Dynamic Bayesian networks for symbolic polyphonic pitch modeling

Stanislaw Raczynski, Emmanuel Vincent, Shigeki Sagayama

Dynamic Bayesian networks for symbolic polyphonic pitch modeling

Stanisław A. Raczyński, Emmanuel Vincent and Shigeki Sagayama

***Abstract*—Symbolic pitch modelling is a way of incorporating knowledge about relations between pitches into the process of analysing musical information or signals. In this paper, we propose a family of probabilistic symbolic polyphonic pitch models, which account for both the "horizontal" and the "vertical" pitch structure. These models are formulated as linear or log-linear interpolations of up to five sub-models, each of which is responsible for modelling a different type of relation.**

The ability of the models to predict symbolic pitch data is evaluated in terms of their cross-entropy, and of a newly proposed "contextual cross-entropy" measure. Their performance is then measured on synthesised polyphonic audio signals in terms of the accuracy of multiple pitch estimation in combination with a Nonnegative Matrix Factorisation-based acoustic model. In both experiments, the log-linear combination of at least one "vertical" (*e.g.*, harmony) and one "horizontal" (*e.g.*, note duration) sub-model outperformed a pitch-dependent Bernoulli prior by more than 60% in relative cross-entropy and 3% in absolute multiple pitch estimation accuracy. This work provides a proof of concept of the usefulness of model interpolation, which may be used for improved symbolic modelling of other aspects of music in the future.

***Index Terms*—Dynamic Bayesian Networks, multipitch analysis, symbolic pitch modelling**

I. Introduction

Symbolic music modelling, also known as *musicological modelling* [1], [2], [3], is the equivalent of language modelling in speech processing. It has the potential to improve the performance of many Music Information Retrieval (MIR) tasks, such as multiple pitch estimation [3], chord and key estimation [2], [4], [5], music structure analysis [1], algorithmic composition [6], [7] and automatic performance [8], [9], as a part of an integrated statistical model of music [10].

A particular MIR task, polyphonic pitch transcription, consists of estimating the pitches, the onset times and the durations of each of the musical notes present in a recorded audio signal. Many techniques have been proposed to this aim: sparse coding [11], auditory filterbanks [3], [12], harmonic amplitude summation [13] or Gaussian mixture models [14], but the most popular methods are based on Nonnegative Matrix Factorisation (NMF) and its variations [15], [16], [17], [18], [19], [20]. Except for [14], all these solutions operate in two subsequent steps (though much of the work focuses only on the first one). First, the salience of each pitch is quantified for every spectro-temporal bin by an *acoustic model* (sparse coder, filterbank, NMF). The salience values are then post-processed in order to detect the musical notes. Without including any prior knowledge about the occurrences of the notes, or *symbolic pitch model* $\mathrm{P}(\mathbf{N})$, this post-processing can be considered as a form of maximum likelihood estimation:

$$\widehat{\mathbf{N}} = \arg\max_{\mathbf{N}} \mathrm{P}(\mathbf{S}|\mathbf{N}), \tag{1}$$

where $\mathrm{P}(\mathbf{S}|\mathbf{N})$ is the salience model. Adding a symbolic model results in an estimation of the notes in the maximum *a posteriori*-like sense:

$$\widehat{\mathbf{N}} = \arg\max_{\mathbf{N}} \mathrm{P}(\mathbf{S}|\mathbf{N})\mathrm{P}(\mathbf{N}). \tag{2}$$

While acoustic modelling has been widely studied, symbolic pitch modelling has been given much less attention so far. Some researchers have used basic musicological models in order to overcome the limitations of current state-of-the-art multiple pitch transcription models: Ryynänen and Klapuri proposed a melody transcription method that uses a Hidden Markov Model (HMM) to model note envelopes, together with a simple musical key model in [21], but their approach was limited to monophonic note sequences. A polyphonic extension was later proposed in [3], but it still lacks modelling of the dependencies between concurrent pitches: the music is treated as a combination of independent and non-overlapping melodic voices. In other MIR areas, Raphael and Stoddard have proposed to use an HMM as a symbolic model for harmonic analysis, *i.e.*, for the estimation of the chord progression behind a sequence of notes [22]. Similar HMMs have also been successfully used for harmonic analysis of audio signals (for a recent paper see, *e.g.*, [4]). These approaches, however, model only chromatic pitch classes and discard the octave information, and the temporal dependencies are modelled between chords, but not between notes.

We propose a family of probabilistic pitch models based on Dynamic Bayesian Networks (DBNs), which account for both the "vertical" dependencies between concurrent notes (harmony) and for the "horizontal" dependencies between notes and chords. The main challenge when building such a model is dealing with the high dimensionality of the resulting distributions that makes training and inference very difficult or even impossible in practice. In our previous work, we applied a series of factorisations and approximations to the conditional note combination distribution and performed inference on a highly reduced solution space [23]. However, that was still problematic because that distribution could not be normalised

Manuscript received October 31, 2012. This work was conducted while E. Vincent was with Inria Rennes and was supported by Inria under the Associate Team Program VERSAMUS (http://versamus.inria.fr/). S. Raczyński is with Inria, 35042 Rennes Cedex, France (e-mail: stanislaw.raczynski@inria.fr). E. Vincent is with Inria, 54600 Villers-lès-Nancy, France (e-mail: emmanuel.vincent@inria.fr). S. Sagayama is with the Graduate School of Information Science and Technology, The University of Tokyo, Tokyo 113-8656, Japan (e-mail: sagayama@hil.t.u-tokyo.ac.jp).

over the entire solution space. The result was not a true probabilistic model and its approximate normalisation was computationally very expensive. In this paper, we effectively deal with this challenge by factorising the note combination distribution into a product of single note distributions, each modelled with several normalised sub-models that are combined by means of linear or log-linear interpolation. Bayesian symbolic music models have been proposed in MIR before: Kashino *et el.* used a Bayesian network for music scene analysis [24]. Mauch *et al.* proposed a DBN for simultaneous estimation of chords, the tonality and the metric structure from audio recordings [25]. In their work they have combined two conditional probability models by multiplying the corresponding probabilities with equal weight. The combined probability distribution in [25, eq. 12] is not normalized so as to sum up to 1, however, which may result in the decoding of erroneous sequences. In our work, we adopt a rigorous approach and we achieve more flexible modelling using a different interpolation weight for each model.

This paper is organised as follows. Section II details the proposed approach and describes the way of combining sub-models by means of interpolation. Particular distributions chosen in this work are discussed in Section III. Section IV describes then the experimental set-up and the results of symbolic and audio evaluations. Finally, the conclusion is given in Section V.

II. General approach

A. Model structure

We model the distribution of the note sequences $\mathrm{P}(\mathbf{N})$ using a Bayesian network with two layers of nodes: a *chord* (harmony) *layer* $\mathbf{C} = (C_1, C_2, \ldots, C_T)$, where C_t is the underlying chord at time t and T is the number of time frames in the analysed note sequence, and a *note activity layer* $\mathbf{N} = (\mathbf{N}_1, \mathbf{N}_2, \ldots, \mathbf{N}_T)$, where $\mathbf{N}_t = (N_{t,1}, \ldots, N_{t,K})$ is a binary vector in which $N_{t,k} = 1$ if pitch k is active at time t and $N_{t,k} = 0$ otherwise. Pitches k follow a discrete pitch scale, such as the chromatic scale, and K is the number of pitches in the analysed range. Denoting by $p : q$ the set of indices from p to q, inclusively, each note activity vector $\mathbf{N}_t$ is assumed to depend on all the previous note activity vectors $\mathbf{N}_{1:t-1}$ and on all the chords up to the current time frame $C_{1:t}$:

$$\mathrm{P}(\mathbf{N}) = \sum_{\mathbf{C}} \prod_{t=1}^{T} \mathrm{P}(\mathbf{N}_t|\mathbf{N}_{1:t-1}, C_{1:t})\mathrm{P}(C_t|C_{1:t-1}). \quad (3)$$

The note activity distribution can be factorised using the chain rule:

$$\mathrm{P}(\mathbf{N}_t|\mathbf{N}_{1:t-1}, C_{1:t}) = \prod_{k=1}^{K} \mathrm{P}(N_{t,k}|\mathbf{N}_{1:t-1}, \mathbf{N}_{t,1:k-1}, C_{1:t}). \quad (4)$$

