# DEFORMING THE OSCILLATOR: ITERATIVE PHASES OVER PARAMETRIZABLE CLOSED PATHS

*Georg Essl* *

Department of Mathematical Sciences
University of Wisconsin – Milwaukee
Milwaukee, U.S.A.
essl@uwm.edu

## ABSTRACT

Iterative phase formulations allow for the generalization of many oscillatory sound synthesis methods from circles to general parametrizable loops, with or without explicit geometric contexts. This paper describes this approach, leading to the ability to perform modulation, feedback and chaotic oscillations over deformed circles that can include ill-behaved geometries, while allowing modulations or feedback to be deformed as well.

## 1. INTRODUCTION

One of the intuitions behind topological formulations is their generality and flexibility. A topological circle captures a closed path, even if the specifics of the path in some substrate space might vary drastically. Everyday intuition already captures topological ideas. For example, we understand that closed loop racing tracks have different shapes, thus providing differing challenges to the driver, but they all allow for the definition of laps, which count the number of times a driver has completed the closed path in one direction. We understand that a less than round tire will still be able to rotate, even though driving on a flat surface may be uneven. Topology captures these intuitions abstractly, and allows for us to formally use them constructively.

In this paper we propose a way to generalize oscillations over the standard metric circle in the plane to oscillations over any parametrizable topological loop. This in turn allows us to use parametric formulations of oscillatory synthesis methods such as iterative phase functions [1] to generate modulation and feedback type oscillatory synthesis methods using these more general loop spaces and their projections. This method, in its full generality does not require a geometric representation, by which we mean that it does not have to be embedded, immersed, or otherwise continuously mapped into some substrate geometric space. However, thinking of the approach geometrically helps strengthen intuition. In particular, in a geometric setting one can think of the reconfigurations of topological loops under varied parametrizations and varied maps into a substrate geometric space as *deformations*. In this sense one can think of race tracks or flat tires as deformed circles. More abstractly, this work provides a sound general setting in which existing modulation and feedback techniques can be applied, and new ones can be formulated.

* This work was made possible in part by a fellowship of the John Simon Guggenheim Foundation.



## 2. RELATED WORK

Oscillators are essential objects in sound synthesis and hence are widely taught [2]. Historically, oscillators are often defined by algorithms of certain functional forms, such as time-indexed computations, though it can be convenient to take alternative formulations. For example, in the context of digital signal processing the description of behavior on the circle via complex numbers called *phasors* can simplify description and elucidate interesting behavior of oscillatory phenomena [3]. Formulating oscillators as iterations of the phase on the circle [1] provides a further description of oscillations with additional insights. In particular this formulation makes clear the relationship of topology of the space, dynamics, and output projections. It allows us to bring a large class of oscillators under one description, including non-linear and chaotic oscillators. This particular formulation of oscillatory synthesis methods provide the foundation of the current paper.

Topological methods have been variably employed in sound synthesis and signal processing. Recently, sheaf-theoretic methods were proposed to attach linear time-invariant filters [4, 5] and non-linear oscillators to a topological space defined by a line-like simplicial complex. This allowed the method to employ paths over a standard torus as well as a parametrization of frequency modulation (FM) to be used as the underlying closed path geometry [6]. The current method can be described without the use of sheaf constructions, and leans closer to existing intuitions of oscillators.

Geometric oscillatory descriptions have long been of interest in sound synthesis. Putnam's thesis proposes *harmonic patterns* as one possible description of path-like geometric shapes in conjunction with sound synthesis [7, 8, 9]. Furthermore, his dissertation provides an expansive review of path parametrized methods in geometric sound synthesis [7]. Given that the proposed method works for any parametrized closed path, this body of work provides many examples and application possibilities. We will employ splines which have already been proposed for smooth transformations of additive synthesis [10], waveform generation and interpolation [11] as well as for control parameter transitions [12].

Sound synthesis based on comb filters also have an inherent loop character. In this context, Trautmann proposed synthesis based on analogous constructions to the Möbius strip [13]. Closed path trajectories form an important aspect of understanding waves on path trajectories for physical models based on waveguides [14].

More broadly, one can view the current work in the context of developments of computational and applied topology [15, 16], a field that seeks to develop computational topological techniques for applied problems. This body of work tends to focus on the analysis of data [17] and methods for analysis of time-series have been developed [18, 19].

## 3. FROM SINUSOIDAL OSCILLATIONS TO PARAMETRIZED TOPOLOGICAL LOOP

The conventional understanding of a circle is that of a closed curve in the Euclidean plane $\mathbb{R}^2$ with coordinates $(x, y)$. For our purposes, the most convenient formulation is given by familiar complex exponential oscillator under time parametrization $t$:

$$(x, iy) = e^{2\pi it} \qquad t \in [0, 1) \tag{1}$$

Circles play an important role in many parts of digital audio processing and sound synthesis as they capture harmonic oscillation under projection. Taking an orthogonal projection with a given angle $\phi$ one gets the undamped harmonic oscillator $y = \sin(2\pi t + \phi)$. Furthermore, there are perceptual reasons to privilege circular oscillations as our hearing seems to be performing a kind of spectral analysis, of which sinusoids are elementary building blocks. Perhaps relatedly, sinusoids are the basis functions of Fourier analysis. For all of these reasons, it is no surprise that a large body of work has developed in which sinusoidal oscillators are used, or have served as starting points for developing enriched oscillatory phenomena which are not purely sinusoidal, such as modulation techniques or wave shaping.

