



WIKIPEDIA
The Free Encyclopedia

[Main page](#)
[Contents](#)
[Featured content](#)
[Current events](#)
[Random article](#)
[Donate to Wikipedia](#)
[Wikipedia store](#)

[Interaction](#)
[Help](#)
[About Wikipedia](#)
[Community portal](#)
[Recent changes](#)
[Contact page](#)

[Tools](#)
[What links here](#)
[Related changes](#)
[Upload file](#)
[Special pages](#)
[Permanent link](#)
[Page information](#)
[Wikidata item](#)
[Cite this page](#)

[Print/export](#)
[Create a book](#)
[Download as PDF](#)
[Printable version](#)

[Languages](#)
[Deutsch](#)
[Italiano](#)
[Edit links](#)

[Create account](#) [Log in](#)

[Article](#) [Talk](#)

[Read](#) [Edit](#) [View history](#)

Karplus–Strong string synthesis

From Wikipedia, the free encyclopedia

(Redirected from [Karplus-Strong string synthesis](#))



This article includes a [list of references](#), but **its sources remain unclear** because it has **insufficient inline citations**. Please help to [improve](#) this article by [introducing](#) more precise citations. *(December 2013)*

Karplus–Strong string synthesis is a method of [physical modelling synthesis](#) that loops a short waveform through a filtered delay line to simulate the sound of a hammered or plucked [string](#) or some types of [percussion](#).

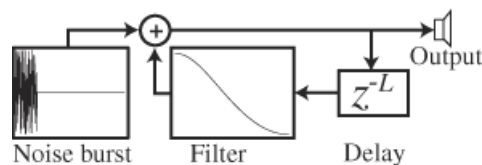
At first glance, this technique can be viewed as [subtractive synthesis](#) based on a [feedback loop](#) similar to that of a [comb filter](#) for [z-transform](#) analysis. However, on the other viewpoint, it can also be viewed as the simplest class of [wavetable](#)-modification algorithms now known as [digital waveguide synthesis](#), because the delay line acts to store one period of the signal.

[Alexander Strong](#) invented the algorithm, and [Kevin Karplus](#) did the first analysis of how it worked. Together they developed software and hardware implementations of the algorithm, including a custom VLSI chip. They named the algorithm "Digitar" synthesis, as a [portmanteau](#) for "digital guitar".

Contents [\[hide\]](#)

- [1 How it works](#)
- [2 Tuning the string](#)
- [3 Refinements to the algorithm](#)
- [4 Musical Applications](#)
- [5 References](#)
- [6 External links](#)

How it works [\[edit\]](#)



1. A short excitation waveform (of length L samples) is generated. In the original algorithm, this was a burst of [white noise](#), but it can also include any [wideband](#) signal, such as a rapid [sine wave chirp](#) or frequency sweep, or a single cycle of a [sawtooth wave](#) or [square wave](#).
2. This excitation is output and simultaneously fed back into a [delay line](#) L samples long.
3. The output of the delay line is fed through a [filter](#). The [gain](#) of the filter must be less than 1 at all frequencies, to maintain a stable positive feedback loop. The filter can be a first-order lowpass filter (as pictured). In the original algorithm, the filter consisted of averaging two adjacent samples, a particularly simple filter that can be implemented without a multiplier, requiring only shift and add operations. The filter characteristics are crucial in determining the harmonic structure of the decaying tone.
4. The filtered output is simultaneously mixed back into the output and fed back into the delay line.

Tuning the string [\[edit\]](#)

The [fundamental frequency](#) (specifically, the lowest nonzero resonant frequency) of the resulting signal is the lowest frequency at which the unwrapped phase response of the delay and filter in cascade is -2π . The required [phase delay](#) D for a given fundamental frequency F_0 is therefore calculated according to $D = F_s/F_0$ where F_s is the sampling frequency.

The length of any digital delay line is a whole-number multiple of the sampling period. In order to obtain a [fractional delay](#), [interpolating filters](#) are used with parameters selected to obtain an appropriate phase delay at the fundamental frequency. Either [IIR](#) or [FIR](#) filters may be used, however FIR have the advantage that transients are suppressed if the fractional delay is changed over time. The most elementary fractional delay is

the [linear interpolation](#) between two samples (e.g., $s(4.2) = 0.8s(4) + 0.2s(5)$). If the phase delay varies with frequency, [harmonics](#) may be sharpened or flattened relative to the fundamental frequency. The original algorithm used equal weighting on two adjacent samples, as this can be achieved without multiplication hardware, allowing extremely cheap implementations.

[Z-transform](#) analysis can be used to get the pitches and decay times of the harmonics more precisely, as explained in the 1983 paper that introduced the algorithm.

A demonstration of the Karplus-Strong algorithm can be heard in the following [Vorbis](#) file. The algorithm used a loop gain of 0.98 with increasingly attenuating first order lowpass filters. The pitch of the note was A2, or 220 Hz.