B. Interpolation

Unfortunately, the note activity probability distribution $\mathrm{P}(N_{t,k}|\mathbf{N}_{1:t-1}, \mathbf{N}_{t,1:k-1}, C_{1:t})$ is too highly dimensional to be trained or used for inference in practice. To deal with this problem, we approximate it using a combination of several simpler *sub-models*. They are combined by means of *linear interpolation*:

$$\mathrm{P}(N_{t,k}|\mathbf{N}_{1:t-1}, \mathbf{N}_{t,1:k-1}, C_{1:t}) \approx \sum_i \lambda_i \mathrm{P}_i(N_{t,k}|\mathcal{X}_{t,k}^{(i)}) \quad (5)$$

with $\sum_i \lambda_i = 1$, or *log-linear interpolation*:

$$\mathrm{P}(N_{t,k}|\mathbf{N}_{t-1}, \mathbf{N}_{t,1:k-1}, C_{1:t}) \approx Z^{-1} \prod_i \mathrm{P}_i(N_{t,k}|\mathcal{X}_{t,k}^{(i)})^{\lambda_i}, \quad (6)$$

where $\mathcal{X}_{t,k}^{(i)} \subset \{\mathbf{N}_{1:t-1}, \mathbf{N}_{t,1:k-1}, C_{1:t}\}$ is a small subset of the conditioning variables, $\lambda = \{\lambda_i\}$ are the interpolation coefficients, P_i are the sub-models and Z is the normalisation factor, which depends on the values of the conditioning variables:

$$Z = \sum_{l=0}^{1} \prod_i \mathrm{P}_i(N_{t,k} = l|\mathcal{X}_{t,k}^{(i)})^{\lambda_i}. \quad (7)$$

Note that the coefficients for the log-linear interpolation do not need to sum up to 1. Each sub-model is responsible for modelling a different musicological aspect of the note sequences, such as relation to the current chord $\mathcal{X}_{t,k}^{(i)} = \{C_t\}$, local polyphony $\mathcal{X}_{t,k}^{(i)} = \{N_{t,1:k-1}\}$ or note durations $\mathcal{X}_{t,k}^{(i)} = \{N_{t-1,k}\}$.

Linear interpolation of models was first proposed in the context of spoken language modelling by Jelinek and Mercer [26], while log-linear interpolation was proposed much later by Klakow [27]. Due to the focus on spoken language modelling, most model interpolation studies deal with different temporal dependencies within a word sequence. For the sake of modelling polyphonic pitches, as well as the underlying harmony, we extend the concept of model interpolation to arbitrary dependencies including "vertical" dependencies between the notes or between the notes and the chords.

C. Training

When training all of the sub-models $\mathrm{P}_i(N_{t,k}|\mathcal{X}_{t,k}^{(i)})$ and the chord model $\mathrm{P}(C_t|C_{1:t-1})$, a simple, additive smoothing [28] was used in order to avoid overfitting. This consists of pretending that every combination of variables occurred at least α_i times:

$$\mathrm{P}_i(N_{t,k}|\mathcal{X}_{t,k}^{(i)}) = \frac{\mathrm{O}(N_{t,k}, \mathcal{X}_{t,k}^{(i)}) + \alpha_i}{\mathrm{O}(\mathcal{X}_{t,k}^{(i)}) + 2\alpha_i}, \quad (8)$$

where $\mathrm{O}()$ is the number of occurrences of a particular combination of variable values in the training set. This way, the obtained probability tends to 0.5 if no training data is available and to the real occurrence probability for large amount of data. The smoothing parameters α_i are optimised for each model separately to maximise its log-likelihood on the validation data set, which is disjoint from the training and the test sets. The same procedure is applied to the chord model:

$$\mathrm{P}(C_t|C_{1:t-1}) = \frac{\mathrm{O}(C_t, C_{1:t-1}) + \alpha_C}{\mathrm{O}(C_{1:t-1}) + D\alpha_C}, \quad (9)$$

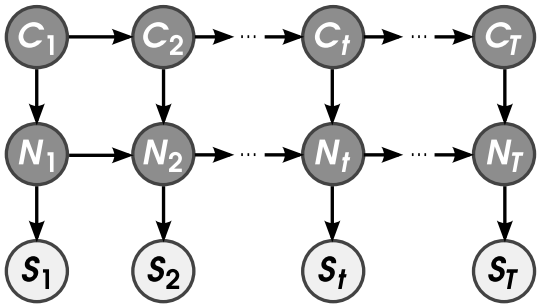

Fig. 1: Proposed Dynamic Bayesian Network structure for polyphonic pitch modelling with three layers of variables: the hidden chords C_t and note combinations $\mathbf{N}_t$, and the observed salience $\mathbf{S}_t$.

where D is the number of chord symbols.

The linear and log-linear interpolation weights λ_i in (5) and (6) are then optimised by maximizing their log-likelihood (regular cross-entropy):

$$\widehat{\lambda} = \arg\max_{\lambda} \log \mathrm{P}(\mathbf{N}|\lambda), \tag{10}$$

also calculated on the validation data set. Because the log-likelihood is convex [27], any optimization algorithm can be used. In this work, the optimisation is performed using a non-negatively constrained limited-memory Broyden-Fletcher-Goldfarb-Shanno (BFGS) method (a quasi-Newton optimisation), built into the GNU R environment as the `optim()` function [29]. The initial values were all set to $\lambda_i = 1$, $i \in \{1, \ldots, 5\}$.

III. Considered sub-models

In this work, we assumed that both the chord and note combination sequences are first-order Markovian. This is a common assumption in building models of harmony [4], [5] and has also been used to build note sequence models in [3]. Investigating the effect of using longer term dependencies is not the goal of this work and has been studied before, *e.g.*, by Scholz [30]. The note combination prior is therefore given by

$$\mathrm{P}(\mathbf{N}) = \sum_{\mathbf{C}} \mathrm{P}(C_1)\mathrm{P}(\mathbf{N}_1|C_1) \prod_{t=2}^{T} \mathrm{P}(\mathbf{N}_t|\mathbf{N}_{t-1}, C_t)\mathrm{P}(C_t|C_{t-1}) \tag{11}$$

and the corresponding DBN structure is presented in Fig. 1.

We define 5 sub-models as a proof of concept: the *harmony sub-model* is responsible for modelling relations between chords and pitches; the *note duration sub-model* deals with note and silence durations; the *voice movement sub-model* models melodic intervals in voices; the *neighbour sub-model* handles relations between vertically neighbouring pitches; finally the *polyphony sub-model* accounts for the degree of polyphony in each time frame. Other sub-models are naturally possible, but we believe that the above set covers most of the aspects of music that are important for multiple pitch analysis. In addition, the *chord model* incorporates knowledge about chord progressions.

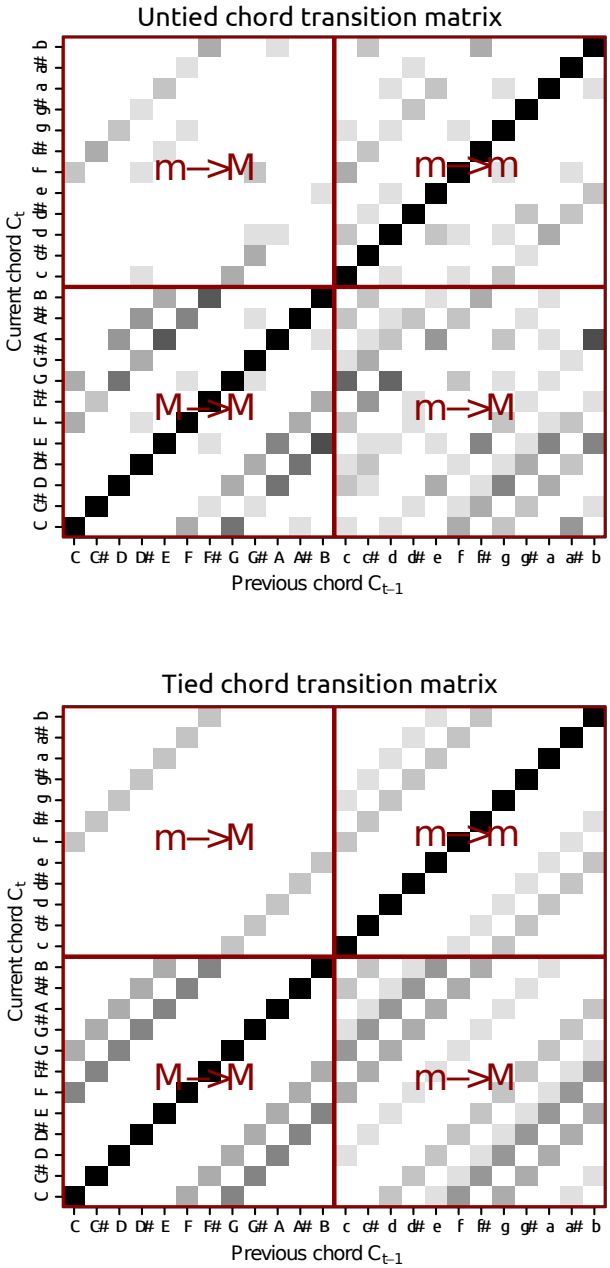

Fig. 2: Chord transition probability $\mathrm{P}(C_t|C_{t-1})$ when state tying is not used (top) and when the transition probabilities are tied (bottom). Darker colour represents higher probability values. Minor chords are annotated with lower case (m) and major chords with upper case (M).

