

FREEDOM IN TAPESTREA! VOICE-AWARE TRACK MANIPULATIONS

Tom Lieber
Princeton University
lieber@princeton.edu

Ananya Misra
Princeton University
amisra@cs.princeton.edu

Perry Cook
Princeton University
prc@cs.princeton.edu

ABSTRACT

Sinusoidal tracks in TAPESTREA have too long been oppressed, unable to choose which other tracks they hang out with, and only able to be modified in fixed groups. They had no say in their own lives, no individuality. That ends today. We advocate FREEDOM in TAPESTREA by way of VTAPS, where every track can be interacted with as a distinct, FREE entity. Only thus can it contribute fully to the larger sound scene. Please join our movement by downloading TAPESTREA at

<http://taps.cs.princeton.edu/>.

TAPESTREA is an environmental audio editing system that encourages the user to work with pitched sounds in terms of salient frequencies that modulate over time, otherwise called tracks. While TAPESTREA has provided the means for manipulating these tracks as groups (called “templates”), the aim of this project – VTAPS – is to allow the user to manipulate tracks individually or in arbitrary groups. This allows for advanced manipulations to be performed quickly and with more control than when templates were indivisible units. It inspires and enables features for manipulating the audio in ways that are useful for voice recordings and musical instruments.

1. INTRODUCTION

TAPESTREA [7] is a framework for creating music by re-composing existing sounds. It lets users extract desired portions of a recording, and stores them as *sound templates* of different types based on the modeling technique best suited for the selection. TAPESTREA supports the three-step workflow of extracting reusable sound templates from selected time-frequency regions in a recording, transforming them independently, and re-synthesizing the transformed templates to create new music.

The types of templates extracted can roughly correspond to foreground and background components of an audio recording. *Stochastic background* is the ambient noise that characterizes the background “din” of an environment. *Transient events* are brief, noisy, non-pitched audio events like door slams or claps. *Sinusoidal events* correspond to stable sinusoidal components of a sound. Voices, bird chirps, and other components perceived as pitched foreground events work well as sinusoidal events.

Typically, a sinusoidal event template in TAPESTREA can be interactively extracted and later transformed as a single unit. The current work, VTAPS, facilitates more in-

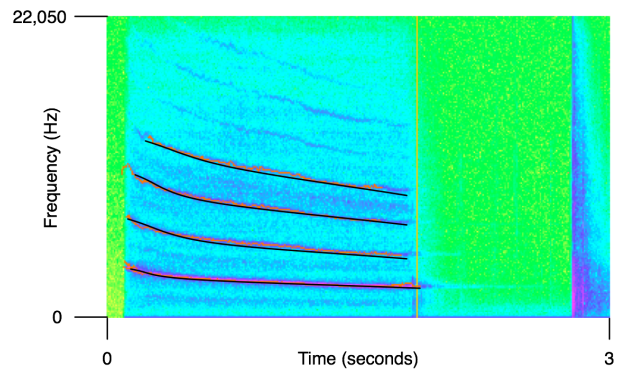


Figure 1. Spectrogram from the analysis face of TAPESTREA. The sound is a shooting firework with an explosion at the end. Four of the harmonic tracks associated with the firecracker’s whistling sound have been marked in bold.

depth manipulation of sinusoidal templates. VTAPS adds a user interface area (or “face”¹) that allows the user to interact with the building blocks of sinusoidal templates, tracks. In particular, VTAPS adds track-level manipulations relevant for voice processing. In this paper, we discuss the place of tracks in TAPESTREA, how they are interacted with by the user and internally by the software, and some applications of the new track selection and modification functionality.

2. RELATED WORK

TAPESTREA makes use of the Spectral Modeling Synthesis (SMS) paradigm [9], which in turn extends the original sinusoidal modeling algorithm [6] by introducing the concept of “sines plus noise”. SMS depends on the idea that some parts of a sound suit a sinusoidal model, while others are closer to spectrally shaped noise. It originally modeled musical instrument sounds, but TAPESTREA applies it to arbitrary sounds at the user’s discretion.

Related existing software includes the CLAM library [1], Lemur [3], SPEAR [4], and AudioSculpt [2], which allow high-level sinusoidal analysis, transformation, and re-synthesis. However, they lack the interactive control over all stages of processing offered by TAPESTREA.

¹ TAPESTREA’s interface is imagined to be spread across the faces of a cube, where each face is a different screen and caters to a different step of the editing process.