### 3.1. Discrete Dynamics and Circle Maps

Under the time-parametrization $t$ one can interpret the oscillator as a dynamical process. Dynamics refers to a time-dependent change or evolution. If the dynamics consists of discrete steps, then it is a discrete dynamical process. Furthermore, one can think of the dynamics to be performed on a given domain. In this paper we will use only discrete dynamical descriptions and we will use the index $n$ to identify a notion of discrete time, phase, or position of this discrete dynamics. The circle $\mathbb{S}^1$ is one such possible domain and its unit parametrization $[0, 1)$ with identification of 1 and 0 allows one to define a position $x_n \in [0, 1)$ on the parametrization. Then discrete iterations on the circle can be written in the following form [1]:

$$\underbrace{y_n}_{\text{Time Series}} = \underbrace{p}_{\text{Projection}} \underbrace{(x_n = f(x_{n-1}) \overbrace{\mod 1}^{\text{Circle Topology}})}_{\text{Iterative Phase Function}} \tag{2}$$

The current position on the parametrization $x_n$ is computed from the previous position $x_{n-1}$ via a given map $f$. The discrete-time output of the oscillator $y_n$ is the result of a projection function $p$, which converts the position in the parametrization into sample amplitudes.

This formulation forms the basis for connecting the sound synthesis literature with the existing literature in discrete dynamical systems. Maps from one location on the circle to another are usually called *circle maps* [20, 21] in the dynamical systems literature [22]. More interestingly, this connection led to the discovery [1] that the sine circle map as studied in the 1960s by Arnold [23] and onward is nothing but feedback frequency modulation [24]. This means that a wealth of results in dynamical systems for discrete dynamics on the circle apply to sound synthesis methods and characterizations such as Arnold tongues [23] apply and can be expanded for auditory use [1].

There are additional benefits to using this particular formulation of discrete oscillators. Projection $p$ and iterative phase functions $f$ are independent of each other. The iterative phase function $f$ can be fruitfully be interpreted as merely living on some cyclic parametrication in the interval of $[0, 1)$ which one can also think of as the quotient $\mathbb{R}/\mathbb{Z}$, which is the integer repetition of the interval in $\mathbb{R}$ and this in turn has the connectivity of a topological circle $\mathbb{S}^1$. Hence the iterative phase function really lives on a topological space. A geometric interpretation in some ambient space is suggested by the projection function $p$. The flexibility of some map into a substrate geometric space and a suitable projection map $p$ is of central interest of this paper. Given that these choices leave the topology of the loop parametrization and discrete iterative maps on them undisturbed, these two aspects can be treated independently. For this reason, we will emphasize maps with nonlinear chaotic feedback in examples in this paper as they serve to illustrate that indeed the construction is robust under all the variations of the given geometric and projection constructions. At the same time, we will also discuss sinusoidal oscillation and its underlying linear dynamics to provide simple base cases for comparisons. These examples constitute an illustrative subset, and, in fact, any iterative phase function, or cascades thereof, can be used and have the same structural stability and independence as the examples given.

### 3.2. Sinusoidal Oscillation (or Bare Circle Map)

A *discrete sinusoidal oscillator* can be written in this form with the following choices of iterative phase function $f$ and projection $p$ [1]:

$$x_n = x_{n-1} + \Omega \mod 1 \tag{3}$$
$$y_n = \sin(2\pi x_n) \tag{4}$$

The constant $\Omega$ is a discrete step taken and can be related to the frequency $\omega$ of an complex exponential oscillator of equation (1) $\Omega = \omega/\omega_s$ with a given sampling frequency $\omega_s$.

A conventional interpretation of this dynamical process is that of a discrete sinusoidal oscillator as a discrete dynamical process that moves around a Euclidean circle with constant steps $\Omega$. In the dynamical systems literature this equation is also sometimes known as the *bare circle map* [22] and it will be helpful for our purposes to refer to the discrete sinusoidal oscillator requiring the particular projection equation 4 while we think of the bare circle map not requiring this particular projection.

### 3.3. Feedback FM (or Sine Circle Map)

The iterative formulation of *feedback frequency modulation* (*feedback FM*) or equivalently the *sine circle map* under a sinusoidal projection is defined as follows [24, 20, 1]:

$$x_n = x_{n-1} + \Omega + \frac{k}{2\pi}\sin(2\pi x_{n-1}) \mod 1 \tag{5}$$
$$y_n = \sin(2\pi x_n) \tag{6}$$

We observe, that this equation is a perturbation of the sinusoidal oscillator (3). The strength of the perturbation is given by the constant $k$ ($k$ is normalized such that $k = 1$ corresponds to the point of invertibility [23, 20]. This constitutes the main difference to typical modulation formulations in the sound synthesis literature, where the corresponding modulation strength parameter is usually not normalized (compare [1]). The perturbation itself is sinusoidal and could be interpreted as the projection of another oscillator onto the phase of the perturbed one. Just as with the sinusoidal oscillator, it is useful to make a distinction between feedback FM as

defined with the specific projection equation (6) and the sine circle map as not requiring it.

### 3.4. Projection from Coordinates

So far our projection function is a map from $[0, 1) \to [-1, 1]$ assuming peak normalized audio samples. While the geometry of a circle in the plane may be suggestive for equations (3-4), technically this equation does not specify a continuous map into the plane. The standard circle is a continuous map into the plane according to equation (1). This equation gives two coordinates in the plane and a projection would be a reduction of dimensionality $\pi : \mathbb{R}^2 \to \mathbb{R}$. In more generality, we can have an continuous map from a parametric loop into $\mathbb{R}^n$ with $n \geq 2$. Moreover a planar representation of any higher dimensional continuous maps can be achieved by a projection down onto the plane $\pi_2 : \mathbb{R}^n \to \mathbb{R}^2$.

