Sound Cross-synthesis and Morphing Using Dictionary-based Methods

Collins, Nick; Sturm, Bob L.

Published in:
Proceedings of the International Computer Music Conference

Publication date:
2011

Document Version
Early version, also known as pre-print

Link to publication from Aalborg University

Citation for published version (APA):
Collins, N., & Sturm, B. L. (2011). Sound Cross-synthesis and Morphing Using Dictionary-based Methods. In *Proceedings of the International Computer Music Conference* ICMA.

SOUND CROSS-SYNTHESIS AND MORPHING USING DICTIONARY-BASED METHODS

Nick Collins

Department of Informatics
University of Sussex, Brighton, UK
N.Collins@sussex.ac.uk

Bob L. Sturm

Dept. Architecture, Design and Media Technology
Aalborg University Copenhagen, Denmark
bst@create.aau.dk

ABSTRACT

Dictionary-based methods (DBMs) provide rich possibilities for new sound transformations; as the analysis dual to granular synthesis, audio signals are decomposed into 'atoms', allowing interesting manipulations. We present various approaches to audio signal cross-synthesis and cross-analysis via atomic decomposition using scale-time-frequency dictionaries. DBMs naturally provide high-level descriptions of a signal and its content, which can allow for greater control over what is modified and how. Through these models, we can make one signal decomposition influence that of another to create cross-synthesized sounds. We present several examples of these techniques both theoretically and practically, and present on-going and further work.

1. INTRODUCTION

With a dictionary-based method (DBM), we can construct a flexible and parametrically-rich interface to audio content by modeling a signal in terms of parametric "atoms," each representing mid-level content of interest, e.g., time-frequency content having a particular time-domain envelope. A DBM specifies how to linearly combine atoms from a "dictionary" to reproduce a given sound, which makes it essentially the analytical equivalent of granular synthesis [11, 27]. Previous work demonstrates how a DBM can be used to transform audio signals, such as time stretching, pitch shifting [26], and various granular synthesis effects [11, 27]. In this paper, we explore how we can use models found by a DBM to create a type of cross-synthesis and morphing of sounds. We use "cross-synthesis" to refer to a sharing of characteristics between sounds, and "morphing" to refer to the transformation of one sound to others over time.

Of course, there exist many approaches to creating such effects. For example, with autoregressive modeling (linear prediction) [18], we can decompose a sound into a source and filter such that we can apply the filter to another source to make, e.g., a chainsaw having speech formants. Audio modeled by a combination of parametric sinusoids and noise allows one to effectively morph one sound into another [1, 14], e.g., to turn ringing bells into singing voices. Another method is adaptive concatenative sound synthesis [21, 24], which allows one to imitate the timbral material of one sound by others to reassemble, e.g., saxophone into speech.

In this paper, we describe ways to combine the characteristics of audio signals through a DBM, specifically using scale-time-frequency dictionaries. This can provide a multiresolution signal model that gives numerous possibilities for synthesis and analysis since each atom is associated with meaningful parameters, such as scale and time-frequency location. For these reasons, DBMs have been used for the analysis of signals having content spanning multiple time-scales, such as music [5, 27, 19], environmental sound [2], and biomedical signals [6]. In the end, we hope to use a DBM to obtain mid-level parametric models that facilitate the sharing of qualities between two or more signals with content difficult to model with mono-resolution and frequency-domain methods, e.g., drums. We first review DBMs, and then present several approaches to the cross-synthesis and analysis of sounds using a DBM. Then we present experiments, and discuss current research directions. Sound examples in this paper are available at: http:// www.cogs.susx.ac.uk/users/nc81/crossanalysiscross synthesis.html.

2. DICTIONARY-BASED METHODS

DBMs, more formally known as methods for "sparse approximation" [16], attempt to model a signal with a small number of atoms drawn from a user-defined dictionary, such as a family of scale-time-frequency atoms:

$$\mathscr{D} \triangleq \{d_\gamma(t) \triangleq Y_\gamma g(t-u;s)\cos(t\omega+\phi)\} \qquad (1)$$

where t is time, $g(t;s)$ is a lowpass function of time with scale $s > 0$, u is a time translation, ω is a modulation frequency, and ϕ is a phase offset. Each atom in $\mathscr{D}$ is indexed by $\gamma = (s,u,\omega,\phi) \in \Gamma$, which describes the atom parameters. Γ denotes the set of parameters possible in $\mathscr{D}$. The scalar Y_γ makes each atom have unit length, i.e., the inner product of any atom with itself is 1.

