Dynamic Convolution Modeling, A hybrid synthesis strategy.

David Bessell

ICCMR
Plymouth University
Drake Circus
Plymouth
Devon
PL4 8AA
UK.
David.Bessell@Plymouth.ac.uk
https://sites.google.com/site/davebessellmusic/home

Abstract
An outline of a hybrid approach to the synthesis of percussion sounds. The synthesis method described here combines techniques and concepts from physical modeling and convolution to produce audio synthesis of percussive instruments. This synthesis method not only achieves a high degree of realism in comparison to audio samples but also retains some of the flexibility associated with waveguide physical models. When the results are analyzed, the method outlined exhibits some interesting detailed spectral features which have some aspects in common with the
behavior of acoustic percussion instruments. In addition to outlining the synthesis process there is some discussion of the more creative possibilities inherent in this approach. For example the use, and free combination of excitation and resonance sources from beyond the realms of the purely percussive examples given.

Introduction

Physical modeling offers the promise of many interesting and desirable technical and creative attributes to the contemporary composer. Conceptually it straddles the worlds of traditional acoustic and electronic composition seeming to offer possibilities to both. (To give just one example, the virtual creation of hybrid instruments made from disparate components derived from real world models but impossible to actually build in practice). It is perhaps surprising then that it has not become more widely used in contemporary musical works to date (2012). Might there be some barrier that is limiting the use of this technology by the wider body of composers and instrumentalists?

To fully explore the potential offered by physical modeling requires the acquisition of quite high level computing, mathematical, mechanical and electrical engineering skills. Allowing the composer to recreate conventional acoustic or new expressive instruments with specific timbral characteristics suited to an individual piece raises the question at what technical level must a musician engage to fully exploit the possibilities? It is desirable for a composer or player to have as much control over the model as possible to truly be able to fit the instrument to the musical task at hand. However the acquisition of traditional musical skills is no trivial task in itself, if we add to this the demands of computing, mathematics and
associated fields, there comes a point even for the most gifted where the time involved and learning curve become problematic. This has led to a noticeable division between technically orientated researchers and software developers on the one hand and practicing musicians and composers on the other. The practical outcome of this is that important advances in physical modeling research are generally not being utilized creatively in the majority of new music pieces.

Currently there are a variety of levels at which a player or composer could engage with physical modeling technologies. There are the pre built commercial instruments with fixed interface options such as Applied Acoustics String Studio (applied-acoustics 2011), the modular systems with more or less pre built building blocks exemplified by Native Instruments Reaktor (Native Instruments 2011), or Ircam’s Modalys (http://forumnet.ircam.fr/701.html?&L=1), graphic programming environments such as Max/msp (cycling74 2011), libraries for various text based software languages such as Cook and Scavone’s C++ synthesis toolkit (Cook, Scavone 2004), and finally work from first principles implemented as mathematical expressions in text based code such as that found in the work of Stefan Bilbao (Bilbao 2009). Each of these falls somewhere along a continuum between flexibility and ease of use. The approach proposed here, a variant on physical modeling and convolution synthesis methods, offers a further option to this list. The work focuses on percussive sounds in which an attack (source) part and a decay part (filter) are obtained from recorded sounds. The amplitude of the attack component is used to derive an envelope for a noise source.

A Hybrid software approach.
Dynamic Convolution Modeling is an approach to realistic sound synthesis combining techniques and paradigms derived from the fields of physical modeling (Välimäki, et al 2006), real time convolution (Stockham, 1966, Gardner 1995) and audio sampling (Rabiner and Gold 1975). Audio samples of the process in action were realized with an implementation in Max/msp. The Max/msp implementation can be downloaded here https://sites.google.com/site/davebessellmusic/software-maxmsp.