We will now describe the chord model and each of these sub-models in detail and show the corresponding probabilities, as trained on the data described in Section IV.

A. Chord model

This model is responsible for modelling the progression of chords. The chord transition probability $\mathrm{P}(C_t|C_{t-1})$ is easy to model with a categorical probability distribution. This approach is common in MIR tasks that deal with chord progression, *e.g.*, in chord recognition [4]. It is also common

to assume $D = 24$, *i.e.*, a 24-word chord dictionary: 12 major and 12 minor chords. We have adopted this approach as well, so the chord transition distribution is described in terms of a 24×24 transition matrix.

The upper part of Fig. 2 shows the chord transition matrix trained on the entire available data set. Unfortunately, the obtained transition probabilities are biased, as some keys, and therefore some chord progressions, are sparsely represented in our data set, while others dominate. However, we can assume that the chord transitions have the same distribution in all keys if observed in relation to the tonic, which is reasonable since any song can be transposed to an arbitrary key without any loss in musical correctness. In other words, we assume that the same probability should be given to, *e.g.*, the transition from C-major to F-major chord (I→IV transition in C-major key) and the transition from A♭-major to D♭-major (I→IV transition in A♭-major key), as in [5]. In that case, the chord transition probability is a function of the interval between chord roots and the chord types

$$\mathrm{P}(C_t|C_{t-1}) \propto \mathrm{P}(\,\mathtt{I}\{\mathtt{R}\{C_t\};\mathtt{R}\{C_{t-1}\}\},\mathtt{M}\{C_t\},\mathtt{M}\{C_{t-1}\}), \tag{12}$$

where $\mathtt{I}\{\}$ is the chromatic interval operator (disregarding the octave information), $\mathtt{R}\{\}$ is the root note operator and $\mathtt{M}\{\}$ is the mode operator, *i.e.*, major or minor. The transition matrix obtained by tying distributions in the above way is presented in the lower part of Fig. 2.

Furthermore, because key is not considered in our model, we assume a uniform distribution of the initial chord $\mathrm{P}(C_1) = {}^1\!/_{24}$, which in classical Western music is generally the tonic.

B. Harmony sub-model

This sub-model models the relation between the notes and the underlying chord sequence. Similarly to the chord model, in order to avoid overfitting, we tie together the probabilities of notes that share certain musicological functions: we assumed that notes have identical distribution with respect to the chord's root notes. This distribution depends on the chord type:

$$\mathrm{P}_1(N_{t,k}|\mathbf{N}_{t-1},\mathbf{N}_{t,1:k-1},C_{1:t}) = \mathrm{P}(\,\mathtt{I}\{k;\mathtt{R}\{C_t\}\}|\mathtt{M}\{C_t\}). \tag{13}$$

This approach is similar to the Pitch Class Profiles proposed by Fujishima [31], which are 12-tone chromatic (disregarding the octave information) note activity vectors commonly used in audio-based chord estimation.

The corresponding probability distribution is presented in Fig. 3. Unsurprisingly, the interval distribution for major chords peaks at the root (R), the major third (M3) and the perfect fifth (P5), *i.e.*, the intervals that constitute a major triad, while the distribution for minor chords peaks at the *minor* third (m3), which is the interval that differentiates a minor triad from a major one.

C. Duration sub-model

This sub-model deals with the durations of individual notes and silence. The individual note activities are assumed to be

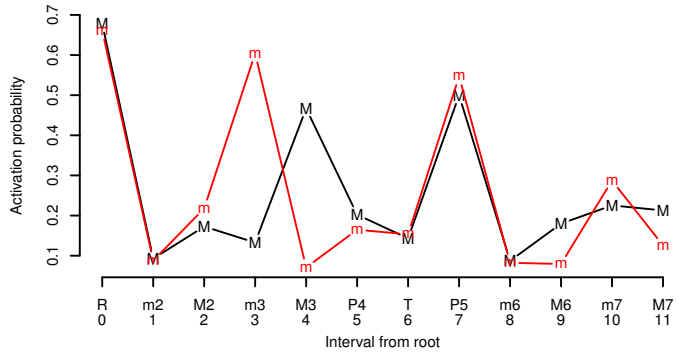

Fig. 3: The harmony sub-model: *M*'s mark the pitch class probability distribution as a function of the interval from the chord's root note for major chords $\mathrm{P}(\,\mathrm{inter}\{k;\mathrm{root}\{C_t\}\}|\mathrm{major})$ and *m*'s the distribution for minor chords, $\mathrm{P}(\,\mathrm{inter}\{k;\mathrm{root}\{C_t\}\}|\mathrm{minor})$.

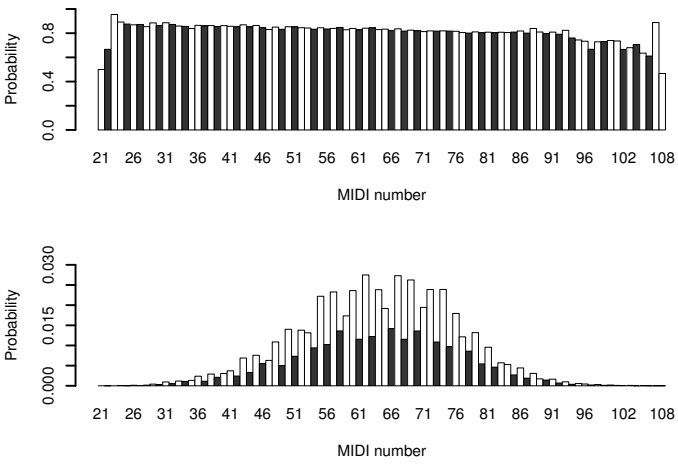

Fig. 4: The duration sub-model: $\mathrm{P}(N_{t,k} = 1|N_{t-1,k} = 1)$ (top) and $\mathrm{P}(N_{t,k} = 1|N_{t-1,k} = 0)$ (bottom). Black and white bars correspond to black and white piano keys, respectively.

dependent only on the previous state of the same pitch (first-order Markovian):

$$\mathrm{P}_2(N_{t,k}|\mathbf{N}_{t-1},\mathbf{N}_{t,1:k-1},C_{1:t}) = \mathrm{P}(N_{t,k}|N_{t-1,k}). \tag{14}$$

Its parameters are presented in Fig. 4. Its upper part shows the note sustain probabilities $\mathrm{P}(N_{t,k} = 1|N_{t-1,k} = 1)$ that seem to decrease almost linearly with increasing frequency, this property being disturbed only for the very low and the very high pitches due to sparsity of training data. This means that the low-frequency notes tend to have longer durations. The note onset probabilities $\mathrm{P}(N_{t,k} = 1|N_{t-1,k} = 0)$, shown on the bottom, exhibit a bell-shaped curve not unlike the note activity priors from Fig. 8, with black piano keys being less likely to be played than the white ones.

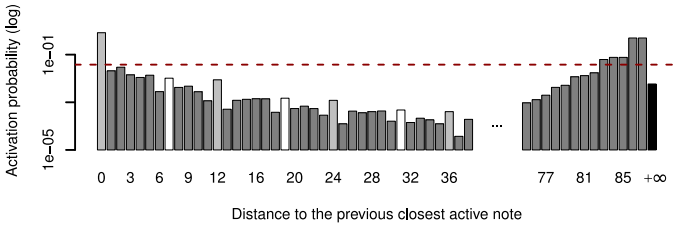

Fig. 5: The voice sub-model $\mathrm{P}(N_{t,k} = 1|M_{t,k})$. The dashed line marks the marginal note activity probability $\mathrm{P}(N_{t,k} = 1)$. Dark grey is used for unison and octave intervals, white colour marks the simple and compound perfect fifths and the black bar represents the infinite interval $\mathrm{P}(N_{t,k} = 1|M_{t,k} = +\infty)$.

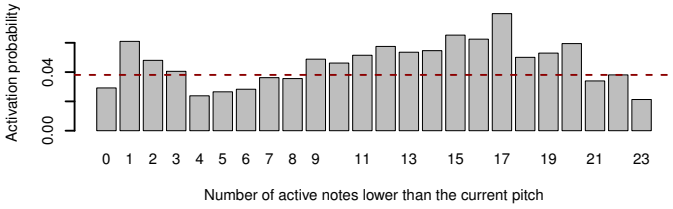

Fig. 6: The polyphony sub-model $\mathrm{P}(N_{t,k} = 1|L_{t,k})$. The dashed line marks the marginal note activity probability $\mathrm{P}(N_{t,k} = 1)$.

