Electronic instruments
Chris Chafe

chapter to appear in *The Oxford Handbook of Music Performance: Insights from Education, Psychology, Musicology, Science and Medicine*. G. McPherson (ed.) (2020) Oxford University Press

Abstract

Electronic musical instruments using electrical signals as a sound source have existed for over 150 years. The chapter on Electronic Instruments focuses on digital musical instruments and explores the designs of computer-based technologies in the most common present of electronic synthesizers. As these machines advance, so do the capabilities for expressive performance. Instruments may be devices with keyboards, microphones and / or other kinds of sensors attached or they may exist solely as software applications running on many possible kinds of computers. The chapter describes how performance gestures and sound are treated as numbers and are represented in formats such as the MIDI protocol and sound files. Musical sound is produced by samplers and synthesizers which are programmed for tone generation. The basics of spectrum analysis and synthesis methods are covered in order to understand what a particular tone generator algorithm provides as sonic dimensions to control and which dimensions of control might be available to the performer.

Introduction

> 'I shall consider an evolution aided by a specific class of machines.' The Architecture Machine, "Preface to a Preface." (Negroponte, 1969, p. 11)

Electronic instruments encompass everything from wired versions of traditional instruments such as electric guitars to fully digital systems for live performance. Many are hybrid combinations comprising mechanical, electronic and software components. This chapter focuses on digital music instruments (DMI's) and will be best appreciated with some familiarity having been gained by playing digital keyboards, drums or other DMI's. To a designer of these kinds of instruments, everything is a signal. As we step through the layers involved, I'll describe processes for sensing, transduction, signal processing, and sound generation. The discussion wraps up with some new technologies using mobile devices and web browsers

Gesture as numbers

The Musical Instrument Digital Interface (MIDI) protocol is a numerical format for encoding gestures as data. MIDI was proposed in 1982 as a means for connecting multiple devices and classes of hardware such as connecting controllers to tone generators. Its rapid adoption propelled the growth of DMI's and it continues to have an effect on DMI designs. MIDI is a long-lived standard as standards go and is well-documented (Huber, 2012).

Earlier computer music formats for representing music information used a keyboard-like paradigm which represented musical notes as isolated events (Mathews, 1963; Mathews & Pierce, 1987). Scores were in the form of time-ordered lists of notes. Each pitch or event to be played was bundled with loudness and other parameters including those describing timbral effects. While MIDI inherited some of these aspects, it is oriented toward real time performance and allows continuous gestures. A key press initiates a 'note-on' message on a particular key number which is transmitted as soon as possible and can be followed by 'continuous controller' messages for shaping an already sounding note. A 'note-off' message occurs when the key is let up. The polyphonic nature of keyboards creates thick streams of events during playing, each message signaling physical manipulations which are either key presses or various modifications to already sounding tones. Bundled as part of the note-on message is the velocity with which the key was pressed. A single MIDI stream can carry up to 16 sources distinguished by their MIDI channel number. Receiving devices listen on a particular channel and are generally either a tone generator, a MIDI recorder or some form of MIDI data processor (for example, an automatic arpeggiator). Incoming 'program change' messages instruct the receiver to switch its function, for example, altering a tone generator's instrumental timbre.

Besides keyboard activity, the format transmits actuation of buttons and switches, pedals, wheels and knobs. The console of a very large theater organ can be streamed over one MIDI connection and encompass, for example, all the manuals, shutters, stops, pedals, and effects such as klaxon buttons that are at the disposal of the organist. I composed music for a resurrected 1920's Wurlitzer Theater Organ which was performed by a computer connected to it by a single MIDI cable. Originally, it had been tethered to its console by an enormous bundle of electrical wires with each wire linking an individual pipe or sound effect. When the organ was retired, someone had taken an ax to the bundle. It was eventually restored at a new location and retrofitted using MIDI for communication from the console. Each one of the original signal wires was represented with a unique MIDI code. Choosing to rebuild it in this way made it possible to swap in a computer in place of the console and play the organ live from software.

At its core, MIDI carries the music by conveying gestures in time. Peering closely at gestures in numerical form data point by data point is a challenging way to see the expressive subtleties which we deem the essences of performance. These musical substances, so elusive when viewed as isolated events, are a goal of training and musicianship. They can even be difficult to describe with words. Keyboardist Jordan Rudess says it beautifully in writing about instrumental performance, "I think the goal for musical expression is to be connected mentally and physically with the sound of your instrument at every moment in time. We want the sensitivity of every touch (and every thought) to be recognized and have meaning. The human voice is thought of as the most 'connected' and expressive instrument to many, so examining it can clarify this idea. We use the power of our breath and vibrate our vocal cords at different frequency rates and control the tone through the natural resonating chambers of our throat, nose and mouth. Creating and controlling sound this way is the most organic musical experience for a human being as possible! Being in touch with the sound you are making with full continuous control of every moment of its vibratory existence is the goal of expression" (Rudess, 2017). It is that goal which has been driving the evolution of electronic instruments.

Digital music instrument design

On the input side, a multiplicity of types of transducers exist for rendering expressive gestures as digital signals. A performer manipulates switches, knobs and other electronic inputs and their instrument may be equipped with sensors to read acceleration or pressure of the fingers, hands, breath or lips. Some signals consist of instantaneous triggers (from switches) while others are continuous (from smoothly varying analog transducers). Signals of the latter type are converted to a numerical form with analog-to-digital conversion (ADC) which makes them available for signal processing by software. Hybrid instruments use analog audio transducers and ADC's to incorporate sound from their strings or other mechano-acoustical parts. Examples include digital guitars with pickups, vocal effects processors and microphones, drums with contact microphones. On the output side, the reverse happens. Digital-to-analog conversion (DAC) of the signal produced by the tone generator or effects processor changes it from its discrete, digital form into a continuous electrical signal which, when amplified, can drive output transducers like loudspeakers and headphones.

