# Granular Sound Spatialization Using Dictionary-Based Methods

Aaron McLeran*, Curtis Roads*, Bob L. Sturm†, John J. Shynk†
*Media Arts and Technology Program, University of California, Santa Barbara, CA 93106-6065, USA
†Department of Electrical and Computer Engineering, University of California, Santa Barbara, CA 93106-9560, USA

***Abstract*—We present methods for spatializing sound using representations created by dictionary-based methods (DBMs). DBMs have been explored primarily in applications for signal processing and communications, but they can also be viewed as the analytical counterpart to granular synthesis. A DBM determines how to synthesize a given sound from any collection of grains, called *atoms*, specified in a *dictionary*. Such a granular representation can then be used to perform spatialization of sound in complex ways. To facilitate experimentation with this technique, we have created an application for providing real-time synthesis, visualization, and control using representations found via DBMs. After providing a brief overview of DBMs, we present algorithms for spatializing granular representations, as well as our application program *Scatter*, and discuss future work.**

## I. Introduction

Sound can be spatialized on multiple time scales [1]. In classic electronic music, many compositions are characterized by a global spatial perspective, such as a uniform blanket of reverberation applied to the entire macrostructure of a composition, e.g., Oskar Sala's *Elektronische Impressionen* (1978). In other works, spatial variations articulate mesostructural boundaries: phrases and sections. For example, Stockhausen's *Kontakte* (1960) contrasted sounds in foreground/background relationships on a time scale of phrases within moments [2].

Later, through the development of music programming languages and digital audio editors, the time scales of spatial transformations were reduced down to the level of individual sound objects. A cascading sequence of sound objects, each emanating from a different virtual space, provides the dimension of spatial depth to an otherwise flat perspective and articulates a varying topography.

Below the level of individual sound objects is the world of microsound [1]. Gabor proposed that all sound could be decomposed into a family of functions obtained by time and frequency shifts of acoustic "quanta" [3], [4]. The composer Xenakis extended Gabor's theory and proposed its inverse: any given sound can be composed, or synthesized, by elementary sonic *grains* [5]. Today, it is possible to decompose and recompose sound by a variety of means. Some methods, such as granulation, work directly in the signal time domain [1]. However, in the dictionary-based methods (DBMs) described later in this paper, a granular representation of a signal is provided through time-frequency analysis. By means of these techniques, spatialization can now be explored down to the microsound level of sonic structure, where individual spatial positions are assigned to every sonic grain.

## II. Dictionary-Based Methods

DBMs provide an alternative to time-frequency signal representations, such as those made by short-term Fourier and wavelet analyses. While Fourier analysis is built upon complex sinusoids, and wavelet analysis uses the dilation of a mother wavelet, DBMs allow for any set of functions – collectively called the *dictionary*. The general idea behind DBMs is to avoid making an a priori decision about a basis that best represents a particular signal; instead, the representation basis is allowed to adapt to the signal statistics [6]. This can result in representations that are more sparse, efficient, and meaningful than those found by standard analysis methods [6], [7]. So far, DBMs have been primarily applied in applications of communications and signal processing (e.g., see [8]–[10]). Research using DBMs for sound transformation applications has only recently begun [11]–[13].

In DBMs, a signal is represented as a linear combination of waveforms chosen from a predefined dictionary of possible waveforms. Let the signal be denoted by the $K$-dimensional column vector $\mathbf{x}$, and let the dictionary be denoted by the matrix $\mathbf{D}_{K\times N}$, where each column is an individual waveform. The signal $\mathbf{x}$ can thus be written as

$$\mathbf{x} = \mathbf{D}\mathbf{s} \tag{1}$$

where $\mathbf{s}$ is a column vector of $N$ weights. Observe that if $\mathbf{D}^H$ is the complex conjugate transpose of the orthonormal discrete Fourier transform matrix, then $\mathbf{s}$ is simply the Fourier transform of $\mathbf{x}$. In general, however, $N \gg K$ and $\mathbf{D}$ is overcomplete, meaning that rank$(\mathbf{D}) = K$. This implies that there will always exist at least one solution $\mathbf{s}$ satisfying (1), and possibly an infinite number of solutions. In general, without specifying any constraints, finding a solution to (1) is an ill-posed problem. Constraining the solution $\mathbf{s}$ to have the minimum number of nonzero elements creates an NP-hard problem [14]. A more relaxed constraint involves minimizing the $\ell_1$-norm of $\mathbf{s}$, which creates a convex problem solvable by a linear program [7]. An entirely different set of methods for solving (1) are based on gradient descent [6], [15].