Software development in this project is ongoing and this Max/msp iteration is currently designed primarily to recreate realistic percussion instruments that can be played from a standard midi keyboard, although it already has some capabilities beyond simple replication of realistic percussion. The motivation for attempting to create a hybrid synthesis method was to try and combine the strengths of some current staples of computer audio manipulation in to a new expressive tool which could be easily managed by non expert users or more traditionally orientated composers while still maintaining sophisticated audio and performance flexibility. During research into a practical implementation of this concept it was discovered that, in comparison to the large and somewhat unwieldy sample resources typically used for realistic and expressive drum emulation in commercial libraries, comparable audio results could be achieved with a significant data reduction while retaining a quite modest demand on cpu. load. In comparison to a well known and popular drum sample library which uses 1.6MB of data compressed samples to recreate a velocity sensitive snare drum, the method proposed here uses just 4KB (uncompressed) for the snare strike sample and 152KB for the convolution impulse sample. This potential for data reduction in physical modeling based approaches has been noted by Rauhala. (Rauhala et al 2007)

**Limitations of existing techniques used in isolation.**
Although each of the three fields mentioned above have their own well established body of techniques, each also has its own strengths and limitations. For example, convolution as normally implemented has a tendency to produce rather static aural results, sampling likewise can only produce static snapshots of audio which require further manipulation to render them more expressive in a musical context. Physical modeling can be powerfully expressive, but at the expense of some fairly high order complexity both in control and implementation. As noted, this complexity has led to some resistance amongst non specialist musicians in its use, see for example the relative commercial failure of the introduction of Yamaha’s VL 1 physical modeling synthesizer based on research by Julius Smith which was launched in 1994. (Smith 2005, Stanford 2011) Dynamic Convolution Synthesis as it is presented here attempts to marry some of the conceptual and practical ease of use found in typical audio sampling software with some of the realism of convolution filtering and reverberation software, without entirely loosing the flexibility and expressiveness of physical modeling.

**Precedents**

A number of previous studies have elements in common with the methods outlined here. There are precedents for the use of physical modeling for drums and percussion such as the use of 2D waveguide percussion models by Van Duyne and Smith (Van Duyne and Smith 1993), 3D waveguides in the work of Laird (Laird 2001) or the ‘physically informed’ percussion synthesis approach of Perry Cook (Cook 1997). A number of approaches based on source filter decomposition have also been proposed (Laroche and Meillier 1994, Sandler 1990). Further work on physical modeled percussion encompasses both modal approaches and finite difference models, (Avanzini and Marogna 2010, Rabenstein et al. 2010, Bilbao 2012)
and there are precedents for the use of noise inputs in a physical modeling context such as (Karplus and Strong 1983). (Demoucron 2008, Demoucron and Rasamimanana 2009) have presented a violin system that uses physical modeling for the input combined with convolution. Early work on convolution as a synthesis tool has been done by Curtis Roads (Roads 1993). (Mackenzie et al. 1995) have investigated percussion synthesis using a variety of filters excited by a noise source. In this case the result of the uniform noise excitation is then amplitude scaled after the filter stage. Similarly (Karjalainen et al. 2002) have investigated the use of auto regression, moving average, linear prediction and frequency zooming in a similar context. (Macon et al. 1998) have implemented percussion synthesis using an all pole model which emphasizes efficiency.

**Component Parts**

The hybrid synthesis approach proposed here can be broken down into a number of component parts as outlined below.

**Physical Modeling**

The principle element that the proposed hybrid strategy takes from physical modeling is the concept of separating the excitation source from the rest of the process. In particular, techniques of noise excitation seen in waveguide model implementations were the starting point. (Smith 2006)

‘…the excitation wavetable signals obtained by inverse filtering (deconvolution) of the recorded sound by the SDL response. For practical synthesis, only the initial transient part of the inverse-filtered excitation is used typically covering several tens of milliseconds.’ (Välimäki et al 2006.)
However the particular form it takes in the current article is a little different to that commonly seen in waveguide models. In this case a noise source is amplitude modulated by an envelope derived directly from an initial short transient audio sample. (See the Excitation Sample, Excitation Envelope and Pink Noise sections of fig 1.) This amplitude modulated noise source is then used as the excitation for an impulse response derived from an audio sample of a resonant body. (fig 1. Drum Sample) In order to generate this excitation envelope the initial transient sample waveform is first rectified. This process is performed in order to create a less colored result when the amplitude scaled noise source is convolved with the drum resonance impulse. The excitation sample might typically be drum beater strike noise generated by the user, see the section on sampling below for further details. The architecture of the whole process can be seen in fig 1. This architecture draws generally on ideas found in the work of Välimäki and Smith but is implemented here in a way specific to this application, more on the implications of this later.