For purposes of illustration throughout the following I'll refer to a 'keyboard trumpet,' an example design with no counterpart from the world of earlier instruments but one that makes a perfectly plausible DMI. I first heard such an instrument played by Stanley Jungleib of Qwire, an ensemble from the mid-1990's in which I was the electric cellist. While the concept of joining keyboard technique with trumpet sounds might seem somewhat odd, bear in mind that what matters most is the musical expression afforded by the instrument and that what a performer wishes to evoke with it are genre and style specific. I would also contend that 'keyboard' and 'trumpet' are loaded concepts inherited from an earlier organology. Such design choices should not, however, be construed as limitations due to the technology. Computers are indeed a wide open medium and a part of the fascination in designing DMI's is the risk taking associated with innovation. The unforeseeable and sometimes profound effects such innovation may have on the music which gets produced is the other part of the fascination. The following treatment will be as style agnostic as possible.

Figure 1.1: Keyboard trumpet excerpt. Sampler used for tone generation. Play from a phone's bar code reader.

Sound as numbers

Computers use strings of numbers arranged sequentially through time to represent sound. Sound data can be acquired, transformed, stored and played back. It can be observed and analyzed either while happening (in real time) or later (out of real time), particularly if it is being edited after the fact. The data plot shown in Figure 1a is a trumpet passage recorded with a microphone in an anechoic room. The values correspond to 4 seconds worth of air pressure variation measured at a sampling rate of 44,100 points per second (a DMI generates audio samples at a similarly high data rate).

The accompanying graph in Figure 1b shows a zoomed-in portion of one note. It represents a very brief slice of time, two waveform periods long. The individual, exact values of the numerical samples can be seen. The trumpet note at this point has reached a 'steady state' between its attack and the next transition. Sound snapshots like this will be useful later in two ways: as a wavetable oscillator that can generate a steady tone with the harmonic structure of the original at a given point in time, and as data for spectral analysis often graphed as a sequence of such snapshots which show the evolution of the components of a tone.

The audio signal we see in Figure 1a has been digitally sampled at a rate fast enough to have full frequency bandwidth resolution. In other words, there are enough data points per second to represent the highest frequencies we can hear. And it has full dynamic range resolution because the range of numeric values of the data is sufficient to represent the softest and loudest sounds a trumpet might produce.

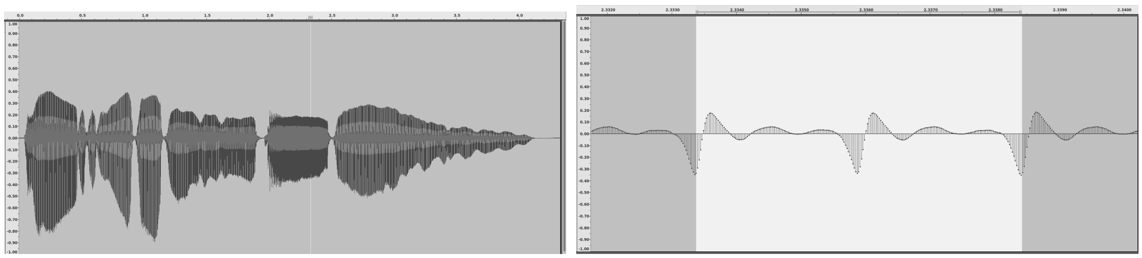

Figure 1.2: a) trumpet passage b) zoomed in on 2 periods.

Figure 1.3: Sound of Figure 1.2a. Play from a phone's bar code reader.

Samplers

A sampler is a DMI that plays back individual, isolated tones from a database of pre-recorded tone samples. The database resides in its inboard computer memory. For the most basic keyboard trumpet design, tone samples for each chromatic tone are collected and stored that can then be triggered by a key press corresponding to the desired pitch. In the initial collecting phase playing and recording should be as uniform as possible, if not the tones in the database will need to be matched for volume and length in post processing. The emitted sound will resemble the original trumpet tones within the limits of the choices made for the recording setup, for the audio data format and for output transduction in the listening environment. Fidelity can be quite good given a high-quality microphone placed close to the instrument, an ADC producing, for example, linear, 48 kHz, 16-bit audio samples and on the performance side, a high-quality PA system (public address is a common name for a stage loudspeaker system).

The most basic keyboard trumpet sampler provides only one dimension of performer control, namely choice of pitch.

Dimensions of performer control

The key number of the keyboard selects which stored tone sample to play out. Key numbers correspond to the MIDI convention of integer semi-tones (with 60 equal to 'middle-C'). A design with a restricted keyboard range having fewer octaves might include a transposition control to provide access to a wider range of pitches. Analogous to the 'Shift' keys on computer keyboards that give access to a greater number of characters, octave switches add a second dimension of performer control. Such an interface design now has two: pitch and octave transposition. The full set for our example design is shown in Table 1.

gesture	attribute	parameters	tone generator
key down	pitch	(multi-sample)	sample to play
octave switch			..
key up	duration		fade or loop
key velocity	dynamic		sample bank
...	portamento	range	pitch shift
pitchbend wheel	glissando	...	...
legato overlap		(mono mode)	continuous tone
after touch	vibrato	ranges	LFO
modulation wheel	tremolo	ranges	...

Table 1.1: Dimensions of performer control (keyboard sampler example).

Duration and dynamic level

Designing for greater possibilities of musical expression involves adding further dimensions of performer control. Duration is an essential one. With the simple design so far, there's only the key down gesture that triggers a tone of a fixed length. It ignores the subsequent key up (let off). Carillons are similar. Their bells are struck by levers (batons) that afford dimensions of pitch and dynamic. In their case, duration depends on the latter. Other instruments, like the piano, have dampers that provide an independent dimension of duration. The keyboard trumpet sampler's solution for duration is necessarily a bit of a compromise. If the key up happens before the end of the tone sample, the tone will be truncated by a volume fade out. If it's longer, it will be artificially lengthened by looping. Neither truly replicates trumpet sound. But sampling a comprehensive set of tones of different durations is obviously impractical, not to mention the fact that in a real-time sampler the correct tone sample would need to be selected before the duration is known.

The tone generator will engage either its release envelope or a looping wavetable as a function of when the key up happens. For the former, a tapering ramp reduces the volume to zero over a fixed length of time. For the latter effect, tone samples will have been examined ahead of time to select a steady-state wavetable comprising one waveform period (one cycle of the data shown in Figure 1b). If the note continues to be held past the loop point, the wavetable loops (oscillates) until the key is let off.