### *A. Matching Pursuit Algorithm*

The matching pursuit (MP) algorithm is quite simple, and fast implementations exist [16]. MP iteratively builds the representation basis by choosing atoms from a given dictionary $\mathbf{D} = [\mathbf{d}_1|\mathbf{d}_2|\cdots|\mathbf{d}_N]$, where each column $\mathbf{d}_i$ is a unique waveform. At step $n+1$, a column is selected from $\mathbf{D}$ that has the largest magnitude inner product with the $n$th-order residual signal

$$\mathbf{g}_n = \arg\max_{\mathbf{d}\in\mathbf{D}} \left|\mathbf{d}^T\mathbf{r}(n)\right|/||\mathbf{d}|| \tag{2}$$

where $\mathbf{r}(n) = \mathbf{x} - \widetilde{\mathbf{x}}(n)$ ($\mathbf{r}(0) \equiv \mathbf{x}$), and $\widetilde{\mathbf{x}}(n)$ is the $n$th-order approximation waveform ($\widetilde{\mathbf{x}}(0) \equiv \mathbf{0}$). The complexity of finding each atom in MP is on the order of computing a fast Fourier transform of the entire signal [6], [16]. After choosing $\mathbf{g}_n$, its corresponding weight is computed as

$$a_n = \mathbf{g}_n^T\mathbf{r}(n)/||\mathbf{g}_n||. \tag{3}$$

The $(n+1)$st-order residual signal is then given by

$$\mathbf{r}(n+1) = \mathbf{r}(n) - a_n\mathbf{g}_n, \tag{4}$$

and the algorithm repeats until some stopping criterion is met. After $n$ iterations, the $n$th-order approximation of the original signal $\mathbf{x}$ is given by:

$$\widetilde{\mathbf{x}}(n) = [\mathbf{g}_0|\mathbf{g}_1|\cdots|\mathbf{g}_{n-1}] \begin{bmatrix} a_0 \\ a_1 \\ \vdots \\ a_{n-1} \end{bmatrix} \triangleq \mathbf{G}(n)\mathbf{a}(n). \tag{5}$$

If the dictionary is at least complete, i.e., $\text{rank}(\mathbf{D}) = K$, then $\widetilde{\mathbf{x}}(n)$ will converge to the original signal $\mathbf{x}$ [6]. While MP does not guarantee that this will occur after a finite number of steps, orthogonal MP does [15] – but at a higher computational cost. For our applications, however, the approximations created by MP provide a useful and meaningful representation of the original signal.

### *B. Building and Specifying Dictionaries*

Dictionaries are often constructed from a combination of discretized, scaled, translated, and modulated lowpass functions. For instance, a dictionary element can be parametrically described by

$$g(k) = h(k-u;s)\cos\big([k-u]\omega(k-u) + \phi(k-u)\big) \tag{6}$$

where $0 \le k \le K-1$ is a time index, $0 \le u < K - s/2$ is a translation, $1 \le s \le K$ is a scale in samples, and $0 \le \omega(k) \le \pi$ and $0 \le \phi(k) < 2\pi$ are the modulation frequency and phase, respectively, which might depend on time, such as chirps [17]. The function $h(k;s)$ can be likened to a window function. For a Gabor atom [3], [6], $h(k;s)$ is the Gaussian function

$$h(k;s) = \begin{cases} \exp\big(-\frac{(k-s/2)^2}{2(\alpha s)^2}\big), & k = 0,1,\ldots,s-1 \\ 0, & \text{else} \end{cases} \tag{7}$$

where $\alpha$ controls the variance, and $s$ is the scale. A plot of an example Gabor atom is shown in Fig. 1.

