Abstract
A method for expressive melody synthesis is presented seeking to capture the prosodic (stress and directional) element of musical interpretation. An expressive performance is represented as a note-level annotation, classifying each note according to a small alphabet of symbols describing the role of the note within a larger context. An audio performance of the melody is represented in terms of two time-varying functions describing the evolving frequency and intensity. A method is presented that transforms the expressive annotation into the frequency and intensity functions, thus giving the audio performance. The problem of expressive rendering is then cast as estimation of the most likely sequence of hidden variables corresponding to the prosodic annotation. Examples are presented on a dataset of around 50 folk-like melodies, realized both from hand-marked and estimated annotations.

1 Introduction
A traditional musical score represents music symbolically in terms of notes, formed from a discrete alphabet of possible pitches and durations. Human performance of music often deviates substantially from the score’s cartoon-like recipe, by inflecting, stretching and coloring the music in ways that bring it to life. Expressive music synthesis seeks algorithmic approaches to this expressive rendering task, so natural to humans.

A successful method for expressive synthesis would breathe life into the otherwise sterile performances that accompany electronic greeting cards, cellphone ring tones, and other mechanically rendered music. It would allow score-writing programs — now as common with composers as word processors are to writers — to play back compositions in pleasing ways that anticipate the composer’s musical intent. Expressive synthesis would provide guiding interpretive principles for musical accompaniment systems and give composers of computer music a means of algorithmically inflecting their music. Utility aside, we are attracted to this problem as a basic example of human intelligence, often thought to be uniquely human. While humans may be the only ones that can appreciate expressively inflected music, we doubt the same is true for the construction of musical expression.

Most past work on expressive synthesis, for example [Widmer and Goebl, 2004], [Goebl et al., 2008], [Todd, 1995], [Widmer and Tobudic, 2003], as well as the many RENCON piano competition entries, has concentrated on piano music for one simple reason: a piano performance can be described by giving the onset time, damping time, and initial loudness of each note. Since a piano performance is easy to represent, it is easy to define the task of expressive piano synthesis as an estimation problem: one must simply estimate these three numbers for each note. In contrast, we treat here the synthesis of melody, which finds its richest form with “continuously controlled” instruments, such as the violin, saxophone or voice. This area has been treated by a handful of authors, perhaps with most success by the KTH group [Sundberg, 2006], [Friberg et al., 2006]. These continuously controlled instruments simultaneously modulate many different parameters leading to wide variety of tone color, articulation, dynamics, vibrato, and other musical elements, making it difficult to represent the performance of a melody. However, it is not necessary to replicate any of these familiar instruments to effectively address the heart of the melody synthesis problem. We will propose a minimal audio representation we call the theremin, due to its obvious connection with the early electronic instrument by the same name [Roads, 1996]. Our theremin controls only time-varying pitch and intensity, thus giving a relatively simple, yet capable, representation of a melody performance.

The efforts cited above are examples of what we see as the most successful attempts to date. All of these approaches map observable elements in the musical score, such as note length and pitch, to aspects of the performance, such as tempo and dynamics. The KTH system, which represents several decades of focused effort, is rule-based. Each rule maps various musical contexts into performance decisions, which can be layered, so that many rules can be applied. The rules were chosen, and iteratively refined, by a music expert seeking to articulate and generalize a wealth of experience into performance principles, in conjunction with the KTH group. In contrast, the work of [Widmer and Goebl, 2004], [Widmer and Tobudic, 2003] takes a machine learning perspective by auto-
matically learning rules from actual piano performances. We share the perspective of machine learning. In the latter example, phrase-level tempo and dynamic curve estimates are combined with the rule-based prescriptions through a case-based reasoning paradigm. That is, this approach seeks musi-
cal phrases in a training set that are “close” to the phrase being synthesized, using the tempo and dynamic curves from the best training example. As with the KTH work, the perform-
ance parameters are computed directly from the observable score attributes with no real attempt to describe any interpre-
tive goals such as repose, passing tone, local climax, surprise,
etc.