We call a map $f(t)$ parametric if it consists of a set of $n$ coordinate maps $f_0, f_1, \ldots, f_{n-1}$ in each component of $\mathbb{R}^n$ that are each parametrized by $t$ on the interval $[0, 1)$. It is a parametric map of the circle $\mathbb{S}^1$ if parameters 0 and 1 are identified. We call the map $f$ a parametric embedding if it is an injection (the map is one-to-one). This implies that the map does not self-intersect in $\mathbb{R}^n$. With this we arrive at the following diagrammatic relationship between the parametric embedding $f(t)$, direct projection $p$, embedding projection $\pi$ and planar projection $\pi_2$:

$$\begin{array}{ccccc} \mathbb{S}^1 \cong \mathbb{R}/\mathbb{Z} & \xrightarrow{f(t)} & \mathbb{R}^n & \xrightarrow{\pi_2} & \mathbb{R}^2 \\ & \searrow^{p} & \downarrow{\scriptstyle \pi} & & \\ & & [-1, 1] & & \end{array} \tag{7}$$

### 3.5. Properties of Projections

Many different continuous maps into a substrate space can yield the same output under projection. This is captured in the diagram (7) by the following composition:

$$p = \pi(f(t)) \tag{8}$$

In general, projection maps will lose information because distinct higher dimensional or parametric information will often be mapped to multiple instances of the same coordinate point in the lower dimensional image of the projection. This includes the direct projection $p$. Consider the example of the sine projection. Ranges $[0, \pi/2)$ and $[\pi/2, \pi)$ both map to $[0, 1]$ and ranges $[\pi, 3\pi/4)$ and $[3\pi/4, 2\pi)$ map to $[-1, 0]$, hence the map maps to distinct coordinates only at two isolated points and maps to a multiplicity of 2 at all points except an additional exception of 0 which maps to a multiplicity of 4. Geometrically, this phenomenon corresponds to intersections or overlap between stretches of a parametrized path under the projection. We call this *degenerate* and its existence a *degeneracy*. A map into a substrate space is an *embedding* if the map is injective (one-to-one) and therefore does not contain any degeneracies. We call a map an *immersion* if it is injective except at isolated crossing points. Finally, maps can fail to be embeddings or immersions, but still be the image of a continuous map from a parametrization into the substrate space. In figure 1 we see examples (left) embeddings (without degeneracies), (middle) immersions (only isolated degeneracies,) and (right) long stretches of overlap, hence degeneracies being typical, these are neither embeddings nor immersions, yet they still can be a continuous image

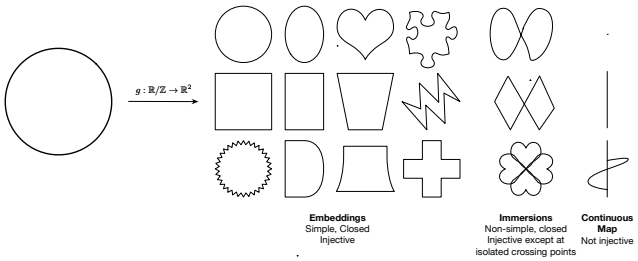


Figure 1: Examples of images of continuous maps $g$ from topological circle into the plane: (left) an embedding, (middle) an immersion, or (right) neither an embedding or an immersion. The top right example shows an isolated point.

of a map from a loop parametrization. An example that is not an image of a continuous map from the parametric circle into the plane are two non-touching points. We will call any continuous map $g$ of a topological space into a geometric space its *geometric realization*.

### 3.6. Equivalent Projections

It is useful to consider *equivalent projections*. These are the projections, which are the set of all possible projections from the parametrization or any geometric realization which create the same sample outputs. This can be used to either identify simple cases for convenient implementation, or flexibility in higher dimensional representation for the purpose of visualization [25] or to define control parameters that are convenient [12].

As an example, consider the projection $\pi : \mathbb{R}^2 \to [-1, 1]$, $\pi(x, y) = x$. Take $x = \sin(2\pi t)$. Clearly there is substantial freedom on the $y$ coordinate under this projection. The constant function $y = c$ but also any continuous function $y = cos(2\pi kt)$ where $k \in \mathbb{N}$ are permissible. The constant case is of further interest as it illustrates the important property that higher dimensional space or even the notion of embedding is not a requirement for the correct outcome. The constant case is equivalent to the "sideways" projection in $\mathbb{R}^3$ and it contains the topological connectivity of the circle oscillation even under the projection. While the side-projection itself no longer captures a circle topology, the underlying parametrization of the sine function does. This is the key to the utility of separating parametrized paths from projections. The parametrization disambiguates the multiplicities in the geometric realization in the substrate space. Hence we are only concerned if a circle-like closed path is parametrizable, and not if the path can be properly embedded in some higher dimensional space.

However, embeddings and immersions can be desirable restricted classes of geometric realizations for visualization. Embeddings into the plane allow us to uniquely identify each position on the parametrization visually. Immersions can offer a good compromise as the points of intersection are isolated. Hence it is helpful for visualizations and control parameter constructions.

## 4. DEFORMATION

The class of possible continuous maps permissible in this setup is vast. Hence it is helpful to restrict the case of maps to an inuitively more accessible subclass of maps. In sound synthesis the ability to gradually vary parameters is considered attractive [12], hence

we will adopt a deformation perspective. Concretely we will discuss the case of a time-parametric piecewise spline construction. We chose a spline type that allows a one-parameter variation of spline shape. Hence we arrive at two distinct forms of deformation: one is geometric from changed placements of control points of the spline, and the other stems from varying the spline shape between control points through parametric control.

### 4.1. Parametrized Loops via Tensioned Catmull-Rom Splines

As an exemplar of a parametrized closed curve construction, we use the tensioned version of the Catmull-Rom spline [26] — also known as cardinal splines [27, 28, 29] — in a closed curve configuration.