Musical dynamic is a dimension commonly tied to key press velocity. MIDI note-on messages encode velocity with a range of 127 possible levels. The actual implementation of the effect is a choice in the design of the tone generator. A simplistic volume effect could be implemented by mapping the velocity value directly to output gain (amplitude). However, the fact that in many instruments dynamic level affects not only loudness but also timbre calls for a more sophisticated solution. A keyboard sampler's tone generator can use velocity information to select from different tone banks recorded at different dynamic levels. A design might use three, Pianissimo, Mezzo Forte and Fortissimo, each bank with a full complement of pitches. Across these sets there are clearly noticeable timbral differences.

Bends, swells and modulations

Expressive dimensions can include skew and modulation effects affecting pitch and loudness: respectively, portamento and accent, glissando and dynamic swell, and vibrato and tremolo, all performed with some type of real-time gesture on the part of the performer. Similarly, there are many kinds of timbral effects which can be invoked which are specific to the type of instrument being designed. The manipulations that a performer makes are on a time scale which is slower than 'automatic' features belonging to the timbral quality of a note. For example, fast, automatic pitch skew that quickly reaches the target pitch can be a characteristic of an instrument and is something 'built-in' to its sound, outside the volition of the performer. But even these automatic features can become dimensions of performer control if they are brought out in the design. Suppose that accents or portamento in the attack are programmed to be linked to key velocity in a way which further enhances the impression of dynamic level. In our example with three tone banks, the range of 127 velocity levels can be partitioned in a way which leaves considerable room for such expansion.

Polyphonic keyboard instruments can be configured to have a 'mono mode' option that constrains them to play single lines only. For legato-capable instruments like trumpet, a mono mode implementation automatically detects note overlaps which would then translate on a sampler into smooth transitions between wavetable loops (rather than a re-attack). Automatic portamento could be inserted at the transitions, if desired.

The tone generator can take over the work required for modulation effects like vibrato and tremolo. A low-frequency oscillator (LFO) is assigned to pitch or dynamic level to create continuous fluctuations (low-frequency refers to typical vibrato rates, for example, up to 8 Hz). The performer doesn't need to rapidly vibrate a wheel or pressure sensor to directly modulate the tone. Instead, at a level one level higher, their gesture will control dimensions of the LFO, for example, its amplitude (excursion) and / or frequency. These have large expressive potential, especially when combined with other pitch, loudness and timbre manipulations.

Further methods to provide timbre variation in samplers include adding variable filters and variable mixtures of samples from different tone banks. Imagine a dynamic swell, possibly one mapped to increasing aftertouch pressure, and how it speaks more effectively if it gets brighter (spectrally richer) with more weight in the higher harmonics as it gets louder.

Spectrum analysis

A signal can be represented in both the time and frequency domains. In Figure 2a the same trumpet sound clip from Figure 1a has been converted to the frequency domain and displayed as a sonogram. Fourier analysis (computed with the Fast Fourier Transform – or FFT – algorithm) detects sinusoidal components within very short snapshots of the signal and measures their frequencies and strengths. By smoothly plotting sequences of these snapshots, we can build intuition about the time-varying spectral character of a sound and particularly its timbre and articulations. The trumpet's frequency components are largely harmonically- related to the pitch and can be seen to evolve over time.

What the analysis process 'sees' in a spectral portrait like this becomes data that can drive a resynthesis of the original. Figure 2b shows the components that have been produced from a synthesizer made of a bank of sinusoidal oscillators 'performed' by the analysis data. Carefully tracking the components of the original sound creates a replica that looks and sounds identical in its timbral details. However, the tuning of the analysis itself imposes a limitation to its fidelity. For example, the window size (duration in time) of the FFT snapshots and the rate (at which they occur) affects the resolution with which components can be distinguished. Different tunings track some components better in some cases and worse in others, and only those which are well-resolved can be resynthesized.

Risset's mid-60's Computer Study of Trumpet Tones was "...undertaken to determine the important physical correlates of trumpet timbre. Musical fragments played by a professional trumpet player have been recorded ... for subsequent spectral analysis. The computer analysis has yielded displays of the amplitude of each harmonic as a function of time. . . tones were synthesized that proved to be indistinguishable from the real tones by skilled musicians. By systematically altering the tones'

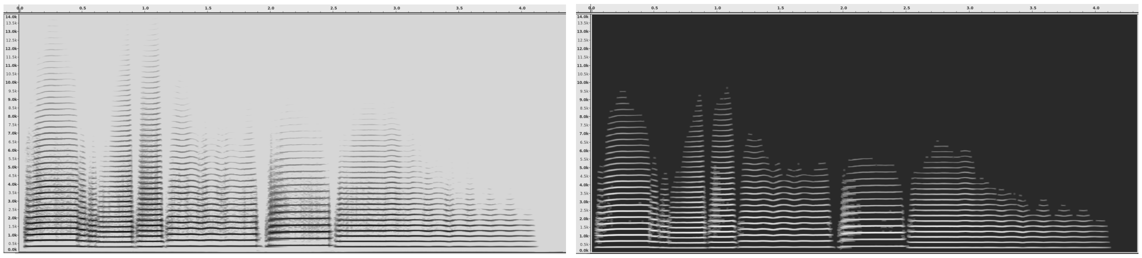

Figure 1.4: a) sonogram analysis of Fig. 1.2a. b) resynthesis from analysis, partial-by-partial.

Figure 1.5: Sound of Fig. 1.4b. Play from a phone's bar code reader.

parameters, it was found that a few physical features were of utmost aural importance: the attack time (which is shorter for the low-frequency harmonics than for the high-frequency ones), the fluctuation of the pitch frequency (which is of small amplitude, fast, and quasi-random), the harmonic content (which becomes richer in high- frequency harmonics when the loudness increases). From only these features, it was possible to synthesize brass-like sounds" (Risset, 1965). Further descriptions of early synthesis experiments to recreate trumpet and other tones are reported in (Risset & Mathews, 1969).

Resynthesis and modification in the frequency domain

Frequency domain representations of tones are made audible by converting them back to the time domain. A variety of DSP techniques that perform this conversion are used in spectrally- based DMI's. Not all spectral methods are driven directly from analyzed data, others are instead methods which are guided by observations of characteristics noted from analysis.