Table 1 shows an example of the parameters used in defining a scale-time-frequency dictionary (used for many of the simulations in this paper), where the window $g(t;s)$ is a zero-mean Gaussian function of finite-length s samples. For instance, the first row of this table specifies that $\mathscr{D}$ has atoms of scale 5.8 ms spaced in time every 2.9 ms, and with modulation frequencies spaced 43.1 Hz

s (samples/ms)	Δ_u (samples/ms)	Δ_f (Hz)
256/5.8	128/2.9	43.1
512/11.6	256/5.8	43.1
1024/23.2	512/11.6	43.1
2048/46.4	1024/23.2	21.5
4096/92.9	2048/46.4	10.8
8192/185.8	4096/92.9	5.4
16384/371.5	8192/185.8	2.7

Table 1. Scale-time-frequency dictionary parameters for a sampling rate of $F_s = 44.1$ kHz: scale s, time resolution Δ_u, and frequency resolution Δ_f.

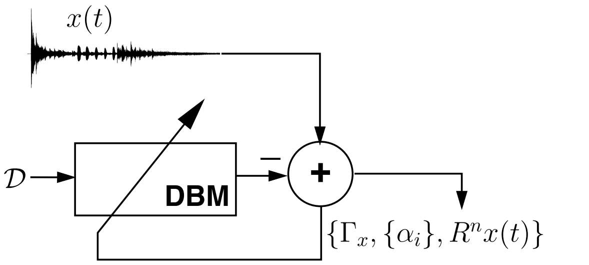

Figure 2. A DBM adaptively builds a model of $x(t)$ using atoms from the dictionary $\mathscr{D}$. The output is the set of indices Γ_x into the dictionary, the weights of the atoms in the model $\{\alpha_i\}$, and an error signal $R^n x(t)$.

from 0 to the Nyquist frequency. Figure 1 shows an example atom from this row. For a signal of duration t seconds, the number of atoms with this scale in $\mathscr{D}$ is about $176640t$. Similarly, if we performed a short-term Fourier transform (STFT) of the same signal with these parameters, then we would have about $176640t$ complex values. The total number of atoms in $\mathscr{D}$ for this signal, however, is about $494163t$.

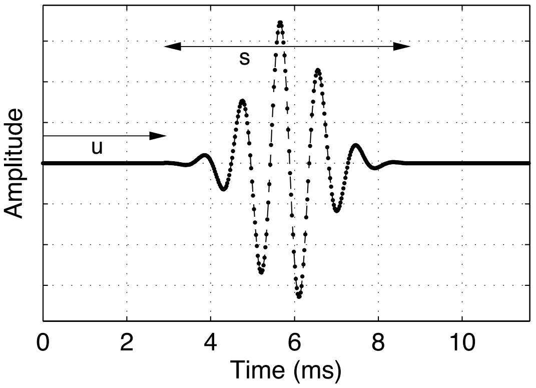

Figure 1. An example atom from the dictionary in Table 1. This scale $s = 5.8$ ms atom is translated to $u = 2.9$ ms, and has a modulation frequency of $\omega = 25(2\pi 43.1)$ rad/s.

Given a signal $x(t)$, the output of a DBM using a dictionary $\mathscr{D}$ is the model

$$x(t) - \sum_{i=1}^{n} \alpha_i d_{\gamma_i}(t) = R^n x(t),\ \gamma_i \in \Gamma_x \subset \Gamma \qquad (2)$$

where $R^n x(t)$ is an error, Γ_x is a set of indices pertaining to n atoms in the dictionary, and the set $\{\alpha_i : i = 1, 2, \ldots, n\}$ are the model weights. Figure 2 depicts an iterative and adaptive DBM, such as one from the matching pursuit family [16]. Here, the DBM selects one atom at each iteration based on an error, adds it to the model, and repeats the process with the new error until the model reaches an acceptable state. The quality of this model can be gauged, for instance, by some measure on the error $R^n x(t)$, or on the set of atoms selected [28]. There is a large variety of dictionary based methods, e.g., [10, 7, 8, 3, 16, 25]. Since the dictionary plays a crucial role in the performance of these methods, we use the name "dictionary based methods" to highlight this fact. For audio signals, scale-time-frequency dictionaries appear to be highly relevant perceptually and physically for efficiently modeling the underlying oscillatory phenomena [13, 23].