SPEAR [4] supports the selection of sinusoidal tracks (partials) from a graphical interface, and various track modifications such as frequency shifting and transposition and time translation. AudioSculpt [2] also allows analysis and editing of partials, but VTAPS has the advantage of being free and open-source, and thus easily extensible. Further, VTAPS is specially designed to modify sinusoidal templates within TAPESTREA. These modified templates can then be combined with stochastic background, transients, and other templates to create a rich piece of music or sound design. VTAPS also offers certain track-level manipulations not available in the other software, including some short-cuts geared towards voice processing. Another difference from the other tools is that TAPESTREA offers the ChucK [8] programming language for precise scripting, which is especially powerful on re-synthesis.

3. TRACK EDITING IN TAPESTREA

TAPESTREA uses a Fast Fourier Transform (FFT) algorithm over a sliding window to determine the strength of a wide range of frequencies, then graphs it for the user in the form of a spectrogram on the analysis face.

A human looking at a graph like that in Figure 1 can see that there are several discrete sounds occurring at the same time on the left; the roughly horizontal, evenly-spaced “lines” are the harmonic tones of the whistling of a firecracker. A human can think of these as discrete tones, and so can TAPESTREA by representing them as tracks. The tracks are extracted according to parameters entered on the analysis face, the most important of which is the number of tracks to detect. At each of the stopping points of the sliding FFT window, that number of frequency peaks with the highest magnitude are stored. They are then matched with similarly detected peaks from adjacent frames. There are rules for determining whether a peak is the start of a track, when a track may die, and so on [7]. The end result (see Figure 2) is that the tracks that have been created to represent the sinusoidal audio can be used to approximately recreate the original sound without the other noise. More importantly to this paper, tracks are a convenient representation for the user to interact with in TAPESTREA, as will be shown.

3.1. Track Internal Representation

Tracks do not stand on their own in TAPESTREA. TRACK objects are contained within objects of the DETERMINISTIC class, which is a sub-class of TEMPLATE².

Templates have a few parameters like gain, pan, and time stretch which are mapped to controls on the synthesis face. This information can be changed without modifying the original underlying representation of the audio in the template. In the case of sinusoidal audio in the DETERMINISTIC sub-class, the underlying representation is a set of tracks.

² DETERMINISTIC and TEMPLATE are actually C++ struct types, but they can be thought of as classes for the purposes of this paper.

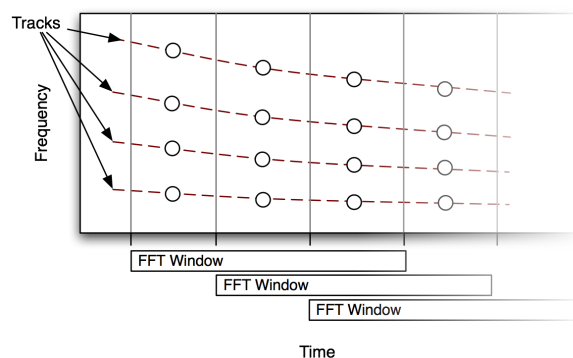


Figure 2. Adjacent frequency peaks are connected to create tracks.

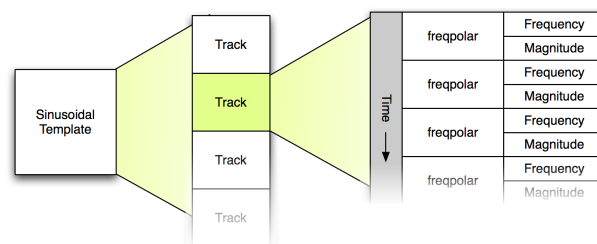


Figure 3. A sinusoidal template is a container for one or more tracks. The tracks each contain an array of ‘freqpolar’ objects with frequency and magnitude values.

Tracks have a small bit of data about their start time, end time, and phase, but most importantly they have an array of FREQPOLAR objects. The FREQPOLAR objects are used for storing the frequency, the time at which it was detected, and the magnitude of the frequency peak. See Figure 3.

3.2. Tracks to the User – The Group Face

The added functionality in VTAPS for track-based editing is on the “group” face, a screenshot of which is in Figure 4. The group face consists of a display similar to a spectrogram, a library of available templates, buttons to control playback, a button to add vibrato, a button to invert the track selection, a checkbox to control whether harmonic tracks are selected when a track is clicked, and buttons for other added features.

3.2.1. The Group Face Spectrogram

The analysis face has a spectrogram that reflects the strength of all frequencies in the scanning range in the color that is drawn. The spectrogram on the group face does the same thing for only the frequencies and times that are present in the currently displayed tracks. The detected peaks at each time are colored according to their strength, but other frequencies are not colored at all.