Our work differs significantly from these, and all other past work we know of, by explicitly trying to represent aspects of the interpretation itself. Previous work does not represent the interpretation, but rather treats the consequences of this interpre-
tation, such as dynamic and timing changes. We introduce a
hidden sequence of variables representing the prosodic inter-
pretation (stress and grouping) itself by annotating the role of
each note in the larger prosodic context. We believe this hidden sequence is naturally positioned between the musical
score and the observable aspects of the interpretation. Thus
the separate problems of estimating the hidden annotation
and generating the actual performance from the annotation
require shorter leaps, and are therefore easier, than directly
bridging the chasm that separates score and performance.

Once we have a representation of interpretation, it is pos-
sible to estimate the interpretation for a new melody. Thus,
we pose the expressive synthesis problem as one of statistical estimation and accomplish this using familiar methodol-
dy from the statistician’s toolbox. We present a deterministic transformation from our interpretation to the actual theremin parameters, allowing us to hear both hand labeled and esti-
imated interpretations. We present a data set of about 50 hand-
annotated melodies, as well as expressive renderings derived
from both the hand-labeled and estimated annotations. A
brief user study helps to contextualize the results, though we
hope readers will reach independent judgments.

2 The Theremin

Our goal of expressive melody synthesis must, in the end, produce actual sound. We focus here on an audio representa-
tion we believe provides a good trade-off between expressive power and simplicity.

Consider the case of a sine wave in which both frequency,
f(t), and amplitude, a(t), are modulated over time:

\[ s(t) = a(t) \sin(2\pi \int_0^t f(\tau)d\tau). \]  \hspace{1cm} (1)

These two time-varying parameters are the ones controlled in the early electronic instrument known as the theremin. Con-
tinuous control of these parameters can produce a variety of musical effects such as expressive timing, vibrato, glissando,
variety of attack and dynamics. Thus, the theremin is capa-
bale of producing a rich range of expression. One significant
aspect of musical expression which the theremin cannot cap-
ture is tone color — as a time varying sine wave, the timbre of
the theremin is always the same. Partly because of this weak-
ness, we have allowed the tone color to change as a function of amplitude, leading to the model

\[ s(t) = \sum_{h=1}^{H} A_h(a(t), f(t)) \sin(2\pi h \int_0^t f(\tau)d\tau) \]  \hspace{1cm} (2)

where the \( \{A_h\} \) are fixed functions, monotonically increasing in the first argument. The model of Eqn. 2 produces a variety of

tone colors, but still retains the simple parameterization of the signal in terms of \( f(t) \) and \( a(t) \). The main advantage
this model has to that of Eqn. 1 is that subtle changes in \( a(t) \) are more easily perceived, in effect giving a greater effective
dynamic range.

Different choices of the \( A_h \) functions lead to various instru-
mental timbres that resemble familiar instruments on oc-
casion. If this happens, however, it is purely by accident,
since we do not seek to create something like a violin or sax-
ophone. Rather we simply need a sound parameterization that
has the potential to create expressive music.

3 Representing Musical Interpretation

There a number of aspects to musical interpretation which we
cannot hope to do justice to here, though we describe several
to help place the current effort in a larger context.

Music often has a clearly defined hierarchical structure
composed of small units that group into larger and larger
units. Conveying this structure is one of the main tasks of in-
terpretation including the clear delineation of important struc-
tural boundaries as well as using contrast to distinguish struc-
tural units. Like good writing, not only does the interpretation
need to convey this top-down tree-like structure, but it must
also flow at the lowest level. This flow is largely the domain
of what we call musical prosody — the placing, avoidance,
and foreshadowing of local (note-level) stress. This use of
stress often serves to highlight cyclical patterns as well as
surprises, directing the listener’s attention toward more im-
portant events. A third facet of musical interpretation is af-
fect — sweet, sad, calm, agitated, furious, etc. The affect
of the music is more like the fabric the interpretation is made of,
as opposed to hierarchy and prosody, which are more about
what is made from the fabric.