The sparsity of the models produced by a DBM, i.e., n in (2), provides a release from coefficient-heavy transforms, such as the STFT, or wavelet transforms, as long as the content we want to represent can be modeled well in $\mathscr{D}$. For this reason, DBMs are very robust to noise, or more generally, content that is not well-correlated with the dictionary [16]. We make this more concrete with the following example. A DBM might model (to a useful quality) a 10 second audio signal sampled at 44100 Hz using 5000 atoms — each with five associated parameters (scale, translation, frequency, phase, amplitude) if using the dictionary in Table 1. For an STFT using a window size of 1024 samples, a time-resolution (hop-size) of 512 samples, and no zeropadding (which contains the same atoms as described by the third row in Table 1), the resulting number of amplitudes and phases will be $2 * 513 * \lceil 10 * 44100/512 \rceil = 884412$. This is not only a doubling of the signal dimension, but also a 35-fold increase from the number of parameters in the model produced by the DBM. While $\mathscr{D}$ contains nearly 5 million atoms for this signal, a DBM selects in some sense only a few of the "best" ones. The STFT, on the other hand, contains all projections onto the set of atoms specified in the third row in Table 1.

DBMs have several shortcomings, however. First, the process is computationally expensive when the dictionary does not admit algorithms as fast as, for example, implementations of the discrete Fourier transform, or other decompositions over orthogonal bases. Among the methods for sparse decomposition, there are ones faster than others, e.g, matching pursuit [12, 15], and which can be implemented in a parallel architecture [4]. For the purposes of off-line audio effects and analysis, non-realtime performance does not pose a problem, as long as rendering completes within a 'reasonable' time. The computer simulations we discuss below took on the order of minutes to an hour. Second, not all elements of the resulting model (2) represent content in a signal. Some may result from mismatches between the signal and dictionary, or the greediness of a particular DBM [9, 27, 28]; this can creates audible artifacts when processing sparse models [26, 27] This problem is not yet settled; here we take a pragmatic approach and continue to explore the transformative possibilities.

3. ATOMIC CROSS-SYNTHESIS AND ANALYSIS

We are interested in ways to share and cross-influence the characteristics of two or more sounds. We assume their atomic models are expressed as in (2) produced by a DBM using, e.g., a scale-time-frequency dictionary $\mathscr{D}$ as in (1). In this section, we formally present several approaches to cross-synthesis and cross-analysis, first in the time-domain, and then in a sparse domain. We present computer simulations in Section 4.

3.1. Time-domain Transformation

Given the model of $x(t)$ in (2) produced by a DBM we can take another signal $y(t)$ and simply substitute for the coefficients $\{\alpha_i\}$ the inner products $\{\langle y, d_{\gamma_i}\rangle(t) : \gamma_i \in \Gamma_x\}$, where

$$\langle y, d_{\gamma_i}\rangle \triangleq \sum_t y(t) d_{\gamma_i}(t) \qquad (3)$$

is the projection $y(t)$ onto the atom $d_{\gamma_i}(t)$. This gives the model

$$y(t) - \sum_{i=1}^{n} \langle y, d_{\gamma_i}\rangle d_{\gamma_i}(t) = R^n y(t) \qquad (4)$$

where $R^n y(t)$ is the error. With the dictionary in (1), this can reinforce the scale-time-frequency content in $y(t)$ also present in $x(t)$, while suppressing other content. We can set the depth of the effect with the mixture

$$y_{px}(t) = (1-p)y(t) + p\sum_{i=1}^{n} \langle y, d_{\gamma_i}\rangle d_{\gamma_i}(t) = y(t) - pR^n y(t) \qquad (5)$$

where $0 \le p \le 1$. With $p = 0$ we have the original signal; and with $p = 1$ we have the model of $y(t)$ in terms of the model of $x(t)$. We can produce dynamic variation by varying p over time.

Rather than this direct resynthesis of $y(t)$ with its inner products with the atoms indexed by Γ_x, we can run the iterative decomposition process as in matching pursuit [16], but retain the order of the atoms in Γ_x. This produces the model

$$y(t) - \sum_{j=1}^{n} \beta_j d_{\gamma_j}(t) = R^n y(t). \qquad (6)$$

where the jth weight in (6) for $j \le n$ is given by

$$\beta_j = \langle R^j y(t), d_{\gamma_j}\rangle(t) = \langle y, d_{\gamma_j}\rangle - \sum_{i=1}^{j-1} \beta_i \langle d_{\gamma_i}, d_{\gamma_j}\rangle. \qquad (7)$$

This approach is used in the Matching Pursuit Dissimilarity Measure (MPDM) [17] to compare two signals through their sparse approximations. Equation 7 arises in this form by taking the inner product of 6 on both sides by $d_{\gamma_j}(t)$.

