1:N)}_{1:K}$ and $S^{(1:N)}_{1:K}$, where the K in the subscript is understood to be piece dependent: i.e., $X^{(1:N)}_{1:K} = \{X^{(1)}_{1:K^{(1)}}, X^{(2)}_{1:K^{(2)}}, \ldots, X^{(K)}_{1:K^{(K)}}\}$, with the same subscript interpretation for $S^{(1:N)}_{1:K}$.

The efficient use of corpus data requires additional assumptions on the joint distribution over $X^{(1:N)}_{1:K}$ and $S^{(1:N)}_{1:K}$ given the collection of all α-parameters, which we simply denote as α. We assume that joint observation-state sequences are mutually conditionally independent across different pieces:

$$P(S^{(1:N)}_{1:K}, X^{(1:N)}_{1:K} \,|\, \alpha) = \prod_{n=1}^{N} P(S^{(n)}_{1:K^{(n)}}, X^{(n)}_{1:K^{(n)}} \,|\, \alpha) \tag{5.1}$$

Because the basic AR-HMM factors according to (2.15), the joint corpus distribution factorizes as follows:

$$P(S_{1:K}^{(1:N)}, X_{1:K}^{(1:N)}|\alpha) = \prod_{n=1}^{N} P(S_1^{(n)})P(X_1^{(n)}|S_1^{(n)}) \prod_{i=2}^{K^{(n)}} P(S_i^{(n)}|S_{i-1}^{(n)})P(X_i^{(n)}|X_{i-1}^{(n)}, S_i^{(n)}, \alpha) \tag{5.2}$$

Ideally, α is chosen to maximize the likelihood of the corpus:

$$\hat{\alpha}_{MLE} \in \operatorname*{argsup}_{\alpha} P(X_{1:K}^{(1:N)} \,|\, \alpha) \tag{5.3}$$

Unfortunately, direct maximization of (5.3) becomes difficult due to the marginalization over $S_{1:K}^{(1:N)}$. In such situations, the expectation-maximization (EM) algorithm [29] is often applied.

Beginning with an initial guess $\alpha^{(0)}$, the EM algorithm comprises a sequence of updates $\alpha^{(j)} \to \alpha^{(j+1)}$ for which the sequence $\{\alpha^{(j)}\}, j \geq 0$ converges to a local maximum of the likelihood objective $P(X_{1:K}^{(1:N)} \,|\, \alpha)$ as a function of α. If $\alpha^{(0)}$ is sufficiently close to $\hat{\alpha}_{MLE}$, the convergence will be to the global maximum, $\hat{\alpha}_{MLE}$, as desired.

Each update $\alpha^{(j)} \to \alpha^{(j+1)}$ for the EM algorithm consists of two steps. The first, or *expectation* step forms the objective $Q(\alpha \,|\, \alpha^{(j)})$:

$$Q(\alpha \,|\, \alpha^{(j)}) = E_{P(S_{1:K}^{(1:N)}|X_{1:K}^{(1:N)}, \alpha^{(j)})} \log P(S_{1:K}^{(1:N)}, X_{1:K}^{(1:N)} \,|\, \alpha) \tag{5.4}$$

where E denotes the expectation operator. The second, *maximization* step chooses $\alpha^{(j+1)}$ as the value of α maximizing that objective:

$$\alpha^{(j+1)} = \operatorname*{argmax}_{\alpha} Q(\alpha|\alpha^{(j)}) \tag{5.5}$$

According to the factorization in (5.2),

$$\log P(S_{1:K}^{(1:N)}, X_{1:K}^{(1:N)} \,|\, \alpha) = \text{const} + \sum_{n=1}^{N} \sum_{i=2}^{K^{(n)}} \log P(X_i^{(n)}|X_{i-1}^{(n)}, S_i^{(n)}, \alpha) \tag{5.6}$$

where the const value collects all terms that do not depend on α.

Maximizing $Q(\alpha \,|\, \alpha^{(j)})$ thus becomes equivalent to maximizing

$$\begin{aligned} Q'(\alpha \,|\, \alpha^{(j)}) &= E_{P(S_{1:K}^{(1:N)} | X_{1:K}^{(1:N)}, \alpha^{(j)})} \sum_{n=1}^{N} \sum_{i=2}^{K^{(n)}} \log P(X_i^{(n)} \,|\, X_{i-1}^{(n)}, S_i^{(n)}, \alpha) \\ &= \sum_{n=1}^{N} \sum_{i=2}^{K^{(n)}} E_{P(S_i^{(n)} | X_{1:K^{(n)}}^{(n)}, \alpha^{(j)})} \log P(X_i^{(n)} \,|\, X_{i-1}^{(n)}, S_i^{(n)}, \alpha) \\ &= \sum_{n=1}^{N} \sum_{i=2}^{K^{(n)}} \sum_{S_i^{(n)} \in \mathcal{S}} P(S_i^{(n)} | X_{1:K^{(n)}}^{(n)}, \alpha^{(j)}) \log P(X_i^{(n)} \,|\, X_{i-1}^{(n)}, S_i^{(n)}, \alpha) \quad (5.7) \end{aligned}$$

Hence, the EM update reduces as follows:

$$\alpha^{(j+1)} = \underset{\alpha}{\operatorname{argmax}} \sum_{n=1}^{N} \sum_{i=2}^{K^{(n)}} \sum_{S_i^{(n)} \in \mathcal{S}} P(S_i^{(n)} | X_{1:K^{(n)}}^{(n)}, \alpha^{(j)}) \log P(X_i^{(n)} | X_{i-1}^{(n)}, S_i^{(n)}, \alpha) \quad (5.8)$$

To actually perform this EM update, we begin by using the convex optimization approach discussed in Chapter 4 to obtain $P(X_i^{(n)} | X_{i-1}^{(n)}, S_i^{(n)}, \alpha^{(j)})$, the maximum entropy rate distribution corresponding to the collection of parameters $\alpha^{(j)}$; this distribution is identical for all pieces (all superscript n). We then use that transition distribution in a standard Bayesian smoothing inference, as detailed in Section 7.3, to compute $P(S_i^{(n)} | X_{1:K^{(n)}}^{(n)}, \alpha^{(j)})$ for all N pieces in the corpus, i.e., for all $n \in 1\!:\!N$. This smoothing step must only be done once per EM update. Next, the solution of the maximization in (5.8) is obtained via an iterative, nonlinear, hill-climbing algorithm, such as gradient ascent; several such algorithms are described in [30, 39]. Computing the gradient at any given value of α requires the evaluation of the objective in (5.8) at nearby values, α' (two such values per parameter in the collection α). For each α', the entire distribution $P(X_i \,|\, X_{i-1}, S_i, \alpha')$ is precomputed by the entropy maximization approach, then the evaluation of $P(X_i^{(n)} \,|\, X_{i-1}^{(n)}, S_i^{(n)}, \alpha')$ simply selects elements of that maximum entropy rate solution. Each step of this learning process is demonstrated in the following section.

5.2 Learning example

To demonstrate each step of the EM algorithm, we will once again work within the context of the simple first-order Markov chain shown in Figure 2.4, where the note transition distribution is determined by a single rule combining nondiatonic and repeat constraints, $\alpha_{nondiatonic}$ and α_{repeat}; these constraints are defined in Section 4.4. Maximum entropy rate distributions corresponding to three values of α_{repeat} are displayed in Figure 4.4, and distributions for two values of $\alpha_{nondiatonic}$ are displayed in Figure 4.6. As in those figures, the note space $\mathcal{N}$ in this example contains the thirteen notes spanning the octave from C_4 to C_5.

As demonstrated in Section 4.5, constraints corresponding to $\alpha_{repeat} \geq 1/13$ have no effect. Similarly, constraints corresponding to $\alpha_{nondiatonic} \geq 5/13$ have no effect. These regions are clearly visible in Figure 5.1a, which displays the entropy rate of the maximum entropy distribution corresponding to pairs of parameter values.

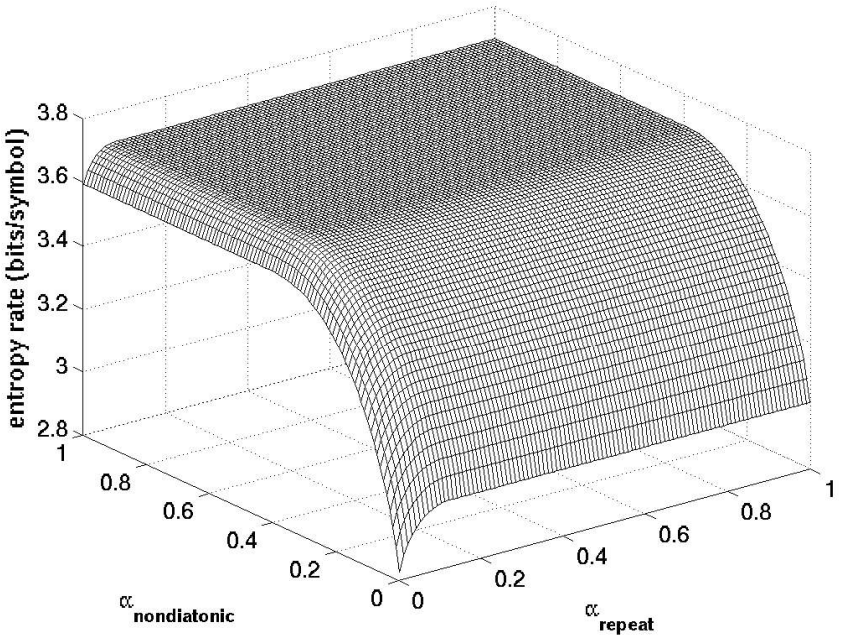
When $\alpha_{nondiatonic} < 5/13$ and $\alpha_{repeat} < 1/13$, all constraints are active, so we will test the learning process using a value of $\alpha = [\alpha_{nondiatonic}\ \alpha_{repeat}]$ in that active region. To demonstrate the learning process, we algorithmically compose a corpus of melodies using known value of α, then initialize the EM iterations by starting at some other choice of parameter values in the same region of the space, and verify that the EM iterations converge to the true α.

Given the distributional specifications for a first-order Markov chain modeling note sequences, we can compose a melody using several different probabilistic sampling methods. Perhaps the simplest is a *random walk* method involving the following steps:

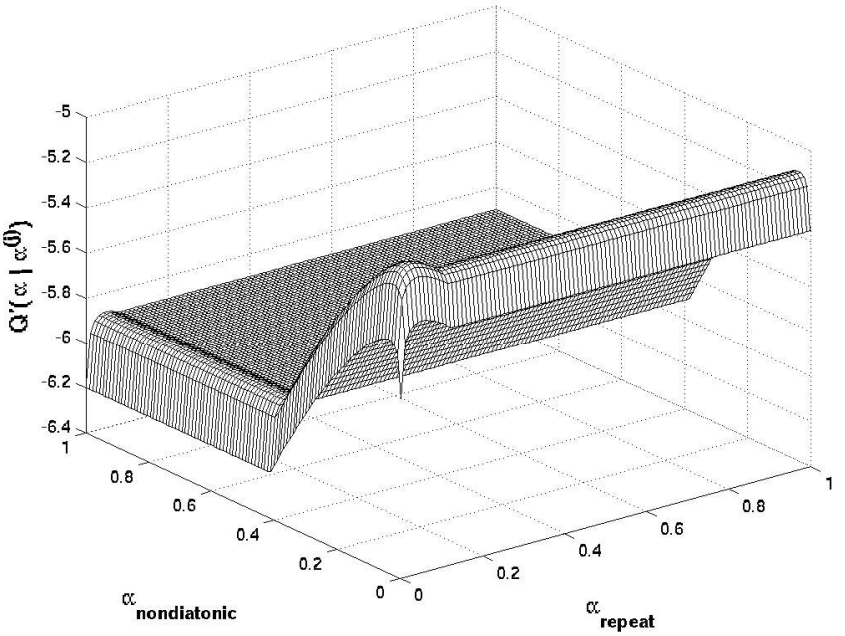
1. Sample from the prior distribution $P(N_1)$ to determine the first note of the piece, n_1.

2. Sample from the transition distribution $P(N_{i+1} \,|\, n_i)$ to determine the value of n_{i+1}.

3. Repeat step 2 until the melody reaches the desired length.



(a) Maximum entropy rate of the distribution obtained using pairs of $\alpha_{nondiatonic}$ and α_{repeat} values. The maximum entropy rate distribution is completely uniform when $\alpha_{nondiatonic} \geq 5/13$ and $\alpha_{repeat} \geq 1/13$.



(b) Objective function $Q'(\alpha \,|\, \alpha^{(0)})$ formed by the first expectation step. The maximization step chooses $\alpha^{(1)}$ to be the α value maximizing this objective.

Figure 5.1: Entropy rate and objective surfaces for learning example, where the synthesized corpus corresponds to $\alpha_{nondiatonic} = 0.04$ and $\alpha_{repeat} = 0.05$

We obtain the maximum entropy rate distribution corresponding to $\alpha_{nondiatonic} = 0.04$ and $\alpha_{repeat} = 0.05$, and use this random walk process to synthesize 500 melodies, each containing a sequence of 50 notes. If the learning process is successful, it should recover the value of α used to generate that set of 500 melodies.

We begin the EM iterations by initializing $\alpha^{(0)} = [5/13\ 1/13] \approx [0.3846\ 0.0769]$, the α value on the corner of the region in which both constraints become active. We then compute $P(X_i \,|\, X_{i-1}, S_i, \alpha^{(0)})$, the corresponding maximum entropy rate distribution, and use it to perform Bayesian smoothing on each of the N pieces. The output of the smoothing stage is a set of state estimates $P(S_i^{(n)} \,|\, X_{1:K^{(n)}}^{(n)}, \alpha^{(j)})$, which are used in the expectation step (5.7) to form the objective $Q'(\alpha \,|\, \alpha^{(0)})$. Figure 5.1b displays the surface $Q'(\alpha \,|\, \alpha^{(0)})$ as a function of α. This surface contains constraint activity regions that match those in Figure 5.1a, and a global maximum, which we locate using an iterative hill-climbing approach.

Figure 5.2 displays the hill-climbing steps chosen by the `fmincon` function in Matlab's optimization toolbox; we use this to minimize $-Q'(\alpha \,|\, \alpha^{(0)})$ subject to boundary constraints ensuring that $0 < \alpha_{nondiatonic} \leq 1$ and $0 < \alpha_{repeat} \leq 1$.[1] Although Figures 5.1–5.2 display an entire surface for $Q'(\alpha \,|\, \alpha^{(0)})$, each hill climbing step only involves evaluating the several points necessary to determine the gradient, and to perhaps perform a line search in the direction of steepest ascent. The maximization step displayed in Figure 5.2 starts with $\alpha^{(0)} = [0.3846\ 0.0769]$ and obtains $\alpha^{(1)} = [0.0383\ 0.0492]$. This $\alpha^{(1)}$ starts the next iteration, requiring smoothing all files using the maximum entropy rate transition matrix corresponding to $\alpha^{(1)}$, then forming $Q(\alpha \,|\, \alpha^{(1)})$ and finding its maximum value. This process continues until each element of $\alpha^{(j)}$ and $\alpha^{(j+1)}$ are within some tolerance of one another; this tolerance might just be an absolute maximum difference if all α parameters are about the same size, or it could represent a percentage of the range of each variable, if the range of parameter values differs significantly. As defined in this example, the parameters α_{repeat} and $\alpha_{nondiatonic}$

[1] Although this choice of maximization algorithm is convenient, because we have a license for the Matlab optimization toolbox, and its usage is well documented, future research may involve finding or developing a maximization algorithm that performs especially well on surfaces containing regions where the slope along one or more dimensions is zero. As seen in Figure 5.1, the objective surface can have large flat areas due to parameter values producing constraints that are satisfied by a uniform distribution, effectively rendering those constraints inactive.

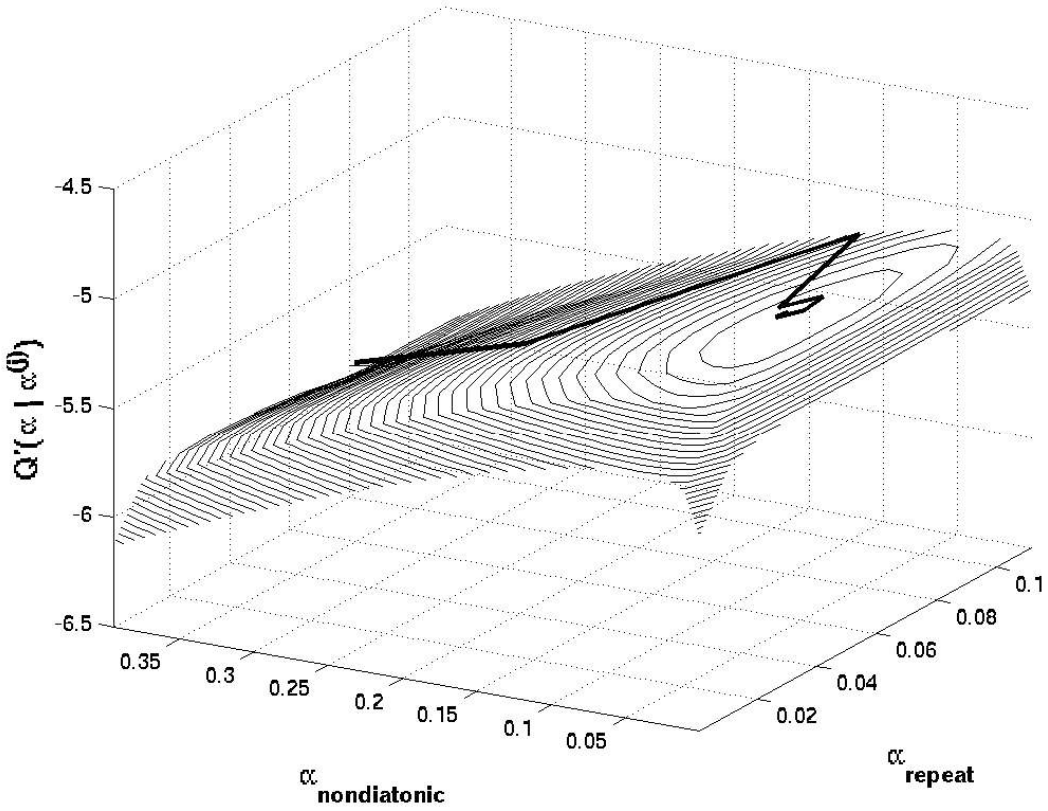


Figure 5.2: Hill-climbing steps to determine $\alpha^{(1)} = \text{argmax}_\alpha \; Q'(\alpha \,|\, \alpha^{(0)}) = [0.0383\ 0.0492]$.

both operate within the range $[0, 1]$, so we choose a small absolute convergence tolerance, $1 \cdot 10^{-4}$. At the end of the second EM update, $\alpha^{(2)} = [0.0383\ 0.0492]$, which is within that tolerance (equal in displayed precision to $\alpha^{(1)}$). The EM iterations thus converge to a solution, $\hat{\alpha}_{MLE} = [0.0383\ 0.0492]$, closely matching the parameter values used to synthesize the musical corpus.

Chapter 6

Inferring Rule Activation and Violation

The previous chapter explained a maximum entropy rate framework for encoding parameterized musical rules, and showed how a convex optimization approach facilitates the creation of rule combinations that include the constraints from multiple individual rules. A set of such maximum entropy rate distributions can work together in the context of the basic AR-HMM model, Figure 2.13, to create rule-based musical expectations. If the hidden state S_i in the basic AR-HMM model, contains a single random variable R_i that indicates which rule is responsible for for choosing the note at time i. If we step through a piece using Bayesian filtering, we can compute the posterior distribution $P(R_i|x_{1:i})$, allowing us to visualize the relative contribution of each rule in determining note x_i, and to identify at any point in the piece which rules are active and which are violated.

The ability to observe the participation of individual musical tendencies allows us to do a detailed comparison of the listening experience of hypothetical listeners with differing musical expectations, where each listener's prior experience is captured by the set of rules of which they aware. Our framework in this way can be used to compare music theories, observing musical situations in which their predictions agree or differ, and observing which notes in a piece are surprising given the assertions of each theory.

6.1 Musical forces

To demonstrate inference of rule activation and violation, we encode three musical forces described in Larson [47] and Larson and VanHandel [48]: *gravity*, *magnetism*, and *inertia*. As in [48], which focuses on assessing stepwise, diatonic patterns, our encoding of these musical forces assumes that they only apply to stepwise, diatonic transitions, and this assumption is reflected in two ways. First, for any note n, the constraints for each of the rules only affect the note a diatonic step above n and the note a diatonic step below n. Second, in the maximum entropy rate optimization, we assume that the constraints encoding each of the forces are added to a common set of repeat, stepwise, and nondiatonic constraints, described in Section 4.4. The maximum entropy distribution for stepwise, diatonic motion, achieved by setting $\alpha_{repeat} = .001$, $\alpha_{nondiatonic} = .001$, and $\alpha_{stepwise} = 1000$, on the space of notes spanning C_4 to C_5, is displayed in Figure 6.1. All diatonic steps have equal probability close to 0.5, with the exception of $P(N_i^{(0)} = \mathrm{D}_4 \,|\, N_{i-1}^{(0)} = \mathrm{C}_4)$ and $P(N_i^{(0)} = \mathrm{B}_4 \,|\, N_{i-1}^{(0)} = \mathrm{C}_5)$, which have probability close to 1.0, because steps in the opposite direction are not possible in the space.

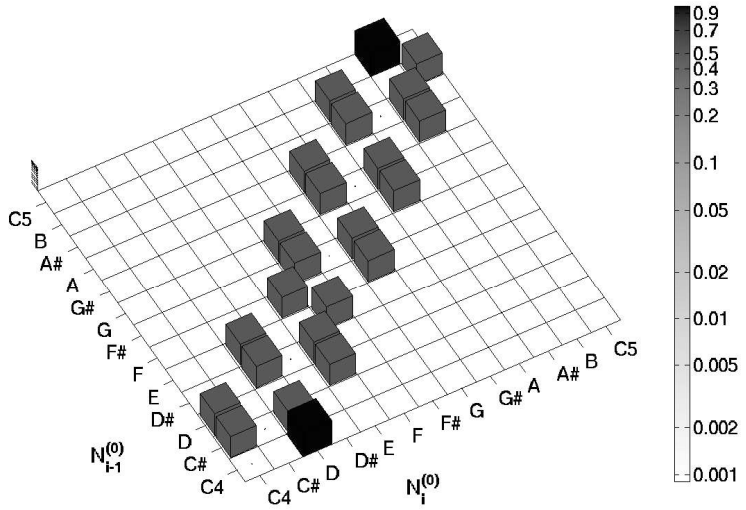


Figure 6.1: Maximum entropy rate distribution under constraints enforcing diatonic, stepwise motion. The constraints encoding musical forces will be added to the set of constraints producing this distribution.

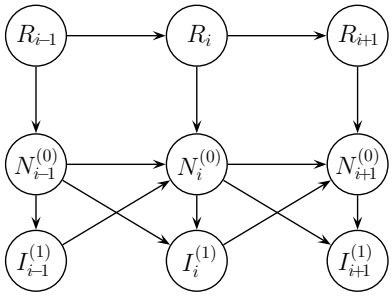


Figure 6.2: Directed acyclic graph used to model musical forces

As will be discussed in Sections 6.1.2 and 6.1.3, gravity and magnetism can be encoded using only a first-order note dependence. Inertia, discussed in Section 6.1.4, is the tendency of a note pattern to continue moving in the same manner, so in order to predict a note at index i, we need to know how the note at $i-1$ was reached. In examples containing a space of just thirteen semitones, maintaining one additional note state might not result in any noticeable difference in rule optimization or inference speed. If the size of the note space were larger (e.g, all eighty-eight notes on a piano keyboard), or the length of the history grew beyond a couple of notes, it would be computationally helpful, or even necessary, to compress the variable space of the history in a way that it stores only the information really needed by the model.