Catmull-Rom splines have some practical advantages as they can be specified merely by control points, and the effect of moving a control point is localized. This makes the curve construction straightforward to use in practice as well as easy to implement. For our purpose it is also desirable that tensioned Catmull-Rom splines do not have "nice" guarantees such as differentiability and that they can be made equivalent to piecewise linear construction by chosing the tension parameter $\tau$ to be 1. This helps demonstrate that these good properties are not at all necessary for parametrized loop construction to be sensible. At the same time, tensioned Catmull-Rom splines have good relationships to our signal processing understanding of interpolation [29] including converging to sinc interpolation in the limit [30].

The tensioned Catmull-Rom spline is computed from an ordered set of four local control points. The temporal parameter $t_i$ describes the location of interpolation between the spline control point positions $i$ and $i+1$ (the second and third control point) as follows [27]:

$$P_i(t_i) = \begin{bmatrix} 1 & t_i & t_i^2 & t_i^3 \end{bmatrix} \begin{bmatrix} 0 & 1 & 0 & 0 \\ -s & 0 & s & 0 \\ 2s & s-3 & 3-2s & -s \\ -s & 2-s & s-2 & s \end{bmatrix} \begin{bmatrix} P_{i-1} \\ P_i \\ P_{i+1} \\ P_{i+2} \end{bmatrix}$$

We will refer to this equation as (9). Any variable named $P$ is a point in $\mathbb{R}^n$ and above equation holds independently in each coordinate of the point coordinates, hence each coordinate can be independently and equivalently computed allowing for the interpolation to be applied in any number of dimensions. We will confine ourselves to $n = 2$. $P_i(t_i)$ is the interpolated coordinate of the path between control points $P_i$ and $P_{i+1}$ at parameter position $t_i \in [0, 1)$. This path will be referred to as *segment*. These relationships of control points in an ordered set to computed spline segment are illustrated in Figure 2. For the remainder of the paper we will simply say spline and spline interpolation, taking it to mean tensioned Catmull-Rom splines and their interpolation if the discussion refers to concrete computations.

It is customary to use a rescaled variation of the $s$ parameter for control. This new parameter $\tau = 1 - 2s$ is called *tension*. The classical Catmull-Rom spline [26] is recovered for a tension $\tau = 0$ [27] and piecewise linear interpolation is achieved at $\tau = 1$. We will use[1] $\tau$ in the range of $[-1, 5] \in \mathbb{R}$.

[1] All our rendered computations use Processing's tensioned Catmull-Rom implementation via their *curvePoint* interface. `https://processing.org/reference/curvePoint_.html`

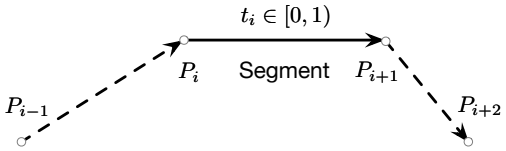


Figure 2: Catmull-Rom splines use four control points $P_{i-1}, \cdots, P_{i+2}$ to compute a spline segment parametrized by $t_1 \in [0, 1)$ between the second and third control points.

### 4.2. Indexing and Parametrization of Spline Segments

All our splines will be closed. This means that if we choose a configuration of $N$ points, point $N-1$ is connected to point 0. Hence the indices of all points are in the quotient $\mathbb{Z}/\mathbb{Z}_{\mathbb{N}}$ (usually referred to as $\mathbb{Z} \mod N$). Hence all index computations will be computed mod $N$. The index $i$ always refers to the second of the four points needed to compute a spline interpolation, consistent with equation (9). For example, assume the index position is $i = 0$ and $N = 4$. To compute the interpolation between $P_0$ and $P_1$ the two additional control points are computed to be $P_{-1} \mod 4 = P_3$ and $P_2$. Computing spline segments for each index $i \in [0, \cdots, N-1)$ will hence compute a closed spline. The total parametrization of the whole closed loop is $t = [0, 1)$ which is divided evenly between each control point pair. Hence the currently addressed control pair index $i$ is computed from $t$ as follows: $i = \lfloor N \cdot t \rfloor \mod N$. The time parameter $t_i$ used for each pair is computed from the loop parameter as $t_i = t \cdot N \mod 1$. This implies that all $t_i \in [0, 1)$. Points in a given spline shape will be identified with a given superscript $P_i^{\circ}$. If we refer to a complete sequence of points the index subscript will be omitted.

### 4.3. Examples 1: Four Point Diamond

The first concrete example of a closed spline loop is defined by the following control point coordinate notated as pair of coordinates in an ordered sequence:

$$P^{\diamond} = [(0, 1), (1, 0), (0, -1), (-1, 0)] \tag{10}$$

The total number of control points in this cycle is $N = 4$. The location of the control points correspond to a square in a diamond configuration as can be seen in center of Figure 3. Henceforth we will simply refer to this configuration as *diamond*. The figure shows additional spline interpolations for different tension parameters $\tau$. While the linear realization and deformation under tension for some parameters remain convex, convexity does break down for tension greater than 1. The shape however retains a high degree of symmetry and hence presents as rather regular.

### 4.4. Example 2: Four Point Irregular Concave L-shaped

The second concrete example of a spline captures a substantial reconfiguration of the diamond four point loop defined in the previous section. It is defined by the following ordered sequence of $N = 4$ control point coordinates:

$$P^{L} = [(0.65, 0.65), (0.75, -0.7), (-0.8, -0.85), (0.6, -0.6)] \tag{11}$$

The piecewise linear rendering of the control point configuration for this shape can be seen in center of Figure 4. Due to its

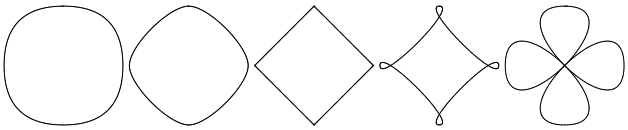

Figure 3: Closed loop tensioned Catmull-Rom splines with four points in a diamond arrangement. $\tau = -1, 0, 1, 2, 5$ from left to right. The center configuration corresponds to a linear connection between control point positions.