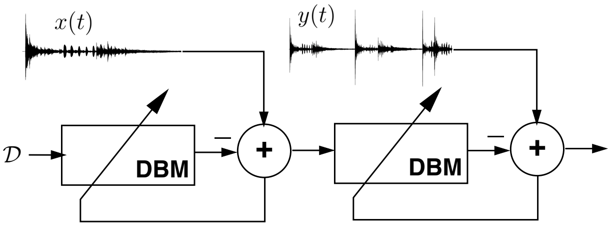

Figure 3. Sparse model of $x(t)$ is used by a DBM in the decomposition of $y(t)$.

Another possibility, depicted in Fig. 3, is to use the atoms indexed by Γ_x to define a new dictionary for a DBM in building a model of $y(t)$

$$\mathscr{D}_x \triangleq \{d_\gamma(t) \in \mathscr{D} : \gamma \in \Gamma_x\}. \qquad (8)$$

In this case, the decomposition of $y(t)$ by a DBM will use the scale-time-frequency domain content of $x(t)$ but will pay no attention to the ordering of Γ_x. For matching pursuit [16], this will give a signal model

$$y(t) - \sum_{j=1}^{n} \beta_j d_{\gamma_j}(t) = R^n y(t) \qquad (9)$$

where here each weight β_j is determined by the DBM using an intermediate residual. We can also create a larger dictionary by freely varying some parameters of $\mathscr{D}_x$, for example, the atom translations and phases and keeping the scales and modulation frequencies

$$\mathscr{D}_x \triangleq \{d_{\gamma_i'}(t) \in \mathscr{D} : \gamma_i' = (s_i, u, \omega_i, \phi), \{s_i, \omega_i\} \in \Gamma_x\}. \qquad (10)$$

We can of course vary all of these time-domain transformation methods by using any subset of the atoms indexed by Γ_x, for instance, using only atoms larger than a specific scale, redefining the atom selection criteria of the DBM, and so on. We can also incorporate the residual with the above transformations, which can restore fine details lost in the approximation process.

3.2. Sparse Domain Transformation

Given that we have two signals described by models like (2) found by a DBM, we can modify and compare the atom parameters in a sparse domain, which is depicted by Fig. 4. One simple process is to “fade-in” one sound while “fading-out” the other by weighting the atom amplitudes as a function of time. Based on the atom parameters for the dictionary in (1) we can do this as a function of atom modulation frequency and scale as well. These are essentially a type of “granular crossfade” [20, 11, 27].

We can cross-synthesize two sounds in a sparse domain by considering the parameters of atoms from both models. For instance, assume we have separated the large-scale atoms from the short-scale atoms in each model so we can limit our transformation to signal content with

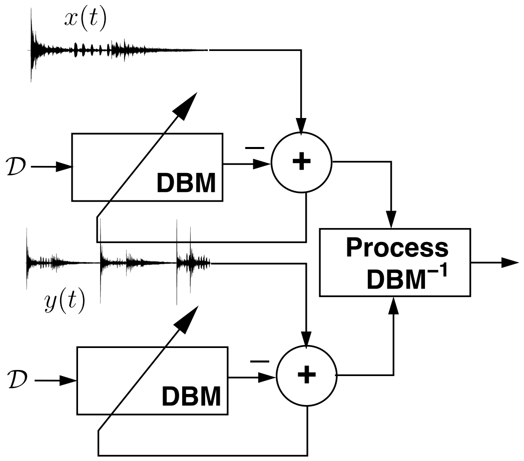

Figure 4. Sparse models of $x(t)$ and $y(t)$ are combined and possibly inverted to create another signal.

statistics that do not vary over time scales shorter than $s_{\min}$. Now, based on some scheme, e.g., minimizing a center-time-frequency distance, we pair the remaining atoms in each model $\{(\gamma_{x,i},\gamma_{y,j}) : \gamma_{x,i} \in \Gamma_x, \gamma_{y,j} \in \Gamma_y, s_{x,i} > s_{\min}, s_{y,j} > s_{\min}\}$ where Γ_x and Γ_y are the indices into $\mathscr{D}$ for the two signals, and $s_{x,i}$ is the scale of the ith atom modeling $x(t)$. For these atom pairs then, we can change the parameters of one to be closer to the other, for example, by averaging the modulation frequencies of each pair. We could also make each atom "chirp" between the two frequencies over its scale.