This spectrogram is also different in that it allows the individual tracks to be selected or deselected with the mouse

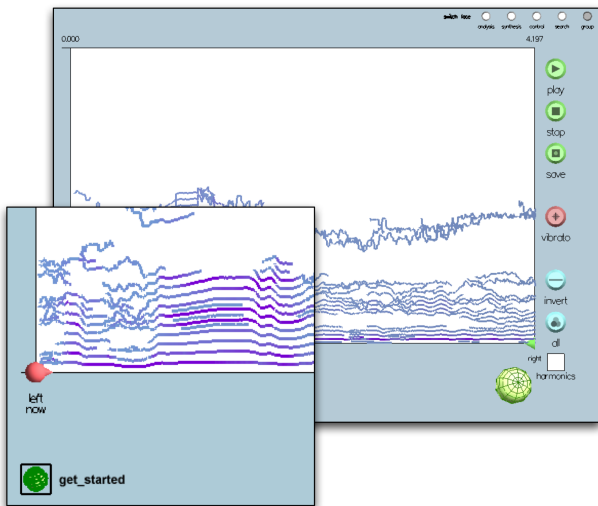


Figure 4. Screenshot of the “group” face. Colors have been adjusted for print.

by clicking. Only tracks that have been selected will be played back (the others are lightly colored), so “what-if” experiments can be quickly performed by clicking tracks and templates.

There are also buttons for inverting the current selection (deselecting selected tracks and vice-versa) and selecting all tracks.

3.2.2. The Template Library

Unlike in other faces, the template library on the group face only shows sinusoidal templates, because those are the only templates with tracks. Also unique to this face is the ability for multiple templates to be selected at one time. When a template is selected, its tracks are added to the spectrogram, and are removed when the template is deselected.

When the save button is pressed, currently selected tracks are saved to a new sinusoidal template regardless of which template they were originally a part of. The new template can be modified without disturbing the originals.

3.2.3. Vibrato

The “vibrato” button adds a vibrato to selected tracks. The vibrato effect is added by iterating over each of the selected track’s `FREQPOLAR` objects and multiplying the frequency by $1 + \alpha \sin(2\pi f_v t) + \beta * r$, where α is the amount of periodic vibrato, f_v is the vibrato frequency, β is the amount of random vibrato, and r is a random sequence whose parameters can be set by the user. The vibrato of all tracks will be synchronized.

3.2.4. Harmonic Selection

The checkbox labeled “harmonics” enhances track selections by also applying the selection to tracks that appear to be harmonics (multiples) of the one that was clicked.

Clicking a deselected track will select that track as well as any tracks that, at the time along the axis at which you clicked, also have a frequency within a small percentage of an integer multiple of the original track’s frequency.

By only considering each track’s frequency at that time, the user is given the ability to select tracks which are harmonic at one point even if they are not harmonic at others. This is convenient because of errors in detecting tracks that cause non-pitched sounds to be merged into tracks in the form of random-seeming modulation. If the whole track were considered, this noise would cause some harmonic tracks to be left undetected.

4. APPLICATIONS AND EXAMPLES

What follows are in-depth descriptions of a few of the techniques enabled by the addition of VTAPS.

4.1. Arbitrary Filtering

One obvious application of the group face is the filtering of high or low frequencies of a piece of audio. This is easily done by removing the tracks above (low-pass) or below (high-pass) a certain frequency. This is more stringent than actual high- and low-pass filters, that only reduce the amplitude of filtered frequencies, not always eliminating them. The effect of a traditional filter can be emulated by reducing the gain on the high- and low-frequency tracks. Since track synthesis in TAPESTREA is accomplished by additive sinusoids, any filtering, no matter how extreme, is guaranteed not to cause aliasing components.

4.2. Clean Extraction

The method of extracting tracks from audio by block selection on the spectrogram has a problem dealing with unwanted tracks. These are tracks which intersect the smallest rectangle one can draw to encompass the tracks one actually wants. Decreasing the number of tracks to detect solves this problem for low-power unwanted tracks, but high-power tracks that are more prominent than the wanted tracks will be extracted in their stead.

The group face is useful for solving this problem. One can set the number of tracks to detect (the number of simultaneous tracks allowed at any point in time) to the number of wanted tracks plus some extra tracks. The unwanted tracks can then be removed by clicking them, leaving only the wanted tracks.

4.3. Time, Amplitude, and Frequency Manipulations

The ability to recombine tracks gives way naturally to the desire to adjust them to mix together better. Shifting individual tracks in time allows one to rearrange a template and to flexibly combine selected tracks from different parts of different templates. Similarly, it is useful to adjust the amplitude of tracks of some tracks to emphasize certain frequencies or make others stand out less. A track

or set of tracks can also be individually time-stretched or shifted in frequency.