Figure 4: Closed loop tensioned Catmull-Rom splines with four points in an irregular non-convex L-shaped arrangement. $\tau = -1, 0, 1, 2, 5$ from left to right. The center configuration corresponds to a linear connection between control point positions.

shape we will referred to this configuration as *L-shaped*. None of the coordinates align and the configuration shape is concave to begin with. Under the same tension variations as in the previous section all configurations are concave and exhibit no obvious symmetries in the Euclidean plane.

## 5. PROJECTION OF SPLINE LOOPS

Our given examples are closed spline loops in the Euclidean plane $\mathbb{R}^2$. All our examples are bounded within the interval $[-1, 1] \in \mathbb{R}$ for each coordinate. The general requirement of a projection $\pi$ for our purposes is to be a map from $\mathbb{R}^2 \to [-1, 1]$. Hence we can use an orthogonal projection along the Cartesian coordinates and satisfy the domain requirements of this map. Throughout all examples we will project onto the second Cartesian coordinate $(x, y) \to y$ matching the y axis in depictions of figures if they display either loop geometries or sample amplitudes. The general case of all possible orthogonal projections can be derived by applying a rotation in the plane and normalizing the maxima along projection direction.

## 6. AUDIO OUTPUT USING SPLINE LOOPS

Given that we have discrete iterative phase functions, geometric path constructions via splines, and suitable projections, we are ready to compute audio samples. All computations were performed at a rate of 44100 Hz. Each computed iteration provides one sample. All samples are equally spaced.

For all computations involving the fast Fourier transform (FFT) a length of 4096 bins was used. The signal was weighted using a Blackman window function. For all parameter plane computations over nonlinear feedback maps (the sine circle map and its spline variations), the first 1000 iterations were computed but skipped to eliminate short-lived non-linear transients, such as rapid fixed point convergences [31]. Fourier spectra are normalized to sampling frequency (SR) and plotted to Nyquist frequency (SR/2) and amplitudes are normalized.

### 6.1. Bare Circle Map

The simplest example of an interactive phase function is the bare circle map of equation (3). For a sinusoidal projection function we would call this oscillator sinusoidal. With the spline construction we can now replace the base sinusoidal projection with dynamics on spline loops. Hence, this case constitutes a deformed generalization of the sinusoidal oscillator to other loop shapes.

Figures 6 shows the Fourier spectrum of the diamond spline of equation (10) under orthogonal projection for tension parameters $\tau = -1$, 1 and 5 (compare Figure 3 for the respective spline shape). The same tension parameters were also rendered for the L-shaped spline of equation (11) (see Figure 4 for its shapes). The resulting Fourier spectrum is depicted in Figure 7. As was to be expected, none of the spectra shows a sinusoidal oscillator. Rather we see rich yet discrete spectra of varying regularity and strength corresponding to the underlying spline deformations. The rightmost spectra for each shape show aliasing peaks as no anti-aliasing[2] has been applied.

### 6.2. Sine Circle Map

As with with standard sinusoidal oscillation, the canonical form of feedback FM uses sinusoidal projections. We can however use the underlying dynamics of the sine circle map to generalize feedback FM to use deformed circles as "carriers".

Figure 5 shows a comparison of various deformation of a closed loops from the circle with the discrete dynamics of the sine circle map after equation (5). The figure uses parameter planes with perceptual measures [31]. Each pixel color represents the value of an aggregate measure of spectral content of the audio signal at a given parameter pair. We employ the PeakSparsity measure, which is defined as follows [31]:

$$\text{area}_n = \sum_{m=1}^{n} \text{sort}_m(f) \qquad \text{sort}_n(f) \text{ are sorted FFT bins}$$

$$\text{c}_n = \begin{cases} 1 & \text{if } \sum_{m=1}^{n} \text{area}_m \leq \text{mean} \\ 0 & \text{otherwise} \end{cases}$$

$$D_{\text{PS}} = \sum_{n=1}^{N} c_n \tag{12}$$

The PeakSparsity measure is designed to characterize how peak-dominated a spectrum is and it leans on an integration notion of the spectral information. The core idea is to compute the area underneath the spectrum, which captures the mean of the spectrum. Peaks will contribute more to the overall area than a flat spectrum or valley areas between peaks. Hence sorting the spectrum by peak heights and counting the number of peak bins that are required to reach the mean will differentiate spectra with strong peaks that dominate the integral. A flat spectrum will have a count of half the bins. A single peak in an otherwise zero spectrum will count 1 bin. Hence we normalize the range between 1 and the number of bins over two to get the dynamic range of the measure which is mapped onto a color palette as can be seen on the right in Figures 5 and 8. Blue corresponds to a single dominant peak, while red corresponds to a flat spectrum. We expect simple oscillatory and mode locking regions to show near blue and regions captured

[2] Anti-aliasing strategies for this setting are beyond the scope of the this paper.