We can also adjust the parameters of each atom modeling $y(t)$ based on the atoms modeling $x(t)$. For example, for each atom parameter $\gamma_{y,j} \in \Gamma_y$ in the model of $y(t)$, we add to its modulation frequency $\omega_{y,j}$ a value δ computed from a weighted average of the frequencies in Γ_x and the set of atom weights $\{\alpha_i\}$ modeling $x(t)$

$$\delta = \sum_{i=1}^{|\Gamma_x|} (\omega_{y,j} - \omega_{x,i})\alpha_i e^{-r_u|u_{y,j}-u_{x,i}|} e^{-r_\omega|\omega_{y,j}-\omega_{x,i}|} \qquad (11)$$

where $r_u \geq 0$ and $r_\omega \geq 0$ weight the influence of atoms located in time and frequency, respectively, to $\gamma_{y,j}$. We can apply this transformation selectively again, such as only on large-scale atoms.

4. COMPUTER EXPERIMENTS

To explore these techniques, we conducted a series of experiments with a number of diverse audio signals and the DBM matching pursuit (MP) algorithm [16], primarily using the scale-time-frequency dictionary defined in Table 1. We altered the software library MP Toolkit (MPTK) [12], which is a core C++ library that efficiently implements MP for audio signal processing. To describe our practical work we follow the terminology of MPTK: a *block* essentially encompasses the set of functions describing a STFT with a particular scale (window size), time-resolution (hop size), and frequency resolution (zero padding); a *frame* is an indexed window position (time location) in a block; an *index* denotes the atom selected from the dictionary (γ from above) within a block at a given frame; and finally a *book* refers to all the atoms found by MP that constitute a signal model, which is essentially the model $\{\Gamma_x, \{\alpha_i\}, R^n x(t)\}$.

Since MPTK does not have much flexibility in selecting particular atoms for resynthesis or analysis, we modified the functionality of the library, as well as the MP decomposition process. To enable the process seen in Fig. 3 we added an auxilliary file mechanism which, for a given MP decomposition, stores the block, frame, and index choices of each iteration. This permits a previous signal decomposition (book) to guide that of another signal.

These source code modifications to MPTK 0.5.6 are available from the site accompanying this paper already mentioned under sound examples.

4.1. Time-domain Transformation

Equations (4)–(6) will amplify the scale-time-frequency characteristics of one signal that are common to another signal. The audible differences between (4) and (6) are very subtle, but in the former case we observed much more clipping in the results due to not subtracting the contribution of each atom before considering the next, as done in MP. We observed some effective results when using the mixture of the two syntheses in (5). The process depicted in Fig. 3 produced similar results using either dictionary in (8) or (10). The performance of the DBM, with respect to residual energy decay, is extremely diminished when using (8), which is no surprise when the two signals do not share the same scale-time-frequency structures. This performance increases with the dictionary in (10), but of course is not as good as that when using the overcomplete dictionary in (1). As an audio effect, however, these cross-synthesis methods led to some interesting hybrid sounds.

In some cases, using a book as a dictionary in a DBM generates a signal closer to the original book's source rather than the new signal, as seen in Fig. 5. Here the book produced from decomposing signal A was used in the decomposition of signal B, but with the dictionary given by (10), i.e., only the scales and modulation frequencies were constrained. We see a large similarity between the wivigrams of signal B decomposed with this book, and that of signal A; though elements of signal B can certainly be heard in the cross-synthesis, A is dominant. This can be reversed by switching the role of each signal. To examine this as an effect, we limited the use of the book to certain iterations of MP, and for the others we used a generalized scale-time-frequency dictionary. In such a case, we could easily hear the new sound with brief glimpses of the other sound where the dictionaries were switched.

4.2. Sparse Domain Transformation

If we combine the atoms of two books and then synthesize, we effectively mix the two signal approximations;