Overlap Add (OLA)

The overlap add (OLA) method uses the inverse FFT (IFFT) algorithm to reconstitute short segments of sound directly from successive spectral snapshots. The density of the snapshots determines the time resolution and if they overlap they can be layered (added) together as time progresses. FFT / IFFT processing of a signal can be high fidelity and extremely efficient. A signal can be reconstructed faithfully or manipulated by modifying the frequency domain data (altering component frequencies and their strengths). One major drawback, and something which affects the performer's experience with OLA, is latency. Because the snapshot segments must be long enough for the FFT to see at least two periods of a given frequency, these short segments of signal actually turn out to be relatively long. Real time responsiveness is affected. Waiting for processing which renders the low 'E' on a contrabass would delay the signal by almost 50 milliseconds.

Spectral Modeling Synthesis (SMS)

Spectral Modeling Synthesis (SMS) analyzes tones into a mixture of sinusoids and residual 'noise'. The so-called deterministic (aka periodic) partials of a sound are tracked component- by-component.

Tracking is done with extremely fine resolution in frequency and time. The deterministic model is then used to create an additive resynthesis as shown in Figure 2b, a very clean portrait of the original that includes all the periodic components accounted for by the tracking algorithm. This re-creation lacks non-periodic elements which would exist in sounds having breath or bow noise, piano hammer or guitar pick sounds. These noisier so- called stochastic components can be isolated from the original by simple subtraction of the resynthesized deterministic signal from the original signal in the time-domain, sample-by- sample. Once separated into deterministic and stochastic parts, interesting manipulations become possible. For example, the pitch of a vibraphone model can be transposed while preserving the details of the mallet impact, or a voice can be made breathier by increasing the stochastic proportion in the mix of the two parts.

Frequency Modulation (FM) and waveshapers

Looking again at the looping wavetables first mentioned above in the context of samplers, these constitute oscillators with waveforms which can have complexes of harmonics. Distorting the waveform of the oscillator changes its spectral components. Frequency Modulation (FM) is one technique for distorting an oscillator's waveform by disturbing it using a second oscillator. This can be used to vary the number of spectral components through time. Like other waveshapers, FM has its own unique set of dimensions available for performer control. The FM 'parameter space' specifies relationships between the oscillators in terms of their frequencies and amplitudes.

Any process applied to a waveform which creates new frequency components is non-linear by definition. Modulation-based waveshapers (like FM) create non-linear distortions using a separate signal (the modulator) to distort the first signal (the carrier). Amplitude modulation (AM) and ring modulation (RM) involve multiplication of the two signals, while FM and phase modulation (PM), use the output of the modulator to modify the (instantaneous) frequency of the carrier. All are mathematical operations carried out sample-by-sample on the input wave.

Non-linear distortion which doesn't involve a waveshaping signal are common in digital effects processors and can also be useful in synthesis applications. Clipping, in which the input wave (or oscillator) is clamped at a certain value or cubing the wave's samples (power- law distortion), are examples.

An FM trumpet algorithm is described in John Chowning's early introduction to FM synthesis: "Risset demonstrated in his revealing analysis of trumpet tones a fundamental characteristic of this class of timbres; the amount of energy in the spectrum is distributed over an increasing band in approximate proportion to the increase of intensity" (Chowning, 1973; p. 532). Chowning goes on to list aspects of timbre to be emulated in setting the FM parameters:

1. The frequencies in the spectrum are in the harmonic series
2. Both odd and even numbered harmonics are at some times present
3. The higher harmonics increase in significance with intensity
4. The rise-time of the amplitude is rapid for a typical attack and may 'overshoot' the 'steady state'

Without going into great detail, a keyboard DMI design based on Chowning's observations would be compact. Only two sinusoidal oscillators and an attack-decay-sustain-release (ADSR) envelope are required. Parameters to be set according to his observations are the oscillators' frequencies and amplitudes and the ADSR's time-varying contour: its attack duration, attack amplitude, sustain amplitude and release duration. The modulating oscillator's amplitude controls the amount of modulation effect produced (modulation index). Greater modulation produces a greater number of new frequency components and because the ADSR envelope is applied to modulation index as well as volume, it creates a coupled effect of becoming brighter while getting louder. This dependency is exhibited both within the 'automatic' micro-structure of the envelope's attack segment and as a feature of longer-term, volitional expressive changes in the hands of the performer, such as crescendo.

Vocoders

During the same period which produced the early frequency domain-based models by Risset, Chowning and others, synthesis of intelligible vocal sound was underway using a filter-based approach. Models of this type produce a time-varying spectral result using subtractive rather than additive means. Components representing the larynx and vocal tract use oscillators and resonant filters. To get a speech-like sound, a buzz-like, spectrally-rich wavetable mimicking the laryngeal source is fed through a complex set of filters representing the vocal tract. Recorded speech is analyzed for pitch and amplitude inflections and resonances are tracked to provide data which can drive the source and filter settings. In fact, any sort of sound signal can be substituted for the excitation source and have resonance filters applied to it to sculpt its spectrum. With other kinds of input (either synthesized or direct from an external instrument source) this creates the well-known vocal-like ('vocoder') digital effect.

Modal synthesis

Physically speaking, the periodic spectral components we observe in instruments arise from excitation of their modes of vibration. Imagine a struck metal plate and the partial tones that can be heard. Each mode is an audible resonance at a certain frequency with a certain strength and decay rate (damping factor). As with the vocal tract resonances, this modal structure can be reproduced using a bank of resonant filters. When a source signal excites the filters, they will ring until they completely decay either because of internal damping (which is a property of each mode) or due to external damping applied to the instrument by the performer (a dimension of control).

A subtractive model can be used for modal synthesis of instruments but this differs from vocal synthesis in two ways. For the voice, the broad resonances called formants are highly- damped and inharmonic. These ring so briefly when impulsed that it's hard to hear any true duration or hear out their frequencies. For example, impulse your cheek by snapping it with a finger and the 'pop' will seem related to mouth shape. Its modal complex (the vowel we hear for that shape) is comprised of multiple formant resonances. Now, try impulsing a uniform cylindrical tube by tapping on its end. Its pop will suggest a brief but definite pitch because the modes are harmonically related. Flautists and trumpeters can play percussion melodies using key clicks or tongue ram techniques to excite these resonances (whose pitch varies with tube length). Stretched strings are resonators in a different medium (mechanical rather than air) with significantly less damping. Tapping or plucking a string excites modes which ring considerably longer.