Because our encoding of musical forces only constrains stepwise motion, we can reduce the size of the extra history state; we only have to know whether the note at index $i-1$ was reached by upward or downward diatonic step. Rather than storing the additional note history variable $N^{(1)}_{i-1}$, we store a reduced interval variable $I^{(1)}_{i-1}$, which takes semitone values from $\mathcal{I} = \{-2, -1, 1, 2, other\}$. The resulting model is displayed in Figure 6.2, and the graph encodes the following factorization:

$$P(X_{1:K}, R_{1:K}) = P(X_1 \,|\, R_1)P(R_1)\prod_{i=2}^{K} P(X_i \,|\, R_i, X_{i-1})P(R_i \,|\, R_{i-1}) \tag{6.1}$$

$$= P(N^{(0)}_{i-1}, I^{(1)}_{i-1} \,|\, R_1)P(R_1)\times \tag{6.2}$$

$$\prod_{i=2}^{K} P(I^{(1)}_i \,|\, N^{(0)}_i, N^{(0)}_{i-1})P(N^{(0)}_i \,|\, N^{(0)}_{i-1}, I^{(1)}_{i-1}, R_i)P(R_i|R_{i-1}) \tag{6.3}$$

where

$$P(I_i^{(1)} = i_i^{(1)} \mid N_i^{(0)}, N_{i-1}^0) = \begin{cases} 1 & \text{if } |n_i^{(0)} - n_{i-1}^{(0)}| \notin \{1,2\} \text{ and } i_i^{(1)} = \textit{other}, \text{ or} \\ & |n_i^{(0)} - n_{i-1}^{(0)}| \in \{1,2\} \text{ and } i_i^{(1)} = n_i^{(0)} - n_{i-1}^{(0)} \\ 0 & \text{otherwise} \end{cases} \quad (6.4)$$

6.1.1 Note sets and operator definitions

In order to encode the set of musical forces, it is helpful to define several subsets of the space of all notes. These set definitions appear in Table 6.1.

Symbol	Description
$\mathcal{N}$	set of all notes in the space
$\mathcal{D}(k)$	subset of $\mathcal{N}$ containing all diatonic notes in the given key
$\mathcal{A}(k)$	set of notes that are nondiatonic in the given key: $\mathcal{A}(k) = \mathcal{N} \setminus \mathcal{D}(k)$
$\mathcal{S}(k)$	subset of $\mathcal{N}$ containing all stable notes in the given key; in major and minor keys, stable notes are those in the tonic triad
$\mathcal{U}(k)$	set of notes that are unstable in the given key: $\mathcal{U}(k) = \mathcal{N} \setminus \mathcal{S}(k)$

Table 6.1: Sets of notes used to encode musical forces

We also define several operators to facilitate the encoding process. In order to provide examples of each operator, let $\mathcal{N}$ to be the set of all MIDI note numbers on a standard 88-note piano keyboard (notes 21–108, where 60 corresponds to middle C).

- $\mathrm{D}^+(k, n)$: the next higher diatonic pitch from note n in $\mathcal{D}(k)$; e.g., $\mathrm{D}^+(\mathrm{C}_{\mathrm{Maj}}, 62) = 64$ and $\mathrm{D}^+(\mathrm{C}_{\mathrm{min}}, 62) = 63$.
- $\mathrm{D}^-(k, n)$: the next lower diatonic pitch from note n in $\mathcal{D}(k)$; e.g., $\mathrm{D}^-(\mathrm{C}_{\mathrm{Maj}}, 60) = 59$.
- $\mathrm{A}^+(k, n)$: nearest attractor, the closest note to n in $\mathcal{S}(k)$; e.g., $\mathrm{A}^+(\mathrm{C}_{\mathrm{Maj}}, 65) = 64$. If two such stable notes are equidistant from n, the choice is arbitrary.
- $\mathrm{A}^-(k, n)$: the opposing attractor, the closest stable note in the opposite direction of $\mathrm{A}^+(k, n)$; e.g., $\mathrm{A}^-(\mathrm{C}_{\mathrm{Maj}}, 65) = 67$.

- $\mathrm{M}^{+}(k, n)$: diatonic step from n in the direction of the nearest attractor, $\mathrm{A}^{+}(k, n)$; e.g., $\mathrm{M}^{+}(\mathrm{C}_{\mathrm{Maj}}, 69) = 67$.
- $\mathrm{M}^{-}(k, n)$: diatonic step from n in the direction of the opposing attractor, $\mathrm{A}^{-}(k, n)$; e.g., $\mathrm{M}^{-}(\mathrm{C}_{\mathrm{Maj}}, 69) = 71$.

6.1.2 Gravity

Musical gravity is the tendency of a note a step above a stable note to descend to the stable note. A rule stating, "The probability that a note one diatonic step above a stable pitch descends to the stable pitch is greater than α_G times the probability that the unstable note ascends a diatonic step," can be represented using the following set of constraints:

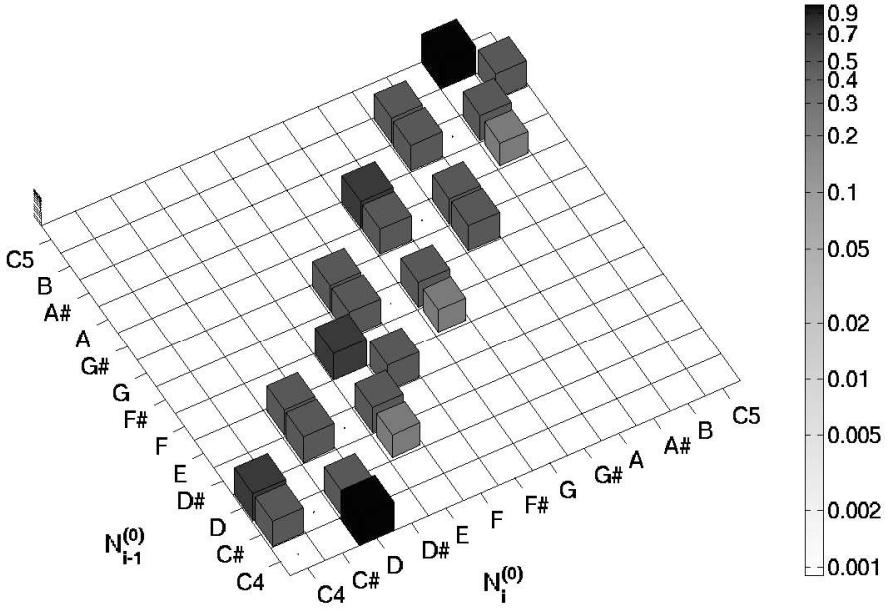
$$\alpha_G \, P(N_i^{(0)} = \mathrm{D}^{+}(k, n) \mid K_i = k, N_{i-1}^{(0)} = n, I_{i-1}^{(1)}) \\ \leq P(N_i^{(0)} = \mathrm{D}^{-}(k, n) \mid K_i = k, N_{i-1}^{(0)} = n, I_{i-1}^{(1)}) \quad (6.5)$$

for all values of k and $I_{i-1}^{(1)}$, and all n for which $\mathrm{D}^{-}(k, n) \in \mathcal{S}(k)$. Gravitational force only depends on the previous note, so for a given key, we add one constraint per value of $I_{i-1}^{(1)}$ for each value of N_{i-1} that is a diatonic step above any of the stable pitches in the key.

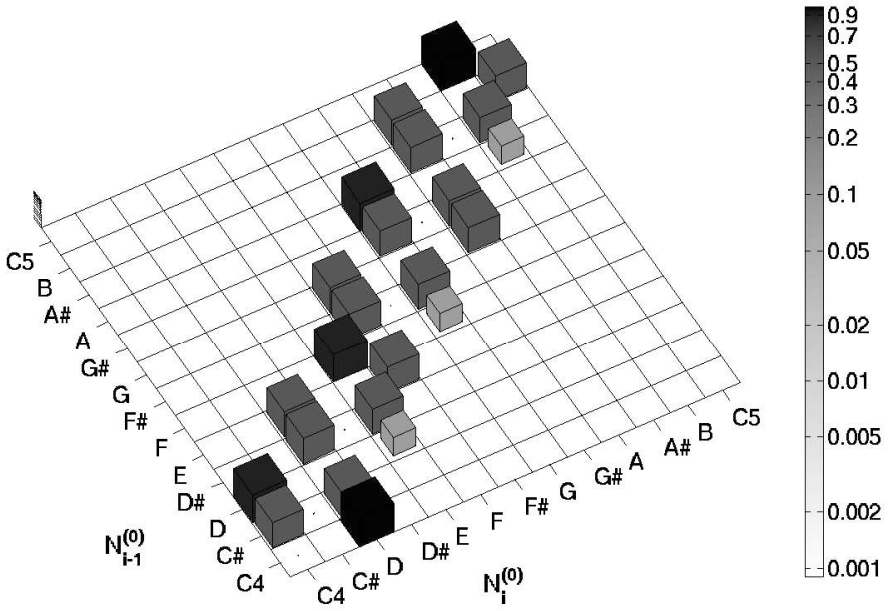
For example, if $k = \mathrm{C}_{\mathrm{Maj}}$, then then $\mathcal{S}(k)$ contains the tonic triad, $\{\mathrm{C}, \mathrm{E}, \mathrm{G}\}$, in all octaves. There are three notes per octave, $\{\mathrm{D}, \mathrm{F}, \mathrm{A}\}$, that are a diatonic step above those stable pitches. Figure 6.3 displays maximum entropy rate distributions obtained using two different values of α_G. Because gravity does not depend on the interval leading to $N_{i-1}^{(0)}$, the distributions displayed in this figure are identical for all six values of $I_{i-1}^{(1)}$.

6.1.3 Magnetism

Musical magnetism is the tendency of an unstable note to move toward the nearest stable note. The shorter the distance between the unstable and stable note, the



(a) Gravity rule with $\alpha_G = 3$. In addition to constraints enforcing stepwise, diatonic transitions, this solution satisfies three gravity constraints:
$3\,P(60|\mathrm{C_{Maj}}, 62, I^{(1)}_{i-1}) \leq P(64|\mathrm{C_{Maj}}, 62, I^{(1)}_{i-1})$; $3\,P(64|\mathrm{C_{Maj}}, 65, I^{(1)}_{i-1}) \leq P(67|\mathrm{C_{Maj}}, 65, I^{(1)}_{i-1})$;
$3\,P(67|\mathrm{C_{Maj}}, 69, I^{(1)}_{i-1}) \leq P(71|\mathrm{C_{Maj}}, 69, I^{(1)}_{i-1})$



(b) Gravity rule with $\alpha_G = 10$. In addition to constraints enforcing stepwise, diatonic transitions, this solution satisfies three gravity constraints:
$10P(60|\mathrm{C_{Maj}}, 62, I^{(1)}_{i-1}) \leq P(64|\mathrm{C_{Maj}}, 62, I^{(1)}_{i-1})$; $10P(64|\mathrm{C_{Maj}}, 65, I^{(1)}_{i-1}) \leq P(67|\mathrm{C_{Maj}}, 65, I^{(1)}_{i-1})$;
$10P(67|\mathrm{C_{Maj}}, 69, I^{(1)}_{i-1}) \leq P(71|\mathrm{C_{Maj}}, 69, I^{(1)}_{i-1})$

Figure 6.3: Musical gravity, encoded using two values of $\alpha_{gravity}$.

stronger the magnetic force becomes. Larson [47] quantifies the magnetic pull on note n in key k as the inverse square of the distance in semitones from the closest stable attractor, $\mathrm{A}^+(k,n)$, minus the inverse square of the distance to the opposing attractor, $\mathrm{A}^+(k,n)$:

$$M = \frac{1}{\left|n - \mathrm{A}^+(k,n)\right|^2} - \frac{1}{\left|n - \mathrm{A}^-(k,n)\right|^2} \tag{6.6}$$

A special case of this force is the well-known tendency of the leading tone to be pulled upward to the tonic of a key. As it turns out, using (6.6), the leading tone achieves the maximal magnetism value, with an an overall pull of 0.9375 in the direction of the tonic ($M = 1/1^2 - 1/4^2 = 15/16$). Bharucha [9], like Larson, describes a magnetic force, but states that the pull on an unstable note is inversely proportional to the distance to the closest stable note, rather than the square of that distance.

To encode this rule as a parameterized conditional distribution, we generalize the assertions of Larson and Bharucha by letting $\alpha_{\mathrm{M_{pow}}}$ be the power to which the inverse distances are raised (2 for Larson, 1 for Bharucha), and let α_{M} represent an unknown overall strength of the magnetism tendency. We then use these parameters in the following set of linear constraints:

$$\begin{aligned}\alpha_{\mathrm{M}} \left|n - \mathrm{A}^-(k,n)\right|^{\alpha_{\mathrm{M_{pow}}}} P(N_i^{(0)} = \mathrm{M}^-(k,n) \mid K_i = k, N_{i-1}^{(0)} = n, I_{i-1}^{(1)}) \\ \leq \left|n - \mathrm{A}^+(k,n)\right|^{\alpha_{\mathrm{M_{pow}}}} P(N_i^{(0)} = \mathrm{M}^+(k,n) \mid K_i = k, N_{i-1}^{(0)} = n, I_{i-1}^{(1)})\end{aligned} \tag{6.7}$$

for all values of k and $I_{i-1}^{(1)}$, and for $n \in \mathcal{D}(k) \cap \mathcal{U}(k)$, where $|n - \mathrm{A}^+(k,n)| \neq |n - \mathrm{A}^-(k,n)|$. Like gravity, magnetism does not depend on the interval leading to N_{i-1}, so for a given key, we add one constraint per value of $I_{i-1}^{(1)}$ for each value of N_{i-1} that is diatonic and unstable but not an equal distance between the two closest stable pitches; in equidistant cases, where $|n - \mathrm{A}^-(k,n)| = |n - \mathrm{A}^+(k,n)|$, the opposing magnetic forces cancel one another out.

For example, if $k = \mathrm{C_{Maj}}$, then $\mathcal{S}(k)$ contains the tonic triad, $\{\mathrm{C}, \mathrm{E}, \mathrm{G}\}$, in all octaves. There are three notes per octave, $\{\mathrm{F}, \mathrm{A}, \mathrm{B}\}$, that are diatonic and unstable but not the same number of semitones from the nearest pair of stable notes. If we

look at one of the leading tones, $n = \mathrm{B}_4 = 71$, then (6.7) evaluates as:

$$\begin{aligned}\alpha_{\mathrm{M}} \left|71 - \mathrm{A}^{-}(\mathrm{C}_{\mathrm{Maj}}, 71)\right|^{\alpha_{\mathrm{M_{pow}}}} P(N_i^{(0)} = \mathrm{M}^{-}(\mathrm{C}_{\mathrm{Maj}}, 71) \,|\, K_i = \mathrm{C}_{\mathrm{Maj}}, N_{i-1}^{(0)} = 71, I_{i-1}^{(1)}) \\ \leq \left|71 - \mathrm{A}^{+}(\mathrm{C}_{\mathrm{Maj}}, 71)\right|^{\alpha_{\mathrm{M_{pow}}}} P(N_i^{(0)} = \mathrm{M}^{+}(\mathrm{C}_{\mathrm{Maj}}, 71) \,|\, K_i = \mathrm{C}_{\mathrm{Maj}}, N_{i-1}^{(0)} = 71, I_{i-1}^{(1)})\end{aligned} \tag{6.8}$$

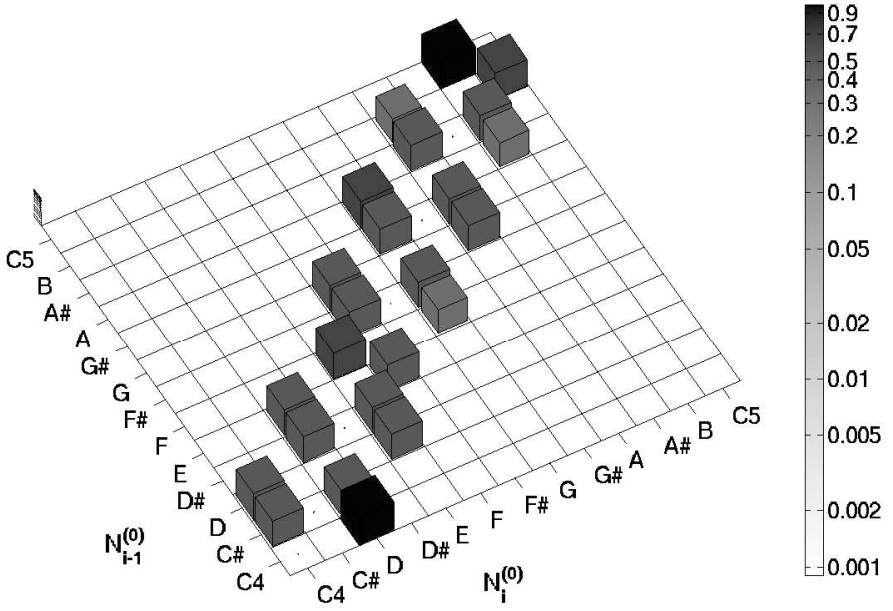
$$\begin{aligned}\alpha_{\mathrm{M}} |71 - 67|^{\alpha_{\mathrm{M_{pow}}}} P(N_i^{(0)} = 69 \,|\, K_i = \mathrm{C}_{\mathrm{Maj}}, N_{i-1}^{0} = 71, I_{i-1}^{(1)}) \\ \leq |71 - 72|^{\alpha_{\mathrm{M_{pow}}}} P(N_i^{(0)} = 72 \,|\, K_i = \mathrm{C}_{\mathrm{Maj}}, N_{i-1}^{0} = 71, I_{i-1}^{(1)})\end{aligned} \tag{6.9}$$

If we set $\alpha_{\mathrm{M_{pow}}} = 2$, then

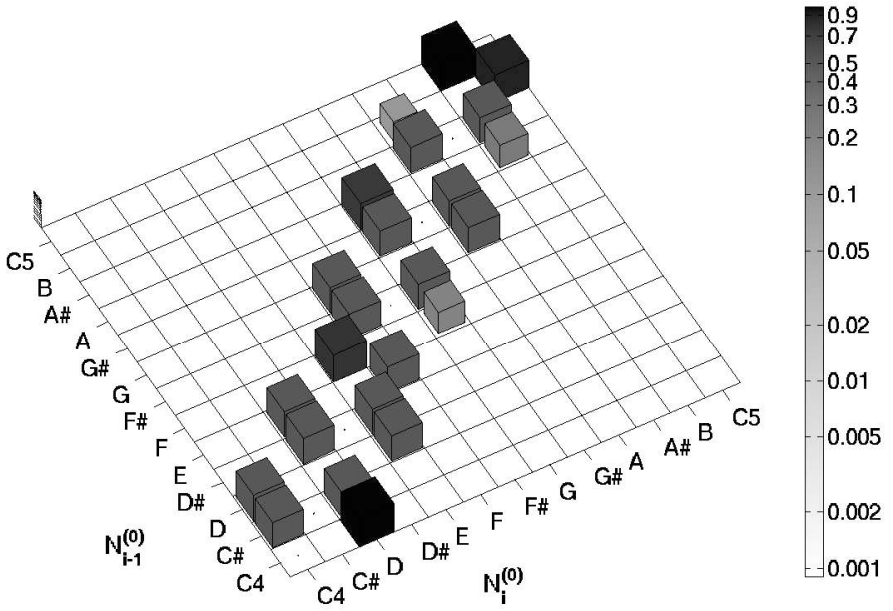
$$\begin{aligned}16\,\alpha_{\mathrm{M}}\, P(N_i^{(0)} = 69 \,|\, K_i = \mathrm{C}_{\mathrm{Maj}}, N_{i-1}^{(0)} = 71, I_{i-1}^{(1)}) \\ \leq P(N_i^{(0)} = 72 \,|\, K_i = \mathrm{C}_{\mathrm{Maj}}, N_{i-1}^{(0)} = 71, I_{i-1}^{(1)})\end{aligned} \tag{6.10}$$

The probability of the leading tone resolving upward to the tonic, C, is thus at least $16\,\alpha_{\mathrm{M}}$ times as great as the probability of it resolving downward to A. Following the same procedure, with $\alpha_{\mathrm{M_{pow}}} = 2$, we obtain that, in all octaves, the probability of $F \searrow E$ is at least $4\,\alpha_{\mathrm{M}}$ as great as $F \nearrow G$, and the probability of $A \searrow G$ is at least $9/4 \cdot\ \alpha_{\mathrm{M}}$ as great as $A \nearrow B$.

Figure 6.4 displays the maximum entropy rate optimization results, holding $\alpha_M = 2$ for three different values of $\alpha_{\mathrm{M_{pow}}}$. When $\alpha_{\mathrm{M_{pow}}} = 0$, as in Figure 6.4a, the magnetic pull on values of $N_{i-1}^{(0)}$ only depends on which stable note is closest; the number of semitones from the two nearest stable notes does not otherwise factor into the constraints. When $\alpha_{\mathrm{M_{pow}}} = 1$, as in Figure 6.4b, the calculation of magnetic forces involves inverse distances from attractors, and when $\alpha_{\mathrm{M_{pow}}} = 2$, Figure 6.4c, the calculation involves the inverse of squared distances. Because magnetism does not depend on the interval leading to $N_{i-1}^{(0)}$, the distributions displayed in this figure are identical for all six values of $I_{i-1}^{(1)}$.

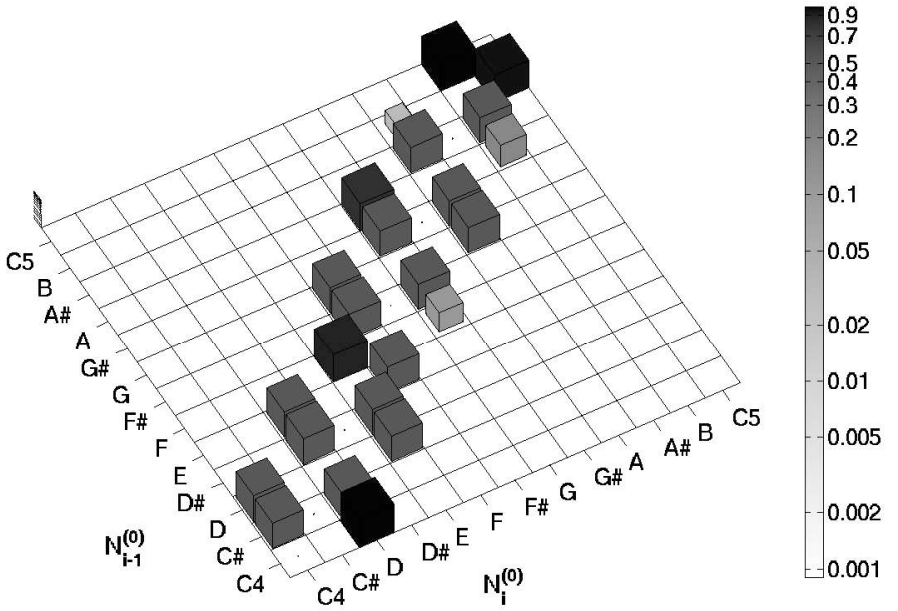


(a) Magnetism rule with $\alpha_M = 2$ and $\alpha_{\mathrm{M_{pow}}} = 0$. In addition to constraints enforcing stepwise, diatonic transitions, this solution satisfies three magnetism constraints: $2\,P(69|\mathrm{C_{Maj}}, 71, I^{(1)}_{i-1}) \leq P(72|\mathrm{C_{Maj}}, 71, I^{(1)}_{i-1})$; $2\,P(71|\mathrm{C_{Maj}}, 69, I^{(1)}_{i-1}) \leq P(67|\mathrm{C_{Maj}}, 69, I^{(1)}_{i-1})$; $2\,P(67|\mathrm{C_{Maj}}, 65, I^{(1)}_{i-1}) \leq P(64|\mathrm{C_{Maj}}, 65, I^{(1)}_{i-1})$



(b) Magnetism rule with $\alpha_M = 2$ and $\alpha_{\mathrm{M_{pow}}} = 1$. In addition to constraints enforcing stepwise, diatonic transitions, this solution satisfies three magnetism constraints: $8\,P(69|\mathrm{C_{Maj}}, 71, I^{(1)}_{i-1}) \leq P(72|\mathrm{C_{Maj}}, 71, I^{(1)}_{i-1})$; $3\,P(71|\mathrm{C_{Maj}}, 69, I^{(1)}_{i-1}) \leq P(67|\mathrm{C_{Maj}}, 69, I^{(1)}_{i-1})$; $4\,P(67|\mathrm{C_{Maj}}, 65, I^{(1)}_{i-1}) \leq P(64|\mathrm{C_{Maj}}, 65, I^{(1)}_{i-1})$

Figure 6.4: Musical magnetism, encoded using three values of $\alpha_{\mathrm{M_{pow}}}$.