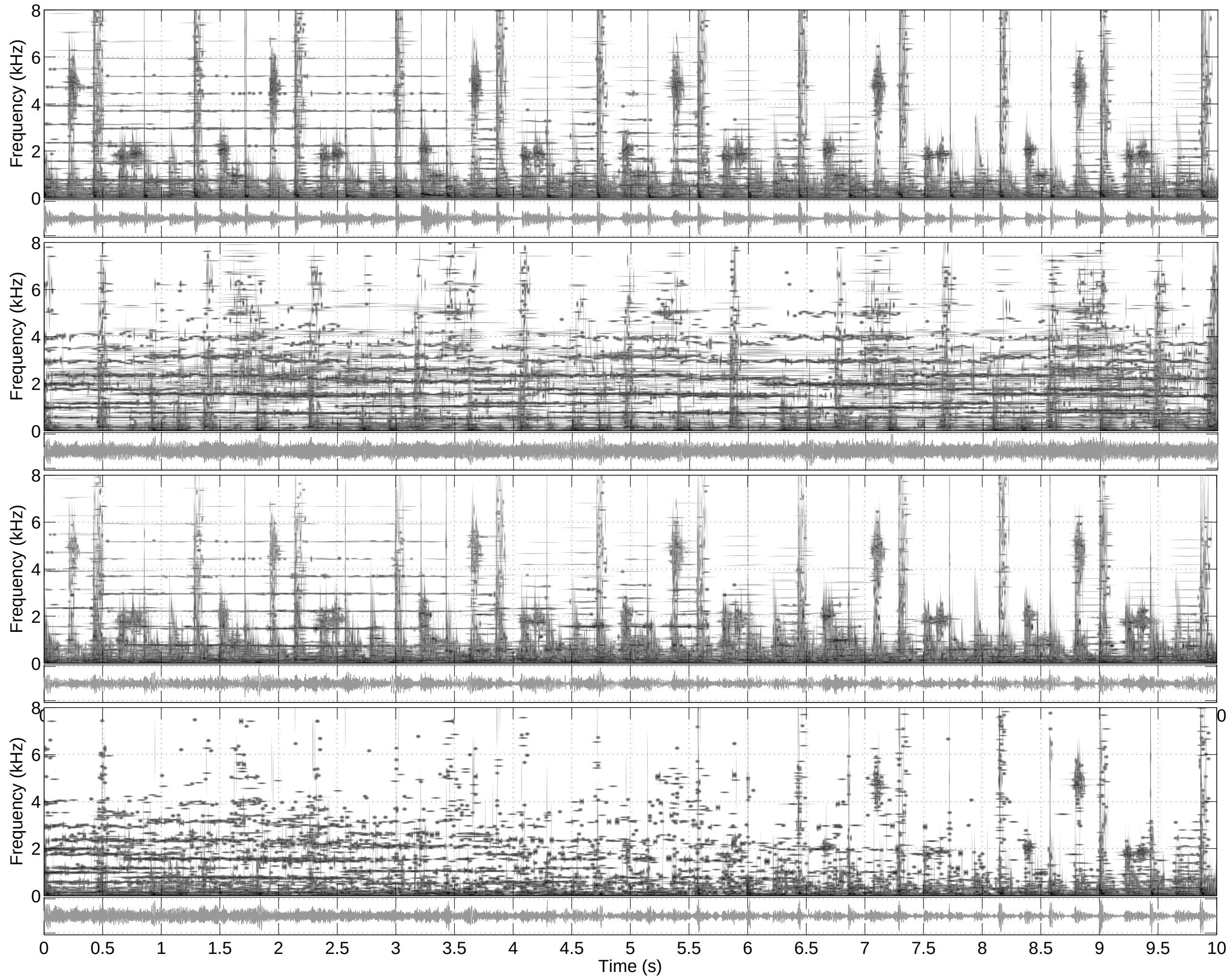

Figure 5. From top to bottom: wivigrams [27] of the atomic models of music signal A; music signal B; B decomposed as in Fig. 3 using (10) derived from the atomic model of A; gradual morph from B to A. Time-domain waveforms are shown below each wivigram.

but we can mix together different subsets as well, for instance based on modulation frequency and scale. Since each atom of a book is associated with a set of unique parameters, it is a simple matter to filter a book such that what remains is a subset of atoms within some range of parameters [27]. Through this, effects such as a granular crossfade can be created, as well as "evaporation," "coalescence," and "cavitation" [20, 11, 27]. We explored this for both monophonic and polyphonic signals, using different methods, for instance, selecting atoms from the two books in a deterministic or random manner. With a gradual transition, we hear one signal become less and less dense ending in a region sparsely populated by atoms, while the other fades in the same way. An example of this is seen at the bottom of Fig. 5.

5. DISCUSSION

Many of the transformations here can resemble the artifacts of poor audio coding; and we should expect nothing else when we use a limited dictionary produced from the analysis of one signal to represent another. For the pragmatic composer though, it does not matter if errors sometimes increase, or if decomposition convergence is broken, as long as interesting sounds result. And we have shown in this paper how DBMs can enable a variety of transformations, many of which have familiar counterpoints in existing granular synthesis techniques.

Though it is not possible in its present form to take sparse approximations of audio signals and produce high-quality cross-synthesis effects like those generated from high-detail parametric sinusoidal models [22, 1], we can, however, see DBMs as an intermediate step for producing such parametric models. Fundamentally, a DBM is, after all, an approximation method. This stands in contradiction to "high-quality" and "high-fidelity" audio signal processing. Of course, it is possible to reach any approximation error by a DBM as long as the dictionary is complete; but the interpretation of the model becomes difficult as the order of the model grows.