That the resonators used in modal synthesis can be independently controlled is particularly useful for producing the inharmonic, lightly-damped tones characteristic of ringing metal (bells, cymbals and the like). Associated dimensions of control include pitch or pitch complex, time-varying mode frequencies, mode damping and type of excitation. The latter is often implemented using a brief audio sample made by recording different mallets, hammers, picks and so on. An excitation sample with no residual sound of the resonator works best, one in which only the sound of the strike, pop or pluck itself should be heard. Using accelerometers as transducers is an effective means to obtain resonance-less recording, also physically damping the strings or membrane of the instrument. The SMS method mentioned above which isolates stochastic sounds can be applied to 'purify' the excitation source.

Modal synthesis is thus additive, subtractive and physical. Considered as a spectral method, it is an additive technique because of its superposition of modes as well as being a subtractive technique because filters are used for the resonances. Finally, it is a physical modeling method because modal structures can be directly derived from analysis of the physics of an instrument's resonant system.

Time domain physical models

Waves which propagate within an instrument's resonant medium (e.g., strings, air column,) can be simulated numerically using time domain physical models (McIntyre, et al., 1983). The driving force that initiates wave motion is, as with modal synthesis, accounted for by some form of excitation. The dimensions of performer control in time domain physical models are mechanical in nature and intuitive for those familiar with the instrument. However, if the instrument is one that takes time to learn to play with consistently good tone it will present similar difficulties of control in the model. This level of detail is both a blessing and a curse. Synthesized transients and timbral nuance can be very convincing, especially in how the result follows a player's gestures, but tone control can be a challenge.

A 'first principles' trumpet algorithm of this sort combines elements functioning as the lips and bore. The following description, specific to the trumpet, can be modified to model other instruments which are also capable of self-sustained mechanical oscillations. The lips part of the circuit represents the active, driving component that is powered by air pressure delivered from the lungs and manipulated by the cheeks and tongue. This initiates waves traveling in the trumpet bore. The length of the air column, manipulated by the valves, governs the resulting pitch and is directly related to the round-trip time of a recirculating wave in the bore. Simple delay lines that return their output back to their input are used for this part of the circuit. As the wave reflects up and down the bore, some energy is emitted at the bell end. There are other losses as well and these are modeled with filters (see further, Moorer, 1979). The entire model consists of a feedback loop in which the lips and air column interact. Each explosion of the lips emits a new pressure 'push' into the bore and the timing of when that happens is governed by the returning (delayed) wave pressure from the previous push.

A keyboard trumpet physical model can be designed in two ways: either with chromatic tube lengths (which is simpler and has a different length for each key but is not a true replica) or with seven lengths (for the sake of realism). The latter, valved version has the advantage of being able to render sonic effects resulting from the dynamics of overblowing. Doing so dictates inclusion of a pitch-to-valve table that maps from the desired pitch to one of the seven possible tube lengths. Trumpeters manipulate the tension of their lips to select pitches at overtones that are supported by the (harmonically-related) resonant modes of the tube. The given fundamental frequency of the overtone series will correspond to one of seven possible frequencies that are set by the seven possible valve combinations. Tensioning the lips as they 'buzz' changes their natural frequency and when that coincides with an overtone resonance of the bore a stable tone is produced. The most basic model of the lips simply uses another sharply resonant filter with its own characteristic frequency that selects the overtone. The resonance frequency of the lips filter is tuned to the pitch selected by the keyboard. Key velocity can be mapped to an ADSR envelope for mouth pressure.

The state of the physical model at a given instant is described by

1. mouth pressure (driven by the lungs);

2. bore pressure returning to the lips (from the wave propagating in the waveguide); and

gesture	attribute	high level	1D waveguide physical model	detailed model
key down	pitch	direct mapping	tube length	
		pitch-to-valve	...	valve positions
	...		lips resonance frequency	lip tension
modulation wheel			...	
				embouchure
key up	duration	ADSR	mouth pressure	lungs
key velocity	dynamic	...		
				cheeks
				tongue

Table 1.2: Dimensions of performer control (keyboard trumpet, physical models).

3. differential pressure (which is mouth pressure minus returning bore pressure).

Self-sustained DMI's driven by lips or bow, reed, or edge tone use circuit elements that create new frequency components in their interaction with the resonating element. The same kind of non-linear distortion employed by these components was mentioned above in regard to waveshaping using power-law functions. In the present example, squaring the lips filter's output provides the non-linearity.

The process representing the lips

1. filters the differential pressure with the lips resonance filter;

2. squares the filter's output; and

3. inserts the squared result back into the bore (to create the outgoing "push").

Manipulation of lips resonance frequency is, from a trumpeter's point-of-view, related to lip tension, embouchure and more. Their sense of the instrument's state and adjustments to it (a 'control feedback loop') involves regulating these parameters by the sound produced and mechanical feel. In comparison to a keyboard, the similarity ends there. A keyboard model lacks haptic sensations from the lips or the feel of the valves under the fingers. But increasingly an evolving technological shift to simulated 'feel' is part of our lives. Cars now substitute electronic controls for strictly mechanical ones. To clarify this with an analogy, let's call it 'apples for oranges'. What used to be the feel of the road through the mechanical steering system has been replaced with a steering wheel which is essentially a large dial. But, since we're used to resistances to help regulate steering effort, a better solution is 'simulated oranges for oranges' in which the vehicle simulates the resistances. Using a keyboard for pitch or knobs for lips tension presents a similar problem and suggests a similar solution: simulate the haptics of trumpet lips or valves somehow in the keys and knobs. Such active controls in DMI's are still at the experimental stage.

The synthesis model's parameters are shown in Table 2. The ones marked 1D waveguide physical model can be performed in two ways. The simplest is direct control via knobs and sliders tied to the physical model's parameters. But because this is a keyboard trumpet, not a 'Theremin trumpet', two higher-level functions intervene between keyboard and tone generator (these are the pitch-to-tube-length mapping table and mouth pressure ADSR envelope described above). Both direct and key-mapped control in tandem are possible without much additional complexity. A knob or wheel that runs up and down the overtone series during a held note adds an attractive dimension. A keyboard's modulation wheel can be mapped as an offset to the lips resonance frequency (which was initially set by the key note-on).