Our focus here is on musical prosody, clearly only a piece
of the larger interpretive picture. We make this choice be-
cause we believe the notion of “correctness” is more mean-
ingful with prosody than with affect, in addition to the fact
that musical prosody is somewhat easy to isolate. The mus-
ic we treat consists of simple melodies of slow to moderate

tempo where legato (smooth and connected) phrasing is ap-
propriate. Thus the range of affect or emotional state has been
intentionally restricted, though still allowing for much divers-
ity. In addition, the melodies we choose are short, generally
less than half a minute and tend to have simple binary-tree-
like structure.

We introduce now a way of representing the desired mu-
sicality in a manner that makes clear interpretive choices and
conveys these unambiguously. Our representation labels each
melody note with a symbol from a small alphabet,

\[ A = \{l^-, l^+, l^0, l^-, l^+\} \]
4 From Labeling to Audio

Ultimately, the prosodic labeling of a melody, using symbols from $A$, must be translated into the amplitude and frequency functions we use for sound synthesis. We describe here how $a(t)$ and $f(t)$ are computed from the labeled melody and the associated musical score.

Let $t_n$ for $n = 1, \ldots, N$ be the onset time for the $n$th note of the melody, in seconds. With the exception of allowing extra time for breaths, these times are computed according to a literal interpretation of the score. We let

$$f(t) = c_0 2^{(f^{\text{ vib}}(t)+f^{\text{ atk}}(t))/12}$$

where $c_0$ is the frequency, in Hz., of the C lying 5 octaves below middle C. Thus, a unit change in either the note profile, $f^{\text{ atk}}(t)$, or the vibrato profile, $f^{\text{ vib}}(t)$, represents a semitone.

![Figure 1: Amazing Grace (top) and Danny Boy (bot) showing the note-level labeling of the music using symbols from $A$.](image)

![Figure 2: A graph of the frequency function, $f(t)$, between two notes. Pitches are bent in the direction of the next pitch and make small glissandi in transition.](image)
where each $Q_n$ term is a quadratic function, depending on only one or two of the components of $\lambda$. In general, the objectives of the $\{Q_n\}$ may conflict with one another, which is why we pose the problem as optimization rather than constraint satisfaction.

For example, if $x_n = l^1$ we want the amplitude to increase over the note. Thus we define a term of Eqn. 3

$$Q_n(\lambda) = \bar{\alpha} (\lambda^1_n - \lambda^1_{n-1} - \bar{\alpha})^2$$

to encourage the difference in amplitude values to be about $\bar{\alpha} > 0$ while $\bar{\beta} > 0$ gives the importance of this goal. $\bar{\alpha}$ may depend on the note length. Similarly, if $x_n = l^{-1}$ we define a term of Eqn. 3

$$Q_n(\lambda) = \bar{\alpha} (\lambda^1_n - \lambda^1_{n-1} - \bar{\alpha})^2$$

to encourage the decrease in amplitude associated with receding notes. In the case of $x_n = l^2$ we have

$$Q_n(\lambda) = \beta_0 (\lambda^1_n - \lambda^1_{n-1} - \alpha^0_0)^2 + \beta_1 (\lambda^1_n - \lambda^1_{n-1} - \alpha^1_1)^2$$

where $\alpha^0_0 > 0$ and $\alpha^1_1 > 0$ encourage the nth note to have greater amplitude than either of its neighbors. If $x_n = l^{-1}$ we have $J(n) = 2$ and a term

$$Q_n(\lambda) = \beta_0 (\lambda^2_n - \lambda^2_{n-1} - \alpha^2_0)^2 + \beta_1 (\lambda^2_n - \lambda^2_{n-1} - \alpha^2_1)^2$$

with an identical form, but different constants for the other two stresses $l^+$ and $l^-$. Such terms seek an absolute value for the peak intensity. An analogous term seeks to constrain the intensity to a low value for the first note labeled as $l^{-1}$ following a stress or receding label.