D. Voice sub-model

The voice sub-model accounts for voice and melody movements in the music. It assumes that the note activity depends only on the distance to the closest active pitch in the previous frame:

$$\mathrm{P}_3(N_{t,k}|\mathbf{N}_{t-1}, \mathbf{N}_{t,1:k-1}, C_{1:t}) = \mathrm{P}(N_{t,k}|M_{t,k}), \qquad (15)$$

where $M_{t,k} = |k - j|$ is the interval between the given pitch k and the closest active pitch j in the previous time frame. If there was no active pitch in the previous time frame, then $M_{t,k} = +\infty$. If the pitch k was active in the previous time frame, this model acts as a duration model, otherwise it is a simple voice movement model.

The trained parameter values for this sub-model are depicted in Fig. 5. As the distance increases, the probabilities quickly decrease—but with peaks at, *e.g.*, the perfect fifth and the octave—then increase again as the training data sparsity increases, tending to a uniform distribution (0.5) due to the effect of the smoothing (see Subsection II-C).

E. Polyphony sub-model

The polyphony sub-model models the number of simultaneously active notes:

$$\mathrm{P}_4(N_{t,k}|\mathbf{N}_{t-1}, \mathbf{N}_{t,1:k-1}, C_{1:t}) = \mathrm{P}(N_{t,k}|L_{t,k}), \qquad (16)$$

where $L_{t,k} = \sum_{m=1}^{k-1} N_{t,m}$.

The resulting distribution is plotted in Fig. 6. For small values of $L_{t,k}$ the activity probability is increased above the marginal (dashed line) for values 1, 2 and 3 (which correspond to a local polyphony $L_{t,k+1}$ of 2, 3 and 4, respectively) and then drops below the marginal. This reflects the most common polyphony values in the training set, *i.e.*, 2, 3 and 4 that account for 65% of the data, with the mean value of 3.4. For larger values of $L_{t,k}$, the probabilities increase above the marginal again, this time due to the sparsity of high-polyphony data and hence the tendency towards the uniform distribution.

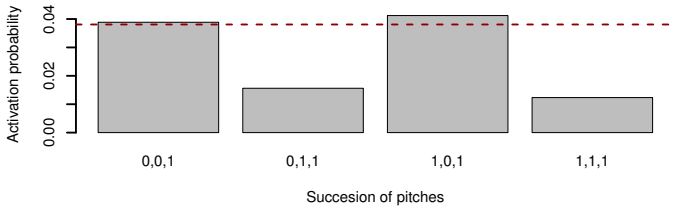

Fig. 7: Neighbour model $\mathrm{P}(N_{t,k}|N_{t,k-1}, N_{t,k-2})$. The dashed line marks the marginal note activity probability $\mathrm{P}(N_{t,k})$.

F. Neighbour sub-model

This sub-model captures the note probability given the note activities directly below it:

$$\mathrm{P}_5(N_{t,k}|\mathbf{N}_{t-1}, \mathbf{N}_{t,1:k-1}, C_{1:t}) = \mathrm{P}(N_{t,k}|N_{t,k-1}, N_{t,k-2}). \qquad (17)$$

It is a binary trigram model designed to help avoiding false positives at the minor second interval that sometimes occur in polyphonic pitch transcription due to spectral leakage of note onsets. Its trained parameter values are presented in Fig. 7. As expected, sequences of two or more active notes in a row—(0,1,1) and (1,1,1)—are strongly discouraged by this model.

IV. Experiments

We evaluated the pitch models in two different ways: first, in terms of their modelling power as measured by cross-entropy and by a newly proposed "contextual cross-entropy" on symbolic data; second, in terms of their multiple pitch estimation accuracy in combination with an NMF-based acoustic model, as measured by the $\mathcal{F}$-measure on audio data.

The symbolic experiments were performed for:

- individual note activity sub-models: harmony (H), harmony + chord (HC), duration (D), voice (V), polyphony (P) and neighbour (N) model,
- model tandems that combine one "horizontal" and one "vertical" model: duration + neighbour (DN) and harmony + chord + voice (HCV) models,
- multiple models: HCDPV and HCDVPN,
- two reference models for comparison: an i.i.d. Bernoulli model $\mathrm{P}(N_{t,k}) \sim \mathrm{Bernoulli}(p)$ with the parameter value $p = 0.03807$ trained on the training set, and an independent, pitch-dependent Bernoulli model $\mathrm{P}(N_{t,k}) \sim \mathrm{Bernoulli}(p_k)$. The values of p_k are shown in Fig. 8.

The Bernoulli models are simply probabilistic formulations of post-processing NMF results with simple thresholding to detect notes: with a fixed threshold value (Bernoulli) or with a pitch-dependent threshold value (pitch-dependent Bernoulli).

In the audio experiments, the average $\mathcal{F}$-measure was obtained for 7 different models: individual models HC, D, V,

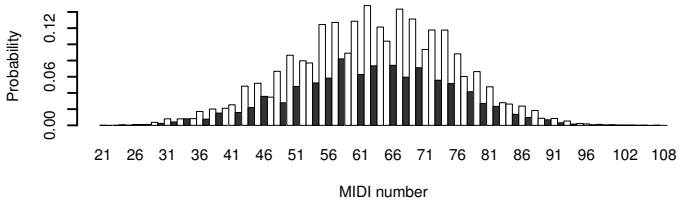

Fig. 8: Parameters p_k of the independent note activity model. Black and white bars correspond to black and white piano keys, respectively.

P, N, a tandem HCV and the full model HCVDPN, as well as the reference pitch-dependent Bernoulli model.

The code for reproducing all our experiments is available at: http://versamus.inria.fr/software-and-data/multipitch.tar.bz2.

A. Data

Two data sets were used in the experiments: the widely used RWC Classical Music Database [32] and the Mutopia Project data set [33]. All symbolic data was score-like (as opposed to real performance data) and time-quantised so that 1 frame corresponded to $1/6^{\text{th}}$ of a beat.

The 61 pieces of the RWC Classical Music Database had been annotated with detailed harmony labels that include: keys and modulations, and chords with their roots, inversions, types and various modifications [34]. This data uses abstract, tempo-independent musical time (measures and beats), and served as the chord ground-truth for training the harmony and chord models.

The Mutopia data set contains music played on a variety of instruments: chordophones (piano, guitar, cello, shamisen, violin, viola), aerophones (church, rock and reed organs, clarinet, oboe, French horn, bassoon, pan flute, recorder, trumpet), as well as voices singing in chorus. It consists of 1468 files, that we divided into 3 sub-sets: for training (1268 files), validation (100 files) and test (100 files). The training set was used to train all remaining sub-models, while the smoothing parameters and the interpolation weights from (5) and (6) were trained on the validation set. The results were assessed on the test data set.

B. Trained interpolation coefficients

The trained values of the interpolation coefficients λ_i are listed in Tables I and II.

Coefficient	Model	DN	HCV	HCDPN	HCDVPN
λ_1	Harmony	—	0.939	0.896	0.907
λ_2	Duration	0.980	—	0.863	0.272
λ_3	Voice	—	0.847	—	0.570
λ_4	Polyphony	—	—	0.000	0.000
λ_5	Neighbour	0.024	—	0.000	0.000

TABLE I: Trained interpolation coefficients for different combinations of the sub-models, obtained for the log-linear interpolation.

Coefficient	Model	DN	HCV	HCDPN	HCDVPN
λ_1	Harmony	—	0.0000	0.0000	0.0000
λ_2	Duration	1.0000	—	0.9998	0.3766
λ_3	Voice	—	1.0000	—	0.6234
λ_4	Polyphony	—	—	0.0002	0.0000
λ_5	Neighbour	0.0000	—	0.0000	0.0000

TABLE II: Trained interpolation coefficients for different combinations of the sub-models, obtained for the linear interpolation.

From the log-linear interpolation coefficient values we can see that in each case at least one "vertical" (Harmony, Polyphony or Neighbour) and one "horizontal" (Duration or Voice) sub-model were given a non-zero weight. When more than one vertical sub-model was used, the Polyphony and Neighbour sub-models would be given very low, or even zero weights, which means that either the information they hold overlaps with other used models, or that they were not able to capture much useful information about the notes.

For the case of linear interpolation, the horizontal sub-models would dominate the vertical ones completely and only the Duration and Voice sub-models would acquire non-zero weights.

C. Symbolic evaluation

We first evaluated the ability of the models to predict the symbolic test data.