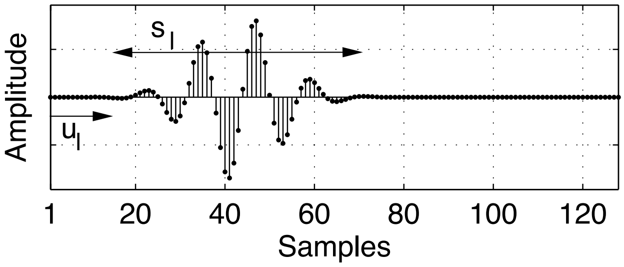

Fig. 1. Example Gabor atom with scale $s_l$ and translation $u_l$.

A dictionary is created by combining numerous atoms with various scales, translations, and modulation frequencies. In contrast to Fourier and wavelet transforms, this produces a dictionary which tiles the time-frequency plane in multiple ways [18].

## III. Spatialization Using Dictionary-Based Methods

After the MP algorithm is performed to a satisfactory signal-to-residual ratio (SRR), the results of the decomposition, i.e., the chosen atoms and weights, are stored as a collection of indices from the dictionary in what is called a *book*. Because each atom is parameterized, many unique sound transformations are possible [11]–[13]. This paper presents recent experiments using novel atom spatialization techniques and a variety of basis functions such as Gabor atoms, or damped sinusoids.

### *A. Two-Dimensional Spatialization Coordinate*

In order to simplify our initial experimentation, spatialization was restricted to a circular two-dimensional (2D) array of $m$ channels. A general 2D spatial coordinate is specified by the parameter $p \in [0,1]$. For the case of a stereo channel ($m = 2$), $p$ is the stereo panning parameter. In a more general case, $p$ is interpolated across the $m$ channels such that each channel contains a number which represents the amount of an atom in that channel.

Assuming that the 2D array of $m$ channels is circular, $p$ wraps around to remain within its specified range. For example, $p = 1.1$ wraps to $p = 0.1$. In this paper, we assume that $p$ is a singular spatial point and not a spatial distribution.

### *B. Random Scattering*

If the atoms are scattered by an amount $\sigma \in [0,1]$, the spatial coordinate $p$ for each atom becomes simply

$$p = \sigma. \tag{8}$$

If $\sigma$ is a number generated from a uniform distribution and is unique for each atom, the result is maximum spatial scattering, because each atom occupies a unique position in space. If every atom is spatialized by the same $\sigma$, either randomly generated or manually set, then the result is the opposite of scattering: instead, the entire book is localized to a singular spatial position.

### C. Blur

In order to achieve spatial blur, another spatial parameter is added to $\sigma$ in (8), yielding

$$p = \sigma + \beta \tag{9}$$

where $\beta$ is a number generated from any desired probability distribution supported on the bounded interval $[-r, r]$. If $\sigma$ is the same for all atoms in a book and $\beta$ is uniquely generated for each atom, then the result is a spatial blur localized at $\sigma$.

### D. Convergence and Divergence

If in (9) the interval range of $\beta$ is reduced to zero ($r \longrightarrow 0$), the spatialization will simply become (8). If this occurs over some time interval, and $\sigma$ is the same for every atom of a book, the effect is spatial convergence to the spatial location specified by $\sigma$. On the other hand, if $r \longrightarrow x$ where $x \in [0, 1]$, the result is spatial divergence.

### E. Panning

Panning is distinct from scattering, convergence, and divergence, in that sound appears to move dynamically through space. Because atoms are typically of a very short duration, dynamically changing $p$ for each atom has no perceivable effect. Therefore, the illusion of panning is achieved by individually spatializing atoms such that each atom's spatial coordinate $p$ is set according to a global function $f(u)$ where $u$ is the atom's time translation from (6). Thus, we can write

$$p = f(u) \tag{10}$$

where $f(u)$ is defined according to any desired process. For example, it might be a slowly varying low-frequency oscillator (LFO), a manually defined break-point function set from a graphical user interface (GUI), or some other algorithmic or stochastic process.