There are several other situations which we will not exhaustively list, however, the general prescription presented here continues to hold. Once we have included all of the $\{Q_n\}$ terms, it is a simple matter to find the optimal $\lambda$ by solving the linear equation $\nabla H = 0$. We then extend $a^in(t)$ to all $t$ by linear interpolation with some additional smoothing. Figure 3 shows an example of $a^in(t)$ and $f(t)$ on the same plot.

5 Does the Labeling Capture Musicality?

The theremin parameters, $f(t)$, $a(t)$, and hence the audio signal, $s(t)$, depend entirely on our prosodic labeling, $x$, and the musical score, through the mapping described in Section 4. We want to understand the degree to which $x$ captures musically important interpretive notions. To this end, we have constructed a dataset of about 50 simple melodies containing a combination of genuine folk songs, folk-like songs, Christmas carols, and examples from popular and art music of various eras. The melodies were chosen to have simple chords, simple phrase structure, all at moderate to slow tempo, and appropriate for legato phrasing. Familiar examples include Danny Boy, Away in a Manger, Loch Lomond, By the Waters of Babylon, etc.

Each melody is notated in a score file giving a symbolic music representation, described as a note list with rhythmic values and pitches, transposed to the key of C major or A minor. The files are prefaced by several lines giving relevant global information such as the time signature, the mode (major or minor), and tempo. Measure boundaries are indicated in the score, showing the positions of the notes in relation to the measure-level grid. Chord changes are marked using text strings describing the functional role of the chord, such as I,IV,V,V/V, annotated by using a variety of sources including guitar tabs from various web collections and the popular “Rise Up Singing” [Blood and Patterson, 1992] folk music fake book, while some were harmonized by the author. Most importantly, each note is given a symbol from our alphabet, $A$, prescribing the interpretive role of the note, painstakingly hand-labeled by the author. We used a single source of annotation hoping that this would lead to maximally consistent use of the symbols. In addition, breaths (pauses) have also been marked.

We rendered these melodies into audio according to our hand-marked annotations and the process of Section 4. For each of these audio files we provide harmonic context by superimposing sustained chords, as indicated in the scores. While we hope that readers will reach independent conclusions, we found many of the examples are remarkably successful in capturing the relevant musicality.

We do observe some aspects of musical interpretation that are not captured by our representation, however. For example, the interpretation of Danny Boy clearly requires a climax at the highest note, as do a number of the musical examples. We currently do not represent such an event through our markup. It is possible that we could add a new category of stress corresponding to such a highpoint, though we suspect that the degree of emphasis is continuous, thus not well captured by a discrete alphabet of symbols. Another occasional shortcoming is the failure to distinguish contrasting material, as in O Come O Come Emmanuel. This melody has a Gregorian chant-like feel and should mostly be rendered with calmness. However, the short outburst corresponding to the word “Rejoice” takes on a more declarative affect. Our prosodically-oriented markup simply has no way to represent such a contrast of styles. There are, perhaps some other general shortcomings of the interpretations, though we believe there is quite a bit that is “right” in them, especially consider-
6 Estimating the Interpretation

The essential goal of this work is to algorithmically generate expressive renderings of melody. Having formally represented our notion of musical interpretation, we can generate an expressive rendering by estimating the hidden sequence of note-level annotations, \(x_1, \ldots, x_N\). Our estimation of this unobserved sequence will be a function of various observables, \(y_1, \ldots, y_N\), where the feature vector \(y_n = y_{n}^{l}, \ldots, y_{n}^{f}\) measures various attributes of the musical score at the \(n\)th note.

Some of the features we considered measure surface level attributes such as the time length of the given note, as well as the first and second differences of pitch around the note. Some are derived from the most basic notion of rhythmic structure given by the time signature: from the time signature we can compute the metric strength of the onset position of the note, which we tabulate for each onset position in each time signature. We have noted that our score representation also contains the functional chords (I, V, etc.) for each chord.