1) Evaluation metrics: The models Λ are compared by calculating the cross-entropy:

$$\mathrm{H}(\Lambda) = -\frac{1}{KT}\log_2 \mathrm{P}(\mathbf{N}|\Lambda), \tag{18}$$

which is the negative log-likelihood of the observed note data normalised by the number of frames T and the number of pitches K, and therefore expressed in bits per semitone-frame. It can be interpreted as the average number of bits needed to encode a single pitch activity (Shannon's optimal code length). In other words, the lower the cross-entropy, the better the model is able to predict pitch data, with 0 meaning that the model can predict absolutely all pitch activity and 1 meaning that the pitch data is completely random given the model. Cross-entropy is a common way of evaluating spoken language models [35] and it is believed that lower cross-entropy correlates with better performance in applications [28].

If the chord model is not used, we can calculate the cross-entropy as

$$\begin{aligned}\mathrm{H}(\Lambda) &= -\frac{1}{KT}\log_2 \prod_{t=1}^{T}\prod_{k=1}^{K} \mathrm{P}(N_{t,k}|\mathbf{N}_{1:t-1}, N_{t,1:k-1}) \\ &= -\frac{1}{KT}\sum_{t=1}^{T}\sum_{k=1}^{K} \log_2 \mathrm{P}(N_{t,k}|\mathbf{N}_{t-1}, N_{t,1:k-1}).\end{aligned} \tag{19}$$

If the chord model is used however, we need to integrate over all possible chord sequences:

$$\mathrm{H}(\Lambda) = -\frac{1}{KT}\log_2 \sum_{\mathbf{C}} \mathrm{P}(\mathbf{N}|\mathbf{C},\Lambda)\mathrm{P}(\mathbf{C}|\Lambda). \tag{20}$$

This integration is done with the Forward/Backward algorithm [36]. The forward probability vector $\mathbf{f}_t$ for frame t is defined as the joint distribution of all notes observed up to the current time frame and the chord value C_t at time t: $f_{t,i} = \mathrm{P}(\mathbf{N}_{1:t}, C_t = i)$, where $i \in \{1, 2, \ldots, 24\}$. Its normalised form $\hat{\mathbf{f}}_t$ is the chord distribution given all previously observed notes: $\hat{f}_{t,i} = \mathrm{P}(C_t = i|\mathbf{N}_{1:t})$. Let us now denote the initial chord probability as $\pi_i = P(C_1 = i)$, the chord transition probability as $A_{i,j} = \mathrm{P}(C_t = i|C_{t-1} = j)$ and the note posterior as

$$\begin{aligned} d_{t,i} &= \mathrm{P}(\mathbf{N}_t|C_t = i, \mathbf{N}_{t-1}) \\ &= \prod_{k=1}^{K} \mathrm{P}(N_{t,k}|\mathbf{N}_{t-1}, \mathbf{N}_{t,1:k-1}, C_t = i). \end{aligned} \quad (21)$$

The forward vectors are calculated as

$$\hat{f}_{1,i} = p_1^{-1} d_{1,i} \pi_i, \quad (22)$$

$$\hat{f}_{t,i} = p_t^{-1} d_{t,i} \sum_{j=1}^{24} A_{i,j} \hat{f}_{t-1,j}, \quad (23)$$

where p_t is the normalising factor:

$$p_t = \sum_{i=1}^{24} \mathrm{P}(\mathbf{N}_t, C_t = i|\mathbf{N}_{1:t-1}) = \mathrm{P}(\mathbf{N}_t|\mathbf{N}_{1:t-1}). \quad (24)$$

Because $\prod_{t=1}^{T} p_t = P(\mathbf{N}_{1:T})$, the normalising factors can be used to calculate the cross-entropy:

$$\mathrm{H}(\Lambda) = -\frac{1}{KT} \log_2 \prod_{t=1}^{T} p_t. \quad (25)$$

However, comparing pitch models of different structure using the regular cross-entropy turns out to be difficult (see Fig. 9), because the values are biased by the abundance of silence in the activity matrices. We therefore propose a new metric to observe the cross-entropy only in specific contexts. The averaging in (19) can be done over specific pitches in each time frame, such as active pitches (notes), inactive pitches (silence), onsets or offsets only, which yields

$$\mathrm{cH}(\Lambda) = -\frac{1}{\sum_{t=1}^{T} |S_t|} \sum_{t=1}^{T} \sum_{k \in S_t} \log_2 \mathrm{P}(N_{t,k}|\mathbf{N}_{t-1}, N_{t,1:k-1}), \quad (26)$$

where S_t is a set of pitches of interest and $|S_t|$ denotes its size. We will refer to this new measure as the *contextual cross-entropy*.

When the chord layer is presented, we need to perform the integration over chords as in (20). For this, we define the following probabilities:

$$\mathring{h}_{t,i} = \mathrm{P}(\mathbf{N}_{t,k \notin S_t}, C_t = i|\mathbf{N}_{1:t-1}), \quad (27)$$

$$r_t = \sum_{i=1}^{24} \mathring{h}_{t,i} = \mathrm{P}(\mathbf{N}_{t,k \notin S_t}|\mathbf{N}_{1:t-1}), \quad (28)$$

$$\hat{\mathring{h}}_{t,i} = r_t^{-1} \mathring{h}_{t,i} = \mathrm{P}(C_t = i|\mathbf{N}_{1:t-1}, \mathbf{N}_{t,k \notin S_t}), \quad (29)$$

$$\dot{h}_{t,i} = \mathrm{P}(\mathbf{N}_{t,k \in S_t}, C_t = i|\mathbf{N}_{1:t-1}, \mathbf{N}_{t,k \notin S_t}), \quad (30)$$

$$q_t = \sum_{i=1}^{24} \dot{h}_{t,i} = \mathrm{P}(\mathbf{N}_{t,k \in S_t}|\mathbf{N}_{1:t-1}, \mathbf{N}_{t,k \notin S_t}). \quad (31)$$

q_t can be obtained from the forward vectors:

$$\hat{\mathring{h}}_{t,i} = r_1^{-1} \mathring{d}_{t,i} \sum_{j=1}^{24} A_{i,j} \hat{f}_{t-1,j}, \quad (32)$$

$$q_t = \sum_{i=1}^{24} \dot{d}_{t,i} \hat{\mathring{h}}_{t,i}, \quad (33)$$

where $\mathring{d}_{t,i} = \prod_{k \notin S_t} \mathrm{P}(N_{t,k}|\mathbf{N}_{t-1}, \mathbf{N}_{t,1:k-1}, C_t = i)$ and $\dot{d}_{t,i} = \prod_{k \in S_t} \mathrm{P}(N_{t,k}|\mathbf{N}_{t-1}, \mathbf{N}_{t,1:k-1}, C_t = i)$.

If $S_t = \emptyset$ then we assume $\dot{d}_{t,i} = 1$. Finally, the contextual cross-entropy is obtained as the product of the normalising factors q_t:

$$\mathrm{cH}(\Lambda) = -\frac{1}{\sum_{t=1}^{T} |S_t|} \log_2 \prod_{t=1}^{T} q_t. \quad (34)$$

2) Results: Models were combined using either the linear or the log-linear interpolation. Table III compares the regular cross-entropies obtained with both interpolation methods. Note that the contextual cross-entropy is only used to gain more insight into the results and the interpolation coefficients were trained using the regular cross-entropy. The log-linear interpolation was able to produce lower cross-entropies than the linear one for all combined models with a difference of 2.7 mb (milibits) per pitch and per frame for the model consisting of all sub-models (HCDVPN). For the Duration + Neighbour sub-model combination (DN), the difference is particularly large because the Neighbour model is not used in the case of linear interpolation (see Table II), so the resulting cross-entropy is identical to that of the Duration model alone. Even though the Neighbour model does not contain much information about the pitches, the log-linear combination DN achieves lower cross-entropy due to the smoothing effect of the non-unit exponential weight given to the Duration model.

The difference between interpolations is even bigger if the cross-entropy is measured on the note onsets only, as shown in Table IV. For the model consisting of all sub-models (HCDVPN) we have obtained a difference of 93.7 mb per onset. The only model for which the log-linear interpolation was not better is the Duration + Neighbour model (DN). In this case the smoothing effect of the exponential interpolation weight had a negative effect on the contextual cross-entropy, as it was optimised to minimise the regular cross-entropy.

The resulting contextual cross-entropy values for log-linear interpolation are presented in Fig. 9. Comparing the cross-entropy for all pitches with the contextual cross-entropy for silence, we immediately see how much the latter dominates the former and why the contextual cross-entropy calculated for notes, onsets or offsets is more apt to assess the prediction capabilities of the models. Compared to the baseline Bernoulli (220 mb per pitch) and pitch-dependent Bernoulli models (181 mb per pitch), we have achieved a 68% and 60% reduction of the regular cross-entropy, respectively, for the log-linear combination of all sub-models (HCDVPN, 73.1 mb per pitch).