### F. Spatializing According to Parametric Filtering

All atomic parameters from (6) are available for the construction of unique spatializing algorithms. For example, because transients are typically composed of very short duration atoms, the following rule spatially moves transient atoms of a book differently than tonal atoms:

$$p = \begin{cases} f(u), & s < \alpha \\ g(u), & \text{else} \end{cases} \tag{11}$$

where $s$ is an atom's scale (duration) value, and $\alpha$ is a tunable threshold below which atoms are most likely part of a transient structure. $f(u)$ and $g(u)$ are different functions which depend on an atom's translation parameter; they can be defined according to any desired procedure. The result is the spatial dislocation of a sound's noisy transients and its harmonic tonals. Many such algorithms for spatial scattering or spatial motion are possible via a desired combination or filtering of atomic parameters.

### G. Stochastic Panning

Setting $\sigma$ and $\beta$ from (9) to be stochastic functions that depend on atomic translation (similar to $f(u)$ in (10)) leads to fully dynamic and stochastic spatialization techniques. For example, multiple clusters of atoms, built from filtering the book according to any number of desired atomic parameters, might expand or contract into spatial clouds which move across a spatial field at unique varying rates.

## IV. Scatter: A Real-Time Application Program for Manipulating Atomic Representations

### A. Implementation Details

The software for Scatter was written in C++ and Objective-C using Mac OS X's Cocoa API. The synthesis was performed using a software toolkit currently under development in the Media Arts and Technology (MAT) Program at UCSB. The implementation assumes traditional block processing of groups of samples at a fixed rate, and follows well-known techniques for real-time granular synthesis [19]. However, instead of scheduling atoms within a block of samples according to purely synthetic procedures, they are scheduled according to their temporal location within a time-sorted decomposition book derived via the MP algorithm.

For a time-sorted dictionary, atom scheduling is usually not problematic as long as the dictionary is queried only for the sample range of the currently executing block as opposed to the entire book, which may contain many thousands of atoms. However, if the oscillators used for atoms are sine waves, scheduling issues may arise when the atom density as a function of time at any point in the book is extreme. Large atom densities typically correspond to complex components within a given signal, such as transients or noise. Therefore, instead of using computed sine waves, atoms are mostly synthesized using a simple sine oscillator based on a two-pole resonator, which requires only one multiply and add per sample computation. The downside to using resonators is that they are expensive if their frequency or phase is changed. Since atoms are typically of very short duration, little benefit is achieved when individual atomic parameters are changed within the block, so this is usually an acceptable compromise.

However, DBMs allow any type of waveform to be included in the dictionary, and it is possible that atoms may have durations on the order of seconds or longer. Long-duration atoms need to have the ability to change their parameters in real-time in order to avoid unwanted artifacts. Therefore, long-duration atoms are synthesized using a relatively simple computed third-order polynomial sine wave that can dynamically change its parameters with essentially no increase in the computation time.

### B. Visualizing Decompositions

In order to accurately represent the energy content of individual atoms, they are represented graphically using their Wigner-Ville distribution (WVD) [20]. The WVD of