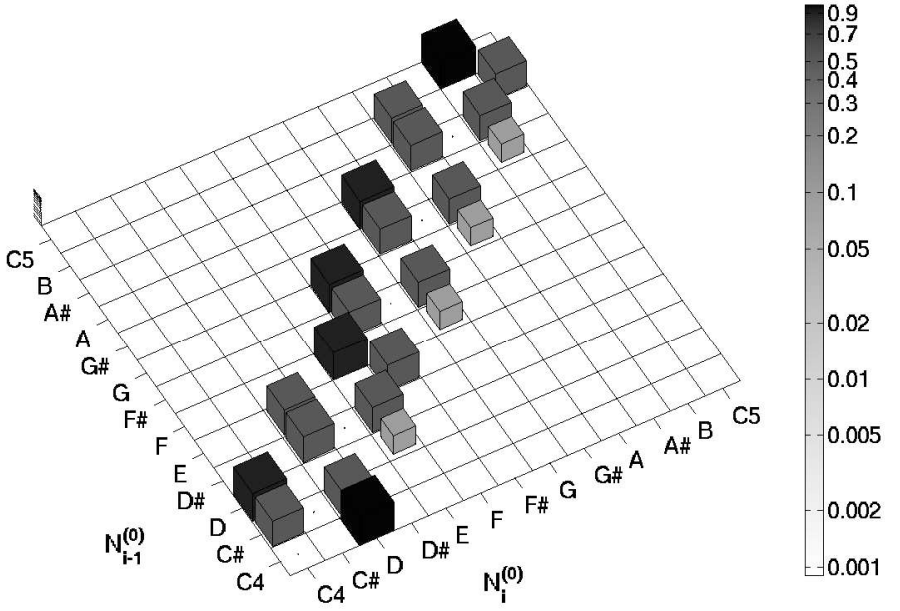


(c) Magnetism rule with $\alpha_M = 2$ and $\alpha_{\mathrm{M_{pow}}} = 2$. In addition to constraints enforcing stepwise, diatonic transitions, this solution satisfies three magnetism constraints: $32\,P(69|\mathrm{C_{Maj}}, 71, I^{(1)}_{i-1}) \leq P(72|\mathrm{C_{Maj}}, 71, I^{(1)}_{i-1})$; $\frac{9}{2}\,P(71|\mathrm{C_{Maj}}, 69, I^{(1)}_{i-1}) \leq P(67|\mathrm{C_{Maj}}, 69, I^{(1)}_{i-1})$; $8\,P(67|\mathrm{C_{Maj}}, 65, I^{(1)}_{i-1}) \leq P(64|\mathrm{C_{Maj}}, 65, I^{(1)}_{i-1})$

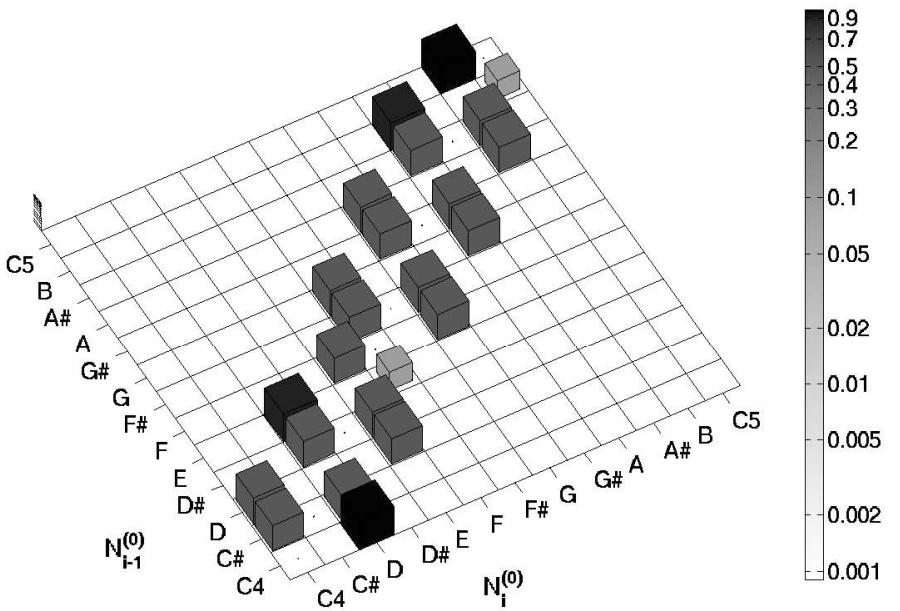
Figure 6.4: Musical magnetism, encoded using three values of $\alpha_{\mathrm{M_{pow}}}$ (cont.)

6.1.4 Inertia

Musical inertia is the tendency for a melody to continue moving in the same manner, where the interpretation of the "same manner" depends on a listener's interpretation of the melodic pattern. In general, this could mean that if listener interpreted a melody as arpeggiating a particular chord, the listener would expect the melody to continue outlining chord tones in the same direction. Because we limited the interpretation to diatonic stepwise motion, the inertia rule might be stated as, "If a melody reaches a note by diatonic step, it is at least α_I times as likely to take another diatonic step in the same direction as it is to return to the previous note." This rule can be encoded using the two sets of constraints, the first for ascending melodies, and

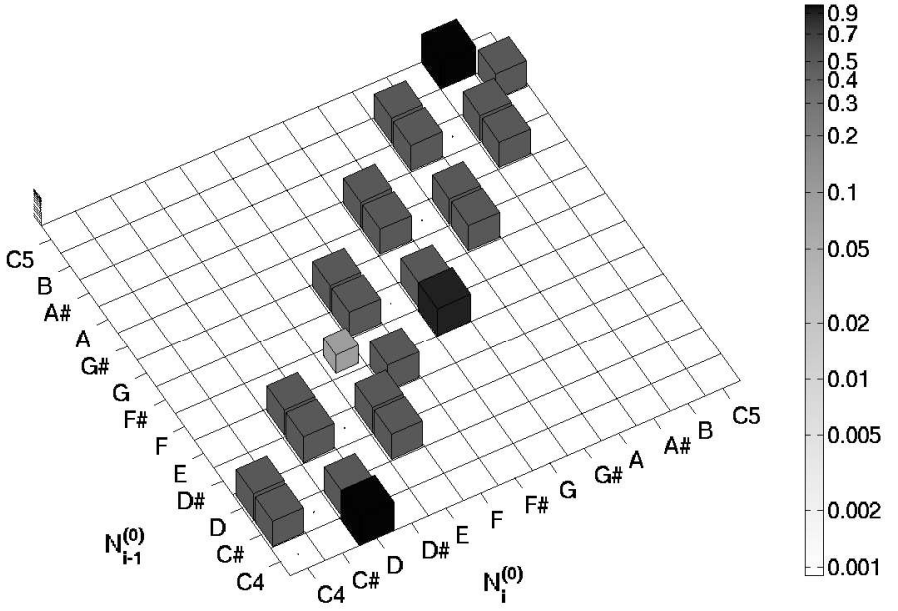


(a) Inertia rule with $\alpha_I = 10$ and $i^{(1)}_{i-1} = -2$. In addition to constraints enforcing stepwise, diatonic transitions, this slice of the distribution satisfies four inertia constraints: $10\,P(71|\mathrm{C_{Maj}}, 69, -2) \leq P(67|\mathrm{C_{Maj}}, 69, -2)$; $10\,P(69|\mathrm{C_{Maj}}, 67, -2) \leq P(65|\mathrm{C_{Maj}}, 67, -2)$; $10\,P(67|\mathrm{C_{Maj}}, 65, -2) \leq P(64|\mathrm{C_{Maj}}, 65, -2)$; $10\,P(64|\mathrm{C_{Maj}}, 62, -2) \leq P(60|\mathrm{C_{Maj}}, 62, -2)$

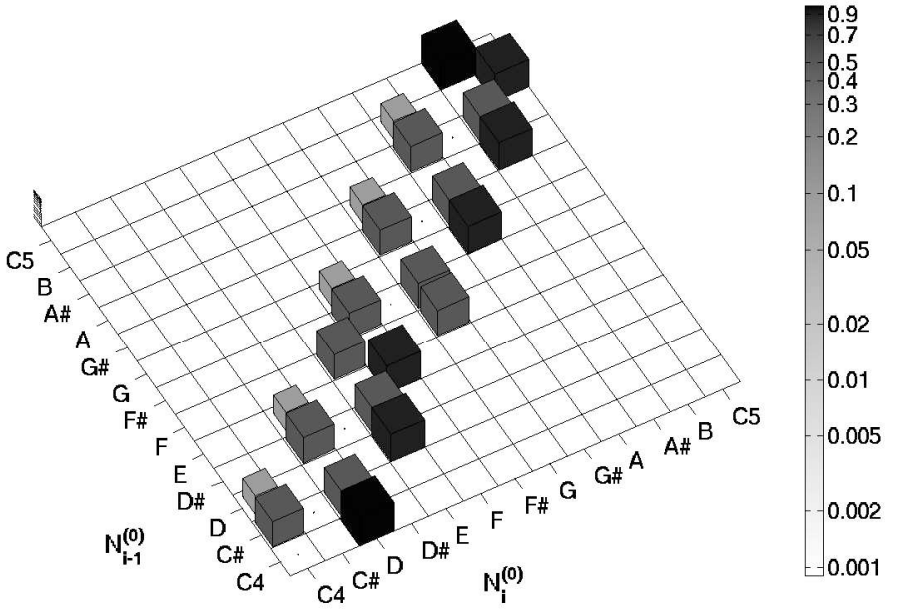


(b) Inertia rule with $\alpha_I = 10$ and $i^{(1)}_{i-1} = -1$. In addition to constraints enforcing stepwise, diatonic transitions, this slice of the distribution satisfies two inertia constraints: $10\,P(72|\mathrm{C_{Maj}}, 71, -1) \leq P(69|\mathrm{C_{Maj}}, 71, -1)$; $10\,P(65|\mathrm{C_{Maj}}, 64, -1) \leq P(62|\mathrm{C_{Maj}}, 64, -1)$

Figure 6.5: Musical inertia, encoded using $\alpha_I = 10$



(c) Inertia rule with $\alpha_I = 10$ and $i^{(1)}_{i-1} = +1$. In addition to constraints enforcing stepwise, diatonic transitions, this slice of the distribution satisfies one inertia constraints: $10\,P(64|\mathrm{C_{Maj}}, 65, +1) \leq P(67|\mathrm{C_{Maj}}, 65, +1)$



(d) Inertia rule with $\alpha_I = 10$ and $i^{(1)}_{i-1} = +2$. In addition to constraints enforcing stepwise, diatonic transitions, this slice of the distribution satisfies four inertia constraints: $10\,P(60|\mathrm{C_{Maj}}, 62, +2) \leq P(64|\mathrm{C_{Maj}}, 62, +2)$; $10\,P(62|\mathrm{C_{Maj}}, 64, +2) \leq P(65|\mathrm{C_{Maj}}, 64, +2)$; $10\,P(65|\mathrm{C_{Maj}}, 67, +2) \leq P(69|\mathrm{C_{Maj}}, 67, +2)$; $10\,P(69|\mathrm{C_{Maj}}, 71, +2) \leq P(72|\mathrm{C_{Maj}}, 71, +2)$

Figure 6.5: Musical inertia, encoded using $\alpha_I = 10$ (cont.)

the second for descending melodies:

$$\alpha_I \, P(N_i^{(0)} = \mathrm{D}^-(k,n) \mid K_i = k, N_{i-1}^{(0)} = n, I_{i-1}^{(1)} = n - \mathrm{D}^-(k,n)) \\ \leq P(N_i^{(0)} = \mathrm{D}^+(k,n) \mid K_i = k, N_{i-1}^{(0)} = n, I_{i-1}^{(1)} = n - \mathrm{D}^-(k,n)) \quad (6.11)$$

$$\alpha_I \, P(N_i^{(0)} = \mathrm{D}^+(k,n) \mid K_i = k, N_{i-1}^{(0)} = n, I_{i-1}^{(1)} = n - \mathrm{D}^+(k,n)) \\ \leq P(N_i^{(0)} = \mathrm{D}^-(k,n) \mid K_i = k, N_{i-1}^{(0)} = n, I_{i-1}^{(1)} = n - \mathrm{D}^+(k,n)) \quad (6.12)$$

for all key values k and note values n. To display example results, we again let $k = \mathrm{C}_{\mathrm{Maj}}$, and let $\alpha_I = 10$, so results are easily visible in Figure 6.5, which displays slices of the the maximum entropy rate distribution corresponding each value of $i_{i-1}^{(1)} \neq other$. Because we only encode stepwise diatonic constraints, the slice of the distribution with $i_{i-1}^{(1)} = other$ is identical to the distribution in Figure 6.1.

6.2 Rule inference results

Recall that the random variable R_i is a switching state that selects which CPD governs the note state transition at time i. If we observe a sequence of notes $x_{1:i}$, then the filtered posterior distribution $P(R_i \mid x_{1:i})$ reflects the weighted participation of all three rules in the determination of x_i. Larson and Van Handel [48] study the response of human listeners to eight three-note patterns. We specify the following set of constraint values, which will be used to compute filtered rule posterior corresponding to each of those eight patterns:

- $\alpha_{nondiatonic} = .001$: the probability of any nondiatonic note is less than or equal to 0.001.
- $\alpha_{step} = 1000$: diatonic stepwise motion is 1000 times more likely than other intervals; repeated notes are a special case, and are not constrained.
- $\alpha_{repeat} = .001$: the probability of repeating a note is less than or equal to 0.001.

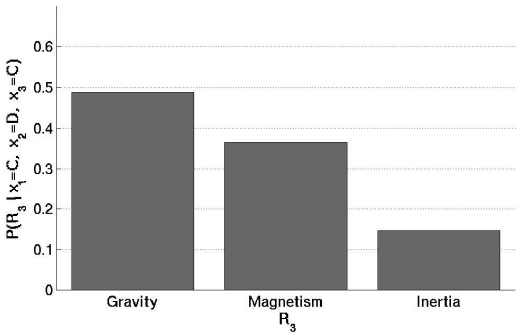
- $\alpha_G = 2$: a note a step above a stable platform is at least twice a likely to move downward by diatonic step.

- $\alpha_{\mathrm{M}} = 1$, $\alpha_{\mathrm{M}_{\mathrm{pow}}} = 2$: calculation of magnetism involves squared distances; the leading tone, with $\alpha_{\mathrm{M}} = 1$ is at least 16 times as likely to resolve to the tonic than step down to the note A.

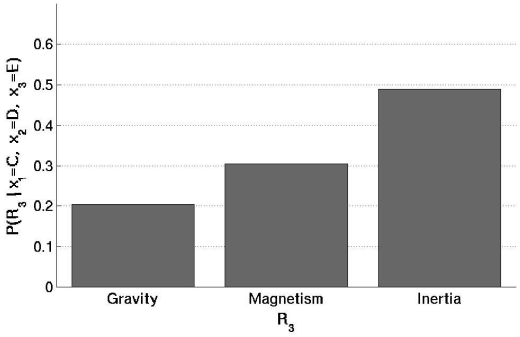
- $\alpha_I = 4$: when inertia is active, the probability of taking another diatonic step in the same direction is at least 4 times as much as the probability of moving opposite inertia.

Nondiatonic, stepwise, and repeat constraints are always active and are combined with each of the three individual musical forces. Figure 6.6 displays the filtered rule posterior $P(R_3 \,|\, x_{1:3})$ corresponding to each of eight three-note patterns. Most of the distributions appear exactly as one would expect, but a couple of them require an explanation. In Figure 6.6d, the magnetism force achieves the maximum value of the posterior, despite that magnetism has no effect on the note D is equidistant from the two nearest stable notes C and E. Because we do not encode any magnetism constraints for the note D, either diatonic step can be chosen with probability near 0.5 without violating the rule. The other two rules, however, are explicitly violated; motion from D $\nearrow$ E is opposite gravity and opposite the inertia implied by the first two notes of that pattern. Similarly, gravity achieves the greatest posterior value in Figure 6.6h, despite the fact that motion from F $\nearrow G$ explicitly violates the direction of encoded gravity. This is because this three-note pattern violates all three of the rules, and the magnetism and inertia forces are specified with α values that make them stronger, so the weakest force is the one least violated, and the one maximizing the posterior.

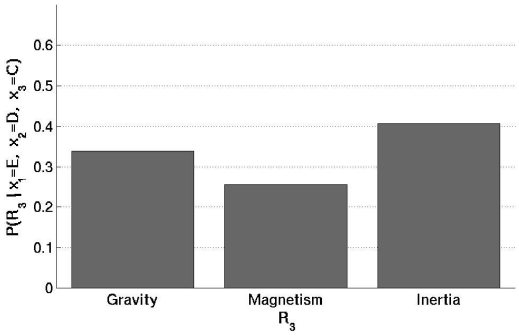
Future research will involve coding more generalized versions of these forces, relaxing the stepwise constraints and allowing inertia to take on other interpretations, as dictated by the musical context.



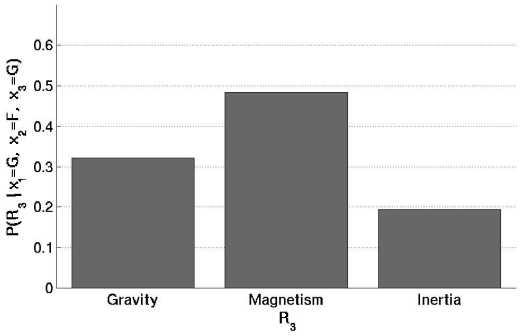
(a) $P(R_3 \mid x_1 = \mathrm{C}, x_2 = \mathrm{D}, x_3 = \mathrm{C})$



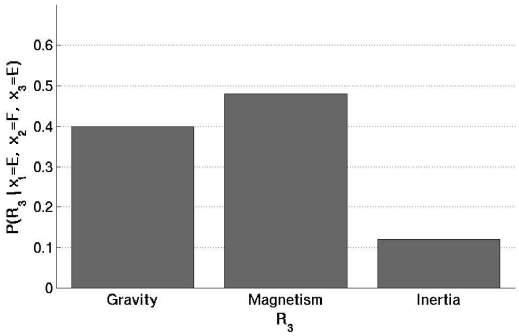
(b) $P(R_3 \mid x_1 = \mathrm{C}, x_2 = \mathrm{D}, x_3 = \mathrm{E})$



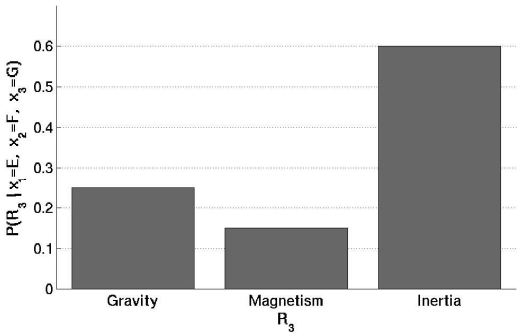
(c) $P(R_3 \mid x_1 = \mathrm{E}, x_2 = \mathrm{D}, x_3 = \mathrm{C})$



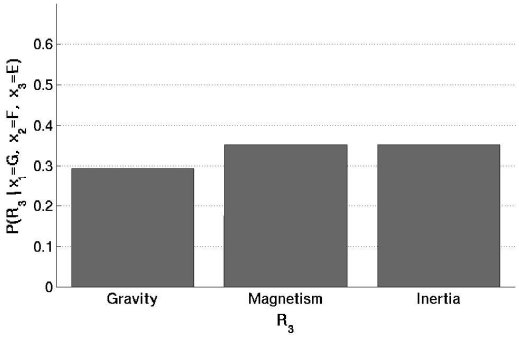
(d) $P(R_3 \mid x_1 = \mathrm{E}, x_2 = \mathrm{D}, x_3 = \mathrm{E})$



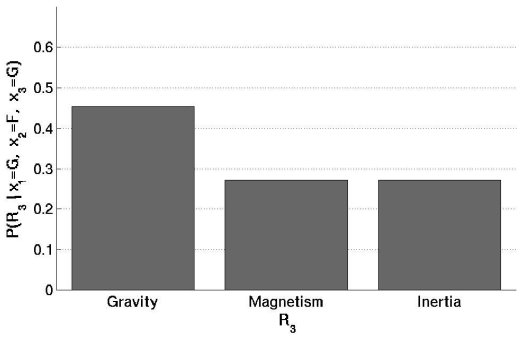
(e) $P(R_3 \mid x_1 = \mathrm{E}, x_2 = \mathrm{F}, x_3 = \mathrm{E})$



(f) $P(R_3 \mid x_1 = \mathrm{E}, x_2 = \mathrm{F}, x_3 = \mathrm{G})$



(g) $P(R_3 \mid x_1 = \mathrm{G}, x_2 = \mathrm{F}, x_3 = \mathrm{E})$



(h) $P(R_3 \mid x_1 = \mathrm{G}, x_2 = \mathrm{F}, x_3 = \mathrm{G})$

Figure 6.6: Activation and violation of musical forces

Chapter 7

Extending Model Hierarchy

All of the probabilistic models presented in preceding chapters contain only a single hidden variable. In Section 3.3, the hidden state represents musical mode, and in Chapter 6, the hidden state is a switching state which probabilistically selects from among a set of note-to-note transition distributions corresponding to encoded musical rules. This chapter extends the basic AR-HMM model by adding several hidden variables, producing a hierarchic model in which note observations depend on harmonic and metric context. This is an important extension, because it allows us to generate expectations not only about *what* events will occur, but also *when* those events will occur.

As discussed in Chapter 2, a hierarchic DBN is capable of encoding arbitrarily complex musical relationships. This chapter does not attempt to present an all-encompassing model of harmony and meter, but rather to demonstrate the power of encoding just two musical tendencies, both related to the recurrent patterns of strong and weak beats from which musical meter emerges:

1. Chord changes occur more frequently on strong beats than on weak beats.

2. Notes occurring on strong beats are more likely to be members of the current chord than notes occurring on weaker beats.

Encoding these two tendencies in a hierarchic generative model allows us to use Bayesian inference to invert the relationships, to infer harmony, meter, and beat

position from the sequence of observed notes.