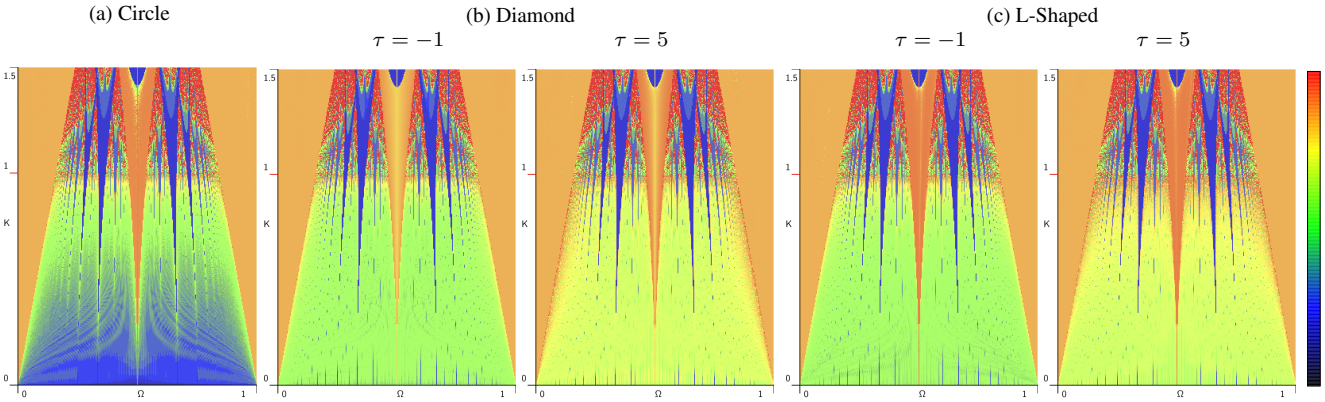


Figure 5: Parameter planes using the PeakSparsity measure computed from 4096 point FFT over parameter ranges $\Omega = [0, 1)$ and $k = [0, 1.5)$ for the sine circle map for (a) circle, (b) diamond shaped spline (c) L-shaped spline. Tensions shown are $\tau = -1$ and $\tau = 5$ if applicable.

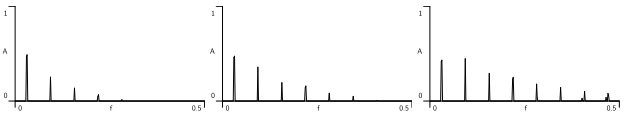

Figure 6: Spectra for the diamond shaped spline for $\tau = -1$, $\tau = 1$ and $\tau = 5$ with $\Omega = 0.03141593$ (corresponding to 1385.4 Hz at 44 100 Hz sampling rate.).

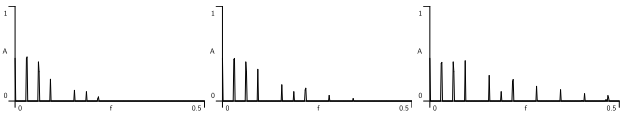

Figure 7: Spectra for the L-shaped spline for $\tau = -1$, $\tau = 1$ and $\tau = 5$ with with with $\Omega = 0.03141593$ (corresponding to 1385.4 Hz at 44 100 Hz sampling rate.).

by fixed points to show in near red. Loosely, more peaks mean a richer spectrum, so the transition from blue to red corresponds to an increasingly rich spectrum. Near red this interpretation can fail due to the potential of silence from fixed points, which would also produce a flat spectrum, but for values between light blue and yellow this ambiguity between flat and silent spectrum does not exist and therefore the intuition is sensible and allows us to characterize spectral changes in visualization.

The parameter plane is filed by sweeping over two parameters to be studied. The two parameters of interest for the sine circle map are $\Omega$, the frequency of the carrier — in the language of feedback FM — and $k$, the modulation strength, which also can be thought of as the strength of the feedback nonlinearity. For $k > 1$ the dynamics transitions into chaotic behavior and emerging triangular "tongues" correspond to regions of mode locking [23, 20].

Figure 5 shows feedback FM on the left. Each spline shape (diamond, and L-shaped) is displayed for two tension parameter choices ($\tau = -1$ and $\tau = 5$). The picture shows that the general structure of the sine circle map is preserved but modified under different loop geometries. For the circle case, one sees that for low nonlinear feedback ($k << 1$) this case shows low spectral richness (blue). For other loop geometries, we observe an increase in spectral richness (green to yellow). The degree to which the spectrum is enriched as a function of $\tau$ mirrors that of the bare circle map depicted in Figures 6 and 7.

## 7. PROJECTION OF FEEDBACK OR MODULATION ONTO AN INTERATIVE PHASE

Modifications of the sine circle map have previously been studied by changing the nonlinear perturbations function to piecewise linear, or Fourier-series functions instead of the sine function [21]. In the spirit of our geometrizing loops and projections instead of functions, we can repeat the process we just employed to change the base or "carrier" loop geometry for the non-linear perturbation function in the iterative phase function that either is used for feedback, as in the case of the sine circle map, or for modulation. To illustrate this process we rewrite the sine circle map equations (5-6) to replace the sine projection with a general one:

$$x_n = x_{n-1} + \Omega + Hp_1(x_{n-1}) \mod 1 \quad (13)$$

$$y_n = p_0(x_n) \quad (14)$$

We have now two projections related to a parametrized loop. The case studied before is now label $p_0(\cdot)$ and the new projection is called $p_1(\cdot)$. This $p_1$ can then be computed by the same processes already discussed. With reference to typical nomenclature in the signal modulation literature, we will refer to topological circles geometrically realized and projected by map $p_0$ as *base* or *carrier*. And we will refer to the same for $p_1$ as *modulation*.

Taking the example of close spline curves, we are now able to take one arbitrary closed spline as carrier and another arbitrary closed spline as a modulation, hence allowing for both their deformation from the circle simultaneously.