Since audio signal transformations through DBMs are naturally limited at the atomic level, it is critical to move beyond atomic level descriptions to, for instance, "molecular" descriptions of signal content [29], or higher-level parameteric models. An open question is how a multiresolution and sparse approximation can guide the creation of a high-level parametric model of an audio signal in terms of sound objects, like other approaches of analysis by synthesis. Another area of research is designing interfaces to make the creative exploration of these methods, and of visualizing the results of DBMs in general, more efficient.

There are plenty of paths to explore further, and which necessitate even more radical changes to the analysis software. Some of our ideas include:

- An "interlinked" MP decomposition of two audio signals, where, for example, we first choose one atom in signal A, and then impose the choice in B's first analysis step. In the second step we choose an atom in signal B, and then impose it in A's second analysis step. This two step process is then iterated.
- Cross-analysis of an audio signal using several books, even iterated through multiple generations of analysis. Each MP iteration can be guided in atom choice from one or more books. Furthermore, if there is still some scope for freedom in selection, as per equation (10), the book from one analysis can go on to influence the creation of a further, and so on.
- Time-stretching: allow use of atoms from some stretched or squashed region (larger or smaller time zone) in an existing book, relative to the region of the sound file currently being approximated.
- Given a set of sounds, assess their similarity via their books (as derived from conventional MP). Then choose books for cross-analysis based on the observed proximities.
- Looping analysis, where a shorter sound's book is "looped" in time (atoms repeated periodically) to analyse a longer sound.
- Incorporation of the residual to use the interesting shadowy sound worlds not captured in the approximation process.
- Use a set of sounds to learn a good set of atoms, and then use these in decomposing other signals not of the set (essentially, vector coding)

A real potential for transformation comes from subselections and substitutions in reading a book, or from combining more than one book to produce cross-syntheses.

6. CONCLUSION

We have presented various methods for the cross-synthesis and analysis of audio signals through DBMs with scale-time-frequency dictionaries. Our practical experiments, enabled by our alterations to the software library MPTK [12], show that the atomic models produced by DBMs can allow radical and interesting transformations of audio signals. The benefit of using a DBM with a scale-time-frequency dictionary over any redundant transformation based on orthogonal transforms, and/or a single time-domain resolution, e.g., the STFT, is that a DBM produces a multiresolution and parametric signal model without arbitrary segmentation, albeit at a higher computational cost, and as an approximation. We have focussed on sound transformations like morphs, but these techniques are also readily applicable to comparison of the content between sounds, for the purposes of analysis.

7. REFERENCES

[1] X. Amatriain, J. Bonada, A. Loscos, and X. Serra, "Spectral processing," in *DAFX — Digital Audio Effects*, U. Zölzer, Ed. Chicester, England: John Wiley and Sons Ltd., 2002, pp. 373–438.

[2] S. Chu, S. Narayanan, and C.-C. J. Kuo, "Environmental sound recognition with time-frequency audio features," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 17, no. 6, pp. 1142–1158, Aug. 2009.

[3] L. Daudet, "Sparse and structured decompositions of signals with the molecular matching pursuit," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 14, no. 5, pp. 1808–1816, Sep. 2006.

[4] ——, "Audio sparse decompositions in parallel: let the greed be shared!" *IEEE Sig. Process. Mag.*, Mar. 2010 (to appear).

[5] L. Daudet and B. Torrésani, "Sparse adaptive representations for musical signals," in *Signal Processing Methods for Music Transcription*, A. Klapuri and M. Davy, Eds. New York, NY: Springer, 2006, pp. 65–98.

[6] P. J. Durka, *Matching Pursuit and Unifiction in EEG analysis*, ser. Artech House Engineering in Medicine and Biology Series. Boston, MA: Artech House, 2007.

[7] R. Gribonval, "Fast matching pursuit with a multiscale dictionary of gaussian chirps," *IEEE Trans. Signal Process.*, vol. 49, no. 5, pp. 994–1001, May 2001.

[8] R. Gribonval and E. Bacry, "Harmonic decompositions of audio signals with matching pursuit," *IEEE Trans. Signal Process.*, vol. 51, no. 1, pp. 101–111, Jan. 2003.

[9] R. Gribonval, E. Bacry, S. Mallat, P. Depalle, and X. Rodet, "Analysis of sound signals with high resolution matching pursuit," in *Proc. IEEE-SP Int. Symp. Time-Freq. Time-Scale Anal.*, Paris, France, June 1996, pp. 125–128.