7.1 Variable definitions

Figure 7.1 displays the directed acyclic graph for a hierarchic model in which the hidden state S_i has been expanded to include multiple variables that take into account harmony, meter, beat position, and observed note durations. Specific roles of each of the following variables will be explained in the Section 7.2, which details the probabilistic relationships among them:

- $\boldsymbol{L_i}$ – ***Meter***: L_i takes values from a discrete space of possible meters, $\mathcal{L}$; e.g. $\mathcal{L} = \{2/4,\ 5/8,\ 3/4,\ 7/8,\ 4/4\}$

- $\boldsymbol{B_i}$ – ***Beat position***: B_i is represented as the position in an equally-spaced timing grid whose resolution equals the smallest possible inter-arrival time between two notes. The smallest inter-arrival time equals the period of the fastest possible musical pulse, called the *tatum* [11].[1] The tatum period is measured in units relative to the duration of a quarter note. The number of positive integers in the beat space, $\mathcal{B}$, is equal to the smallest number of tatum periods that can completely cover the meter with largest number of (possibly-fractional) quarter-notes in $\mathcal{L}$; i.e., if the number of tatum periods it takes to fill a measure with a given meter $l \in \mathcal{L}$ is denoted $|l|$, then $\mathcal{B} = \{1, 2, \ldots, \max_{l \in \mathcal{L}} |l|\}$ This means that some values in $\mathcal{B}$ will be invalid given some values in $\mathcal{L}$; e.g., if $\mathcal{L}$ contains $2/4$ and $4/4$, then the second half of the possible values of B_i will have zero probability when $L_i = 2/4$.

  To provide an example, if we consider a single meter, $4/4$, and restrict possible note durations to whole-notes, half-notes, eighth-notes, and sixteenth-notes, then the tatum is $1/4$ of a quarter note, and the beat space contains the sixteen positive integers $1:16$. A note on the third beat of any bar would correspond to $B_i = 9$. If we add an eighth-note triplet to the possible set of

[1] Blimes [11] explains that he coined the term tatum to mean *t*emporal *atom*, re-spelled in honor of Art Tatum, perhaps the greatest jazz pianist this world has known.

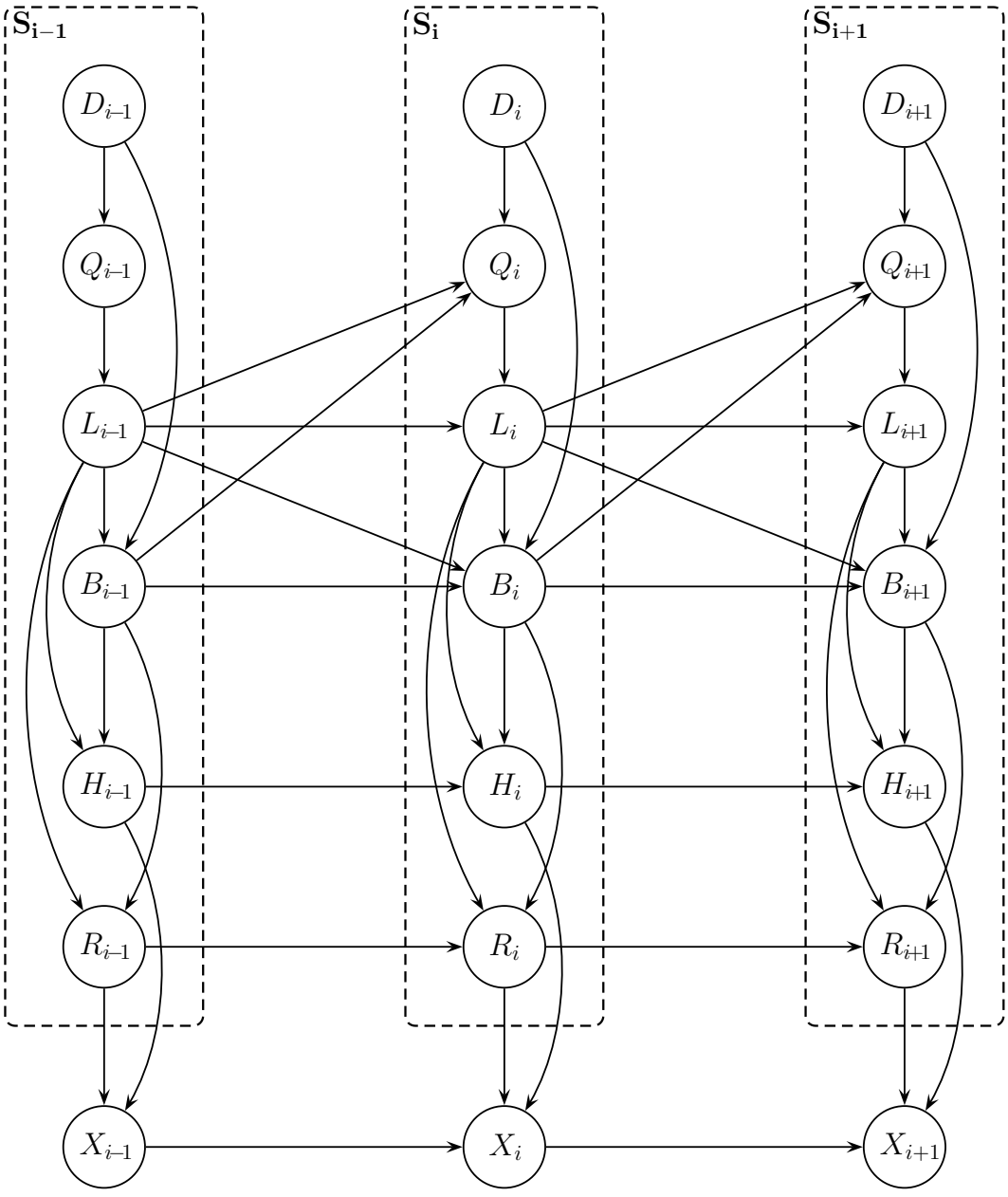


Figure 7.1: DAG for hierarchical model with harmony, meter, and beat position

note durations, then the tatum period is reduced to 1/12, the beat space has three times as many elements, numbered $1:48$, and a note on the third beat would correspond to $B_i = 25$.

- $\boldsymbol{D_i}$ – ***Observed duration***: d_i is the observed duration of the *previous* note at index $i-1$, in tatum units. We will consistently set the value of d_1 equal to zero, since no measurable note comes before it. We choose for now not to explicitly model rests, so the duration of any rest is added to the duration of the note that directly precedes it. If, for example, the beat space contains the 48 tatum positions resulting from modeling music in 4/4 meter with whole-notes, half-notes, quarter-notes, eighth-notes, and eighth-note-triplets, then the observed durations corresponding to the "Shave and a Haircut" melody in Figure 2.8 would be $d_{1:8} = [0, 12, 4, 4, 4, 12, 24, 12]$.

- $\boldsymbol{Q_i}$ – ***Bar crossing indicator***: Q_i is a boolean-valued variable that is *true* if the note at time index i is in a different measure than the note at index $i-1$, and $false$ otherwise. The value of Q_i depends deterministically on the values of observed duration, d_i, previous meter, l_{i-1}, and previous beat position, b_{i-1}.

- $\boldsymbol{H_i}$ – ***Harmony***: H_i represents a chord relative to a single, fixed key. Chords are selected from a finite set of chords, $\mathcal{H}$; e.g. if the key is a major key, the space of chords might be defined as $\mathcal{H} = \{I, ii, iii, IV, V, vi, vii^{\circ}, V^7, V/V\}$. In this example model, we do not attempt to model chord inversions, because our note transition rules only depend on whether a given note is in a particular chord, but not on the voicing of the chord. A straightforward extension to this model would be to add a key variable, which would allow modeling key modulations and chord-key interaction; that extension is left as future research.

- $\boldsymbol{R_i}$ – ***Musical rule***: R_i acts as a switching state, as described in Chapter 6, selecting from a set of maximum entropy rate note transition distributions.

- $\boldsymbol{X_i}$ – ***Observed note state***: As described in Section 2.3.2, X_i can comprise either a single note or a multi-component state memorizing a history of previous notes and intervals.

7.2 Model factorization and distributional specifications

Looking at Figure 7.1, we can just move node-by-node through the DAG and read off the model factorization as:

$$P(X_{1:K}, S_{1:K}) = P(X_1 \mid S_1)P(S_1)\prod_{i=2}^{K} P(X_i \mid S_i, X_{i-1})P(S_i \mid S_{i-1}) \tag{7.1}$$

$$\begin{aligned} = P(X_1|R_1, H_1)P(R_1|L_1, B_1)P(H_1|L_1, B_1)P(B_1|L_1, D_1)P(L_1|Q_1)P(Q_1|D_1)P(D_1)\times \\ \prod_{i=2}^{K}\Big[P(X_i \mid X_{i-1}, R_i, H_i)P(R_i \mid R_{i-1}, L_i, B_i)P(H_i \mid H_{i-1}, L_i, B_i)\times \\ P(B_i \mid B_{i-1}, L_{i-1}, L_i, D_i)P(L_i \mid L_{i-1}, Q_i)P(Q_i \mid L_{i-1}, B_{i-1}, D_i)P(D_i)\Big] \end{aligned} \tag{7.2}$$

where the graph has been unrolled to model a melody K notes long.

7.2.1 Chord membership and context-weighted rules

In order to specify all of the local probability models in (7.2), we must first explain a new musical rule which, when active, enables harmonic context to affect note transitions. In previously demostrated models, containing only a single hidden layer, only the value of the previous rule R_{i-1} could affect R_i. In this extended model, the rule transition CPD is of the form $P(R_i \mid R_{i-1}, L_i, B_i)$. This additional context allows us the ability to create an absolute rule stating, "The observed note at time i is, without exception, a member of the harmony at time i," then encode a dependence on L_i and B_i that determines the probability that this rigid chord-member rule is activated.

Using a concept presented in Lerdahl [50], we map each meter/beat pair, $\{l, b\}$, to a *metric level*, which rates the relative strengths of beats in a given meter; this mapping is accomplished using the function $MetricLevel(l, b)$. In the meter 4/4, for example, if we model all duple subdivisions down to the granularity of the sixteenth note, we obtain a beat space with 16 positive integers, each assigned to one of five

metric levels:

$$MetricLevel(4/4, b) = \begin{cases} 1 & \text{if } b = 1 \\ 2 & \text{if } b = 9 \\ 3 & \text{if } b \in \{5, 13\} \\ 4 & \text{if } b \in \{3, 7, 11, 15\} \\ 5 & \text{if } b \in \{2, 4, 6, 8, 10, 12, 14, 16\} \end{cases} \tag{7.3}$$

We similarly define $MetricLevel(l, b)$ for all meters $l \in \mathcal{L}$ and valid beats in those meters. The dependence of observed notes on harmony takes form when we assert that as the metric level decreases (from weaker to stronger beat positions), the probability that an observed note is a member of the current harmony increases. For example, notes occurring on any of the weakest sixteenth-notes (metric level 5), are more likely to be passing tones than notes falling on downbeats of measures (metric level 1).

The strength of the chord-member rule relative to other encoded musical tendencies is determined by the probability of its activation at each metric level. Recall from Chapter 6 that rules may act in combination with other rules, so the activation probability is distributed evenly among all rule combinations in which the chord-member rule participates. Suppose, for example, that we have encoded two individal rules: chord-member, denoted using $r_i = C$, and repeat, denoted using $r_i = R$. These two rules can combine to create a third rule, $r_i = CR$. Using the levels defined in (7.3), we might, for example, specify the strength of the chord-member rule by asserting that:

$$\sum_{r_i \in \{C, CR\}} P(R_i = r_i \,|\, r_{i-1}, l_i, b_i) = \frac{3}{4\sqrt{MetricLevel(l_i, b_i)}} \tag{7.4}$$

As a result, we would expect the chord-member rule, alone or in combination, to be activated on the downbeat 3/4 of the time, and on the second sixteenth-note roughly 1/3 of the time. The third rule, not involving C, absorbs any remaining probability. To implement these probabilistic relationships, we assume for now that

R_i is independent of R_{i-1}, and let the rule transition CPD, $P(R_i \mid R_{i-1}, L_i, B_i)$, switch among several different metric-level-specific distributions. Figure 7.2 displays the five CPDs produced as a result of the strength specification in (7.30).

7.2.2 Distributional specifications

In order to complete the specification of this hierarchical DBN, we must quantify each of the factors in (7.2). Several of the transition distributions define deterministic relationships that enforce constraints of the musical structure, for example, guaranteeing that a note's starting beat position is not greater than the total number of beats in the measure. Other distributions are user-specified, so we will state the choices used to produce the results in Section 7.4.

Prior distributions

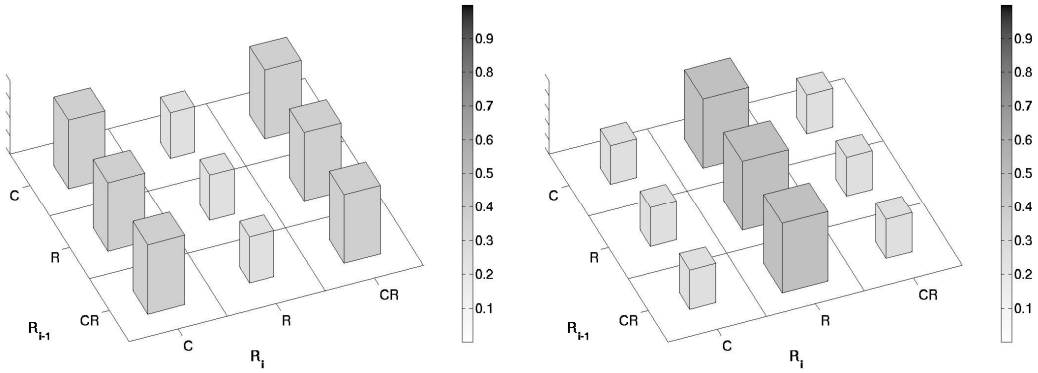
We specify the following prior distributions, several of which simplify the corresponding factors in (7.2):

- $\boldsymbol{P(D_1)}$: Recall that the value of D_i is equal to the duration of the observed note at index $i-1$, so because there are no notes prior to X_1, the value D_1 is irrelevant.

- $\boldsymbol{P(Q_1)}$: We assert that a piece can only start with the bar crossing indicator set to true; i.e., $P(Q_1 = true) = 1$.

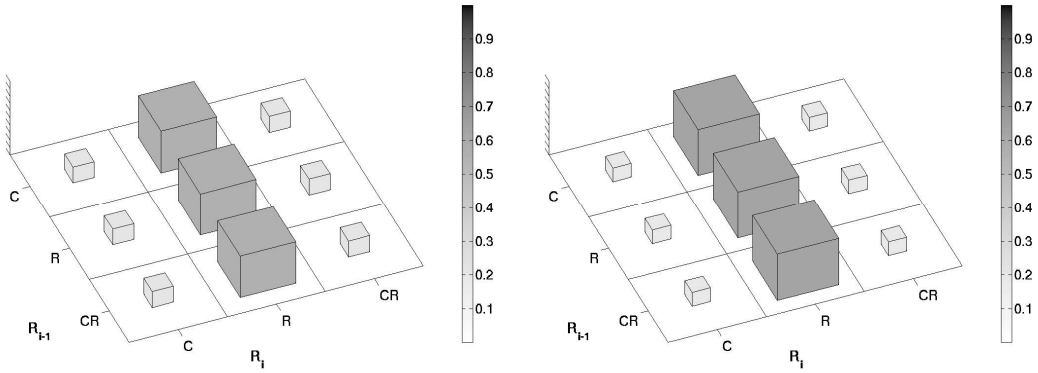
- $\boldsymbol{P(L_1)}$: If a user of the model has a reason to favor certain meters over others (e.g., the user is processing a corpus of waltzes), $P(L_1)$ could reflect that knowledge. Otherwise, as used to produce example results in Section 7.4, we specify that meter is unknown at the start of a piece, and distribute $P(L_1)$ uniformly.

- $\boldsymbol{P(B_1|L_1)}$: For a given meter, all invalid beat positions are assigned zero probability. From among the valid beat positions, a user might have a reason to favor certain choices at the start of a piece. For example, a particular type of



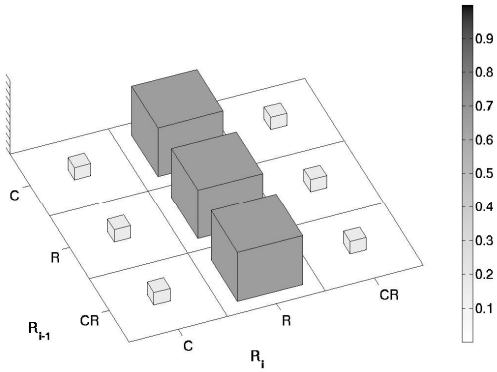
(a) $MetricLevel(l_i, b_i) = 1$
$\sum_{r \in \{C,CR\}} P(R_i = r \,|\, r_{i-1}, l_i, b_i) = 0.375$

(b) $MetricLevel(l_i, b_i) = 2$
$\sum_{r \in \{C,CR\}} P(R_i = r \,|\, r_{i-1}, l_i, b_i) = 0.265$



(c) $MetricLevel(l_i, b_i) = 1$
$\sum_{r \in \{C,CR\}} P(R_i = r \,|\, r_{i-1}, l_i, b_i) = 0.217$

(d) $MetricLevel(l_i, b_i) = 2$
$\sum_{r \in \{C,CR\}} P(R_i = r \,|\, r_{i-1}, l_i, b_i) = 0.188$



(e) $MetricLevel(l_i, b_i) = 2$
$\sum_{r \in \{C,CR\}} P(R_i = r \,|\, r_{i-1}, l_i, b_i) = 0.168$

Figure 7.2: Rule transition distributions for several metric levels

music might tend to start on up-beats. To produce the example results in Section 7.4, we remain maximally noncommittal, distributing valid beat positions uniformly given each value of L_1.

- $\boldsymbol{P(H_1|L_1, B_1)}$: At the start of a piece, certain beat positions might favor certain chords; e.g., a composer might favor particular chords when $\{L_1, B_1\}$ indicates that the first note is a pick-up. In Section 7.4, we choose a uniform harmony prior, independent of the values of L_i and B_i.

- $\boldsymbol{P(R_1|L_1, B_1)}$: Comprises several metric-level-specific distributions, as discussed in Section 7.2.1 and displayed in Figure 7.2.

- $\boldsymbol{P(X_1|H_i, R_i)}$: We simplify this factor by assuming that X_1 is independent of R_1 and H_1, and distribute $P(X_1)$ uniformly.

Transition distributions

The following collection of conditional probability distributions completes the specification of the model. Each of these local probability models is associated with one node in the directed acyclyc graph displayed in Figure 7.1:

- $\boldsymbol{P(Q_i \mid B_{i-1}, B_i, D_i)}$: The bar crossing distribution encodes a deterministic function calculating whether the previous note ended in the same measure in which it began:

$$P(Q_i{=}true \mid b_{i-1}, l_{i-1}, d_i) = \begin{cases} 1 & \text{if } b_i + d_i > |l_i| \\ 0 & \text{otherwise.} \end{cases} \tag{7.5}$$

$$P(Q_i{=}false \mid b_{i-1}, l_{i-1}, d_i) = 1 - P(Q_i{=}true \mid b_{i-1}, l_{i-1}, d_i) \tag{7.6}$$

- $\boldsymbol{P(L_i \mid L_{i-1}, Q_i)}$: The meter transition distribution is constrained so that the meter, M_i, can only change at the start of a measure, which occurs whenever $Q_i = true$. When $Q_i = true$, we specify the probability of changing the meter as α_{meter}, and uniformly distribute that probability among all meter values where

where $l_i \neq l_{i-1}$; the probability of staying in the same meter is thus $1 - \alpha_{meter}$.

$$P(l_i \,|\, l_{i-1}, Q_i = false) = \begin{cases} 1 & \text{if } l_i = l_{i-1} \\ 0 & \text{otherwise} \end{cases} \tag{7.7}$$

$$P(l_i \,|\, l_{i-1}, Q_i = true) = \begin{cases} 1 - \alpha_{meter} & \text{if } l_i = l_{i-1} \\ \frac{\alpha_{meter}}{|\mathcal{L}|-1} & \text{otherwise.} \end{cases} \tag{7.8}$$

- $\boldsymbol{P(B_i \,|\, B_{i-1}, L_i, L_{i-1}, D_i)}$: A deterministic relationship advances the beat position according to the following equation:

$$P(b_i \,|\, b_{i-1}, l_i, l_{i-1}, d_i) = \begin{cases} 1 & \text{if } b_i = \text{mod}(b_{i-1} + d_i - |l_{i-1}| - 1, |l_i|) + 1 \\ 0 & \text{otherwise} \end{cases} \tag{7.9}$$

  For example, crossing the bar line with a meter change, if $|l_{i-1}| = 3$, $b_{i-1} = 3$, $|l_i| = 4$, and $d_i = 2$, then $b_i = \text{mod}(3 + 2 - 3 - 1, 4) + 1 = 2$. Another example, within the same measure and meter, if $|l_{i-1}| = |l_i| = 4$, $b_{i-1} = 1$, and $d_i = 2$, then $b_i = \text{mod}(1 + 2 - 4 - 1, 4) + 1 = 3$. In cases where $b_{i-1} + d_i > |l_{i-1}| + |l_{i-1}|$, we can't in general know which sequence of meters covers the time spanned by d_i. Equation 7.9 encodes the assumption that a transition to meter l_i occurs as soon as possible after the observed note x_{i-1}, and that l_i continues until the observation x_i.

- $\boldsymbol{P(H_i \,|\, H_{i-1}, L_i, B_i)}$: We assign to each value of $MetricLevel(l_i, b_i)$, a probability that the harmony changes at that metric level. The lower the metric level (stronger beats), the greater the probability of harmony change. It is not likely, for example, that the harmony will change on an weak sixteenth note. If $P_{change}(l_i, b_i)$ denotes the probability of a harmony change at $MetricLevel(l_i, b_i)$, then:

$$P(h_i \,|\, h_{i-1}, l_i, b_i) = \begin{cases} 1 - P_{change}(l_i, b_i) & \textit{if } h_i = h_{i-1} \\ P_{change}(l_i, b_i)/(|\mathcal{H}| - 1) & \text{otherwise} \end{cases} \tag{7.10}$$

- $\boldsymbol{P(R_i \mid R_{i-1}, L_i, B_i)}$: See Section 7.2.1 for a detailed discussion of the rule transition probability.
- $\boldsymbol{P(X_i \mid X_{i-1}, R_i, H_i)}$: Note transitions behave as in earlier chapters, with the addition of the new chord-member rule discussed in Section 7.2.1

7.3 Prediction, smoothing, and filtering computations

All of the computations for this more complex hierarchical model exactly parallel the computational steps presented in Section . To improve the readibility of equations in this section, we adopt a shorthand notation in which multiple variables with the same time index can be grouped together as a single text string sharing that common subscript. For example,

$$P(\mathit{DQLBHR}_i) \triangleq P(D_i, Q_i, L_i, B_i, H_i, R_i) \tag{7.11}$$

In each of the steps below, we restate the corresponding inference equation for the basic AR-HMM, then expand the hidden state.

7.3.1 Forward filtering pass

At the start of each filtering iteration, we assume that we can access $P(\mathit{DQLBHR}_i \mid x_{1:i})$; this is initialized at the start of the piece by the prior $P(\mathit{DQLBHR}_1)$. At the end of each iteration, we obtain $P(\mathit{DQLBHR}_{i+1} \mid x_{1:i+1})$, satisfying the starting assumption for the next time step. Although we group D_i with the hidden variables in S_i, so equations correspond directly to those in Chapter 3, all durations are assumed to be in evidence. This is indicated in the following inference steps using a lowercase d_i.