The harmony model suffers the most from the aforementioned dominance of silence in the regular cross-entropies. However, by looking at the contextual cross-entropies obtained

	DN	HCV	HCDPN	HCDVPN
Linear	605.3	76.5	77.2	75.8
Log-linear	77.1	73.4	74.6	73.1
Difference	528.2	3.1	2.6	2.7

TABLE III: Regular cross-entropies (in milibits) and their difference obtained for linear and log-linear combinations of several sub-models.

	DN	HCV	HCDPN	HCDVPN
Linear	1,560.0	4,042.7	4,058.9	3,963.4
Log-linear	6,022.7	3,886.3	3,969.5	3,869.7
Difference	-4462.7	156.4	89.4	93.7

TABLE IV: Contextual cross-entropies (in milibits) for onsets and their difference obtained for linear and log-linear combinations of several sub-models.

for the onsets, we see the benefit of using the harmony sub-model: it offers low cross-entropy, while the other models fail to capture much information about the note onsets and even perform worse than the baseline Bernoulli models. We therefore conclude that the harmony models are very important in multiple pitch estimation, whose sole objective is to detect note onsets.

A similar comment can be made about the other vertical sub-models (Neighbour and Polyphony): they perform poorly in terms of the general cross-entropy, but offer good contextual cross-entropies for onsets and offsets. On the other hand, it is the horizontal sub-models (Duration and Voice) that have the biggest impact in lowering the all-pitch cross-entropy in the interpolated model: the Voice model alone yields 76.4 mb, which is then further lowered by only 3.3 mb when all the other models are used.

D. Audio signal analysis

In the second part of the experimentation, we have used the developed models to perform multiple pitch estimation on audio signals. To obtain the note saliences, we have used the harmonic NMF model proposed in [16], [37] as the acoustic model, with a tempo-synchronous analysis frame size of $\frac{1}{6}^{\text{th}}$ of a beat.

1) Salience model: The observed note saliences are assumed to be i.i.d. given the note activities:

$$\mathrm{P}(\mathbf{S}_t|\mathbf{N}_t) = \prod_{k=1}^{K} \mathrm{P}(S_{t,k}|N_{t,k}). \tag{35}$$

The obtained salience distributions $\mathrm{P}(S_{t,k}|N_{t,k} = 0)$ and $\mathrm{P}(S_{t,k}|N_{t,k} = 1)$ are presented in Fig. 10. Both were estimated by measuring histograms of the detected salience on the training data. Before calculating the histograms, the saliences were non-linearly transformed by applying an exponential factor $\chi = 0.5$ in order to enhance estimation precision for low salience values. The number of histogram bins was set to 500.

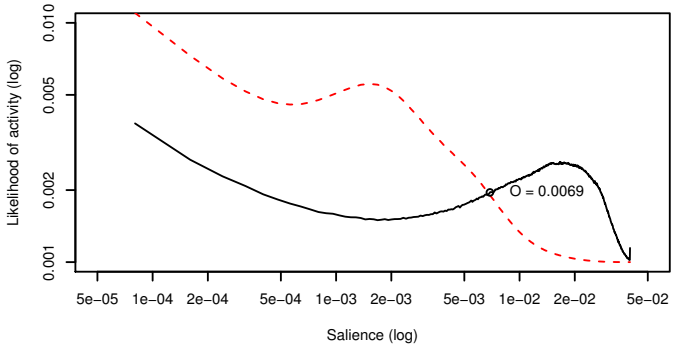

Fig. 10: The salience model $\mathrm{P}(S_{t,k}|N_{t,k} = 0)$ (red dashed line) and $\mathrm{P}(S_{t,k}|N_{t,k} = 1)$ (black solid line).

2) Salience weighting coefficient: The salience model was used with an exponential weighting factor κ, balancing its influence with the symbolic pitch model. Additionally, preliminary experiments showed that the interpolated pitch model provided very good precision, but poor recall (*cf.* Fig. 11), possibly due to the reduction of the search space detailed in the next subsection. This prompted us to interpolate the full model $\mathrm{P}(\mathbf{N})$ with the pitch-dependent Bernoulli model $\mathrm{P}_B(\mathbf{N})$ from Subsection IV-C2 with an interpolation factor μ:

$$\widehat{\mathbf{N}} = \arg\max_{\mathbf{N}} \mathrm{P}(\mathbf{S}|\mathbf{N})^{\kappa} \mathrm{P}_B(\mathbf{N})^{\mu} \mathrm{P}(\mathbf{N})^{(1-\mu)}. \tag{36}$$

3) Decoding: Decoding the most likely sequence of notes was performed with a Viterbi-like modified forward recursion, *i.e.*, a generalisation of the Viterbi algorithm to DBNs, first mentioned by Zweig [38] and Murphy [39] and later formally stated and analysed by Hu *et al.* [40].

However, the algorithm is in this case intractable due to the extremely large size of the solution space: for $K = 88$ (full piano range) there are $2^{88} \approx 3.1 \times 10^{26}$ possible values of $\mathbf{N}_t$. This is dealt with by reducing the search space: only a small number of most likely notes for every analysis frame are taken into account. First, at most Q pitches that are most salient in every frame are selected if their salience is higher than the threshold calculated as the crossing point of the active-note and the inactive-note salience models (in our case 0.0069, see Fig. 10); then, every possible q-combination of the selected notes is created, where $q = 1, \ldots, Q$ and evaluated with the salience model; finally, the L note combinations with the highest likelihood according to the salience model are selected and used in the frontier decoder. The Q and L parameters were set experimentally to 6 and 64 (2^6), respectively.

To reduce the effect of short-time salience fluctuations, the salience matrix was smoothed before selecting the most salient pitches, by applying a single-pole IIR filter to the salience sequence for every pitch with the same parameter a. The optimal value of a was determined experimentally and set to 0.5.

4) Evaluation metric: All multiple pitch estimation results were evaluated using an onset-based $\mathcal{F}$-measure, similarly to the MIR Evaluation Exchange (MIREX) [41]. The $\mathcal{F}$-measure is calculated as the harmonic mean of the precision $\mathcal{P}$ (ratio of the number of correctly detected notes to all detected

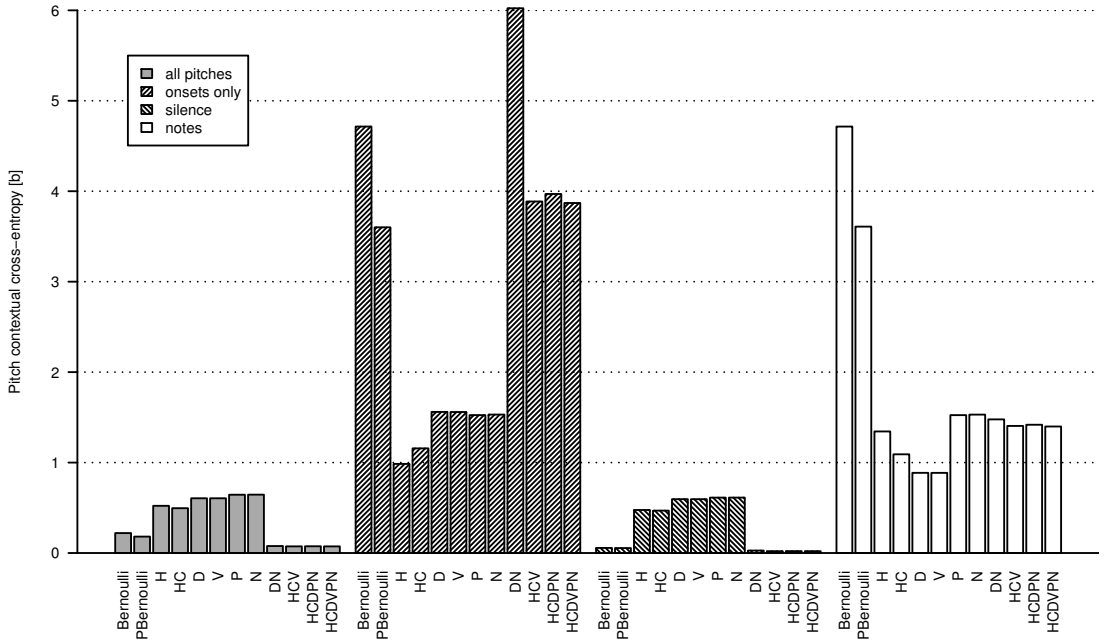

Fig. 9: Contextual cross-entropies (in bits) calculated on the test data set for log-linear interpolation. The contextual entropy for all pitches is identical to the conventional cross-entropy.

notes) and the recall $\mathcal{R}$ (ratio of the number of correctly detected notes to the number of ground-truth notes). A note was considered correctly detected if its pitch was exactly correct and its onset was within 1 frame (93 ms on average) of the correct onset position. The detected offset was ignored, as it is generally believed that an accurate estimation of the note offset is in many cases extremely difficult if not impossible.