**Fig 1. Structure of the synthesis process.**
Convolution

Developments in partitioned convolution implementations (Gardner 1995, Stockham 1996) combined with increases in computing speed now mean that complex filter responses such as those associated with resonant objects can realistically be implemented as part of a real time process. In the Max/msp implementation utilized here, an external object written by Alex Harker (Harker and Tremblay 2012) is used. This implements in Max/msp a similar strategy for real time convolution as that outlined in Gardner’s 1995 paper. The Harker external takes advantage of macintosh specific code to give an efficient realization for this application. (Commercial PC alternatives such as SIR2 or Freeverb3 could be substituted in this context.)

An overview of the real time convolution process that the Harker external employs is as follows. The sample to be used as the impulse response is partitioned, the first partition is convolved in the time domain using a process similar to that used by the Max/msp object buffir~ (FIR filter). An FIR filter convolves an input signal with an impulse response and this can be implemented in a process known as direct convolution. Direct convolution performed in the time domain has no inherent latency but comes with the disadvantage of a high computational cost. This means that long convolutions such as the ones needed for the Dynamic Convolution Modeling process are not viable in real time using just FIR filters. However in this case only the short first partition uses direct convolution. The audio output of this first partition is therefore available with zero latency. Playback of this first partition audio allows time for the delay inherent in the frequency domain convolution of the
second partition to elapse and the second partition audio output can then be appended to that of the first partition. This in turn allows time for the longer delay associated with the larger FFT block size of the third partition and so on in a cascade of just in time calculations.

**Sampling**

The current software implementation takes as its starting point an audio sample of the drum or indeed any resonant body which is to be synthesized. This ‘reverberant body’ sample is used as the impulse response for the convolution stage. Then a second transient ‘strike’ sample is needed as a source for the amplitude envelope used to shape the noise excitation for the convolution stage. This second sample is created by sampling a damped strike on the resonant body used for the first sample. Alternative strike samples derived from impacts on other resonant bodies using a variety of beater materials and attack characters can also be substituted for a wider range of creative results more normally associated with physical modeling. As one of the primary aims was user friendliness, techniques such as deconvolution as a possible means of deriving strike samples from a single overall source sample were not used. Combining strike samples deconvolved from the same sample as that of the resonant body also has the potential to introduce unwanted phase related coloring in to the final audio result and has an associated increase in cpu usage. The two samples, resonant impulse and beater strike sample to be used as excitation envelope, are all that is required to create a sense of realistic dynamic behaviours. In an informal test a small sample of expert listeners (5) were consulted to confirm the subjective audio quality equivalence between the results of the DCM process and conventional commercial sample libraries. The reader can judge for themselves by listening to audio samples created with this process here. http://soundcloud.com/dave-bessell/drum-synthesis
The sum of the parts.

Features

Combining the elements outlined above allows the user control over the following aspects. Separate resonant body impulse and beater strike samples can be loaded. Audio output is triggered by midi note input and the user can specify the level of variability and liveliness at the spectral level. This control of variability in consecutive strikes can be achieved by manipulating various factors including the pitch of the strike sample playback and the balance between the shaped noise excitation and a more conventional single sample excitation impulse. Midi velocity response is implemented by scaling the amplitude of the strike envelope and the frequency cutoff of a one pole low pass filter on the output of this envelope.(fig. 1)

Creative flexibility

Interestingly this architecture retains some of the flexibility and creative possibilities of the type of physical models with which it shares some techniques. A variety of different beater samples can be used to ‘strike’ the drum impulse. This variety of strike excitations is by no means as flexible as the hybrid acoustic/synthesized system proposed in the work of Roberto Aimi (Aimi 2007), or the somewhat similar Korg Wavedrum commercial product. The system proposed here is primarily aimed at a user group more orientated towards sampling technology rather than the live percussion players of Aimi’s study and for this user group the flexibility is extended. The strike excitations can also include non realistic combinations such as the pizzicato gong hybrid in which a pizzicato sample is used as a noise envelope for a gong sample rather than a conventional percussion beater.
In this case the pizzicato excitation sample is created by recording a heavily damped pizzicato on a violin string. The MIDI control on the particular iteration of the software used to create this example is somewhat more elaborate that the examples presented so far. The reader can listen to the example here - http://soundcloud.com/dave-bessell/pizzgong

This kind of creative flexibility was one of the primary design aims for this approach. The general approach is similar to that outlined by Rauhala et al. although the method of excitation is somewhat different.