1. Start the time update by bringing in the hidden state at time $i+1$.

$$P(S_i, S_{i+1} \mid x_{1:i}) = P(S_{i+1} \mid S_i) P(S_i \mid x_{1:i}) \tag{7.12}$$

$$P(DQLBHR_i, DQLBHR_{i+1} \mid x_{1:i})$$
$$= P(DQLBHR_{i+1} \mid DQLBHR_i) P(DQLBHR_i \mid x_{1:i}) \tag{7.13}$$
$$= P(R_i \mid R_{i-1}, L_i, B_i) P(H_i \mid H_{i-1}, L_{i-1}, B_i) \times$$
$$P(B_i \mid B_{i-1}, L_{i-1}, L_i, d_i) P(L_i \mid L_{i-1}, Q_i) P(Q_i \mid L_{i-1}, B_{i-1}, d_i) \tag{7.14}$$

2. Marginalize out $DQLBHR_i$ to produce a one-step state prediction:

$$P(S_{i+1} \mid x_{1:i}) = \sum_{S_i} P(S_i, S_{i+1} \mid x_{1:i}) \tag{7.15}$$

$$P(DQLBHR_{i+1} \mid x_{1:i}) = \sum_{DQLBHR_i} P(DQLBHR_i, DQLBHR_{i+1} \mid x_{1:i}) \tag{7.16}$$

3. Start the measurement update by introducing the observation distribution (no actual note observation at time $i+1$ yet):

$$P(S_i, S_{i+1}, X_{i+1} \mid x_{1:i}) = P(S_i, S_{i+1} \mid x_{1:i}) P(X_{i+1} \mid x_i, S_{i+1}) \tag{7.17}$$

$$P(DQLBHR_i, DQLBHR_{i+1}, X_{i+1} \mid x_{1:i})$$
$$= P(DQLBHR_i, DQLBHR_{i+1} \mid x_{1:i}) P(X_{i+1} \mid x_i, DQLBHR_{i+1}) \tag{7.18}$$
$$= P(DQLBHR_i, DQLBHR_{i+1} \mid x_{1:i}) P(X_{i+1} \mid x_i, H_{i+1}, R_{i+1}) \tag{7.19}$$

4. Compute a one-step observation prediction by marginalizing out the hidden states:

$$P(X_{i+1} \mid x_{1:i}) = \sum_{S_i, S_{i+1}} P(S_i, S_{i+1}, X_{i+1} \mid x_{1:i}) \tag{7.20}$$
$$= \sum_{DQLBHR_i, DQLBHR_{i+1}} P(DQLBHR_i, DQLBHR_{i+1}, X_{i+1} \mid x_{1:i}) \tag{7.21}$$

5. Observe x_{i+1}

6. Compute the surprisal of that observation by evaluating the one-step observation prediction $P(X_{i+1} \mid x_{1:i})$ at the observed note x_{i+1}:

$$\text{Surprisal}(x_{i+1} \mid x_{1:i}) = -\log_2 P(X_{i+1} = x_{i+1} \mid x_{1:i}) \tag{7.22}$$

7. Store the following factor, a smoothed two-slice estimate, for use later the backward smoothing pass:

$$P(S_i, S_{i+1} \mid x_{1:i+1}) = \frac{P(S_i, S_{i+1}, x_{i+1} \mid x_{1:i})}{P(x_{i+1} \mid x_{1:i})} \tag{7.23}$$

$$P(DQLBHR_i, DQLBHR_{i+1} \mid x_{1:i+1}) = \frac{P(DQLBHR_i, DQLBHR_{i+1}, x_{i+1} \mid x_{1:i})}{P(x_{i+1} \mid x_{1:i})} \tag{7.24}$$

8. Obtain the filtered posterior, and store it for use later in smoothing:

$$P(S_{i+1} \mid x_{1:i+1}) = \sum_{S_i} P(S_i, S_{i+1} \mid x_{1:i+1}) \tag{7.25}$$

$$P(DQLBHR_{i+1} \mid x_{1:i+1}) = \sum_{DQLBHR_i} P(DQLBHR_i, DQLBHR_{i+1} \mid x_{1:i+1}) \tag{7.26}$$

7.3.2 Backward, offline smoothing pass

To compute the smoothed posterior, we assume that we have $P(DQLBHR_{i+1} \mid x_{1:K})$ at the start of each iteration, and use the factors stored during the filtering phase, (7.23) and (7.25), to compute $P(DQLBHR_i \mid x_{1:K})$. We initialize the process with the filtered posterior from the last time index in the piece, $P(DQLBHR_K \mid x_{1:K})$, then step backward note-by-note using the following recursion:

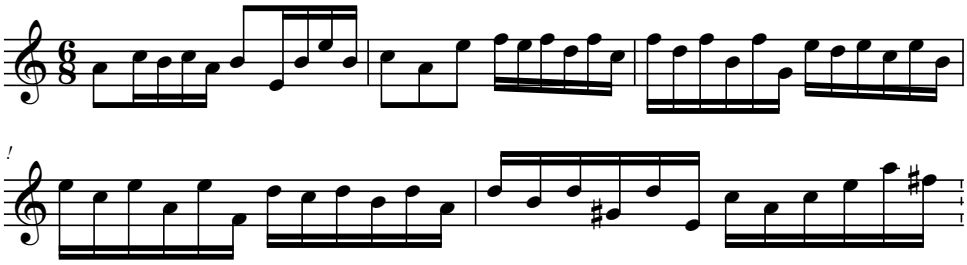


Figure 7.3: First five bars of the fugue from *Prelude and Fugue in A Minor, BWV 543* by J.S. Bach.

$$P(S_i \,|\, x_{1:K}) = \sum_{S_{i+1}} P(S_{i+1} \,|\, x_{1:K}) \frac{P(S_i, S_{i+1} \,|\, x_{1:i+1})}{P(S_{i+1} \,|\, x_{1:i+1})} \tag{7.27}$$

$$P(\mathit{DQLBHR}_i \,|\, x_{1:K}) = \sum_{\mathit{DQLBHR}_{i+1}} P(\mathit{DQLBHR}_{i+1} \,|\, x_{1:K}) \frac{P(\mathit{DQLBHR}_i, \mathit{DQLBHR}_{i+1} \,|\, x_{1:i+1})}{P(\mathit{DQLBHR}_{i+1} \,|\, x_{1:i+1})} \tag{7.28}$$

7.4 A musical example involving harmony and meter

We apply this hierarchical model to a foot-tapping application, in which the simulated listener's objective is to determine the beat position. Figure 7.3 displays the opening five measures of the fugue from *Prelude and Fugue in A Minor, BWV543*, by J.S. Bach. We observe that the tatum period is 1/4 of a quarter note, and there are 12 such periods per 6/8 bar. If the meter space contains just 6/8, then the beat state selects values from $\mathcal{B} = \{1, 2, \ldots, 12\}$. We define metric levels as follows:

$$\mathit{MetricLevel}(6/8, b) = \begin{cases} 1 & \text{if } b = 1 \\ 2 & \text{if } b = 7 \\ 3 & \text{if } b \in \{3, 5, 9, 11\} \\ 4 & \text{if } b \in \{2, 4, 6, 8, 10, 12\} \end{cases} \tag{7.29}$$

We simplify this example by encoding only a single chord-member rule. When this rule is not active, only the distributional simplex constraints are active; i.e., a note will be chosen at random from the space $\mathcal{N} = \{64, 65, \ldots, 81\}$, whose range is determined by the highest and lowest note in the fugue excerpt. We specify the chord-member rule strength for different metric levels as follows:

$$P(R_i = \mathrm{C} \,|\, r_{i-1}, l_i, b_i) = \begin{cases} .8 & \text{if } MetricLevel(l_i, b_i) = 1 \\ .75 & \text{if } MetricLevel(l_i, b_i) = 2 \\ .67 & \text{if } MetricLevel(l_i, b_i) = 3 \\ .33 & \text{if } MetricLevel(l_i, b_i) = 4 \end{cases} \tag{7.30}$$

where C indicates the chord-member rule. We additionally specify the harmony change probability, $P_{change}(l_i, b_i)$, described in (7.10) as follows:

$$P_{change}(l_i, b_i) = \begin{cases} 0.8 & \text{if } MetricLevel(l_i, b_i) = 1 \\ 0.4 & \text{if } MetricLevel(l_i, b_i) = 2 \\ 0.2 & \text{if } MetricLevel(l_i, b_i) = 3 \\ 0.1 & \text{if } MetricLevel(l_i, b_i) = 4 \end{cases} \tag{7.31}$$

With use this collection of parameter settings to compute the filtered and smoothed posteriors, and display the results in Figures 7.4 and 7.5. The top line of Figure 7.4 is a special predictive distribution, $P(B_i{=}1 \,|\, x_{1:i-1})$, which is the probability that the next note is the downbeat of a measure. This represents a virtual listener's desire to tap its foot. Moving down the figure, the rest of the values are the filtered posterior values for beat position $P(B_i \,|\, x_{1:i})$, rule $P(R_i \,|\, x_{1:i})$, and harmony $P(H_i \,|\, x_{1:i})$. The bottom note numbers are a piano-roll representation of the observed melody. The bottom axis of the figure is labeled with the true beat position of each note, rather than note index number, making it easier to determine where strong and weak beats fall.[2]

[2]Note that the first time slice, $i = 1$, does not have any posterior distributions associated with it, because this version of our inference code explicitly sets the starting conditions by prefilling the note

Observe in the filtered posterior that at the start of the piece, the listener cannot discern the beat; the distribution over beat is spread out, and foot-tapping prediction shows confusion. At beat 7 of the first measure, we see the Beat posterior start to trace a consistent diagonal line, indicating that it is starting to latch on, and by the first beat of the next measure, a foot tap is predicted every six beats. Looking at the size and color of the beat posterior as time progresses, we see that the listener becomes more and more certain about its assessment of the beat, but that the listener has shifted the start of the measure out of phase by half a measure with respect to the notated score. If we consider the encoding of the chord member rule and harmonic stability rule, we see that the system will favor changes on downbeats over changes in the middle of the bar. It thus groups each of the sequences which share the same repeated top note into the same measure, transitioning to the next set of repeated notes at what it estimates is the downbeat of the following bar.

Figure 7.5 displays the smoothed posterior distributions $P(B_i \mid x_{1:54})$, $P(R_i \mid x_{1:54})$, $P(H_i \mid x_{1:54})$. Note that the figure does not show the foot-tapping prediction row, because smoothing is done retrospectively, after all notes have been observed. Upon reaching the end of the piece, the simulated listener reassesses its earlier beliefs, confidently asserting the single beat sequence. Future research with this model will involve encoding realistic chord transition patterns rather than using a completely random, uniform transition distribution at points the harmony changes. We will also relax the chord-member constraints to see if we can induce a sense of metrical uncertainty at the midpoint of the second measure where the sequences start.

history delay line with observations rather than assigning a uniform or other prior over previous notes.

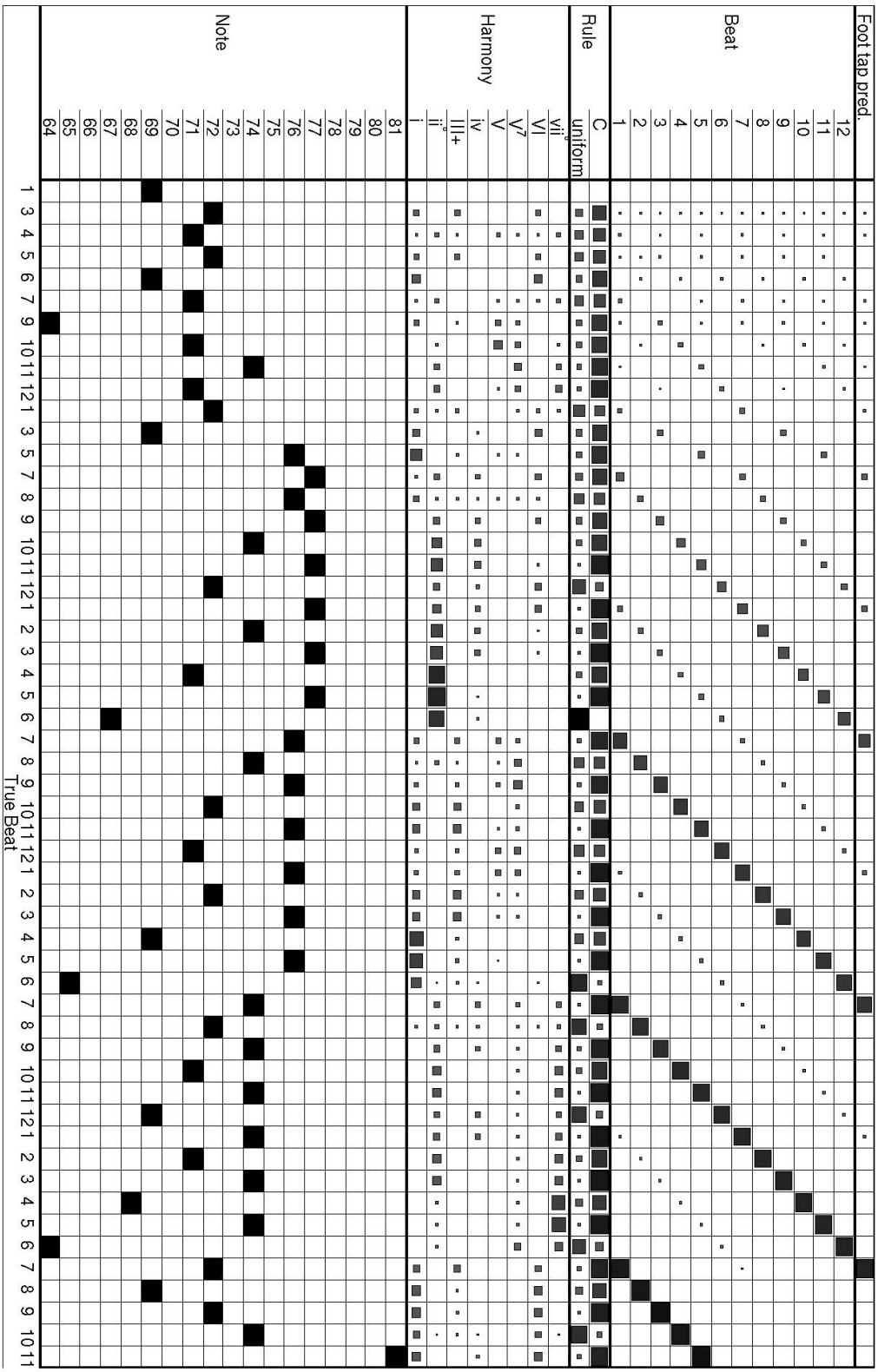


Figure 7.4: Filtered posterior for *BWV543* excerpt.

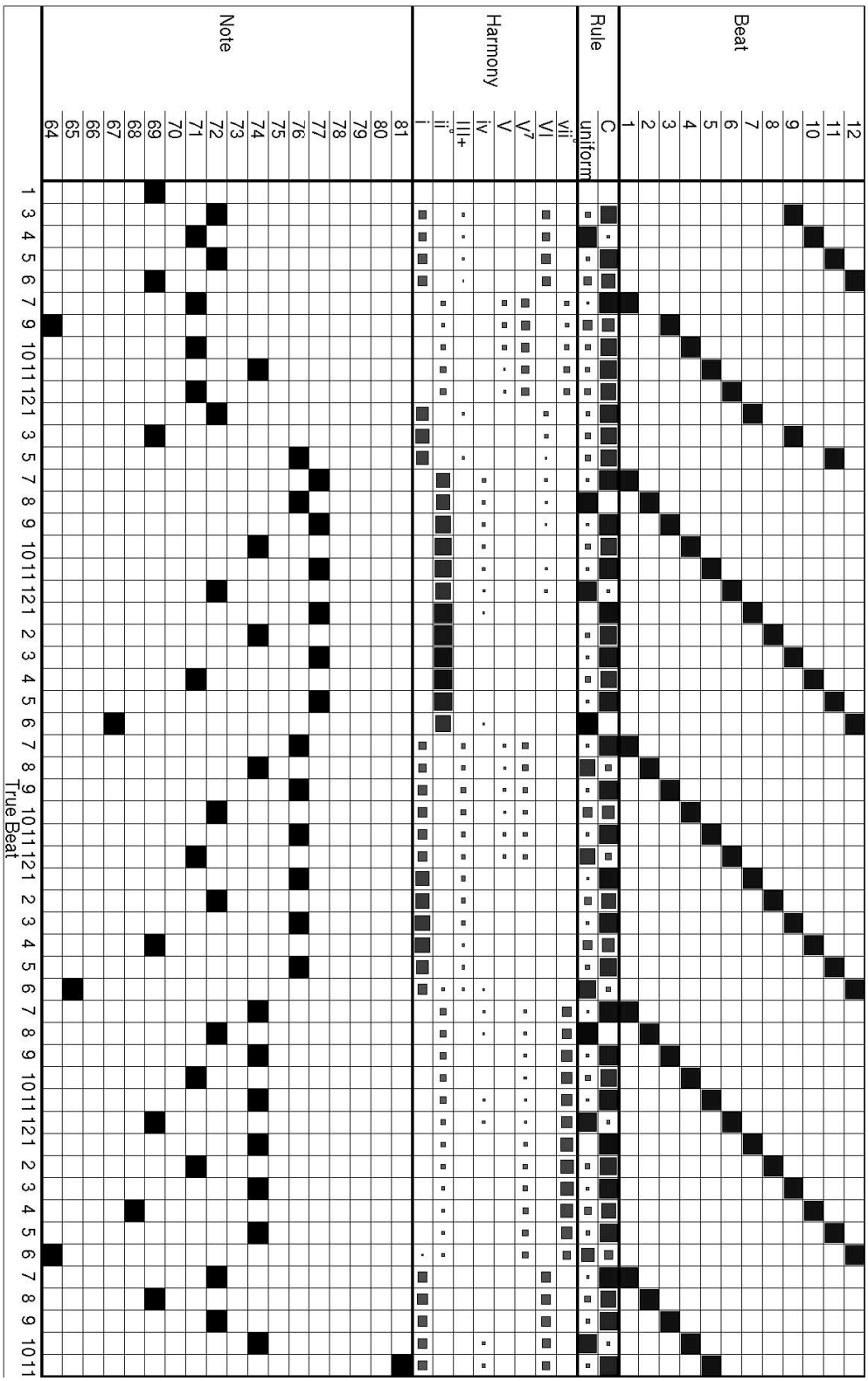


Figure 7.5: Smoothed posterior for *BWV543* excerpt.

Chapter 8

Fusing Symbolic and Signal Information

The preceding chapters present hierarchical probabilistic models of musical expectation that operate on symbolic quantities, such as music-theoretic rules, note numbers, and beat positions. A particularly advantageous feature of these expectation models is that it is possible to seamlessly integrate them into specially designed dynamic Bayesian networks that operate using audio signals as input. In one direction, this fusion of information allows us to infer higher-level musical attributes directly from audio signals. This makes possible the construction of models designed to study aspects of performance practice, for example, linking signal amplitudes with rhythmic accentuation models to study how a pianist tends to articulate a particular musical figure. Jointly, in the other direction, we can improve signal processing results by using musical expectation models to resolve ambiguities caused by a variety of types of signal interference. For example, when identifying the pitch of a note, the signal layers alone might determine that its probability is split almost evenly between two notes an octave apart; in this case, the inclusion of a smooth melodic continuation rule would resolve the ambiguity and could prevent an error in which the transcribed melody suddenly leaps more than an octave.[1]

[1]Of course, if the specific signal under analysis is not likely given encoded musical expectations, adding the unrealized expectations to the system could actually increase the number of identification

This chapter begins by describing a dynamic Bayesian network designed to extract melodies and detect musical onsets in nominally monophonic audio signals. That network, detailed by Thornburg in [86], and summarized by Thornburg, Berger, and the author in [88], represents a discrete-time stochastic process in which the time index increments once per observed short-time Fourier transform (STFT) frame. Using STFT peaks instead of audio samples substantially boosts computational efficiency, while retaining sufficient information for pitch determination and spectral change detection. One of the CPDs in the network quantifies the behavior of note-to-note motion at frames where note onsets occur. In real-world signals, note onsets rarely occur every single STFT frame, but are instead spaced at irregular intervals governed by a musical score or performer's interpretation. The note transition CPD is thus typically inactive for long sequences of STFT frames, and its activation once per note corresponds directly to our discrete-event models of musical expectation, in which we advance one slice in the DBN per note observation. By augmenting the note state in the melody transcription model to include the complete hierarchy of any of the musical expectation models described in Chapters 2–7, then latching changes in expectation model variables to changes in signal model variables, we create a system in which the symbolic and signal layers are mutually informative.

8.1 Extracting melodies and musical onsets from framewise STFT peak data

This section summarizes the probabilistic method of [86, 88] for extracting melodies and detecting onsets in musical audio signals. The system is restricted to modeling signals that arise from a monophonic score, though the actual recordings presented as input may contain significant overlaps in pitch content over time, due to reverberation or legato performance style. Each note is explicitly modeled as a sequence of two regions, transient→pitched, and gaps between notes are represented using a "null" region. A musical piece, as depicted in Figure 8.1, is thus a cyclic succession of

errors.

(transient→pitched→null) regions, where transient and null regions occupy zero or more frames, and pitched regions are constrained to be at least one frame long. The

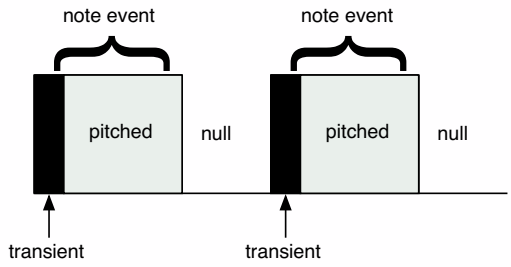


Figure 8.1: Cyclic succession of note event regimes in a monophonic passage

remainder of this section summarizes the model definition and inference goals in [86, 88].

8.1.1 Summary of variable definitions

We define a *musical onset* as a function of two successive frames indicating that the transient boundary has been crossed. To represent the cyclic succession of regimes, as well as the onset incidence, we define the following *modes* which serve to label each frame:

- 'OT' – the beginning frame of a transient region, of which there can be at most one per note event; can only follow a pitched region ('OP','CP') or 'N', and only lead to subsequent modes within the same note ('CT', 'CP').
- 'OP' – the beginning frame of a note event in which the first frame contains salient pitch content, of which there can be at most one per note event; can only follow the pitched region of the preceding note ('OP', 'CP') or 'N', and lead to the onset of the next note ('OT', 'OP'), continuation of the current pitched region ('CP'), or 'N'.
- 'CT' – the continuation of a transient region in the event the transient occupies more than one frame; can only follow a transient mode ('OT' or another 'CT'),

Symbol	Definition	Description
$\mathcal{P}$	{'OP', 'CP'}	pitched modes
$\mathcal{Q}$	{'OT', 'CT', 'N'}	non-pitched modes
$\mathcal{T}$	{'OT', 'CT'}	transient modes
$\mathcal{O}$	{'OT', 'OP'}	onset modes
$\mathcal{C}$	{'CT', 'CP'}	continuation modes

Table 8.1: Definitions of mode groupings

and only lead to subsequent modes within the same note ('CT', 'CP').

- 'CP' – a pitched region following the first frame of a note event; can follow every state except 'N', and only lead to the onset of the next note ('OT','OP') or 'N'.
- 'N' – a silent or spurious frame; can only follow a pitched region ('OP', 'CP') or another 'N', and lead to a new note event ('OT','OP') or another 'N'.

Note onset incidence is represented by the event that either $M_t =$ 'OT' or $M_t =$ 'OP'. In general, grouping modes into sets with common properties proves useful in modeling the evolution of signal characteristics; Table 8.1 summarizes these sets, and we define $\mathcal{M}$ as the set of all modes,

$$\begin{aligned} \mathcal{M} &\triangleq \mathcal{P} \cup \mathcal{Q} \\ &= \mathcal{O} \cup \mathcal{C} \cup \{\text{'N'}\} \end{aligned} \tag{8.1}$$

We associate with the frame at time t a variable $M_t \in \mathcal{M}$, which describes the mode associated with that frame. Therefore, we can represent an onset by $M_t \in \mathcal{O}$, and by the constraints of the mode succession, segment the passage into contiguous note event regions. Note that by construction there can be only a single event $M_t \in \mathcal{O}$ per note event.