Holding the period (= length of the delay line) constant produces vibrations similar to those of a string or bell. Increasing the period sharply after the transient input produces drum-like sounds.



Karplus-Strong #1

Sorry, your browser either has JavaScript disabled or does not have any supported player. You can [download the clip](#) or [download a player](#) to play the clip in your browser. $F_1 = 220\text{Hz}$

Problems playing this file? See [media help](#).

Refinements to the algorithm [\[edit\]](#)

[Alex Strong](#) and [Kevin Karplus](#) realized that the Karplus-

Strong algorithm was physically analogous to a sampling of the transversal wave on a string instrument, with the filter in the feedback loop representing the total string losses over one period. [Julius O. Smith III](#) [\[1\]](#) [↗](#) and others generalized the algorithm to [digital waveguide synthesis](#), which could also be used to model acoustic waves in tubes and on drum membranes. The first set of extensions and generalizations was presented in a paper in 1982 at the International Computer Music Conference in Venice, Italy, and published in more detail in 1983 in *Computer Music Journal* in an article entitled "Extensions of the Karplus Strong Plucked String Algorithm," by David A. Jaffe and Julius O. Smith.^{[\[1\]](#)}

[Alex Strong](#) developed a superior [wavetable](#)-modification method for plucked-string synthesis, but only published it as a patent.

Musical Applications [\[edit\]](#)

The first musical use of the algorithm was in the work *May All Your Children Be Acrobats* written in 1981 by [David A. Jaffe](#), and scored for eight guitars, mezzo-soprano and computer-generated stereo tape, with a text based on [Carl Sandburg's](#) *The People, Yes*. Jaffe continued to explore the musical and technical possibilities of the algorithm in *Silicon Valley Breakdown*, for computer-generated plucked strings (1982), as well as in later works such as *Telegram to the President*, 1984 for string quartet and tape, and *Grass* for female chorus and tape (1987).

The patent was licensed first to Mattel Electronics, which failed as a company before any product using the algorithm was developed, then to a startup company founded by some of the laid-off Mattel executives. They never got sufficient funding to finish development, and so never brought a product to market either. Eventually Yamaha licensed the patent, as part of the Sondius package of patents from Stanford. It is unknown whether any hardware using the algorithm was ever sold, though many software implementations (which did not pay any license fees to the inventors) have been released.

References [\[edit\]](#)

Citations

- ↑ [Jaffe & Smith 1983](#)

Bibliography

- Karplus, Kevin; Strong, Alex (1983). "Digital Synthesis of Plucked String and Drum Timbres". *Computer Music Journal* (MIT Press) **7** (2): 43–55. doi:10.2307/3680062 [↗](#). JSTOR 3680062 [↗](#).
- Jaffe, David A.; Smith, Julius O. (1983). "Extensions of the Karplus-Strong Plucked String Algorithm" [↗](#) (PDF). *Computer Music Journal* (MIT Press) **7** (2): 56–69. doi:10.2307/3680063 [↗](#). JSTOR 3680063 [↗](#).
- US application 4649783 [↗](#), Alexander R. Strong, Kevin J. Karplus, "Wavetable Modification Instrument and Method for Generating Musical Sound [↗](#)", published 1987-03-17
- US application 4622877 [↗](#), Alexander R. Strong, "Independently controlled wavetable-modification instrument and method for generating musical sound [↗](#)", published 1986-11-18
- Moore, F. Richard (1990). *Elements of Computer Music*. Upper Saddle River: Prentice-Hall. ISBN 0-13-252552-6.

External links [edit]

- The Karplus-Strong Algorithm [↗](#)
- Sound Examples [↗](#)
- More sound examples under CC license [↗](#)
- Flash application implementing a guitar [↗](#)
- David A. Jaffe's music, including sound examples [↗](#)

v · t · e	Sound synthesis types	[hide]
	Frequency modulation synthesis · Phase distortion synthesis · Scanned synthesis · Subtractive synthesis · Additive synthesis · Distortion synthesis · Linear Arithmetic synthesis	
Sample-based synthesis	Table-lookup synthesis · Wavetable synthesis · Granular synthesis · Vector synthesis	
Physical modelling synthesis	Digital waveguide synthesis · Banded waveguide synthesis · Karplus–Strong string synthesis · Formant synthesis	

Categories: [Sound synthesis types](#)

This page was last modified on 13 August 2015, at 20:47.

Text is available under the [Creative Commons Attribution-ShareAlike License](#); additional terms may apply. By using this site, you agree to the [Terms of Use](#) and [Privacy Policy](#). Wikipedia® is a registered trademark of the [Wikimedia Foundation, Inc.](#), a non-profit organization.

[Privacy policy](#) [About Wikipedia](#) [Disclaimers](#) [Contact Wikipedia](#) [Developers](#) [Mobile view](#)