The physical model I've described is a one-dimensional waveguide, an approach to modeling physical resonators that can also be used to simulate wave motion in stretched strings, solid bars and air columns (Smith, 2004). Generally-speaking, any resonator that is long and narrow can be approximated with a recirculating delay line which has filters at its endpoints to mimic damping and other intrinsic properties of the medium. This so-called 'lumped circuit' approach can be extended to 2D for thin plates and membranes by constructing meshes of delay lines with filters distributed across the edges. Enhancements to the filters can introduce mild, passive non-linearities such as are found in gongs.

Whether the circuit element representing the lips, bow, reed or edge tone is at the end of a 1D delayline or plugged into the edge of a 2D mesh, the elements are modular and interchangeable. DMI's in this category are interestingly re-configurable and offer novel tone generation possibilities, for example, a gong excited by a flute mouthpiece.

Waveguide models such as the one described are based on approximations and can be improved with more detailed physical models of the parts of the instrument. The right-most column of Table 2 lists parameters for further simulation of trumpet physics. Valve positions (including half-valving), lip tension, embouchure, even tonguing can be modeled numerically. DMI's using finite difference time domain (FDTD) schemes with these levels of detail are now approaching real-time computation speeds. Direct modeling of instrument radiation characteristics with 3D techniques takes this even further. Experimental synthesis techniques are now able to simulate how instrument vibration is directly coupled to (and affected by) the surrounding air.

Practical implications

The Musical Instrument Digital Interface (MIDI)

The musical instrument digital interface, or more commonly referred to as MIDI is fundamental to representing the gestures of performance and in practice a general understanding of what is and what is not possible with it will benefit an electronic instrument performer. As described above, the protocol was engineered for signaling real time events and was largely keyboard inspired. The actions of keys, buttons, knobs and sliders are built into the protocol and there is flexibility which allows it to carry other kinds of signals. That flexibility has made some far-flung applications possible, for show control, stage lighting, fireworks and more.

We are extremely sensitive to slight deviations in musical timing. Practically speaking, MIDI has sufficient throughput to transmit very dense signaling of polyphonic musical streams, but two concerns arise from limitations imposed by the technology of the 1980's, back when it was formulated. Real-time computing and device communication then were different than today and MIDI is ripe for some upgrades. It is a 'best effort' protocol in which events are initiated and responded to as fast as possible but not necessarily as accurately as desired. Any timing jitter affects rhythmic accuracy and the degree to which jitter is apparent depends on several factors. Among them are the topology of connections (e.g., star configurations, daisy chains), and whether MIDI is being used externally for connecting musical devices one to another or is being used internally to route signals within a device's own hardware and software. The sequential nature of MIDI transmits events in succession that are an approximation of simultaneity. At the micro-time level, MIDI converts an attacca into a 'machine arpeggio'. Can this be heard? Do the notes of a chord commence perfectly together or is there some spread? The levels of jitter and arpeggiation encountered seem reasonable for most contexts otherwise MIDI's level of adoption would not be so wide. Masking these slight perturbations, possibly, are larger differences that are inherent in our own production and perception of event timings.

Fine shadings of pitch, loudness and other musical dimensions are manipulated by performers with astonishing resolution. MIDI puts these through a grating which quantizes the information at the limit of its numerical resolution. As with timing, the perceived result seems generally acceptable but today's level of technology can offer much better fidelity.

Under the new MIDI II specification, which at the time of writing this chapter, is in an early adoption phase, real-time events will now carry finely-resolved time stamps and resolution of data values is being improved. These enhancements will aid timing precision and provide higher-definition gesture representation. Moreover, 'intelligent' configuration exchange is being supported between compatible devices (instruments, controllers, software, tone generators). Devices will advertise their capabilities to each other and be able to exchange configurations. MIDI II is backwards compatible with the original MIDI I format.

Samplers, dimensions of performer control, controllers

Samplers are more than simple audio playback devices. Tone samples need to be carefully prepared for uniformity, to preserve fidelity and allow looping if required. Real-time processing is required to provide adjustable duration and to provide expressive dimensions of control. A constraint in designing a sampler is the amount of computer memory required (though the cost of memory is less of a factor over time). If the memory 'footprint' of the tone sample database needs to be reduced, real-time digital pitch shifting is used to thin out the number of stored samples by making one sample serve for more than one pitch (multi- sampling). Often, tones can be shifted up or down within a small compass with negligible audible artifacts. For example, shifting by a minor second allows one tone to be used for three pitches and cuts the required storage by 2/3. Reduction is also possible by using a lower audio sample rate or dynamic range if the resulting fidelity is deemed acceptable.

Adding idiomatic articulations becomes problematic using sampler technology. Consider the entire set of possible ways of starting or ending a note on a given instrument: such a large palette leads to an explosion in the size of the database. Trumpet players, for example, have independent control of lip tension, breath pressure, tonguing, valving, mute motions and even their pointing direction. Further effects would include shake and double- or triple-tonguing. Interactions between these dimensions give rise to a vast number of possible articulations of a given note or legato passage. When factoring in how these articulations also depend on tessitura and dynamic level the set becomes even larger.

Table 1 includes controllers often available in keyboard instruments. Two wheels are commonly provided that are positioned to the left of the keys: one for pitchbend (typically mapped to glissando) and one for modulation (often for vibrato). Keyboards can also offer 'aftertouch' control from a pressure sensor at the bottom of the keybed. Additional touch sensors include horizontal ribbon controllers spanning the instrument (length-wise in the same 'X' direction as the keys) or per-key finger position sensors on the keys themselves ('Y' direction). All such controllers are assignable. During a sustained note, for example, aftertouch pressure or sliding along a touch sensor can be mapped to pitch effects (glissando or vibrato), dynamic level or timbral manipulations. Whether these are preset assignments or are configurable is also a consideration in the design.