In order to represent signal characteristics necessary for performing the melody extraction and segmentation task, we define *state quantities* of note value, tuning, and amplitude as follows:

- $N_t \in \mathcal{S}_N = \{N_{min}, N_{min}+1, \ldots, N_{max}\}$, where each element of $\mathcal{S}_N$ is an integer

representing the MIDI note value (e.g., $\mathcal{S}_N = \{21, 22, \ldots, 108\}$ corresponds to a standard 88-key piano keyboard, where 60 corresponds to middle-C).

- $T_t \in \mathcal{S}_T$, where $\mathcal{S}_T$ is a uniformly spaced set of tuning offset values in $[-0.5, 0.5)$, with the minimum value equal to -0.5 semitones.

- $A_t \in \mathcal{S}_A$, where $\mathcal{S}_A$ is an exponentially spaced set of reference amplitude values active when $M_t \in \mathcal{P}$. This reference amplitude governs the individual amplitudes of harmonically related spectral peaks during construction of templates used to evaluate STFT-peak likelihoods in pitched frames.

- $A_t^Q \in \mathcal{S}_{A^Q}$, where $\mathcal{S}_{A^Q}$ is an exponentially spaced set of reference amplitudes active when $M_t \in \mathcal{Q}$.

We define S_t, the *state* at time t, to be the collection of valid possibilities for all state quantities:

$$S_t \in \mathcal{S} \quad = \quad \mathcal{S}_N \otimes \mathcal{S}_T \otimes (\mathcal{S}_A \cup \mathcal{S}_{A^Q}) \,. \tag{8.2}$$

which is to say, *either* $S_t = \{N_t, T_t, A_t\}$ if $M_t \in \mathcal{P}$ *or* $S_t = \{N_t, T_t, A_t^Q\}$, if $M_t \in \mathcal{Q}$.

Finally, we represent the observations Y_t as the set of all STFT peak frequencies and amplitudes in frame t. Peaks are chosen from overlapping, Hamming-windowed, zero-padded frames following the quadratic interpolation method described in [83].

8.1.2 Model factorization and distributional specifications

The joint distribution of all variables of interest, $P(M_{0:K}, S_{0:K}, Y_{1:K})$, where K is the number of frames, factors over the directed acyclic graph as shown in Figure 8.2, i.e.:

$$\begin{aligned} P(M_{1:K}, S_{1:K}, Y_{1:K}) = P(M_1)P(S_1, M_1)P(Y_1|S_1) \\ \times \prod_{t=2}^{K} P(M_t|M_{t-1})P(S_t|S_{t-1}M_{t-1}, M_t)P(Y_t|S_t) \end{aligned} \tag{8.3}$$

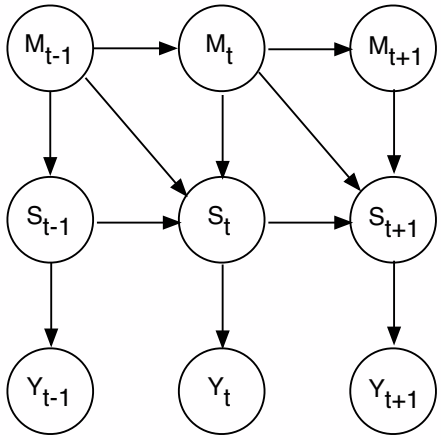


Figure 8.2: Directed acyclic graph for melody extraction and segmentation model

The prior encodes information about the first frame of the recording. We specify $P(M_1)$ as uniform and $P(S_1|M_1)$ factoring independently among components of S_0:

$$\begin{aligned} P(S_0|M_0 \in \mathcal{P}) &= P(T_0)P(N_0)P(A_0) \\ P(S_0|M_0 \in \mathcal{Q}) &= P(T_0)P(N_0)P(A_0^Q) \end{aligned} \tag{8.4}$$

where $P(T_0)$, $P(N_0)$, $P(A_0)$, and $P(A_0^Q)$ are all uniform.

The mode transition distribution, $P(M_t|M_{t-1})$, is based on a grammar encoding the mode transition constraints specified in Section 8.1.1; Several elements of $P(M_t|M_{t-1})$ are ideally constrained by that grammar be zero; e.g., 'OT' cannot progress to modes other than 'CT' or 'CT'. In practice, we set elements corresponding to invalid transitions to some small, fixed probability on the order of $1 \cdot 10^{-3}$ to allow for spurious behavior, such as a tape splice that suddenly breaks the normal cyclic succession of modes. Nonzero elements are then learned via an EM algorithm; see [86] for a complete description of this learning process and convergence results.

The state transition distribution, $P(S_t|S_{t-1}, M_{t-1}, M_t)$, describes the expected consistency between S_{t-1} and S_t as a function of M_{t-1} and M_t. For fixed $M_{t-1}, M_t \in$

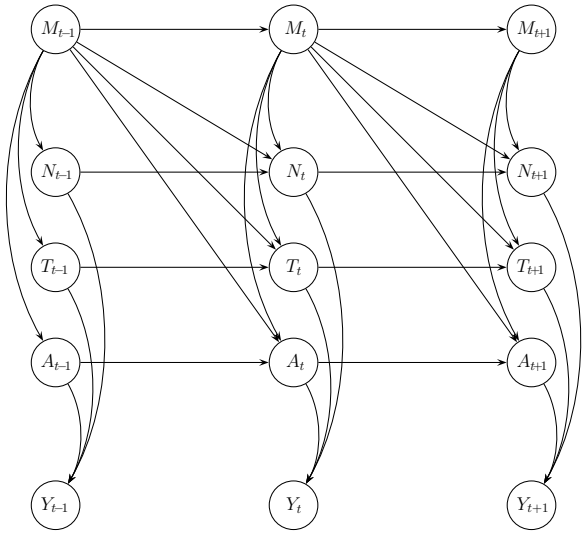


Figure 8.3: DAG for melody extraction and segmentation model, with state S_t unpacked into independently factored components. Note that A_t is replaced by $A_t^{\mathcal{Q}}$ whenever $M_t \in \mathcal{Q}$.

$\mathcal{P}$, the transition behavior factors independently over the components of S_t:

$$P\left(S_t|S_{t-1}, M_{t-1} \in \mathcal{P}, M_t \in \mathcal{P}\right) = P\left(T_t|T_{t-1}, M_{t-1} \in \mathcal{P}, M_t \in \mathcal{P}\right) \times \; P\left(N_t|N_{t-1}, M_{t-1} \in \mathcal{P}, M_t \in \mathcal{P}\right) P\left(A_t|A_{t-1}, M_{t-1} \in \mathcal{P}, M_t \in \mathcal{P}\right) \quad (8.5)$$

and factors similarly for all combinations with $M_{t-1} \in \mathcal{Q}$ and/or $M_t \in \mathcal{Q}$, although in these cases A_{t-1} is replaced by $A_{t-1}^{\mathcal{Q}}$, or A_t is replaced by $A_t^{\mathcal{Q}}$. The factorization (8.5) assumes no interdependence between state components when M_{t-1} and M_t are in evidence.[2] Figure 8.3 displays a DAG of the model with the state S_t unpacked, to show the independent factorization of its components N_t,T_t, and A_t.

Because none of the expectation models presented in Chapters 2–7 take amplitude or tuning into account, the remainder of this chapter will concentrate on only the note transition factor $P(N_t|N_{t-1}, M_{t-1}, M_t)$ on the right hand side of (8.5) and second-from-top layer of Figure 8.3; distributional specifications for all other terms

[2]Situations may arise which do indicate interdependence (e.g., $\{T_t = 0.4\bar{9}, N_t = 60\}$ and $\{T_t' = -0.5, N_t = 61\}$ refer to the same pitch hypothesis), but in most practical situations it seems unnecessary to model this interdependence.

are detailed in [86].

8.1.3 Inference goals and visualization of results

A discussion of the note transition factor, $P(N_t|N_{t-1}, M_{t-1}, M_t)$, will benefit from a visualization of inference results for the melody extraction and onset detection task. The inference process, discussed in [86], can be summarized as follows:

1. Use an approximate Viterbi approach to determine the optimal mode sequence as the global maximum *a posteriori* (MAP) trajectory:

$$M^*_{1:N} = \underset{M_{1:N}}{\operatorname{argmax}} \; P(M_{1:N}|Y_{1:N}) \tag{8.6}$$

   This approach preserves the integrity of the entire mode sequence, because it minimizes the probability that $M^*_{1:N}$ differs *anywhere* from the true $M_{1:N}$ regardless of how many frames differ. If a maximized smoothed posterior were applied to minimize the expected number of mode symbol errors, the system may hedge in favor of declaring onsets in two successive frames, in order to incur at most one symbol error. Onsets in successive frames would clearly be a serious transcription error, failing to produce even the correct number of note events. In order to preserve the integrity of the modal sequence, the approximate Viterbi inference may shift an estimated onset location by one frame, but this usually has a negligible effect.

2. Individual state components, i.e., $N_{1:N}$, $T_{1:N}$, and ($A_{1:N}$ or $A^Q_{1:N}$) are then chosen to minimize the expected number of attribute errors given $M^*_{1:N}$. That is, if Z_t represents a particular state component for the t^{th} frame, we choose:

$$Z^*_t = \underset{Z_t}{\operatorname{argmax}} \; P(Z_t|Y_{1:N}, M^*_{1:N}) \tag{8.7}$$

3. A postprocessing stage, described in [86], uses the optimal mode sequence and maximal values of the smoothed posterior to produce a score with the start time, note number, and duration of each note.

Figure 8.4 displays the global MAP mode trajectory and smoothed attribute posteriors for a solo, monophonic piano recording. The following discussion of note state transition behavior will refer the reader to relevant aspects of that figure.

8.1.4 Note state transition behavior

Transition within steady-state region of a single note event

If $M_t =$ 'CP' and $M_{t-1} \in \mathcal{P}$, then frames t–1 and t belong to the same note event, and the value of the note state would ideally not change between frames (i.e., $N_t = N_{t-1}$ with probability 1). However, to be robust in cases where the tuning offset approaches ± 0.5, we choose a conditional distribution $P(N_t|N_{t-1}, M_{t-1} \in \mathcal{P}, M_t =$ 'CP') such that N_t concentrates on values close to N_{t-1}. To express this concentration, we define the double-sided exponential distribution:

$$\mathrm{E2}\left(X_1|X_0, \alpha_+, \alpha_-\right) \;=\; \begin{cases} c, & X_1 = X_0 \\ c\alpha_+^{\kappa(X_1)-\kappa(X_0)}, & X_1 > X_0 \\ c\alpha_-^{\kappa(X_0)-\kappa(X_1)}, & X_0 > X_1 \end{cases} \tag{8.8}$$

where c is chosen such that the distribution sums to unity, and $\kappa(X) = k$ means that X is the k^{th} smallest element in the finite set of values for X. For N_t given N_{t-1}, the dependence is symmetric:

$$P(N_t|N_{t-1}, M_{t-1} \in \mathcal{P}, M_t = \text{'CP'}) \;\sim\; \mathrm{E2}(N_t|N_{t-1}, \alpha_N, \alpha_N) \tag{8.9}$$

Ideally $\alpha_N = 0$, but as mentioned above, we allow some small deviation for robustness to tuning values near the edges of the space of tuning parameters. On the other hand, allowing for more extreme tuning variations (e.g., vocal vibrato greater than ± 0.5) requires expanding the tuning space to allow the wider-ranging variation and setting α_N very close to zero the note state does not repeatedly change as the vocal pitch oscillates across semitone boundaries. Figure 8.5 displays the steady-state note transition distribution, $P(N_t|N_{t-1}, M_{t-1} \in \mathcal{P}, M_t =$ 'CP'), for two choices of α_N.

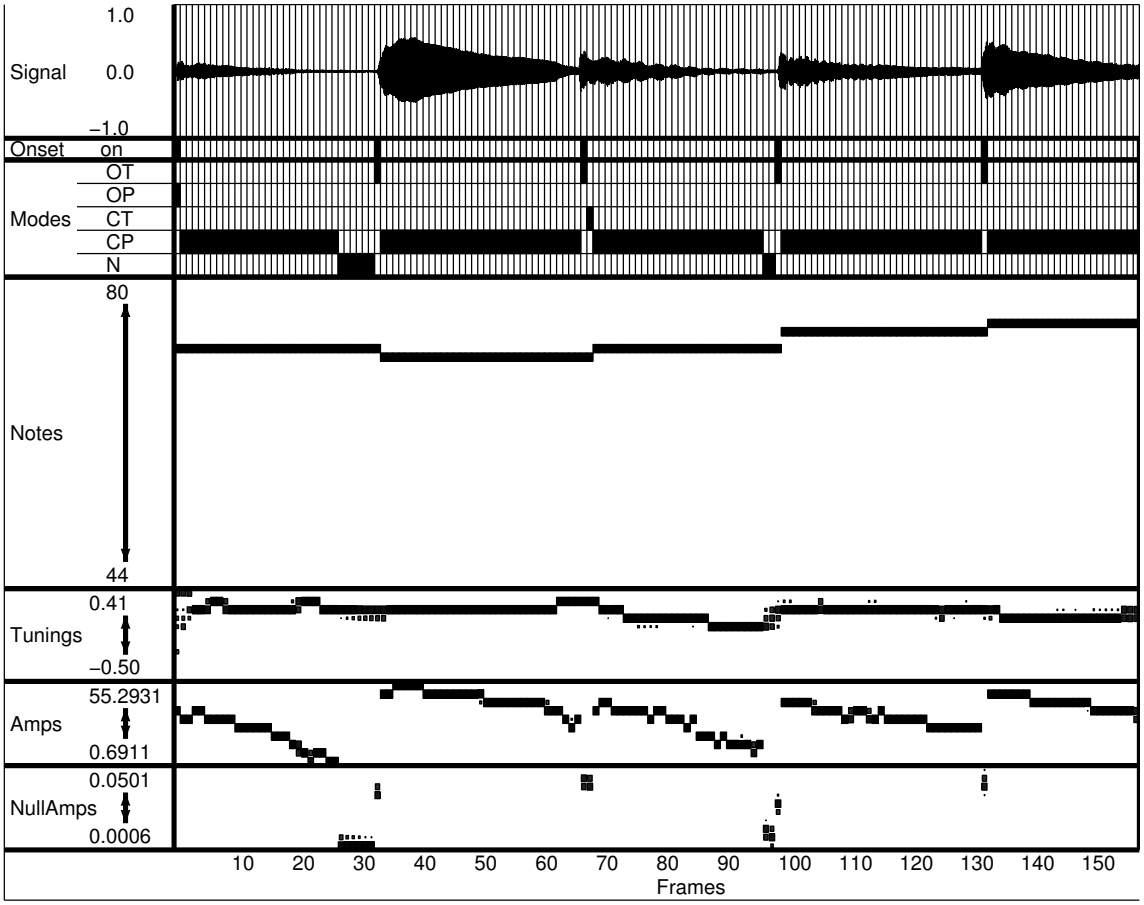
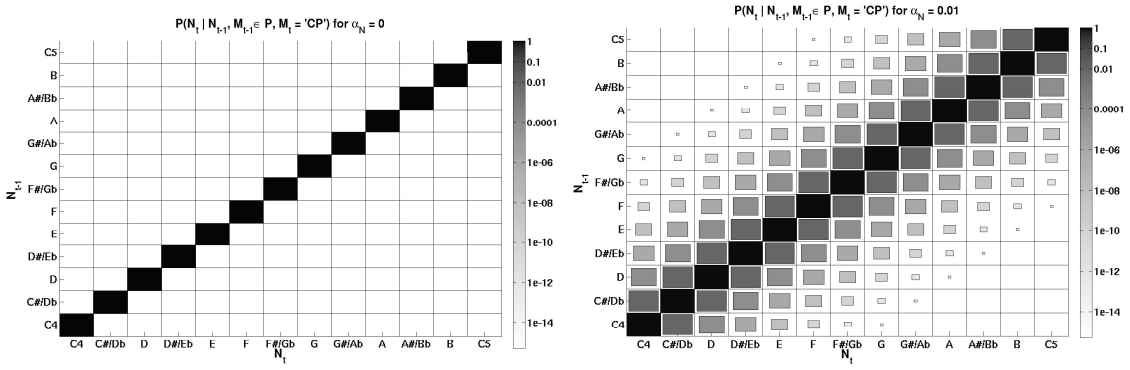


Figure 8.4: Piano example: Introductory motive of Bach's Invention 2 in C-minor (BWV 773), as performed by Glenn Gould. Rectangle sizes vary logarithmically according to probability, with the smallest visible rectangle corresponding to a probability of 0.03, and the largest, 1.0. Observe that the note posterior in any frame for which the M_t-optimal mode is non-pitched ($'OT'$, $'CT'$, and 'N') duplicates the note posterior in the preceding frame, allowing the system to remember notes during transient or null signal segments (see Sect. 8.1.4). The postprocessing stage, however, accounts for this note memorization strategy, correcting note values in leading transient frames by finding the note value that maximizes the posterior during the pitched region of a note event and assigning that value to all frames belonging to the note event.



(a) $\alpha_N = 0$ (b) $\alpha_N = 0.01$

Figure 8.5: Steady state note transition for two different values of α_N. The first, $\alpha_N = 0$ corresponds to the ideal case, where the note state cannot change at all during the region. The second, $\alpha_N = 0.01$, is the value used to generate the results in Figure 8.4. Log scaling of probabilities make the double-exponential behavior in (b) visible; an examination of the values, however, shows that the probabilities die off very quickly.

Transition to a transient or null state

If $M_t \in \mathcal{Q}$, no information about the note is reflected in the observations; the note variable is thus dormant in transient and null frames. Because of the network's first-order Markov structure, if we do not take special precautions in frames where $M_t = \mathcal{Q}$, at frame $t+1$, the system will lose all memory of notes prior to time t, and in the process forfeit its opportunity to apply knowledge of melodic expectations. We thus adopt the convention that whenever $M_t = \mathcal{Q}$, N_t refers to the value of the most recent note event:

$$P(N_t|N_{t-1}, M_{t-1} \in \mathcal{M}, M_t \in \mathcal{Q}) \quad \sim \quad \mathrm{E2}(N_t|N_{t-1}, \alpha_N, \alpha_N) \tag{8.10}$$

By reinterpreting the meaning of N_t whenever $M_t \in \mathcal{Q}$, we can thus memorize the previous note for any length transient/null region. A close examination of Figure 8.4 demonstrates that the value of N_t in the last pitched frame of each note is retained through any trailing silence after the note, as well as through any leading transient frames of the next note event. For example, the first note value in Figure 8.4 continues through six null frames (M_t = 'N') and the 'OT' frame of the second note event. While it may seem unconventional that the concentration of the note posterior does

not instantly change at the very first frame of the next note, it is precisely this state memorization scheme that allows us to tie the event base of music-theoretic models to the time base of an audio signal.

Transition to the first pitched frame of a new note

If $M_t =$ 'OP' or $[M_t =$ 'CP' and $M_{t-1} \in \mathcal{T}]$, then frame t is the first pitched frame of a new note event. Because N_{t-1} always stores the value of the previous note at any such first pitched frame t, if we define a note-to-note dependence $P_{note\_trans}(N_i|N_{i-1})$, where i and $i-1$ are discrete-event indices as in the models described in Chapters 2–7, the conditional distribution of N_t must follow $P_{note\_trans}(N_i|N_{i-1})$:

$$P(N_t|N_{t-1}, M_{t-1} \in \mathcal{M}, M_t = \text{'OP'}) \sim P_{note\_trans}(N_i|N_{i-1}) \quad (8.11)$$

$$P(N_t|N_{t-1}, M_{t-1} \in \mathcal{T}, M_t = \text{'CP'}) \sim P_{note\_trans}(N_i|N_{i-1}) \quad (8.12)$$

If we have no information about note transitions, we define $P_{note\_trans}(N_i|N_{i-1})$ as uniform over N_i Such a uniform distribution was used to generate Figure 8.4; when transitioning to a new note, the system considered any note number 44–80 to be equiprobable. If, however, we are able to specify the event-based note transition $P_{note\_trans}(N_i|N_{i-1})$, we can immediately inject melodic expectations into the the melody extraction model without altering the state S_t or inference equations; we just replace the uniform distribution with a more informative one. For example, if define $P_{note\_trans}(N_i|N_{i-1})$ to be the note transition distribution for major folksongs described in Section 2.2.2, the whole system immediately operates under the melodic expectations encoded by that distribution.[3] The following section describes how to extend the melody extraction model to include the complete hierarchy of variables in more expressive models of melodic expectation.

[3]Because the note space in the Essen folksong discussion of Chapter 2 was collapsed to a twelve-element pitch class representation, we would want to either replicate the 12^2 values across the space of multiple octaves and renormalize to form distributions over the entire space of notes, or go back to the Essen data and recalculate the transition distributions in terms of both scale degree and octave. Furthermore, note that the model presented in [86] does not have a variable for key, so the transition distribution would have to be rotated so that transitions correspond to the key of the observed audio signal.

8.2 Latching changes in an augmented note state

Including the complete hierarchy of variables in a more complete model of musical expectation, utilizing more context than just the previous note, involves two straightforward steps:

1. Augment the state S_t to include all variables in the hierarchy of the musical model.

2. Latch all additional contextual variables together so that they only change when the note state N_t changes; i.e., their values can only change whenever $M_t =$ 'OP' or $[M_t =$ 'CP' and $M_{t-1} \in \mathcal{T}]$. We will call such a frame at time t a *note activation frame*.

At note activation frames, the entire expectation model hierarchy is activated, and distributions are propagated exactly as they were in earlier chapters. Between these change frames, all variables in the expectation hierarchy just memorize values from the preceding frame (transition to the same values with probability one). Note that the memorization operation requires that every state in the expectation model be *persistent*; i.e, the DAG for the expectation model contains a directed edge from X_{i-1} to X_i for each state X in the expectation hierarchy. In cases where states are *transient*, this will require adding edges to the DAG. For example, in Figure 7.1, there are no edges from $Q_{i-1} \rightarrow Q_i$ or $D_{i-1} \rightarrow D_i$. While these edges are necessary for attribute memorization between change frames, at the transition into a change frame, these additional edges are effectively erased, because the conditional distributions are specified to be identical to what they were in the standalone discrete-event model.

Figure 8.6 shows the melody extraction model augmented to include history of one additional musical interval, using $I_t^{(1)}$ as in preceding chapters, and a rule variable R_t. In this case, one extra directed edge must be added to the original expectation model to facilitate state memorization between $I_{t-1}^{(1)}$ and $I_t^{(1)}$. With the addition of these variables, the note transition distribution becomes $P_{note\_trans}(N_i \mid N_{i-1}, I_{i-1}^{(1)}, R_i)$.

Figure 8.7 shows the melody extraction model augmented to include metrical and harmonic knowledge by grafting on the DAG displayed in Figure 7.1; this brings

in the variables D_t, Q_t, L_t, B_t, H_t, and R_t. In this case, the structure of the DAG displayed in Figure 7.1 is modified by adding additional directed edges from $Q_{i-1} \rightarrow Q_i$ and $D_{i-1} \rightarrow D_i$. The note transition distribution for the resulting model becomes $P_{note\_trans}(X_i \,|\, X_{i-1}, R_i, H_i)$.

The ability of the monophonic melody extraction model in [86, 88] to seamlessly link with hierarchical models containing the first-order Markovian relationship $P(N_t \,|\, N_{t-1}, Context_t)$, where $Context_t$ is a collection of musical attributes, leads to a number of interesting future research possibilities exploring applications that associate musical tendencies with signal features. The final chapter discusses a few of these possibilities.