The results of deformations of both base and modulation oscillations are collected in Figure 8. All examples for splines are

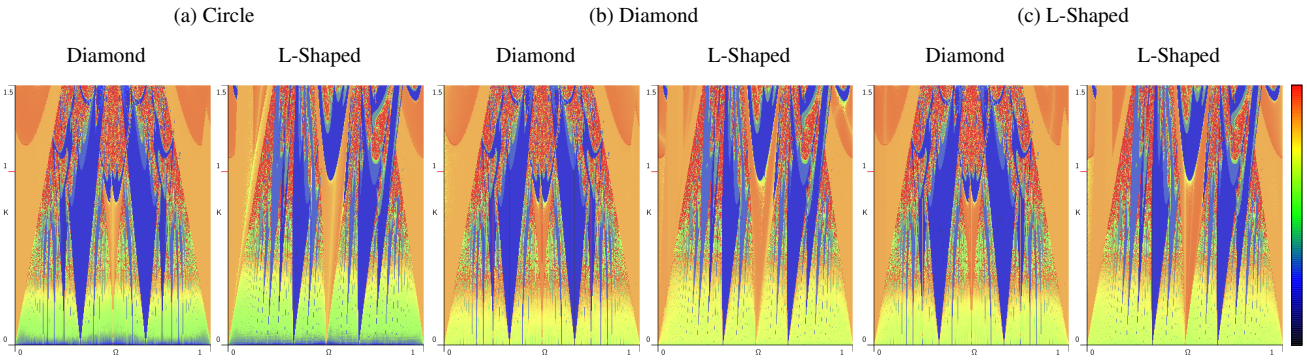


Figure 8: Parameter planes using the PeakSparsity measure computed from 4096 point FFT over parameter ranges $\Omega = [0, 1)$ and $k = [0, 1.5)$ for the general circle map for (a) circle, (b) diamond shaped spline at tension $\tau = 5$, and (c) L-shaped spline at tension $\tau = 5$. For each the figure shows the effect of using a (left) Diamond and (right) L-shaped spline for the feedback projection.

rendered for a tension parameter of $\tau = 5$. Case (a) captures the case were the base is still the Euclidean circle but with modulation by both diamond and L-shaped splines. This case constitutes a $\tau = 5$ deformation of the modulation for the case for the circle depicted in Figure 5 (a). Additionally, the figure shows the case where the base space and the modulation oscillation is deformed. We observe that across all cases the deformation of the modulation oscillation leads to substantial changes in the pattern of the Arnold tongues. If the modulation spline loop is diamond, then the Arnold pattern stays symmetrical. For the L-shaped loop the pattern becomes asymmetrically skewed, a phenomenon also observed for non-symmetric nonlinear functions [1]. Furthermore we observe that deformations of the base oscillation does again show corresponding changes in spectral richness in the low nonlinearity range ($k << 1$). However, additionally, the deformation of the oscillator accelerates the transition into high richness spectra as can be seen comparing Figure 5 (a) and Figure 8.

### 7.1. Cascading Projections onto Multiple Iterative Phase Functions

The process discussed in the previous section can be continued for cascading modulations or feedback:

$$x_n^i = x_{n-1}^i + \Omega + H^i p_{i+1}(f_i(x_{n-1}^i, x_{n-1}^{i+1}, \omega_m^i)) \mod 1 \quad \forall i \in 0, .., N \in \mathbb{N} \tag{15}$$

$$y_n = p_0(x_n^0) \tag{16}$$

where $f_i$ is an iterative phase function composed of feedback, modulation by another iterative phase function[3] $x^{i+1}$ and a potential "modulation frequency" constant $\omega_m^i$. There are $N$ iterative phase functions for each modulation cascade and associated constants $H^i$, $\omega_m^i$ and projection on the previous iterative phase function $p_{i+1}$. The first iterative phase function can be thought of as the carrier and it projects audio data via $p_0$. All these projections can be replaced by the geometric path construction following equation (8).

[3] All superscript notations here are indices and not exponents.

## 8. CONCLUSIONS

Recently oscillatory sound synthesis methods have been reformulated as discrete dynamical systems over a circle topology. In this paper we illustrate the flexibility in terms of geometric realization enabled by this formulation. In particular, many classical modulation and feedback oscillatory techniques can be generalized to use a wide range of closed path constructions in lieu of the classic circular oscillator. Hence, we provide a general context in which many known oscillatory synthesis techniques can be applied. In particular we discuss the role of mapping a path parametrization into a potentially high-dimensional substrate space, and the projection down to audio samples. This yields a clear criterion of which aspects of the geometry are relevant for audio outcomes, and which are free to be used for visual expression. We illustrate the robustness of the construction by demonstrating chaotic oscillations using feedback modulations over non-circular loops in the plane.

In this paper we have ignored aliasing as a topic, though of course, for any practical use, it is relevant. Particularly the sharp edges of linear interpolations as achieved with a tension parameter of $\tau = 1$ will alias. However, over the last decade, a body of research has developed uses spline interpolation for anti-aliasing [32, 33, 34]. Specifically the piecewise linear case in the plane has been investigated already [35]. The full development of this question is future work.

This paper used splines to illustrate constructive examples but did not offer a full theory of the relationship to embedded spline loops under projection to audio outcomes and specifically precise predictions of spectral properties. Such an endeavor is beyond the scope of this paper. This topic is technically rich and should offer a wealth of potential future research avenues.

Ultimately the main aim of this paper is to show that topological constructions can give clarity to synthesis methods and circumscribe proper generalizations of known techniques.

## 9. ACKNOWLEDGMENTS

All figures were rendered interactively using the Processing environment except figures 1 and 2 which was sketched in OmniGraffle. Many thanks to four anonymous reviewers for their detailed comments which helped improve the manuscript.

## 10. REFERENCES


[1] Georg Essl, "Iterative phase functions on the circle and their projections: Connecting circle maps, waveshaping, and phase modulation," in *Perception, Representations, Image, Sound, Music*, Richard Kronland-Martinet, Sølvi Ystad, and Mitsuko Aramaki, Eds. 2021, pp. 681–698, Springer.

[2] Curtis Roads, John Strawn, et al., *The computer music tutorial*, MIT press, 1996.

[3] Ken Steiglitz, *A Digital Signal Processing Primer, with Applications to Digital Audio and Computer Music*, Addison Wesley, 1997.

[4] Michael Robinson, *Topological Signal Processing*, Springer, 2014.

[5] Georg Essl, "Topological IIR Filters Over Simplicial Topologies via Sheaves," *IEEE Signal Processing Letters*, vol. 27, pp. 1215–1219, 2020.