‘A digital filter model for the soundboard has been designed based on recorded bridge impulse responses of the harpsichord. The output of the string models is injected into the soundboard filter that imitates the reverberant nature of the soundbox…’ (Rauhala et al. 2007)

The excitation sample could be manipulated in ways that give comparable results to that associated with varying the hardness of the material that is used to excite the resonant body in waveguide models. This is not implemented in this particular example as it was found that more convincing results were obtained by just using a sample of a different beater material in the first place.

**Variability of spectral detail.**

As a consequence of the noise excitation method used, the audio output also exhibits some properties normally not associated with conventional sampling, for example each audio response although broadly similar is nevertheless unique in its micro detail at the spectral level. For an example compare the spectral detail of two
consecutive Gong strikes in fig. 2 and fig. 3 created with the dynamic convolution modeling software.

Fig 2. DCM Gong strike 1.

Fig 3. DCM Gong strike 2.
This low level spectral variability allows the possibility of the use of multiple simultaneous layers of audio without the obtrusive phase cancellation artifacts associated with static audio samples. Similarly multiple consecutive repeats of a sound are each unique, obviating the need for the ‘round robin’ sample strategies typically found in commercial sample libraries and playback engines. One desirable aspect of conventional sampling and convolution applications that is maintained however is the sense of convincing audio realism, to the extent that it is almost impossible in some contexts to distinguish the result from an actual recording of the original drum/resonant body. Particularly good results were obtained for this implementation with metallic percussion such as cymbals or gongs. Accompanying audio examples illustrate DCM synthesis of a variety of drums: snare, gong, cymbals, tom tom, orchestral bass drum and drum sticks.

http://soundcloud.com/dave-bessell/drum-synthesis
Velocity response

A further consequence of the combination of noise excitation with the velocity sensitivity implementation, (fig 1. MIDI Velocity, Low Pass Filter, Amplitude) which is used as the input for the convolution resonance section, is that to a great degree the static nature of conventional convolution is not apparent in the final result. Providing that the original drum sample used as the impulse response is sampled from a loud (fff) drum hit, a high degree of realism in the velocity response can be achieved. There are some small caveats to this aspect, a response that is ‘louder’ than the original resonance sample cannot be created and at extremely low volumes there is some small subjective deviation from the response of the actual drum in the real world. In most practical musical situations this is likely to be too small to be significant but could easily be rectified by velocity switching to a second set of samples for the low velocity extreme.

This strategy of placing the velocity sensitivity outside the convolution process is conceptually analogous to procedures followed in commuted waveguide models (Smith 2010) and combined with the noise excitation conveys a convincing sense of dynamic response in the otherwise static convolution output.

Notwithstanding some minor novel aspects in the implementation of the noise excitation, the individual elements mentioned above are, largely well established techniques. However the particular configuration presented in this work exhibits some surprisingly realistic, expressive and flexible audio attributes along with a significant reduction in data storage for results which can exceed the realism of a conventional static sample. Typical cpu load figures for this Max/msp implementation on a Macbook Pro (2011) for a single drum are in the range of 1.5 to
2%. In short, the whole is considerably more than the sum of the parts in terms of achievable audio performance and user friendliness for non specialist users.

**Outcomes and areas for further research.**

There is the potential for further research into extending this technique to wind and string instruments but already this implementation allows the free mixing of any resonant body response with any percussive excitation response. Crucially these resonant bodies and percussive excitations can easily be created by the user thus in theory allowing greater flexibility than modular physical modeling approaches that rely on a menu of pre created building blocks. The possible excitation sources include just about anything that can strike, pluck or scrape, and it is possible to imagine a more abstract approach beyond the conventional, such as excitation by water droplets. The user interface can be made conceptually easy to manage even for those without specialist knowledge while maintaining a high degree of audio realism when compared to conventional sampling. Some of the creative aspects of conventional physical modeling techniques such as the possibilities for creating expressive new hybrid instruments are maintained.


Audio, Speech and Language Processing, vol. 18, no. 4, pp. 891-902.