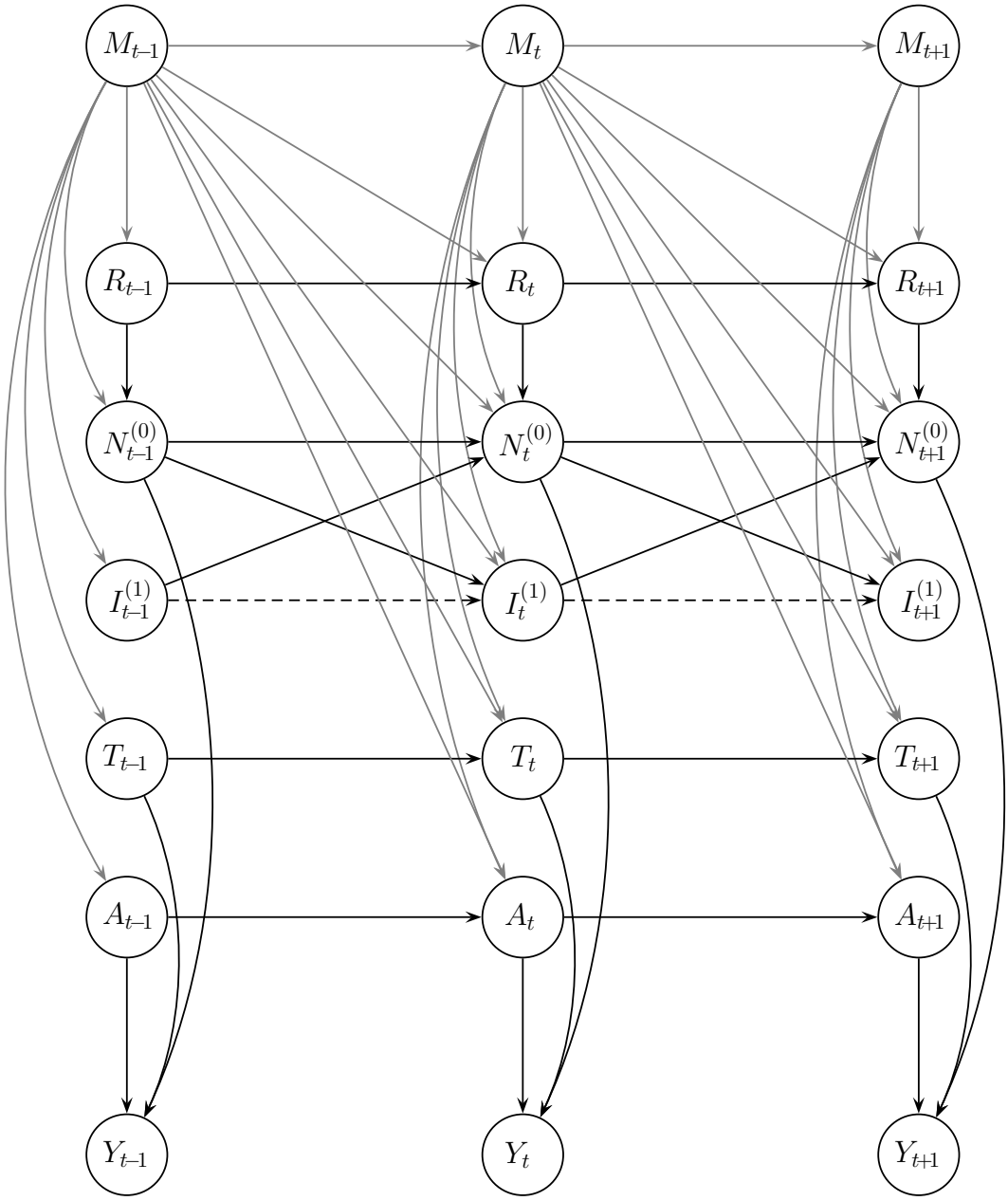


Figure 8.6: DAG for melody extraction and segmentation model, with state S_t augmented to include an additional note history, via $I_t^{(1)}$, and rule variable R_t. The dotted line from $I_{t-1}^{(1)} \rightarrow I_t^{(1)}$ indicates that the edge was added solely to make the state persistent for memorization between note onset frames; in note onset frames, this edge effectively disappears. Directed edges originating at mode states are colored gray to improve readability of the graph.

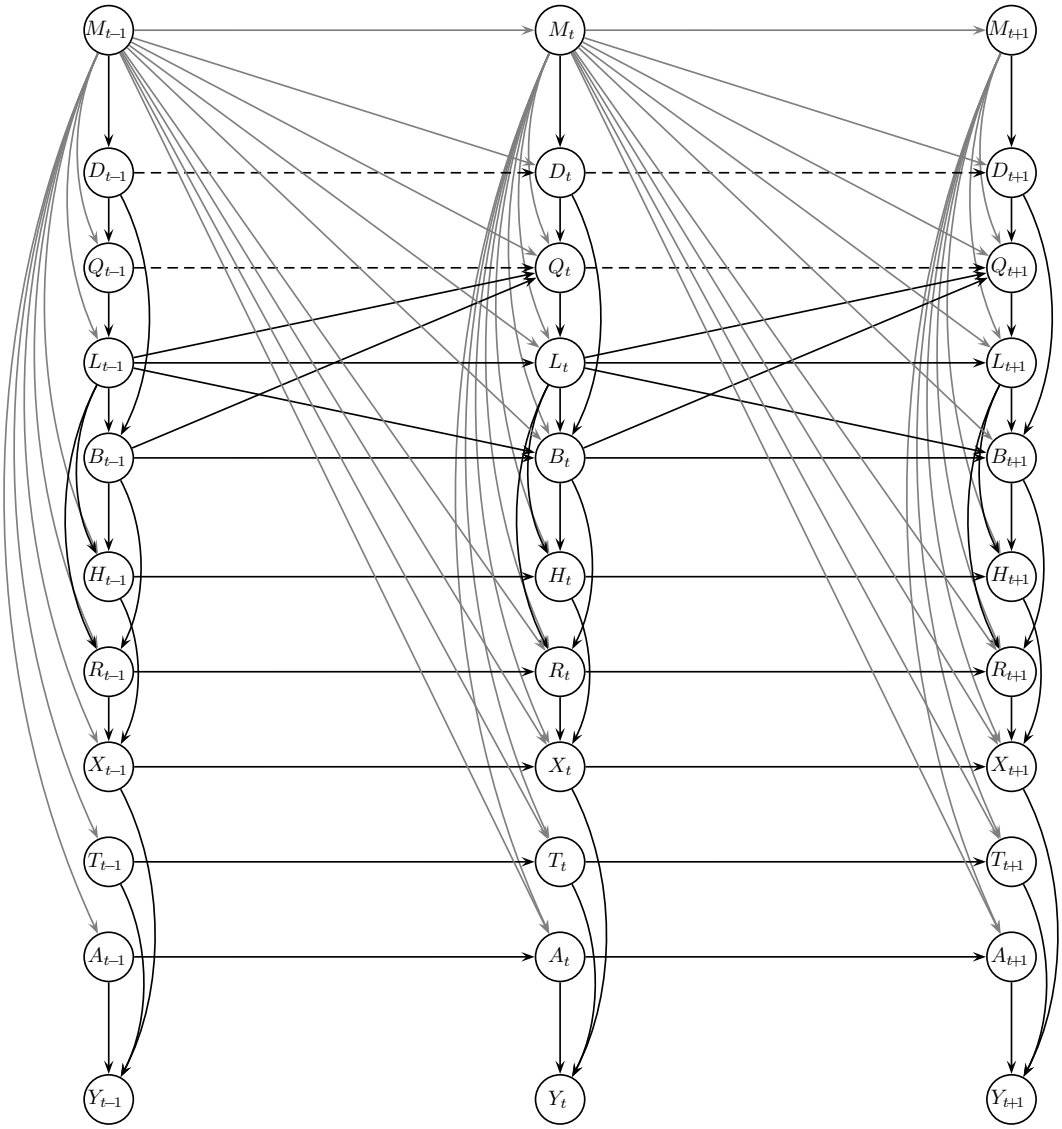


Figure 8.7: DAG for melody extraction and segmentation model, with state S_t augmented to include metrical and harmonic knowledge, by adding D_t, Q_t, L_t, B_t, H_t, and R_t. The dotted lines from $D_{t-1} \rightarrow D_t$ and $Q_{i-1} \rightarrow Q_i$ indicate that the edges were added solely to make the states persistent for memorization between note onset frames; in note onset frames, these edges effectively disappear. Directed edges originating at mode states are colored gray to improve readability of the graph.

Chapter 9

Conclusions and Future Research

9.1 Summary of contributions

The principal contribution of this dissertation is its presentation of a framework for encoding music-theoretic rule sets as maximum entropy rate conditional distributions that can be used in the specification of dynamic probabilistic models. In the music theory literature, musical rules are often inherently incomplete, stated as a set of tendencies whose strengths are governed by a set of unknown parameters. Our framework facilitates the representation of such incomplete rules by encoding their form as a set of parameterized linear constraints in a convex optimization routine that obtains the maximum entropy rate distribution given some choice of the parameter values. Because both the entropy objective and rule constraints are convex, computationally efficient solutions exist which can theoretically handle any number of musical constraints. In maximizing the entropy rate of musical transitions, we maximize the number of pieces likely to be generated by the resultant stochastic process, making the system as widely applicable as possible.

Parameter values can be learned from a corpus of data using the EM algorithm, or simply specified by hand if no such data is available or if a specific theory has already quantified a rule. While the ability to learn the unknown parameter values is itself extremely useful, perhaps the more interesting applications arise from an examination of rule activation and violation within a piece of music. Within the

context of a dynamic Bayesian network, we can create and compare virtual listeners with differing musical experience, as determined by the set of rules of which each listener is aware. Two listeners may be aware of the same set of rules with parameters adjusted according to experience (e.g., same rule set with parameters learned from different musical corpora), or operate using altogether different rule sets. We can then step through a piece of music, having each virtual listener predict upcoming notes, and measuring the surprise associated with observed realizations. In cases where one of the listeners is surprised by a given note, we can examine which of its rules were violated. Because we operate in a probabilistic model, we can measure surprise in information-theoretic terms; a more surprising observation takes more bits to encode.

Musical attribute transitions could be quantified using unconstrained conditional probability distributions learned directly from data, as was demonstrated in Section 2.2.2, before we introduced the maximum entropy rule framework. Whereas such unstructured tables may prove useful in a wide variety of applications where one simply wants to solve a problem without a detailed interpretation of the inner-workings of the system, the ability to peer inside the structure of the conditional distributions and interpret them as rule combinations is a major advantage of this system for those studying music theory and cognition.

By virtue of working in the context of dynamic Bayesian networks, extending the models presented in this dissertation is an intuitive process, because the visual representation of a model's directed acyclic graph serves as a guide, displaying all direct dependencies among variables. Because of the local dependency structure, models are both modular and extensible, and whenever new variables are added, the information they contain is integrated into a model in a consistent way. Perhaps more important than ease of use is the fact that a Bayesian network provides a framework for making optimal decisions. If we use exact Bayesian inference and the model consistently performs poorly, we know that something about our prior assumptions must be incorrect, because no other decision rule using the same assumptions could produce a lower probability of error. Furthermore, because we explicitly specify all of the distributions in the model, and because we can answer arbitrary queries involving any variable at any point in time, we have the means to diagnose problems and make corrections.

The standard types of Bayesian inference, which can be recursively computed in linear time, correspond to different listening modes; for example, filtering corresponds to an active mode of listening in which beliefs about unknown musical attributes are updated in real time, and smoothing corresponds to retrospective assessment.

Finally, all of the music expectation models presented in this dissertation can be seamlessly integrated into specially designed probabilistic models that use audio signal features as input, creating systems in which higher-level symbolic attributes and signal features are mutually informative. This opens the door to new applications in which loosely-specified tendencies governing some aspect of musical structure or performance practice inform the signal processing, and vice versa.

9.2 Future directions

By combining the advantages of dynamic Bayesian networks with maximum-entropy modeling of musical tendencies, we provide a framework that can be extended in several meaningful ways. This section discusses a number of research directions that could follow directly from this dissertation.

Encoding additional music-theoretic rule sets

Chapter 6 presents an example encoding of one music-theoretic rule set and discusses the creation and comparison of multiple virtual listeners. A straightforward area of future research is to create listeners based on several other rule sets, such as the implication-realization model of Narmour [57, 59], Spiral Array model of Chew [22], voice leading rules as described by Huron [38], and the work of Lerdahl and Jackendoff [50, 51]. In addition to modeling Western tonal music, we see this rule encoding framework applied to aspects of a wide variety of world music. One example would be encoding rules describing pitch inflection tendencies in South Indian classical (Carnatic) music [43, 44]. The ability to encode multiple rule sets and examine the experiences of several virtual listeners with differing prior knowledge will lead to a number of enlightening studies comparing musical and cognitive theories.

Integrating more detailed definitions of musical expectation, realization, and response quality

The information-theoretic measure of musical surprise presented in Section 2.2.3 quantifies the degree of fit between an expectation and realization, but does not attempt to assign detailed labels to musical response types. Berger [5] presents a detailed system for qualifying expectations, stating, "Even before the expectation is realized or violated, the strength and specificity of an expectation can generate a wide range of emotional responses." The strength and specificity of predictions, and the response quality of the observed realization could be added to the probabilistic model as random variables, allowing us to explicitly classify each prediction and realization type, rather than just describing qualities of predictive distributions.

Modeling veridical expectations

The hierarchical AR-HMM models presented in this dissertation are designed to handle both data-driven and schematic expectancies. Another type of expectancy, which Bharucha calls *veridical* expectancy [7], arises when a listener is familiar with a given piece. Bharucha states that a piece continues to be surprising even after it has been heard often enough to be familiar because veridical expectations cannot override the more generic, irrepressible, automatic expectations [8]. If our probabilistic models were extended to encode episodic memory, we could observe the interplay between veridical and schematic expectations, and identify points in a piece where the two do not match.

The integration of veridical expectations might be accomplished using a system of triggers, in which transition distributions depend on the previous occurrence of specific features in the music. The DAG for a simple trigger system is depicted in Figure 9.1. The trigger variable τ_i is a variable indicating whether or not some specific event has occurred in a the piece. If the event has occurred, $P(\tau_i = \mathit{triggered}) = 1$; otherwise $P(\tau_i = \mathit{untriggered}) = 1$. At the start of the piece, we assume that the trigger has not occurred yet, so $P(\tau_0 = \mathit{triggered}) = 0$, and the states evolve according to $P(S_i \,|\, S_{i-1}, \tau_{i-1} = \mathit{untriggered})$. The CPDs associated with the trigger

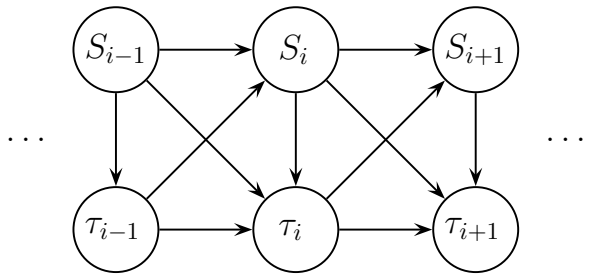


Figure 9.1: Simple trigger model

variables, $P(\tau_i \mid S_{i-1}, S_i)$, listen for specific combinations of the states S_{i-1} and S_i. If the combination has not been seen at time i, then τ_i remains *untriggered*. If the states match a special combination that causes the trigger variable to change, then τ_i equals *triggered*, and remains so for all future time slices. Once the trigger variable changes, states evolve according to a different post-trigger CPD, $P(S_i \mid S_{i-1}, \tau_{i-1} = triggered)$.

This simple model requires pre-defining a specific set of states that cause a change in the trigger variable. A more interesting possibility to pursue is the use of automatically learned triggers such as those described by Rosenfeld [75, 74, 76], which learns the most important trigger sequences in a corpus of text. We might also consider a system which goes through a piece and populates a data structure of observed state sequences, then listens to the piece again, altering its predictions based on whether the current sequence was heard before.

Making sequential decisions

At several points in preceding chapters (e.g., Sections 2.1.2 and 3.3), we demonstrate how to make statistically optimal decisions by maximizing posterior distributions. Such decisions, however, do not accurately reflect many real-time decision processes, in which there is often a cost associated with delaying a decision. A promising direction of research involves testing hypotheses and making decisions in real-time. An example application would be a foot-tapping system capable of listening to a piece and identifying the first point where its certainty about the meter and beat position exceeds a given threshold. This task could be accomplished using an online multiple hypothesis Shiryayev Sequential Probability Ratio Test (SSPRT), which considers the

measurement cost, the cost of a false alarm, and the cost of a miss-alarm [52].

Modeling durations in a real time base, allowing for expressive tempo variation

In Chapter 7, the elapsed time between note events was in units of beats, rather than time units such as seconds or STFT frames. In real-world audio signals, however, events do not typically align exactly with quantized beat positions, but rather arrive at intervals that vary continuously according to the expressive tempo chosen by a performer. A more realistic model of temporal expectations should thus include variables linking expressive timing differences with symbolic beat durations. Toward this end, Thornburg, Swaminathan, Ingalls, and the author have developed a general framework for joint event segmentation and temporal structure inference for partially-observed event sequences in the context of a switching state-space model [85]. Subsequent research will involve integrating that temporal expectation framework into the expectation models presented in this dissertation.

Modeling polyphony and coding more efficient approximate inference strategies

The expectation models presented in this dissertation assumed that we observe only a single melodic line. An important next step is to extend the models to account for multiple voices and rules of voice leading. Similarly, in Chapter 8, we observe only audio signals that might result from a monophonic score. Prior work by the author *et al.* [49] demonstrates that the single-pitch likelihood evaluation strategy of Thornburg and the author [87] can be extended to successfully recognize multiple pitches from STFT peaks. The primary difficulty with a polyphonic extension is the computability of the inference, because the number of mode combinations grows exponentially with the number of notes, and exact inference is quadratic in that number of possibilities. Section 3.10.3 of Thornburg [86] discusses these computational difficulties in depth.

Research is underway to implement a musically-informed automated polyphonic

transcription system using a *particle filter* and related sequential Monte Carlo methods. Other researchers who have applied such techniques to musical audio applications include Cemgil [18], Hainsworth [34, 35], and Godsill and Davy [33]. Another possible approximate inference strategy would involve a data-driven dynamic programming algorithm such as that of Ney, *et al.* [60], in which the computational cost is proportional only to the number of hypothesis generated, rather than the overall size of the potential search space. In either of these approximate inference strategies, the increase in computational efficiency arises from our knowledge of musical tendencies. In a given musical context, we do not typically have to explore the entire space of transition possibilities, but only those transitions which are to some degree probable. For example, if an audio signal contains a note that is highly unlikely given the context, then an approximate inference strategy may not even consider it as a possibility. Because we presume that highly unlikely events will occur infrequently, this is a type of error we are willing to accept in order to make the inference tractable. This example serves as a reminder of the utility of a maximum-entropy approach to modeling musical transitions; any other approach artificially suppresses some of the transitions, decreasing the effective hypothesis search space. While a smaller search space would improve computational speed, a faster inference is only useful if that search space contains the correct note.

Developing musically-informed signal processing applications

The ability to link higher-level musical attributes to signal features in a unified, extensible framework opens the door to a wide range of musically-informed signal processing applications. One example would be an intelligent audio editor that allows users to manipulate notes or phrases in addition to raw audio samples. Figure 8.4 provides a glimpse at what might be possible. Rather than selecting a range of samples in the audio waveform at the top of the figure, a user might be able to simply select one of the rectangles corresponding to a note and move it up or down to change its pitch, move it left or right to change its start time, or stretch it to change its duration. The resulting system would behave much like a MIDI sequencer, but user input would effect changes in the underlying audio signal.

Another extremely promising research direction is studying aspects of performance practice that relate musical context to features in an audio signal. Potential applications would include musically-informed tabla stroke type identification and guitar plucking point detection. Chordia [23] presents a method for the automatic transcription of tabla music. A straightforward adaptation of the model described in Chapter 8 would create a context-aware system for predicting tabla stroke types and identifying likely stroke substitutions. Because the same stroke type is given different names depending on the context, a context-aware system might also be used to identify both the stroke type and label. Traube and Smith [89] present a frequency-domain technique for estimating the plucking point and fingering point on a guitar string from an acoustically recorded signal. A probabilistic translation of that algorithm could be tied to higher-level musical models describing how plucking and fingering points are affected by musical context.

9.3 Final reflections

It is our hope that the framework we have presented will be a significant contribution enabling those studying music theory, cognition, composition, and related fields to analyze music and theories of music in ways that would not otherwise be possible. With this initial phase of our research concluded, we look forward to following the progress of and corresponding with anyone building upon this work, and look forward to making strides along several of the above-outlined future directions as soon as time permits.

Appendix A

Data for Selected Figures

This appendix contains the numeric values displayed in selected figures in this dissertation. In particular, we include all statistics calculated from other sources (e.g., Essen folksong collection).