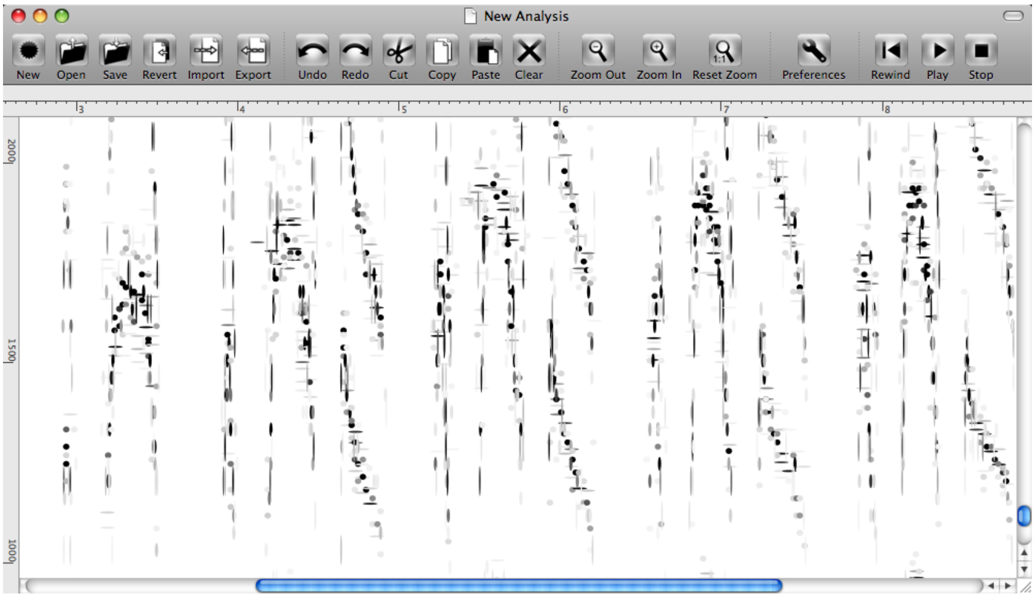

Fig. 2. GUI prototype for Scatter showing WVD plot of a decomposition.

a Gabor atom is a two-dimensional Gaussian waveform centered on a modulation frequency and time translation. Figure 2 is a screenshot of an early prototype of the main GUI for Scatter, which was influenced by SPEAR [21]; the figure illustrates the WVD plot for a decomposition. We call the superposition of WVDs of the atoms in a decomposition a *wivigram*, which has proven to be useful as a means of visualizing and interacting with atomic decompositions.

### C. GUI Components

*Selection, Filtering, Parametric Transformations:* Several options are available for selecting individual atoms within the decomposition. They can be chosen individually, via lasso or box, or by using bounding regions in frequency and time. Once selected, the atoms can be transformed according to any of the atomic parameters such as time and frequency translation, compression, or dilation. Atoms may also be deleted, copied, or pasted.

*GUI for Spatialization:* A set of GUI controls has been designed which allow the user to specifically dictate the various techniques mentioned in this paper for controlling the stochastic spatial parameters. The GUI uses standard controls such as sliders, knobs, and break-point functions, which when combined with any of the selection and editing controls shown in Fig. 2, allow a user to apply any of the previously mentioned spatialization algorithms.

### D. *Extensions to* Scatter

*Molecular Selection and Transformation:* Currently, the implementation allows only selection and transformation at the atomic level. Because books consist of many thousands of atoms, it is often difficult to perform transformations on meaningful structures in a signal. For example, it is currently difficult to select and transform individual harmonics. Thus, current work is focusing on the development of algorithms which automatically construct higher level molecular models of the decomposition and allow for intuitive GUI control and manipulations of molecules. However, these techniques are still experimental and have not yet been implemented for Scatter.

*Analysis Stage:* Real-time synthesis is currently being implemented using books analyzed from the Matching Pursuit Toolkit (MPTK) [16]. This has allowed development efforts to focus on real-time synthesis and GUI interactions and processing rather than the MP implementation. However, in order to fully take advantage of the unique benefits of DBMs, Scatter should include access to the analysis, and allow users to easily customize dictionaries, or set analysis parameters such as SRR to define a desired model order.

## V. Future Work: Micropluriphony in the Allosphere

Stereophony, quadraphony, and octophony refer to sound positioning in a symmetrical lateral array in front of or around the listener. Periphony extends this scheme to the vertical dimension [22]. Using techniques such as wave field synthesis, the notion of periphony is extended to pluriphony: the projection of three-dimensional (3D) sounds from a variety of positions above, below, and within the audience.

MAT's current testbed for spatialization is the Allosphere at UCSB [23]. The Allosphere is a three-story-high spherical instrument in which virtual environments and performances can be experienced with full 360-degree immersion. The space is now being equipped with high-resolution active stereo projectors, a 3D sound system with several hundred speakers, and with tracking and interaction mechanisms.

Our work on spatializing atomic decompositions using DBMs has so far been focused on 2D spatialization techniques. However, current efforts are underway to extend the spatialization methods discussed here to full 3D spatialization within the Allosphere. There are many technical challenges, particularly those of scale. As is common to granular synthesis in general, spatialization of atomic decompositions faces an explosion in the number of parameters that are needed for the control of the position and movement of possibly thousands of sound events per second. A similar problem of scale arises when projection is extended from a 2D spatial field to a fully pluriphonic space with potentially hundreds of channels, such as in the Allosphere.