5) Results: Due to high computational cost of the proposed algorithm, all audio recordings were cut to 320 frames ($53\frac{1}{3}$ beats), which corresponds to roughly 30 seconds in our data set (average tempo of about 108 beats per minute). The computational requirements for a single cut audio recording were still on the order of CPU days and gigabytes of memory, so the test data set was also reduced to 20 audio files.

In preliminary experiments, the average $\mathcal{F}$-measure was obtained for the values of the salience weighting coefficient κ between 0.5 and 2. The optimal value of κ was found to be around 1 for all the models, which suggests that the pitch models were well trained and properly normalised. The value of κ was fixed to 1 in further experiments.

The values of the Bernoulli model weight μ were varied between 0.6 and 1 and the resulting average $\mathcal{F}$-measures are plotted in Fig. 11. For all models the maximal average $\mathcal{F}$-measure is reached for μ between 0.8 and 0.9, with the exception of the Neighbour model, which was "flatter", with a peak around 0.7. The full model was not very sensitive to different μ values and it outperformed the reference pitch-dependent Bernoulli model for all values between about 0.75 and 1. In all cases decreasing the weight of the Bernoulli prior improved the note detection precision further, while at the same time increasing the recall, which suggests that the proposed pitch models play their role well and remove the spurious notes that, though plausible given the detected salience and the global note distribution, were unlikely to appear in the analysed signal in the particular context. This behaviour is highly desirable, because high precision is more important in pitch transcription than high recall as the spurious notes are often dissonant.

	PB	**D**	**N**	**P**	**V**	**HC**	**HCV**	**HCDVPN**
$\mathcal{P}$	73.0%	82.9%	74.2%	76.0%	83.1%	76.0%	83.4%	83.4%
$\mathcal{R}$	83.6%	78.7%	83.9%	82.7%	77.9%	82.8%	77.9%	78.4%
$\mathcal{F}$	76.1%	79.1%	77.2%	77.7%	78.7%	77.6%	78.9%	79.2%

TABLE V: Precision $\mathcal{P}$, recall $\mathcal{R}$ and $\mathcal{F}$-measure values obtained for the tested models for the optimal values of μ, compared with the baseline pitch-dependent Bernoulli (PB) model.

The estimation results, obtained for the optimal value of μ for each model, are summarised in Table V and visualised in Fig. 12. Every model offered a better performance than the baseline pitch-dependent Bernoulli model, with a 3.1% improvement in terms of $\mathcal{F}$-measure for the full HCDVPN model. It can also be observed that the "horizontal" models—V and D—had the biggest impact on this betterment, an observation analogous to that in the symbolic experiments in Subsection IV-C2. This observation, which, to the best of our knowledge, had not been made so far, has important implications for the design of computationally efficient multiple pitch estimation algorithms. Also, we remind that these experiments were made as a proof of concept and that increased accuracy may be achieved in the future using additional or alternative sub-models.

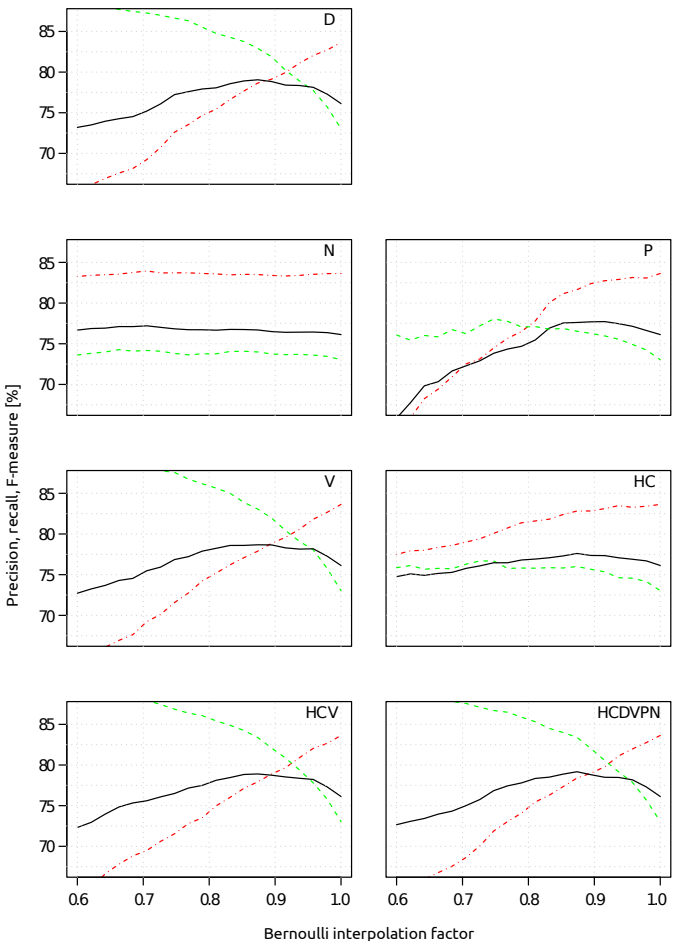

Fig. 11: Onset-based precision (green dashed line), recall (red dashed-dotted line) and $\mathcal{F}$-measure (solid line) against the Bernoulli model weight μ. The value of κ was fixed to 1.

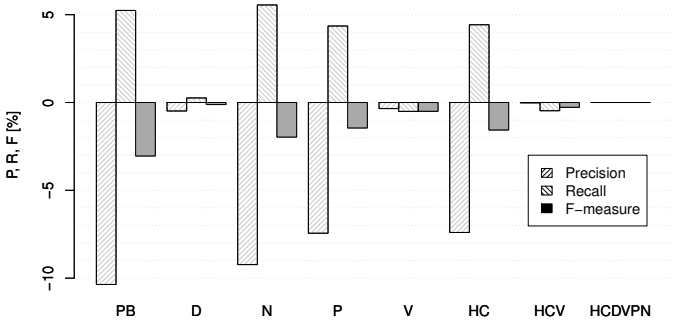

Fig. 12: Precision, recall and $\mathcal{F}$-measure obtained for the tested models, relative to the results of the full HCDVPN model, obtained for a model-dependent optimal value of μ and $\kappa = 1$.

V. Conclusion

In this paper, we have proposed a probabilistic polyphonic pitch model that can be used for multiple pitch estimation. The model is a three-layer dynamic Bayesian network with two hidden layers corresponding to the chords and the notes. The notes are efficiently modelled by means of linear and log-linear interpolation between simpler sub-models, each of which is responsible for modelling a different property of pitch.

The proposed framework was first evaluated in purely symbolic experiments, where we observed the modelling power quantified in terms of the cross-entropy and the contextual cross-entropy; in the acoustic experiments we have performed actual multiple pitch estimation with our proposed model, using a harmonic NMF model as the acoustic model. In both experiments the proposed model offered an improvement over the baseline technique, *i.e.*, a pitch-dependent Bernoulli model (equivalent to thresholding of the salience). Analysing the cross-entropies also showed that it is beneficial to combine sub-models by means of interpolation, as adding models decreases the cross-entropy, especially the contextual cross-entropy for note onsets. Log-linear interpolation, although computationally more demanding due to the need of renormalisation of the composite model, offered higher performance than linear interpolation. The improvement of model interpolation was confirmed in the multiple pitch estimation experiments, where all sub-models performed better than the reference Bernoulli model, while their combinations offered even higher performance, as measured by the note detection $\mathcal{F}$-measure.

This work provides a proof of concept of the usefulness of model interpolation and the models were chosen for their good modelling potential, but also their simplicity. The proposed framework is more general however and we believe that better models must be found and evaluated in the future. Defining and using such models in a way that remains computationally tractable is a significant challenge that lies beyond the scope of this paper. The interpolation of n-gram models with $n > 2$ has already been studied in the context of spoken language processing, which suggests it will also be applicable in the context of music. The use of even longer term models (rhythm history, structure, *etc.*) and additional "vertical" models (key, simultaneous onsets, *etc.*) is currently an open research issue.

In the future work, we will apply our methodology to address the dimensionality issues posed by other symbolic music modelling tasks [10]. Such issues arise when multiple musical variables are jointly modelled when estimating another variable (such as in this work), or when modelling dependencies with infinite-domain variables, *e.g.*, music tags and genres. The latter can also be dealt with by means of interpolating between a finite number of genre-specific models, effectively allowing for an infinite number of possible mixtures of genres.