The full set (if there is such a thing) of idiomatic ways of shaping and connecting tones makes our example sampler design look quite modest. It is not only restricted in terms of the number of common techniques it can produce sonically but if played from a keyboard lacks enough extra appropriate physical controls in its interface. Were the sampler played not from a keyboard but from a trumpet-like controller, at least some of the mismatch between controller dimensions and the tone generator's dimensions of control could be reduced. In summary, questions at this point are, how far can a sampler go in generating the full-range of trumpet technique and to what extent can keyboard gestures stand in for trumpet gestures? MIDI allows for the full representation of dimensions of performer control and assignable 'continuous controller' codes are available that could represent, for example, time-varying lip tension, and breath pressure, which are each a distinct signal assigned its own continuous controller number. The Steiner EVI (1998) represents one solution which trumpeter and composer Mark Isham describes when adopting it as: "The EVI I use is the original Steiner electronic valve instrument with three buttons and a strange little canister that you twist to reproduce the overtones. It's like an electronic trumpet." (Isham, 1999)

Spectral methods, algorithmic tone generation

I've already mentioned how timbre depends on pitch and dynamic. These relationships can be implemented algorithmically in the design of tone generators using spectral methods. Specification of such dependencies in more flexible (but less precise) modeling approaches, like FM and subtractive synthesis, is easier to implement than in the more fine-grained methods, like OLA and SMS. These latter use tone sample analysis and then 'paint their portrait' in the frequency domain, partial-by-partial.

FM's parameter space provides a wider 'paintbrush' producing groups of spectral components. Dependencies between pitch, dynamic and timbre, which give a synthesis instrument its character are implemented at the level of carrier and modulator tunings. Voicing code, produced with these abstractions, can mimic features noted in the analyses by Risset and Chowning.

The trade-off today isn't between economical methods (such as FM) and expensive ones (like SMS) since DMI's are sufficiently powerful to render massive numbers of partials in real time. It is more about gaining dimensions of control that can be applied with the technique. The SMS trumpet example shown in Figure 2b uses upwards of 30 sinusoidal oscillators each with its own individual envelopes for frequency and amplitude. Spectral synthesizers easily compute 1000 oscillators today, which provides for massive polyphonic capabilities. And compared to samplers, such detailed techniques have the advantage of being able to manipulate duration, pitch and timbral dimensions independently of one another. However, these still require added logic to create the dynamic instrumental behavior (and identity) which is readily found in the less-detailed spectral techniques where models often employ interesting levels of abstraction and co-variation.

DMI design

John Pierce described the promise and challenge of software-based synthesis in 1965 (oddly, in Playboy Magazine), "As a musical instrument, the computer has unlimited potentialities for uttering sound. It can, in fact, produce strings of numbers representing any conceivable or hearable sound... Wonderful things would come out of that box if only we knew how to evoke them" (Pierce, 1965, p. 120).

A DMI designer determines the software in the processing layers which map a performer's manipulations onto the output sound. The range of possibilities is wide open (ala Pierce) and provides a compelling canvas for designers who want to innovate, but there are potential concerns. Perry Cook, a designer who's made large contributions to the evolving practice, wrote that, "Much of the design and research of new musical interfaces has focused on abstracting the 'controller' from the 'synthesizer' and then investigating how to best interface those two classes of hardware with each other and the player."(Cook, 2004, p. 315) And observes from experience that, "The major flaw in the controller / synthesizer paradigm is the loss of intimacy between human player and instrument" (Cook, 2004, p. 316). A performer will do well to consider Cook's concern and evaluate DMI's in view of Rudess' goal of "being in touch with the sound with full continuous control" (above). For a designer focused on either making simulations of existing instruments or entirely new sounds, Pierce's words mark the opening of a frontier with untold promise.

The trumpet example has served for comparison of a keyboard-based designs with tone generators employing sampling, additive synthesis, FM and various kinds of physical modeling. Developing appropriate physical controls for tone generator dimensions of control with concern for their tight integration is an area receiving much attention in new designs, both commercial and experimental. Also mentioned above is the question of latency from gesture to sound. What constitutes acceptable latency has been a topic since before MIDI and is to some degree, context dependent. A cathedral organist can experience lags in the 100's of milliseconds, while for other instruments lag is non-existent. In DMI designs, the sum of lags of gesture acquisition, processing and tone generation add up to what the performer experiences. An overall 'lag budget' I often hear from electronic music colleagues is to design for no more than 5 milliseconds overall.

Changing world of DMI's: mobile interfaces, web audio and virtual analog synthesis

New technologies continually emerge in the quest for expressive capabilities. Our present vantage point permits us to appreciate how strongly new DMI designs are influenced by preceding instruments and technologies.

Mobile interfaces

Tablets and mobile phones provide all the hardware necessary for DMI's and offer some compelling new possibilities. Jordan Rudess describes how the sophisticated built-in interfaces of these platforms can be leveraged for music, "Today on your mobile device, you can do things like play four simultaneous notes while adding vibrato to two of the notes while increasing the volume on one of them and bending the pitch of the fourth one. If you walk into a music store today, you will still be hard pressed to find any instrument in any price range that can do that. Luckily for all of us as musicians there are some people who are working on bridging this crazy gap and creating instruments that take advantage of the technology and offer musicians the next level of expression" (Rudess, 2017).

Web audio

In the early 90's, stand-alone hardware-based DMI's gave way to a proliferation of software- based instruments running on laptops. Today's shift to browser-based computing provides a new platform for DMI's. Web Audio technology runs real-time sound computation in the browser itself. The accompanying Web MIDI protocol permits attaching outboard interfaces, like keyboards, to the browser device (e.g., laptop, tablet). At the time of this writing, over 20 million instances of web audio are spawned by browser users daily though most are for web audio applications not related to instrumental music performance. Running a browser for, say a gaming app, is the same as running it for a DMI.

As an example, a spectral simulation of the Roland 808 Drum Machine 'lives' on a web page that is built using wavetables, waveshaping and filters. The page presents a replica of the original interface with identical buttons and knobs (similar projects can be found for other vintage analog synthesizers, for example, the Monotron and Juno).

Ultimately, web technology can support the full range of techniques for DMI's which have been discussed. Entirely novel instruments, replicas of pre-existing instruments across cultures and time, as well as foley sound effects and soundscape instruments are foreseeable web applications.

Virtual analog synthesis

Early analog electronic instruments comprised a mix of novel instruments and instrument emulation. Sampled replicas of many of these vintage analog synthesizers exist now as DMI's in software form. The number of samples needed from the original synthesizer for multi- sampling (to steer clear of pitch shift artifacts) can be quite low, even one sample per octave, so the database can be very compact.