## Acknowledgment

The authors would like to thank MAT Ph.D. students Lance Putnam and Graham Wakefield for consultations on various implementation details of Scatter. This work was supported in part by the National Science Foundation under Grant CCF-0729229.

## References

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

[2] C. Roads, "Decyphering Stockhausen's *Kontakte*," in preparation 2008.

[3] D. Gabor, "Theory of communication," *J. IEE*, vol. 93, no. 3, pp. 429–457, Nov. 1946.

[4] D. Gabor, "Acoustical quanta and the theory of hearing," *Nature*, vol. 159, no. 4044, pp. 591–594, May 1947.

[5] I. Xenakis, "Elements of stochastic music," *Gravensaner Blätter*, vol. 18, pp. 84–105, 1960.

[6] S. Mallat and Z. Zhang, "Matching pursuits with time-frequency dictionaries," *IEEE Trans. Signal Process.*, vol. 41, no. 12, pp. 3397–3415, Dec. 1993.

[7] S. S. Chen, D. L. Donoho, and M. A. Saunders, "Atomic decomposition by basis pursuit," *SIAM J. Sci. Comput.*, vol. 20, no. 1, pp. 33–61, Aug. 1998.

[8] R. M. Figueras i Ventura, P. Vandergheynst, and P. Frossard, "Low-rate and flexible image coding with redundant representations," *IEEE Trans. Image Process.*, vol. 15, no. 3, pp. 726–739, Mar. 2006.

[9] M. Elad and M. Aharon, "Image denoising via sparse and redundant representations over learned dictionaries," *IEEE Trans. Image Process.*, vol. 15, no. 12, pp. 3736–3745, Dec. 2006.

[10] S. Lesage, S. Krstulovic, and R. Gribonval, "Underdetermined source separation: Comparison of two approaches based on sparse decompositions," in *Proc. Int. Conf. Independent Component Analysis Blind Source Separation*, Charleston, SC, Mar. 2006, pp. 633–640.

[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] 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.

[13] B. L. Sturm, C. Roads, A. McLeran, and J. J. Shynk, "Analysis, visualization, and transformation of audio signals using overcomplete methods," in *Proc. Int. Computer Music Conf.*, Belfast, Ireland, Aug. 2008.

[14] G. Davis, S. Mallat, and M. Avellaneda, "Adaptive greedy approximations," *J. Constr. Approx.*, vol. 13, no. 1, pp. 57–98, Jan. 1997.

[15] Y. Pati, R. Rezaiifar, and P. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proc. Asilomar Conf. Signals, Syst., Comput.*, Pacific Grove, CA, Nov. 1993, vol. 1, pp. 40–44.

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

[17] R. Gribonval, "Partially greedy algorithms," in *Trends in Approximation Theory*, K. Kopotun, T. Lyche, and M. Neamtu, Eds., pp. 143–148. Vanderbilt University Press, May 2001.

[18] S. Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, San Diego, CA, 2nd edition, 1999.

[19] B. Truax, "Real-time granular synthesis with a digital signal processor," *Computer Music Journal*, vol. 12, no. 2, pp. 14–26, 1988.

[20] D. Preis and V. C. Georgopoulos, "Wigner distribution representation and analysis of audio signals: An illustrated tutorial review," *J. Audio Eng. Soc.*, vol. 47, no. 12, pp. 1043–1053, Dec. 1999.

[21] M. Klingbeil, "Software for spectral analysis, editing, and synthesis," in *Proc. Int. Computer Music Conf.*, Barcelona, Spain, Sep. 2005.

[22] M. Gerzon, "Periphony: With height sound reproduction," *J. Audio Eng. Soc.*, vol. 21, no. 1, pp. 2–10, 1973.

[23] X. Amatriain, J. Castellanos, T. Höllerer, J. Kuchera-Morin, S. T. Pope, G. Wakefield, and W. Wolcott, "Experiencing audio and music in a fully immersive environment," in *Lecture Notes in Computer Science: Sense to Sound*, K. K. Jensen, R. Kronland-Martinet, and S. Ystad, Eds. Springer Verlag, Berlin, Germany, in press 2008.