References

[1] J. Paulus and A. Klapuri, "Music structure analysis using a probabilistic fitness measure and an integrated musicological model," in *Proc. 9^{th} International Conference on Music Information Retrieval (ISMIR)*, 2008, pp. 369–374.

[2] J. Pauwels and J. Martens, "Integrating musicological knowledge into a probabilistic framework for chord and key extraction," in *Proc. 128^{th} Audio Engineering Society Convention*, 2010.

[3] M. Ryynänen and A. Klapuri, "Polyphonic music transcription using note event modeling," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2005, pp. 319–322.

[4] Y. Ueda, Y. Uchiyama, T. Nishimoto, N. Ono, and S. Sagayama, "HMM-based approach for automatic chord detection using refined acoustic features," in *Proc. IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, 2010, pp. 5518–5521.

[5] H. Papadopoulos and G. Peeters, "Large-scale study of chord estimation algorithms based on chroma representation and HMM," in *Proc. International Workshop on Content-Based Multimedia Indexing (CBMI)*, 2007, pp. 53–60.

[6] S. Fukayama, K. Nakatsuma, S. Sako, T. Nishimoto, and S. Sagayama, "Automatic song composition from the lyrics exploiting prosody of the Japanese language," in *Proc. 7^{th} Sound and Music Computing Conference (SMC)*, 2010, pp. 299–302.

[7] A. Shirai and T. Taniguchi, "A proposal of an interactive music composition system using Gibbs sampler," *Human-Computer Interaction. Design and Development Approaches*, pp. 490–497, 2011.

[8] S. Flossmann, M. Grachten, and G. Widmer, "Expressive performance rendering with probabilistic models," in *Guide to Computing for Expressive Music Performance*, A. Kirke and E. Miranda, Eds. Springer London, 2013, pp. 75–98.

[9] T. Kim, S. Fukayama, T. Nishimoto, and S. Sagayama, "Performance rendering for polyphonic piano music with a combination of probabilistic models for melody and harmony," in *Proc. of Sound and Music Computing Conference (SMC)*, 2010, pp. 23–30.

[10] E. Vincent, S. Raczyński, N. Ono, and S. Sagayama, "A roadmap towards versatile MIR," in *Proc. 11^{th} International Conference on Music Information Retrieval (ISMIR)*, 2010, pp. 662–664.

[11] S. Abdallah and M. Plumbley, "Unsupervised analysis of polyphonic music by sparse coding," *IEEE Trans. Neural Networks*, vol. 17, no. 1, pp. 179–196, 2006.

[12] A. Klapuri, "Multipitch analysis of polyphonic music and speech signals using an auditory model," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 16, no. 2, pp. 255–266, 2008.

[13] ——, "Multiple fundamental frequency estimation by summing harmonic amplitudes," in *Proc. 7^{th} International Conference on Music Information Retrieval (ISMIR)*, 2006, pp. 216–221.

[14] H. Kameoka, T. Nishimoto, and S. Sagayama, "A multipitch analyzer based on harmonic temporal structured clustering," *IEEE Trans. Audio, Speech, and Language Processing*, vol. 15, no. 3, pp. 982–994, 2007.

[15] C. Févotte, N. Bertin, and J. Durrieu, "Nonnegative matrix factorization with the Itakura-Saito divergence: with application to music analysis," *Neural Computation*, vol. 21, no. 3, pp. 793–830, 2009.

[16] S. Raczyński, N. Ono, and S. Sagayama, "Multipitch analysis with harmonic nonnegative matrix approximation," in *Proc. 8^{th} International Conference Music Information Retrieval (ISMIR)*, 2007, pp. 381–386.

[17] P. Smaragdis and J. Brown, "Non-negative matrix factorization for polyphonic music transcription," in *IEEE Workshop on Applications of Signal Processing to Audio and Acoustics (WASPAA)*, 2003, pp. 177–180.

[18] E. Vincent, N. Bertin, and R. Badeau, "Adaptive harmonic spectral decomposition for multiple pitch estimation," *IEEE Trans. Audio, Speech and Language Processing*, vol. 18, no. 3, pp. 528–537, 2010.

[19] B. Niedermayer, "Non-negative matrix division for the automatic transcription of polyphonic music," in *Proc. 9^{th} International Conference on Music Information Retrieval (ISMIR)*, 2008, pp. 544–545.

[20] N. Boulanger-Lewandowski, Y. Bengio, and P. Vincent, "Discriminative non-negative matrix factorization for multiple pitch estimation," in *Proc. 13^{th} International Conference on Music Information Retrieval (ISMIR)*, 2012.

[21] M. Ryynänen and A. Klapuri, "Modelling of note events for singing transcription," in *Proc. ISCA Tutorial and Research Workshop on Statistical and Perceptual Audio*, Jeju, Korea, 2004.

[22] C. Raphael and J. Stoddard, "Harmonic analysis with probabilistic graphical models," in *Proc. 4^{th} International Conference on Music Information Retrieval (ISMIR)*, 2003, pp. 177–181.

[23] S. Raczyński, E. Vincent, F. Bimbot, and S. Sagayama, "Multiple pitch transcription using DBN-based musicological models," in *Proc. 11^{th} International Conference on Music Information Retrieval (ISMIR)*, 2010, pp. 363–368.

[24] K. Kashino, K. Nakadai, T. Kinoshita, and H. Tanaka, "Application of bayesian probability network to music scene analysis," in *Working notes of IJCAI Workshop on CASA*, 1995, pp. 52–59.

[25] M. Mauch and S. Dixon, "Simultaneous estimation of chords and musical context from audio," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 18, no. 6, pp. 1280–1289, 2010.

[26] F. Jelinek and R. Mercer, "Interpolated estimation of Markov source parameters from sparse data," in *Proc. 1^{st} International Workshop on Pattern Recognition in Practice*. Elsevier Science Ltd, 1980, pp. 381–397.

[27] D. Klakow, "Log-linear interpolation of language models," in *Proc. 5^{th} International Conference on Spoken Language Processing*, vol. 5, 1998, pp. 1695–1699.

[28] S. Chen and J. Goodman, "An empirical study of smoothing techniques for language modeling," in *Proc. 34^{th} annual meeting on Association for Computational Linguistics*, 1996, pp. 310–318.

[29] R Development Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2011, ISBN 3-900051-07-0. [Online]. Available: http://www.R-project.org/

[30] R. Scholz, E. Vincent, and F. Bimbot, "Robust modeling of musical chord sequences using probabilistic *n*-grams," in *Proc. IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2009, pp. 53–56.

[31] T. Fujishima, "Realtime chord recognition of musical sound: a system using Common Lisp music," in *Proc. International Computer Music Conference (ICMC)*, 1999, pp. 464–467.

[32] M. Goto, H. Hashiguchi, T. Nishimura, and R. Oka, "RWC music database: Music genre database and musical instrument sound database," in *Proc. 4^{th} International Conference on Music Information Retrieval (ISMIR)*, 2003, pp. 229–230.

[33] The Mutopia Project, "Free classical and contemporary sheet music," http://www.mutopiaproject.org/, March 2011.

[34] H. Kaneko, D. Kawakami, and S. Sagayama, "Functional harmony annotation database for statistical music analysis," in *Proc. International Conference on Music Information Retrieval (ISMIR)*, 2010. [Online]. Available: http://hil.t.u-tokyo.ac.jp/software/KSN/

[35] E. Charniak, *Statistical language learning*. MIT Press, 1996, ch. 2, pp. 34–36.

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

[37] S. Raczyński, N. Ono, and S. Sagayama, "Extending nonnegative matrix factorization—a discussion in the context of multiple frequency estimation of musical signals," in *Proc. 17^{th} European Signal Processing Conference (EUSIPCO)*, 2009, pp. 934–938.

[38] G. Zweig, "A forward-backward algorithm for inference in Bayesian networks and an empirical comparison with HMMs," Master's thesis, Dept. Computer Science, UC Berkeley, 1996.

[39] K. Murphy, "Dynamic Bayesian networks: representation, inference and learning," Ph.D. dissertation, University of California, 2002.

[40] W. Hu, Y. Zhang, Q. Diao, and S. Huang, "An efficient Viterbi algorithm on DBNs," in *Proc. 7^{th} European Conference on Speech Communication and Technology (Eurospeech)*, 2003, pp. 2533–2536.

[41] The MIREX community, "Music information retrieval evaluation exchange (MIREX)," http://www.music-ir.org/mirex/wiki/MIREX_HOME, September 2012.