Beloved analog devices have characteristics which can be more complex than even dense sampling can account for, however. And "circuit bending," the practice in which the original circuitry is creatively modified, adds to the catalog of possibilities. This has inspired physical modeling of analog instruments, known as virtual analog synthesis, as a means of capturing the sonic behavior of such devices. An example is the Roland TR-808 Rhythm Composer and its bass drum circuit. The instrument's unique sound has been crucial in recent music and its digital replica retains its identity. A seminal paper from 2014 presents a new mathematical approach to recreating "...complicated behavior that is impossible to capture through black-box modeling or structured sampling" (Werner, et al., 2014).

Further topics and readings

Several topics that are beyond the scope of this chapter are noted here (in no particular order). These are fundamental to traditional and emerging performance practices in which computers are involved:

- Score following with "smart" accompanist software
- Triggering, especially for percussion (by adapted acoustic instruments)
- Robotic instruments
- Laptop orchestras
- Networked ensembles, split across cities
- Venues for electronic performance and issues around PA systems
- Surround speaker arrays (ambisonics)
- Good micing of acoustic and hybrid instruments
- Artificial reverb
- "Wet rooms" and problems of feedback
- The role of front of house sound technicians for live mixing
- Issues of time for sound check, specifying contract riders, ensuring equipment redundancy in case of failure
- Famous historical instruments e.g., Wurlitzer, Theremin, Moog, Synclavier, Fairlight, DX-7
- Sensors of all kinds (e.g., IR, camera, shaft encoder, biomedical sensors such as GSR and BCI, and environmental registering, for example, light or gas levels)
- Tethered and untethered instruments (power, audio, network)
- Instrument software, is it programmable, upgradable?

- Physical life expectancy of instruments (their capacitors, rotating disk, EPROM)
- Museums with significant Electronic Instrument collections: La Villette (Paris), Yamaha (Hamamatsu), MoPOP (Seattle), National Music Centre keyboard collection (Calgary)
- Currently active DMI design companies e.g., Roli, Dave Smith, Linn, Native Instruments
- Open Sound Control (OSC) is an alternative protocol which can take the place of MIDI for gesture representation

These items form only a partial list. Among them are avenues which promise to take the technology and practice into ever more expressive realms in the future. A case in point is the last item, OSC, which is used for real-time signaling between components. Its importance in present-day performance is significant and its role in creating more 'intimate' DMI's has been a topic for over two decades (Wessel & Wright, 2002). As can be appreciated from the preceding discussion of DMI designs and the topics included in the list, this is a growing field. It has its practical aspects in the present, whose understanding can benefit musicians in performance. New features and capabilities are continually arriving on the horizon and this chapter has been written with the hope of sharing some of the excitement. I close by reprising the epigraph regarding machines from the chapter's beginning.

> 'I shall consider an evolution aided by a specific class of machines.'

Whose author also wrote, '...we will live in them". (Negroponte, 1975, p. 5)

Reflective questions

1. Contrast the performance gestures of a xylophonist and a clarinetist. What is in common and what is unique to each? List possible MIDI messages that could be used for the common and unique categories.
2. You're about to play a synthesizer of some kind in an ensemble. Provide a check-list of what you'll need at the venue and what extra time (if any) you should expect to take (beyond the ordinary).
3. A PC, tablet of mobile device is probably near at hand while you're reading this. Describe a DMI that's available on it and any outboard equipment options that might enhance using it in an ensemble or stage context.
4. Some uses of the instruments (machines) which have been described in the chapter are simply to replicate the functions of existing musical instruments. Pick one of those that is familiar and imagine an entirely new function for the machine to provide that is not possible on the original instrument.
5. The machines described have sensors attached. Imagine a new sensor, or perhaps one from outside of music, and describe something about the music it might be used to create.

References

Chowning, J. M. (1973). The synthesis of complex audio spectra by means of frequency modulation. *Journal of the Audio Engineering Society*, 21(7), 526–534.

Cook, P. R. (2004). Remutualizing the musical instrument: Co-design of synthesis algorithms and controllers. *Journal of New Music Research*, 33(3), 315-320. doi.org/10.1080/0929821042000317877

Huber, D. (2012). *The MIDI manual*. Abingdon: Routledge.

Mathews, M. V. (1963). The digital computer as a musical instrument. Science, 142(3592), 553-557. doi.org/10.1126/science.142.3592.553.

Mathews, M. V., & Pierce, J. R. (1987). The computer as a musical instrument. *Scientific American*, 256(2), 126–133.

McIntyre, M. E., Schumacher, R. T., & Woodhouse, J. (1983). On the oscillations of musical instruments. *The Journal of the Acoustical Society of America*, 74(5), 1325-1345. doi.org/10.1121/1.390157.

Moorer, J. (1985). About this reverberation business. In C. Roads, J. Strawn (Eds.), *Foundations of computer music* (pp. 605-639). Cambridge: MIT Press.

Negroponte, N. (1969). *The architecture machine*. Cambridge: MIT Press.

Negroponte, N. (1975). *Soft architecture machines*. Cambridge: MIT Press.

Risset, J.-C. (1965). Computer study of trumpet tones. *Acoustical Society of America*, 38, 912. https://doi.org/10.1121/1.1939648

Risset, J.-C., & Mathews, M. (1969). Analysis of musical instrument tones. *Physics Today*, 22(2), 23-80. doi.org/10.1063/1.3035399.

Rudess J. (2017) Jordan Rudess on alternative controllers. *Keyboard Magazine*, June (2015), 26.

Smith, J. O. (2004). Virtual acoustic musical instruments: Review and update. *Journal of New Music Research*, 33(3), 283-304. doi.org/10.1080/0929821042000317859.

Werner, K., Abel, J. & Smith, J. O. (2014). A physically-informed, circuit-bendable, digital model of the Roland tr-808 bass drum circuit. *Proceedings of the 17th International Conference on Digital Audio Effects* (pp. 159-166). DAFx. Erlangen, Germany.

Wessel, D., & Wright, M. (2002). Problems and prospects for intimate musical control of computers. *Computer Music Journal*, 26(3), 11–22. doi.org/10.1162/014892602320582945