Our fundamental modeling assumption is that our label sequence has a Markov structure, given the data:

\[
p(x|y) = p(x_1|y_1) \prod_{n=2}^{N} p(x_n|x_{n-1}, y_n, y_{n-1}) \tag{4}
\]

where \(z_n = (y_n, y_{n-1})\). This assumption could be derived by assuming that the sequence of pairs \((x_1, y_1), \ldots, (x_N, y_N)\) is Markov, though the conditional assumption of Eqn. 4 is all that we need. The intuition behind this assumption is the observation (or opinion) that much of phrasing results from a cyclic alternation between forward moving notes, \(\{l^-, l^+\}\), stressed notes, \(\{l^-, l^+, l^{**}\}\), and optional receding notes \(\{l^-\}\). Usually a phrase boundary is present as we move from either stressed or receding states to forward moving states. Thus the notion of state, as in a Markov chain, seems to be relevant. However, it is, of course, true that music has hierarchical structure not expressible through the regular grammar of a Markov chain. Perhaps a probabilistic context-free grammar may add additional power to the type of approach we present here.

We estimate the conditional distributions \(p(x_n|x_{n-1}, z_n)\) for each choice of \(x_{n-1} \in A\), as well as \(p(x_1|y_1)\), using our labeled data. We will use the notation

\[
p_l(x|z) = p(x = x|x_{n-1} = l, z_n = z)
\]

for \(l \in A\). In training these distributions we split our score data into \(|A|\) groups, \(D_l = \{(x_{ti}, z_{ti})\}\), where \(D_l\) is the collection of all (class label, feature vector) pairs over all notes that immediately follow a note of class \(l\).

Our first estimation method makes no prior simplifying assumptions and follows the familiar classification tree methodology of CART [Breiman et al., 1984]. That is, for each \(D_l\) we begin with a “split,” \(z > c\) separating \(D_l\) into two sets: \(D_l^0 = \{(x_{ti}, z_{ti}) : z_{ti} > c\}\) and \(D_l^1 = \{(x_{ti}, z_{ti}) : z_{ti} \leq c\}\). We choose the feature, \(j\), and cutoff, \(c\), to achieve maximal “purity” in the sets \(D_l^0\) and \(D_l^1\) as measured by the average entropy over the class labels. We continue to split the sets \(D_l^0\) and \(D_l^1\), splitting their “offspring,” etc., in a greedy manner, until the number of examples at a tree node is less than some minimum value. \(p_l(x|z)\) is then represented by finding the terminal tree node associated with \(z\) and using the empirical label distribution over the class labels \(\{x_{ti}\}\) whose associated \(\{z_{ti}\}\) fall to the same terminal tree node.

We also tried modeling \(p_l(x|z)\) using penalized logistic regression [Zhu and Hastie, 2004]. CART and logistic regression give examples of both nonparametric and parametric methods. However, the results of these two methods were nearly identical, so we will not include a parallel presentation of the logistic regression results in this sequel.

Given a piece of music with feature vector \(z_1, \ldots, z_N\), we can compute the optimizing labeling

\[
\hat{x}_1, \ldots, \hat{x}_N = \arg \max_{x_1, \ldots, x_N} p(x_1|y_1) \prod_{n=2}^{N} p(x_n|x_{n-1}, z_n)
\]

using dynamic programming. To do this we define \(p^*_n(x_1) = p(x_1|y_1)\) and

\[
p_n^*(x_n) = \max_{x_{n-1}} p_{n-1}^*(x_{n-1})p(x_n|x_{n-1}, z_n)
\]

\[
a_n(x_n) = \arg \max_{x_{n-1}} p_{n-1}^*(x_{n-1})p(x_n|x_{n-1}, z_n)
\]

for \(n = 2, \ldots, N\). We can then trace back the optimal path by \(\hat{x}_N = \arg \max_x p_n^*(x_n)\) and \(\hat{x}_n = a_{n+1}(\hat{x}_{n+1})\) for \(n = N - 1, \ldots, 1\).