[10] R. Gribonval, P. Depalle, X. Rodet, E. Bacry, and S. Mallat, "Sound signal decomposition using a high resolution matching pursuit," in *Proc. Int. Comput. Music Conf.*, Hong Kong, Aug. 1996, pp. 293–296.

[11] G. Kling and C. Roads, "Audio analysis, visualization, and transformation with the matching pursuit algorithm," in *Proc. Int. Conf. Digital Audio Effects*, Naples, Italy, Oct. 2004, pp. 33–37.

[12] S. Krstulovic and R. Gribonval, "MPTK: Matching pursuit made tractable," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Process.*, vol. 3, Toulouse, France, Apr. 2006, pp. 496–499.

[13] M. S. Lewicki, "Efficient coding of natural sounds," *Nature Neuroscience*, vol. 5, no. 4, pp. 356–363, Mar. 2002.

[14] K. F. Lippold Haken and P. Christensen, "Beyond traditional sampling synthesis: Real-time timbre morphing using additive synthesis," in *Analysis, Synthesis and Perception of Musical Sounds*, J. Beauchamp, Ed. New York, NY: Springer, 2007, pp. 122–144.

[15] B. Mailhé, R. Gribonval, F. Bimbot, and P. Vandergheynst, "A low complexity orthogonal matching pursuit for sparse signal approximation with shift-invariant dictionaries," in *Proc. IEEE Int. Conf. Acoustics, Speech, Signal Process.*, Taipei, Taiwan, Apr. 2009, pp. 3445–3448.

[16] S. Mallat, *A Wavelet Tour of Signal Processing: The Sparse Way*, 3rd ed. Amsterdam, The Netherlands: Academic Press, Elsevier, 2009.

[17] R. Mazhar, P. D. Gader, and J. N. Wilson, "Matching pursuits dissimilarity measure for shape-based comparison and classification of high-dimensional data," *IEEE Trans. Fuzzy Syst.*, vol. 17, no. 5, pp. 1175–1188, Oct. 2009.

[18] F. R. Moore, *Elements of Computer Music*. Englewood Cliffs, NJ: P T R Prentice Hall, 1990.

[19] M. Morvidone, B. L. Sturm, and L. Daudet, "Incorporating scale information with cepstral features: experiments on musical instrument recognition," *Patt. Recgn. Lett.*, vol. 31, no. 12, pp. 1489–1497, Sep. 2010.

[20] C. Roads, *Microsound*. Cambridge, MA: MIT Press, 2001.

[21] D. Schwarz, "Concatenative sound synthesis: The early years," *J. New Music Research*, vol. 35, no. 1, 2006.

[22] X. Serra and J. O. Smith III, "Spectral modeling synthesis: a sound analysis/synthesis system based on a deterministic plus stochastic decomposition," *Computer Music J.*, vol. 14, pp. 12–24, 1990.

[23] E. Smith and M. S. Lewicki, "Efficient auditory coding," *Nature*, vol. 439, no. 23, pp. 978–982, Feb. 2005.

[24] B. L. Sturm, "Adaptive concatenative sound synthesis and its application to micromontage composition," *Computer Music J.*, vol. 30, no. 4, pp. 46–66, Dec. 2006.

[25] B. L. Sturm and M. Christensen, "Cyclic matching pursuit with multiscale time-frequency dictionaries," in *Proc. Asilomar Conf. Signals, Systems, and Computers*, Pacific Grove, CA, Nov. 2010.

[26] B. L. Sturm, L. Daudet, and C. Roads, "Pitch-shifting audio signals using sparse atomic approximations," in *Proc. ACM Workshop Audio Music Comput. Multimedia*, Santa Barbara, CA, Oct. 2006, pp. 45–52.

[27] B. L. Sturm, C. Roads, A. McLeran, and J. J. Shynk, "Analysis, visualization, and transformation of audio signals using dictionary-based methods," *J. New Music Research*, vol. 38, no. 4, pp. 325–341, Winter 2009.

[28] B. L. Sturm and J. J. Shynk, "Sparse approximation and the pursuit of meaningful signal models with interference adaptation," *IEEE Trans. Audio, Speech, Lang. Process.*, vol. 18, no. 3, pp. 461–472, Mar. 2010.

[29] B. L. Sturm, J. J. Shynk, A. McLeran, C. Roads, and L. Daudet, "A comparison of molecular approaches for generating sparse and structured multiresolution representations of audio and music signals," in *Proc. Acoustics*, Paris, France, June 2008, pp. 5775–5780.