$P(N) \cdot 10^1$ for major-key Essen folksongs.											
C	C$^\sharp$/D$^\flat$	D	D$^\sharp$/E$^\flat$	E	F	F$^\sharp$/G$^\flat$	G	G$^\sharp$/A$^\flat$	A	A$^\sharp$/B$^\flat$	B
1.8430	0.0074	1.5501	0.0308	1.9112	1.0911	0.0462	2.1424	0.0084	0.7773	0.0379	0.5541

Table A.1: Distribution of all notes in major-key Essen folksongs, displayed in Figure 2.5a

$P(N) \cdot 10^1$ for minor-key Essen folksongs.											
C	C$^\sharp$/D$^\flat$	D	D$^\sharp$/E$^\flat$	E	F	F$^\sharp$/G$^\flat$	G	G$^\sharp$/A$^\flat$	A	A$^\sharp$/B$^\flat$	B
1.9243	0.0512	1.4948	1.7900	0.0157	1.4380	0.0234	2.0050	0.3827	0.1222	0.5336	0.2192

Table A.2: Distribution of all notes in minor-key Essen folksongs, displayed in Figure 2.5b

$P(N_0) \cdot 10^1$ for major-key Essen folksongs.											
C	C$^\sharp$/D$^\flat$	D	D$^\sharp$/E$^\flat$	E	F	F$^\sharp$/G$^\flat$	G	G$^\sharp$/A$^\flat$	A	A$^\sharp$/B$^\flat$	B
3.2404	0.0000	0.0606	0.0000	0.8862	0.0440	0.0000	5.6991	0.0000	0.0422	0.0092	0.0183

Table A.3: Distribution of first notes in major-key Essen folksongs, displayed in Figure 2.7b

$P(N_0) \cdot 10^1$ for minor-key Essen folksongs.												
C	C♯/D♭	D	D♯/E♭	E	F	F♯/G♭	G	G♯/A♭	A	A♯/B♭	B
4.6265	0.0000	0.0917	0.5505	0.0131	0.2097	0.0000	4.0105	0.0262	0.0131	0.4587	0.0000

Table A.4: Distribution of first notes in minor-key Essen folksongs, displayed in Figure 2.7c

$P(N_{i-1}, N_i) \cdot 10^2$ for major-key Essen folksongs.												
B	2.3355	0.0038	0.6454	0.0000	0.0537	0.0552	0.0073	0.4648	0.0054	1.5459	0.0012	0.5373
A♯/B♭	0.0851	0.0004	0.0077	0.0111	0.0008	0.0046	0.0000	0.0499	0.0092	0.1369	0.0802	0.0012
A	0.3394	0.0050	0.2320	0.0008	0.0886	0.4387	0.0472	4.2285	0.0215	1.4692	0.0802	0.9783
G♯/A♭	0.0008	0.0000	0.0004	0.0000	0.0019	0.0050	0.0012	0.0268	0.0092	0.0326	0.0035	0.0042
G	3.3073	0.0015	0.7083	0.0399	3.5988	4.3918	0.2289	6.4700	0.0249	2.4916	0.0686	0.4924
F♯/G♭	0.0054	0.0000	0.0192	0.0004	0.0430	0.0107	0.0445	0.2972	0.0019	0.0449	0.0000	0.0050
F	0.0848	0.0023	1.5317	0.0936	4.7991	1.8435	0.0050	2.0187	0.0046	0.5515	0.0084	0.1952
E	2.3290	0.0054	5.7214	0.0073	4.0206	3.0853	0.1009	3.5048	0.0035	0.3881	0.0004	0.0487
D♯/E♭	0.0364	0.0027	0.1239	0.0407	0.0130	0.0775	0.0000	0.0157	0.0015	0.0000	0.0027	0.0000
D	5.0584	0.0353	3.3249	0.0851	3.9784	0.9016	0.0199	1.4063	0.0000	0.2642	0.0169	0.7168
C♯/D♭	0.0058	0.0054	0.0476	0.0012	0.0069	0.0000	0.0000	0.0000	0.0000	0.0046	0.0019	0.0023
C	4.5502	0.0138	3.4496	0.0345	2.7217	0.3164	0.0173	2.1975	0.0038	0.9975	0.1212	2.6718
N_{i-1} / N_i	C	C♯/D♭	D	D♯/E♭	E	F	F♯/G♭	G	G♯/A♭	A	A♯/B♭	B

Table A.5: Joint distribution $P(N_{i-1},\ N_i)$ for major-key Essen folksongs, shown in Figure 2.5c

$P(N_{i-1}, N_i) \cdot 10^2$ for minor-key Essen folksongs.												
B	1.6189	0.0000	0.1483	0.0121	0.0030	0.0000	0.0030	0.1967	0.0182	0.0635	0.0091	0.1695
A♯/B♭	1.1105	0.0363	0.1816	0.6143	0.0030	0.1604	0.0000	0.9502	0.9320	0.4630	0.9956	0.0061
A	0.0333	0.0000	0.0121	0.0030	0.0000	0.0151	0.0121	0.6052	0.0000	0.0484	0.3994	0.1180
G♯/A♭	0.0726	0.0061	0.0121	0.0878	0.0000	0.4146	0.0091	2.2877	0.4932	0.0030	0.5174	0.0121
G	2.4601	0.0121	0.5780	1.9669	0.0212	5.8825	0.1785	6.1761	1.6825	0.5356	0.8594	0.1180
F♯/G♭	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.0091	0.2269	0.0000	0.0030	0.0000	0.0000
F	0.3329	0.0454	0.9683	5.5648	0.0696	2.4753	0.0061	4.7145	0.3359	0.0151	0.1634	0.0091
E	0.0242	0.0000	0.0363	0.0000	0.0091	0.0726	0.0030	0.0151	0.0000	0.0000	0.0000	0.0000
D♯/E♭	1.9487	0.2481	7.1504	2.8626	0.0030	4.5420	0.0061	1.0984	0.1543	0.0061	0.2512	0.0303
D	6.3576	0.0030	1.7006	4.8325	0.0363	0.5205	0.0061	1.3345	0.0121	0.0182	0.2209	0.2421
C♯/D♭	0.3480	0.0424	0.0091	0.0817	0.0000	0.0151	0.0000	0.0000	0.0000	0.0000	0.0272	0.0000
C	4.3120	0.1301	4.4754	2.1606	0.0121	0.5659	0.0061	1.9820	0.2814	0.0908	1.9094	1.5372
N_{i-1} / N_i	C	C♯/D♭	D	D♯/E♭	E	F	F♯/G♭	G	G♯/A♭	A	A♯/B♭	B

Table A.6: Joint distribution $P(N_{i-1},\ N_i)$ for minor-key Essen folksongs, shown in Figure 2.5c

$P(N_{i-1}, N_i) \cdot 10^1$ for minor-key Essen folksongs.												
B	4.1297	0.0068	1.1412	0.0000	0.0949	0.0976	0.0129	0.8219	0.0095	2.7334	0.0020	0.9500
A♯/B♭	2.2002	0.0099	0.1982	0.2874	0.0198	0.1189	0.0000	1.2884	0.2379	3.5382	2.0714	0.0297
A	0.4280	0.0063	0.2926	0.0010	0.1117	0.5533	0.0595	5.3328	0.0271	1.8529	0.1011	1.2338
G♯/A♭	0.0897	0.0000	0.0448	0.0000	0.2242	0.5830	0.1345	3.1390	1.0762	3.8117	0.4036	0.4933
G	1.5154	0.0007	0.3246	0.0183	1.6490	2.0124	0.1049	2.9646	0.0114	1.1417	0.0315	0.2256
F♯/G♭	0.1137	0.0000	0.4062	0.0081	0.9098	0.2275	0.9423	6.2957	0.0406	0.9504	0.0000	0.1056
F	0.0761	0.0021	1.3752	0.0840	4.3086	1.6551	0.0045	1.8124	0.0041	0.4951	0.0076	0.1753
E	1.2121	0.0028	2.9775	0.0038	2.0924	1.6056	0.0525	1.8240	0.0018	0.2020	0.0002	0.0253
D♯/E♭	1.1600	0.0855	3.9438	1.2943	0.4151	2.4664	0.0000	0.5006	0.0488	0.0000	0.0855	0.0000
D	3.1999	0.0223	2.1033	0.0539	2.5167	0.5704	0.0126	0.8896	0.0000	0.1672	0.0107	0.4534
C♯/D♭	0.7614	0.7107	6.2944	0.1523	0.9137	0.0000	0.0000	0.0000	0.0000	0.6091	0.2538	0.3046
C	2.6617	0.0081	2.0179	0.0202	1.5921	0.1851	0.0101	1.2854	0.0022	0.5835	0.0709	1.5629
N_{i-1} / N_i	C	C♯/D♭	D	D♯/E♭	E	F	F♯/G♭	G	G♯/A♭	A	A♯/B♭	B

Table A.7: First-order transition distribution $P(N_i|N_{i-1})$ for major-key Essen folksongs, shown in Figure 2.6a

$P(N_{i-1}, N_i) \cdot 10^1$ for minor-key Essen folksongs.												
B	7.2200	0.0000	0.6613	0.0540	0.0135	0.0000	0.0135	0.8772	0.0810	0.2834	0.0405	0.7557
A♯/B♭	2.0366	0.0666	0.3330	1.1265	0.0055	0.2941	0.0000	1.7425	1.7092	0.8491	1.8257	0.0111
A	0.2670	0.0000	0.0971	0.0243	0.0000	0.1214	0.0971	4.8544	0.0000	0.3883	3.2039	0.9466
G♯/A♭	0.1855	0.0155	0.0309	0.2241	0.0000	1.0587	0.0232	5.8423	1.2597	0.0077	1.3215	0.0309
G	1.2018	0.0059	0.2823	0.9608	0.0103	2.8736	0.0872	3.0170	0.8219	0.2616	0.4198	0.0576
F♯/G♭	0.0000	0.0000	0.0000	0.0000	0.0000	0.0000	0.3797	9.4937	0.0000	0.1266	0.0000	0.0000
F	0.2264	0.0309	0.6587	3.7855	0.0473	1.6838	0.0041	3.2071	0.2285	0.0103	0.1112	0.0062
E	1.5094	0.0000	2.2642	0.0000	0.5660	4.5283	0.1887	0.9434	0.0000	0.0000	0.0000	0.0000
D♯/E♭	1.0648	0.1356	3.9071	1.5642	0.0017	2.4818	0.0033	0.6002	0.0843	0.0033	0.1372	0.0165
D	4.1596	0.0020	1.1127	3.1618	0.0238	0.3405	0.0040	0.8731	0.0079	0.0119	0.1445	0.1584
C♯/D♭	6.6474	0.8092	0.1734	1.5607	0.0000	0.2890	0.0000	0.0000	0.0000	0.0000	0.5202	0.0000
C	2.4692	0.0745	2.5628	1.2372	0.0069	0.3240	0.0035	1.1350	0.1612	0.0520	1.0934	0.8803
N_{i-1} / N_i	C	C♯/D♭	D	D♯/E♭	E	F	F♯/G♭	G	G♯/A♭	A	A♯/B♭	B

Table A.8: First-order transition distribution $P(N_i|N_{i-1})$ for minor-key Essen folksongs, shown in Figure 2.6b

Bibliography

[1] B. J. Aarden, "Dynamic melodic expectancy," Ph.D. dissertation, The Ohio State University, 2003.

[2] P. H. Algoet and T. M. Cover, "A sandwich proof of the Shannon-McMillan-Breiman theorem," *The Annals of Probability*, vol. 16, no. 2, pp. 899–909, 1988.

[3] J.-J. Aucouturier and M. Sandler, "Segmentation of musical signals using hidden markov models," in *Proceedings of the Audio Engineering Society 110th Convention*, Amsterdam, May 2001.

[4] A. Berger, S. Della Pietra, and V. Della Pietra, "A maximum entropy approach to natural language processing," *Computational Linguistics*, vol. 22, no. 1, pp. 39–71, Mar. 1996.

[5] J. Berger, *Musical Expectations*, unpublished manuscript, 2006.

[6] J. O. Berger, *Statistical Decision Theory and Bayesian Analysis*, 2nd ed. New York: Springer-Verlag, 1985.

[7] J. J. Bharucha, "Music cognition and perceptual facilitation: a connectionist framework," *Music Perception*, vol. 5, pp. 1–30, 1987.

[8] ——, "Tonality and expectation," in *Musical Perceptions*, R. Aiello, Ed. New York: Oxford University Press, 1994, pp. 213–239.

[9] ——, "Melodic anchoring," *Music Perception*, vol. 13, pp. 383–400, 1996.

[10] C. Bishop, *Neural Networks for Pattern Recognition*. New York: Oxford University Press, 1995.

[11] J. A. Blimes, "Techniques to foster drum machine expressivity," in *Proceedings of the 1993 International Computer Music Conference*. Tokyo: Computer Music Association, 1993, pp. 276–283.

[12] R. Bod, "Probabilistic grammars for music," in *Proceedings of the 13th Belgian-Dutch Conference on Artificial Intelligence (BNAIC'01)*, Amsterdam, 2001.

[13] ——, "Memory-based models of melodic analysis: challenging the gestalt principles," *Journal of New Music Research*, vol. 31, no. 1, pp. 27–36, Mar. 2002.

[14] ——, "A unified model of structural organization in language and music," *Journal of Artificial Intelligence Research*, vol. 17, pp. 289–308, Oct. 2002.

[15] S. Boyd and L. Vandenberghe, *Convex Optimization*. New York: Cambridge University Press, 2005.

[16] L. Breiman, "The individual ergodic theorems of information theory," *The Annals of Mathematical Statistics*, vol. 28, pp. 809–811, 1957.

[17] F. P. Brooks, A. Hopkins, P. Neumann, and W. V. Wright, "An experiment in musical composition," *IRE Transactions on Electronic Computers*, vol. 6, no. 1, pp. 175–182, 1957.

[18] A. T. Cemgil, "Bayesian music transcription," Ph.D. dissertation, Radboud University of Nijmegen, Sep. 2004.

[19] ——, "Polyphonic pitch identification and bayesian inference," in *Proceedings of the 2004 International Computer Music Conference*, Miami, FL, 2004.

[20] A. T. Cemgil, B. Kappen, and D. Barber, "Generative model based polyphonic music transcription," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA'03)*, New Paltz, NY, 2003.

[21] W. Chai and B. Vercoe, "Folk music classification using hidden Markov models," in *Proceedings of the International Conference on Artificial Intelligence IC-AI'01*, 2001.

[22] E. Chew, "Towards a mathematical model of tonality," Ph.D. dissertation, Massachusetts Institute of Technology, Feb. 1998.

[23] P. Chordia, "Automatic transcription of tabla music," Ph.D. dissertation, Stanford University, 2005.

[24] D. Conklin, "Music generation from statistical models," in *Proceedings of the AISB 2003 Symposium on Artificial Intelligence and Creativity in the Arts and Sciences*, Aberystwyth, Wales, 2003, pp. 30–35.

[25] T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: John Wiley & Sons, Inc., 1991.

[26] R. G. Cowell, A. P. Dawid, S. L. Lauritzen, and D. J. Spiegelhalter, *Probabilistic Networks and Expert Systems*. New York: Springer, 1999.

[27] I. Cross, "Review of "The analysis and cognition of melodic complexity: the implication-realization model" by E. Narmour, Univ. of Chicago, 1992," *Music Perception*, vol. 12, no. 4, pp. 486–509, 1995.

[28] M. Davy and S. J. Godsill, "Bayesian hamornic models for musical signal analysis," in *Bayesian Statistics VII*, J. M. Bernardo, J. O. Berger, A. P. Dawid, and A. F. M. Smith, Eds. New York: Oxford University Press, 2003.

[29] A. P. Dempster, N. M. Laird, and D. B. Rubin, "Maximum likelihood from incomplete data via the EM algorithm," *Journal of the Royal Statistical Society, Series B*, vol. 39, pp. 1–38, 1977.

[30] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, 2nd ed. John Wiley and Sons, Inc., 2001.

[31] A. S. Durey, "Melody spotting using hidden Markov models," Ph.D. dissertation, Georgia Institute of Technology, Nov. 2003.

[32] S.-C. Fang and H.-S. J. Tsao, "An efficient computational procedure for solving entropy optimization problems with infinitely many linear constraints," *J. Comput. Appl. Math.*, vol. 72, no. 1, pp. 127–139, 1996.

[33] S. Godsill and M. Davy, "Bayesian computational models for inharmonicity in musical instruments," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA'05)*, New Paltz, NY, Oct. 2005.

[34] S. Hainsworth, "Techniques for the automated analysis of musical audio," Ph.D. dissertation, Cambridge University, April 2004.

[35] S. Hainsworth and M. C. Macleod, "Beat tracking with particle filters," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA'03)*, New Paltz, NY, 2003.

[36] C. Huang and A. Darwiche, "Inference in belief networks: a procedural guide," *International Journal of Approximate Reasoning*, vol. 15, no. 3, pp. 225–263, 1996.

[37] D. Huron, *Music Research Using Humdrum: A User's Guide*. Stanford, CA: Center for Computer Assisted Research in the Humanities (CCARH), 1999, Humdrum Toolkit and user guide available online at: http://www.music-cog.ohio-state.edu/Humdrum/.

[38] ——, "Tone and voice: a derivation of the rules of voice-leading from perceptual principles," *Music Perception*, vol. 19, no. 1, pp. 1–64, 2001.

[39] A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*. New York: John Wiley & Sons, Inc., 2001.

[40] E. T. Jaynes, "Notes on present status and future prospects," in *Maximum Entropy and Bayesian Methods*, W. T. Grandy Jr. and L. H. Schick, Eds. Dordrecht the Netherlands: Kluwer, 1991.

[41] F. V. Jensen, *Bayesian Networks and Decision Graphs*. New York: Springer, 2001.

[42] D. Koller, "Probalistic models in artificial intelligence," Stanford University, CS228 Course Reader, Fall 2004.

[43] A. Krishnaswamy, "Application of pitch tracking to South Indian Classical Music," in *Proceedings of the International Conference on Acoustics, Speech, and Signal Processing*, 2003.

[44] ——, "Inflections and microtonality in South Indian Classical Music," in *Proceedings of the International Symposium, Frontiers of Research on Speech and Music (FRSM-2004)*, Chidambaram, India, Jan. 2004.

[45] C. L. Krumhansl and E. J. Kessler, "Tracing the dynamic changes in perceived tonal organization in a spatial representation of musical keys," *Psychological Review*, vol. 89, pp. 334–368, 1982.

[46] C. L. Krumhansl, P. Toivanen, T. Eerola, P. Toiviainen, T. Järvinen, and J. Louhivuori, "Cross-cultural music cognition: cognitive methodology applied to North Sami yoiks," *Cognition*, vol. 76, no. 1, pp. 13–58, 2000.

[47] S. Larson, "Musical forces and melodic expectation comparing computer models with experimental results," *Music Perception*, vol. 21, no. 4, pp. 457–498, 2004.

[48] S. Larson and L. VanHandel, "Measuring musical forces," *Music Perception*, vol. 23, no. 2, pp. 119–136, 2005.

[49] R. J. Leistikow, H. D. Thornburg, J. O. Smith III, and J. Berger, "Bayesian identification of closely-spaced chords from single-frame STFT peaks," in *Proceedings of the 7th International Conference on Digital Audio Effects (DAFx'04)*, Naples, Italy, Oct. 2004.

[50] F. Lerdahl, *Tonal Pitch Space*. New York: Oxford University Press, 2001.

[51] F. Lerdahl and R. Jackendoff, *A Generative Theory of Tonal Music*. Cambridge, MA: MIT Press, 1983.

[52] D. P. Malladi and J. L. Speyer, "A generalized Shiryayev sequential probability ratio test for change detection and isolation," *IEEE Transactions on Automatic Control*, vol. 44, no. 8, pp. 1522–1534, Aug. 1999.

[53] B. McMillan, "The basic theorems of information theory," *The Annals of Mathematical Statistics*, vol. 24, pp. 196–219, 1953.

[54] L. B. Meyer, *Music, The Arts, and Ideas*. Chicago: The University of Chicago Press, 1956.

[55] S. P. Meyn and R. L. Tweedie, *Markov Chains and Stochastic Stability*. New York: Springer-Verlag, 1993, updated monograph available online at http://probability.ca/MT/.

[56] K. P. Murphy, "Dynamic bayesian networks: Representation, inference and learning," Ph.D. dissertation, University of California, Berkeley, 2002.

[57] E. Narmour, *The Analysis and Cognition of Basic Melodic Structures: The Implication-Realization Model*. Chicago: University of Chicago Press, 1990.

[58] ——, "The top-down and bottom-up systems of musical implication: building on Meyer's theory of emotional syntax," *Music Perception*, vol. 9, pp. 1–26, 1991.

[59] ——, *The Analysis and Cognition of Melodic Complexity: The Implication-Realization Model*. Chicago: University of Chicago Press, 1992.

[60] H. Ney, D. Mergel, A. Noll, and A. Paeseler, "Data driven search organization for continuous speech recognition," *IEEE Transactions on Signal Processing*, vol. 40, no. 2, pp. 272–281, Feb. 1992.

[61] A. E. Nicholson and J. M. Brady, "Dynamic belief networks for discrete monitoring," *IEEE Transactions on Systems, Man, and Cybernetics*, vol. 24, no. 11, pp. 1593–1610, 1993.

[62] F. Pachet, "The Continuator: musical interaction with style," in *Proceedings of the 2002 International Computer Music Conference*. Gottenburg, Sweden: Computer Music Association, 2002.

[63] J. Pearl, *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Francisco, CA: Morgan Kaufmann, 1988.

[64] ——, *Causality: Models, Reasoning, and Inference*. New York: Cambridge University Press, 2000.

[65] J. Pickens, "A comparison of language modeling and probabilistic text information retrieval approaches to monophonic music retrieval," in *Proceedings of the 1st International Conference on Music Information Retrieval*, Plymouth, MA, Oct. 2000.

[66] ——, "Harmonic modeling for polyphonic music retrieval," Ph.D. dissertation, University of Massachusetts Amherst, May 2004.

[67] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *Proceedings of the IEEE*, vol. 77, no. 2, pp. 257–286, 1989.

[68] C. Raphael, "Automatic segmentation of acoustic musical signals using hidden Markov models," *IEEE Transactions on PAMI*, vol. 21, no. 4, pp. 360–370, 1999.

[69] ——, "Automatic transcription of piano music," in *Proceedings of the 3rd International Conference on Music Information Retrieval*, Paris, Oct. 2002.

[70] ——, "Aligning musical scores with audio using hybrid graphical models," in *Proceedings of the 5th International Conference on Music Information Retrieval*, Barcelona, Oct. 2004.

[71] ——, "A graphical model for recognizing sung melodies," in *Proceedings of the 6th International Conference on Music Information Retrieval*, London, Sep. 2005.

[72] C. Raphael and J. Stoddard, "Harmonic analysis with probabilistic graphical models," in *Proceedings of the 4th International Conference on Music Information Retrieval*, Baltimore, MD, Oct. 2003.

[73] A. Ratnaparkhi, "A simple introduction to maximum entropy models for natural language processing," Technical Report 97-08, Institute for Research in Cognitive Science, University of Pennsylvania, May 1997.

[74] R. Rosenfeld, "Adaptive statistical language modeling: A maximum entropy approach," Ph.D. dissertation, Carnegie Mellon University, 1994.

[75] ——, "A hybrid approach to adaptive statistical language modeling," in *Proceedings of the Workshop on Human Language Technology*, Plainsboro, NJ, 1994.

[76] ——, "A maximum entropy approach to adaptive statistical language modeling," *Computer, Speech and Language*, vol. 10, pp. 187–228, 1996.

[77] S. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 2nd ed. Upper Saddle River, New Jersey: Pearson Education, Inc., 2003.

[78] M. Saunders and J. Tomlin, "Interior solution of large-scale entropy maximization problems," in *18th International Symposium on Mathematical Programming (ISMP2003)*, Copenhagen, Aug. 2003, see http://www.stanford.edu/group/SOL/software/pdco.html for a digital copy of the presentation slides and associated MATLAB software.

[79] H. Schaffrath, "The Essen Folksong Collecton in Humdrum Kern Format [electronic database]," D. Huron, Ed., Menlo Park, CA: Center for Computer Assisted Research in the Humanities (CCARH), 1995.

[80] R. D. Shachter, "Bayes-ball: the rational pastime (for determining irrelevance and requisite information in belief networks and influence diagrams)," in *Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence (UAI-98)*, 1998, pp. 480–487.

[81] C. Shannon, "A mathematical theory of communication," *Bell System Technical Journal*, vol. 27, pp. 379–423 and 623–656, 1948.

[82] A. Sheh and D. P. W. Ellis, "Chord segmentation and recognition using EM-trained hidden Markov models," in *Proceedings of the 4th International Conference on Music Information Retrieval*, Baltimore, MD, Oct. 2003.

[83] J. O. Smith III and X. Serra, "PARSHL: A program for the analysis/synthesis of inharmonic sounds based on a sinusoidal representation," in *Proceedings of the 1987 International Computer Music Conference*. Champaign-Urbana: Computer Music Association, 1987, also available as Stanford Music Department Technical Report STAN–M–43.

[84] D. Temperley, "Bayesian models of musical structure and cognition," *Musicae Scientiae*, vol. 8, no. 2, pp. 175–205, 2004.

[85] H. Thornburg, D. Swaminathan, T. Ingalls, and R. Leistikow, "Joint segmentation and temporal structure inference for partially-observed event sequences," submitted for consideration to *IEEE International Workshop on Multimedia Signal Processing*.

[86] H. D. Thornburg, "Detection and modeling of transient audio signals with prior information," Ph.D. dissertation, Stanford University, Sep. 2005.

[87] H. D. Thornburg and R. J. Leistikow, "An iterative filterbank approach for extracting sinusoidal parameters from quasi-harmonic sounds," in *Proceedings of the IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA'03)*, New Paltz, NY, Nov. 2003.

[88] H. D. Thornburg, R. J. Leistikow, and J. Berger, "Melody extraction and musical onset detection via probabilistic models of framewise STFT peak data," accepted for publication in *IEEE Transactions on Speech and Audio Processing*, 2006.

[89] C. Traube and J. O. Smith III, "Estimating the plucking point on a guitar string," in *Proceedings of the COST G-6 Conference on Digital Audio Effects (DAFx'00)*, Verona, Italy, Dec. 2000.

[90] M. Tribus, *Thermodynamics and Thermostatics: An Introduction to Energy, Information and States of Matter, with Engineering Applications.* New York: D. Van Nostrand Company Inc., 1961.

[91] F.-W. Tzeng and K.-L. Ma, "Opening the black box – data driven visualization of neural networks," in *Proceedings of IEEE Visualization 2005*, Minneapolos, MN, Oct. 2005, pp. 383–390.