7 Results

We estimated a labeling for each of the \(M = 50\) pieces in our corpus by training our model on the remaining \(M - 1\) pieces and finding the most likely labeling, \(\hat{x}_1, \ldots, \hat{x}_N\), as described above. When we applied our CART model we found that the majority of our features could be deleted with no loss in performance, resulting in a small set of features consisting of the metric strength of the onset position, the first difference in note length in seconds, and the first difference of pitch. When this feature set was applied to the entire data set there was a total of 678/2674 errors (25.3%) with detailed results as presented in Figure 4.

The notion of “error” is somewhat ambiguous, however, since there really is no correct labeling. In particular, the choices among the forward-moving labels: \(\{l^+, l^-\}\), and stress labels: \(\{l^-, l^{**}, l^+\}\) are especially subject to interpretation. If we compute an error rate using these categories, as indicated in the table, the error rate is reduced to 15.3%. The logistic regression model led similar results with analogous error rates of 26.7% and also 15.3%.

One should note a mismatch between our evaluation metric of recognition errors with our estimation strat-
Using a forward-backward-like algorithm it is possible to compute $p(x_n|y_1, \ldots, y_N)$. Thus if we choose $\tilde{x}_n = \arg \max_{x_n \in \mathcal{A}} p(x_n|y_1, \ldots, y_N)$, then the sequence $\tilde{x}_1, \ldots, \tilde{x}_N$ minimizes the expected number of estimation errors

$$E(\text{errors}|y_1, \ldots, y_N) = \sum_n p(x_n \neq \tilde{x}_n|y_1, \ldots, y_N)$$

We have not chosen this latter metric because we want a sequence that behaves reasonably. It the sequential nature of the labeling that captures the prosodic interpretation, so the most likely sequence $\tilde{x}_1, \ldots, \tilde{x}_N$ seems like a more reasonable choice.

In an effort to measure what we believe to be most important — the perceived musicality of the performances — we performed a small user study. We took a subset of the most well-known melodies of the dataset and created audio files from the random, hand, and estimated annotations. We presented all three versions of each melody to a collection of 23 subjects who were students in the Jacobs School of Music, as well as some other comparably educated listeners. The subjects were presented with random orderings of the three versions, with different orderings for each user, and asked to respond to the statement: “The performance sounds musical and expressive” with the Likert-style ratings 1=strongly disagree, 2=disagree, 3=neutral, 4=agree, 5=strongly agree, as well as to rank the three performances in terms of musicality. Out of a total of 244 triples that were evaluated in this way, the randomly-generated annotation received a mean score of 2.96 while the hand and estimated annotations received mean scores of 3.48 and 3.46. The rankings showed no preference for the hand annotations over the estimated annotations ($p = .64$), while both the hand and estimated annotations were clearly preferred to the random annotations ($p = .0002$, $p = .0003$).

Perhaps the most surprising aspect of these results is the high score of the random labellings — in spite of the meaningless nature of these labellings, the listeners were, in aggregate, “neutral” in judging the musicality of the examples. We believe the reason for this is that musical prosody, the focus of this research, accounts for only a portion of what listeners respond to. All of our examples were rendered with the same sound engine of Section 4 which tries to create a sense of smoothness in the delivery with appropriate use of vibrato and timbral variation. We imagine that the listeners were partly swayed by this appropriate affect, even when the use of stress was not satisfactory. The results also show that our estimation produced annotations that were, essentially, as good as the hand-labeled annotations. This demonstrates, to some extent, a success of our research, though it is possible that this also reflects a limit in the expressive range of our interpretation representation. Finally, while the computer-generated interpretations clearly demonstrate some musicality, the listener rating of 3.46 — halfway between “neutral” and “agree” — show there is considerable room for improvement.

While we have phrased the problem in terms of supervised learning from a hand-labeled training set, the essential approach extends in a straightforward manner to unsupervised learning. This allows, in principle, learning with much larger data sets and richer collections of hidden labels. We look forward to exploring this direction in future work, as well as treating richer grammars than the basic regular grammars of hidden Markov models.

### References