Other frequency manipulations on individual tracks include quantizing the frequencies to a fixed scale for interesting vocal effects. Another possibility to explore is the smoothing of a track's frequency trajectory by matching it to a spline or processing it with a low-pass filter.

4.4. Track Slicing

The track extraction algorithms are at most as perfect as the user, so there are times when a track jumps from one perceived frequency peak to another that is inconvenient or unnatural for the user to work with – for example, a track that jumps from one harmonic to the next is inconvenient, but happens when they are spaced closely. Hence, it is helpful for a user to be able to split this track into two by selecting the time at which they are perceptually discontinuous.

4.5. Working with Harmonics

VTAPS allows the selection of harmonics of a given track. These harmonics can then be treated as a group for special processing. For instance, transformations can be applied to the voiced parts (usually with harmonics) of a template, leaving relatively noisy consonants intact.

The pitch of harmonic sounds can be shifted up by one octave simply by deleting the odd harmonic frequencies. While all harmonics are selected by checking the “harmonics” checkbox and clicking on the first harmonic track, clicking the second harmonic track selects all of the even harmonics, bumping the pitch up by one octave. The odd harmonics can be selected by pressing the “invert” button after the even harmonics have been selected.

The ability to select even and odd harmonics allows the user to apply different effects to different harmonic groups within a template. Different vibrato patterns can be applied to different groups, recreating classic examples, and creating new striking examples of auditory stream segregation, fusion, and other psychoacoustic parlor tricks.

5. FUTURE WORK

VTAPS offers interactive track-level manipulation to modify templates in new ways, and has plenty of scope for future expansion—some of which is already underway. The ability to cut, paste, replicate and move selections of tracks together will be helpful. Another desirable feature is the option of adding sub-harmonics to a template. These would be a new set of harmonics based on a fundamental at an integer division of the existing fundamental, usually at a lower amplitude than the main harmonics. Adding sub-harmonics could give a sense of roughness to the voice, as is often heard in some Blues singers [5].

Another avenue to explore is the identification of formants from a set of tracks. When altering the frequency of vocal sounds, the vowels often change as well. Analyzing selected sets of tracks to locate the formants of the vowel

sounds could lead to intelligent frequency transformations that leave the vowels relatively unchanged.

6. CONCLUSIONS

VTAPS brings more freedom into the manipulation of sinusoidal templates in TAPESTREA. Previously, users could control the synthesis parameters of the entire template as a unit. Working with individual tracks allows users to edit tracks independently, remove or combine tracks from different templates, and thus modify templates on a more fundamental level.

The group face provides instant gratification in the sense that experimenting with the inclusion or removal of tracks of a certain frequency can now be done in seconds and with a few clicks. It benefits sound producers who wish to exercise greater control over what sound components go into the final piece, and audio enthusiasts who simply want to understand how audio signals work.

7. REFERENCES

- [1] Amatriain, X. and P. Arumi. “Developing Cross-platform Audio and Music Applications with the CLAM Framework”, *Proc. ICMC*, Barcelona, Spain, 2005.
- [2] Bogaards, N., A. Robel, and X. Rodet. “Sound Analysis and Processing with AudioSculpt 2”, *Proc. ICMC*, Miami, USA, 2004.
- [3] Fitz, K. and L. Haken. “Sinusoidal Modeling and Manipulation Using Lemur”, *Computer Music Journal* 20:4, 1996.
- [4] Klingbeil, M. “Software for Spectral Analysis, Editing, and Synthesis”, *Proc. ICMC*, Barcelona, Spain, 2005.
- [5] Loscos, A. and J. Bonada. “Emulating Rough and Growl Voice in Spectral Domain”, *Proc. 7th Int. Conference on Digital Audio Effects*, Naples, Italy, 2004.
- [6] McAulay, R. J. and T. F. Quatieri. “Speech Analysis / Synthesis Based on a Sinusoidal Representation”, *IEEE Transactions on Acoustics, Speech, and Signal Processing* 34:4, 1986.
- [7] Misra, A., P. R. Cook, and G. Wang. “Musical Tapestry: Re-Composing Natural Sounds”, *Proc. ICMC*, New Orleans, USA, 2006.
- [8] Wang, G. and P. R. Cook. “ChucK: A Concurrent, On-the-fly, Audio Programming Language”, *Proc. ICMC*, Singapore, 2003.
- [9] Xerra, S. *A System for Sound Analysis / Transformation / Synthesis based on a Deterministic plus Stochastic Decomposition*. Ph.D. thesis, Stanford University, 1989.