[6] Georg Essl, "Topologizing Sound Synthesis via Sheaves," in *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, Vienna, 2021.

[7] Lance Putnam, *The Harmonic Pattern Function: A Mathematical Model Integrating Synthesis of Sound and Graphical Patterns*, Ph.D. thesis, University of California, Santa Barbara, 2011.

[8] Lance Putnam, "A Method of Timbre-Shape Synthesis Based On Summation of Spherical Curves," in *Proceedings of the International Computer Music Conference*, 2014.

[9] Lance Putnam, Stephen Todd, and William Latham, "Abstract Shape Synthesis From Linear Combinations of Clelia Curves," in *Proceedings of The 8th ACM/EG Expressive Symposium*, 2019, pp. 87–99.

[10] Axel Röbel, "Adaptive additive synthesis using spline based parameter trajectory models," in *International Computer Music Conference*, 2001, pp. 369–371.

[11] Nick Collins, "SplineSynth: An Interface to Low Level Digital Audio," in *Proceedings of the 1999 Diderot Forum on Mathematics and Music (Vienna)*, 1999.

[12] Oliver R. Bown and Sam Britton, "Methods for the Flexible Parameterisation of Musical Material in Ableton Live," in *Ninth Artificial Intelligence and Interactive Digital Entertainment Conference*, 2013.

[13] Steven Trautmann, "A Physical String Model with a Twist," in *Proceedings of the International Computer Music Conference (ICMC)*, 1995.

[14] Georg Essl, "Aspects of the Topology of Interactions on Loop Dynamics in One and Two Dimensions," in *LNCS 3310 Proceedings of the International Symposium on Computer Music Modeling and Retrieval 2004*, pp. 220–231. Springer Verlag, Esbjerg, Denmark, 2005.

[15] Herbert Edelsbrunner and John Harer, *Computational Topology: An Introduction*, American Mathematical Society, 2010.

[16] Robert Ghrist, *Elementary applied topology*, Createspace, 2014.

[17] Gunnar Carlsson, "Topology and Data," *Bulletin of the American Mathematical Society*, vol. 46, no. 2, pp. 255–308, 2009.

[18] Jose A. Perea and John Harer, "Sliding Windows and Persistence: An Application of Topological Methods to Signal Analysis," *Foundations of Computational Mathematics*, vol. 15, no. 3, pp. 799–838, 2015.

[19] Jose A. Perea, "Topological Time Series Analysis," *Notices of the American Mathematical Society*, vol. 66, no. 5, pp. 686–694, 2019.

[20] Georg Essl, "Circle Maps as a Simple Oscillators for Complex Behavior: I. Basics," in *Proceedings of the International Computer Music Conference (ICMC)*, 2006.

[21] Georg Essl, "Circle Maps as a Simple Oscillators for Complex Behavior: II. Experiments," in *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, Montreal, September 18-20 2006.

[22] Bai-Lin Hao and Wei-Mou Zheng, *Applied Symbolic Dynamics and Chaos*, World scientific, 2nd edition, 2018.

[23] Vladimir I. Arnold, "Small denominators. I. Mapping the circle onto itself," *Izvestiya Rossiiskoi Akademii Nauk. Seriya Matematicheskaya*, vol. 25, no. 1, pp. 21–86, 1961.

[24] Norio Tomisawa, "Tone production method for an electronic musical instrument," Feb. 10 1981, US Patent 4,249,447.

[25] Insook Choi and Robin Bargar, "Interfacing sound synthesis to movement for exploring high-dimensional systems in a virtual environment," in *1995 IEEE international conference on systems, man and cybernetics. Intelligent systems for the 21st century*. IEEE, 1995, vol. 3, pp. 2772–2777.

[26] Edwin Catmull and Raphael Rom, "A class of local interpolating splines," in *Computer aided geometric design*, pp. 317–326. Elsevier, 1974.

[27] David Salomon, *Curves and Surfaces for Computer Graphics*, Springer Verlag, 2007.

[28] Isaac J Schoenberg, *Cardinal Spline Interpolation*, SIAM, 1973.

[29] Michael Unser, "Splines: a perfect fit for signal and image processing," *IEEE Signal Processing Magazine*, vol. 16, no. 6, pp. 22–38, 1999.

[30] Antonio De Sena and Davide Rocchesso, "A Fast Mellin and Scale Transform," *EURASIP Journal on Advances in Signal Processing*, vol. 2007, pp. 1–9, 2007.

[31] Georg Essl, "Exploring the Sound of Chaotic Oscillators via Parameter Spaces," in *Proceedings of the International Conference on Digital Audio Effects (DAFx)*, 2019.

[32] Vesa Välimäki, Jussi Pekonen, and Juhan Nam, "Perceptually informed synthesis of bandlimited classical waveforms using integrated polynomial interpolation," *Journal of the Acoustical Society of America*, vol. 131, no. 1, pp. 974–986, 2012.

[33] Jussi Pekonen, Juhan Nam, Julius O. Smith, and Vesa Välimäki, "Optimized Polynomial Spline Basis Function Design for Quasi-Bandlimited Classical Waveform Synthesis," *IEEE Signal Processing Letters*, vol. 19, no. 3, pp. 159–162, 2012.

[34] Fabián Esqueda, Vesa Välimäki, and Stefan Bilbao, "Rounding corners with BLAMP," in *Proc. Int. Conf. Digital Audio Effects (DAFx-16)*, 2016, pp. 121–128.

[35] Christoph Hohnerlein, Maximilian Rest, and Julian D. Parker, "Efficient Anti-aliasing of a Complex Polygonal Oscillator," in *Proc. Int. Conf. Digital Audio Effects, Edinburgh, UK*, 2017, pp. 125–129.